

# Oracle Fusion Cloud Student Aid Eligibility

---

## **Implementing Student Aid Eligibility**

23D

Oracle Fusion Cloud Student Aid Eligibility  
Implementing Student Aid Eligibility

23D

F85862-01

Copyright © 2023, Oracle and/or its affiliates.

Author: Oracle Student Management Information Development Team

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, and MySQL are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

# Contents

<b>Get Help</b>	<b>i</b>
<hr/>	
<b>1 Getting Started</b>	<b>1</b>
Overview of Student Aid Eligibility	1
Implementation Tasks and Security	1
Supported Browsers	2
Brand the Student Self Service Environment	3
<b>2 Configuration</b>	<b>5</b>
Overview of Configuration	5
Configuration Paradigms	5
Implementing Configuration Workbooks	6
Deploy a Configuration Update	7
Configuration in the FAS User Interface	7
<b>3 Integration</b>	<b>9</b>
Integration Overview	9
Integration Guiding Principles	9
Message Processing Gateway	10
Vocado US Department of Education Gateway	11
Integration Messages	12
Frequently Used Integration Tools	14
Create New Student Self Service User Accounts	14
Integration of SAML 2.0 and SSO	16
Integration with Oracle PeopleSoft Campus Solutions	22
Batch Jobs and TD Client Communications	22
Frequently Asked Integration Questions	22
<b>4 Conversion</b>	<b>25</b>
Overview of Data Conversion	25

---

Conversion Methods	25
Components of Data Conversion	25
Executing Conversion	26
Frequently Asked Conversion Questions	28
<b>5 Implementing Oracle Analytics Server Reports</b>	<b>29</b>
Overview of Reporting	29
Reporting Data Models	29
Delivered Reports	30
User-Defined Reports	30
<b>6 Letters Using Oracle Analytics Server</b>	<b>31</b>
Overview of Letters	31
Letter Processing	31
Letter Templates and Delivered Letters	32
Letter Batch Job	33
At-Will Letters	34
Letters Components	34
Disabling Letters	35

# Get Help

There are a number of ways to learn more about your product and interact with Oracle and other users.

## Get Support

You can get support at [My Oracle Support](#). For accessible support, visit [Oracle Accessibility Learning and Support](#).

## Get Training

Increase your knowledge of Oracle Cloud by taking courses at [Oracle University](#).

## Join Our Community

Use [Cloud Customer Connect](#) to get information from industry experts at Oracle and in the partner community. You can join forums to connect with other customers, post questions, suggest *ideas* for product enhancements, and watch events.

## Learn About Accessibility

For information about Oracle's commitment to accessibility, visit the [Oracle Accessibility Program](#). Videos included in this guide are provided as a media alternative for text-based topics also available in this guide.

## Share Your Feedback

We welcome your feedback about Oracle Applications user assistance. If you need clarification, find an error, or just want to tell us what you found helpful, we'd like to hear from you.

You can email your feedback to [oracle\\_fusion\\_applications\\_help\\_ww\\_grp@oracle.com](mailto:oracle_fusion_applications_help_ww_grp@oracle.com).

Thanks for helping us improve our user assistance!



# 1 Getting Started

## Overview of Student Aid Eligibility

Student Aid Eligibility automates the process of applying for U.S. Department of Education (ED) student financial aid.

Oracle Student Aid Eligibility (SAE) automates the processes of applying for student financial aid funds in accordance with ED guidelines. You can configure automated communications, tasks, and data exchange with ED, third-party systems, and students for minimal staff hands-on processing. A Baseline Configuration is delivered and updated with every release to cover the processing needs for most schools.

Here's what you can do with SAE:

- Configure real-time dynamic checklists and alerts in Student Self-Service to guide students through applying for student financial aid.
- Configure and automate Institutional Student Information Record (ISIR) processing. Document requests, electronic document submission with e-signatures, application verification, and ISIR corrections to determine a student's individual Expected Family Contribution (EFC) can be automated.
- Exercise Professional Judgement overrides.

**Note:** SAE is one of two Student Financial Aid (SFA) products from Oracle Higher Education. The other product, Student Financial Planning (SFP), includes all the features of SAE plus automated cost of attendance calculation, awarding and packaging funds, fund disbursing, and Return to Title IV funds processing. SAE includes both administrative (Financial Aid System; also known as FAS) and Student Self-Service features. When you see "SFA" referenced in Oracle Help topics, it means the topic is applicable to both SFP and SAE. In some shared topics, you may see "SFP only" or similar language if part of a topic is only applicable to SFP.

## Implementation Tasks and Security

Here's some information on Implementation tasks and security.

### Implementation Tasks

This list contains tasks to get prepared for your Student Financial Aid (SFA) implementation.

1. Activate your order. See Oracle Student Financial Planning Cloud Service: How to autoprovision a new SFP instance in the cloud (Doc ID 2725840.1) on [My Oracle Support](#) (MOS).
2. Confirm you received the Welcome e-mail from Oracle.
3. Verify access to your instance of the Student Financial Aid (SFA) application.
4. Verify access to your instance of the Student Self Service (if applicable).
5. Verify access to your instance of the Jaspersoft Web UI.
6. Submit a Service Request in My Oracle Support (MOS) if you want to request your environments only be accessible from specific IP addresses and IP address ranges through the use of IP Allow lists. When you submit the Service Request, provide the CIDR blocks that should be allowed access. This is an optional step.

7. Optionally, if you want to set up firewall rules within your network to allow access inbound or outbound for the Oracle Services Network (OSN), refer to the *OSN Public IP Ranges* documentation. For SFA, Oracle recommends that you use the CIDR blocks for the us-ashburn-1 and us-phoenix-1 regions.

**Note:** Egress traffic sourcing from the SFA Oracle Analytics Server (OAS) implementation uses a static address. Please see Oracle Analytics Server Static Address (Doc ID 2976582.1) on *My Oracle Support (MOS)*.

Use the *OSN Public IP Ranges* documentation as the source of truth for SFA public IP addresses. You can poll the published file to check for new IP address ranges as frequently as every 24 hours. Oracle recommends that you poll the published file at least weekly.

8. Create users in the SFA applications for anyone that will be completing the implementation process and make sure they have the role of System Administrator. To learn how to add users, create and edit roles, and update the password policy, see:
  - o *Security Management* for FAS security.
  - o *Manage Users* and *Manage Role Permissions* for Self Service security.
9. Access the latest Release Notes available on *My Oracle Support (MOS)*: Oracle Fusion Cloud Student Financial Planning: Release Notes (Doc ID 2613016.1). The release notes are organized by year and month. For example 20.10 for 2020, October.
10. Download the latest API assets and the Baseline Configuration Release Extract from *My Oracle Support (MOS)*: Oracle Fusion Cloud Student Financial Planning: Release Notes (Doc ID 2613016.1).
11. Use these additional resources:
  - a. Oracle Fusion Cloud Student Financial Planning: Integration Management (Doc ID: 2514682.1) on *My Oracle Support (MOS)*.
  - b. Oracle Fusion Cloud Student Financial Planning: Data Conversion (Doc ID: 2514676.1) on *My Oracle Support (MOS)*.

## Security Reference

As mentioned in Implementation Task 8, the tasks that people can do and the data that they can see depend on their roles, duties, and privileges. For information about these factors, see:

- *Using Security Management* for FAS security.
- *Manage Users* and *Manage Role Permissions* for Self Service security.

## Supported Browsers

These are the supported world wide web browsers.

This table illustrates which browsers are supported for SFA and SFA Student Self Service.

Web/Mobile Browser	Student Self Service UI	Student Financial Aid UI
Chrome	63 or later	62 or later
Chrome Mobile	63 or later	Not Supported
Firefox	52 or later	52 or later



Firefox Mobile	52 or later	Not Supported
----------------	-------------	---------------

## Brand the Student Self Service Environment

You can brand the delivered Student Self Service environment for consistent look and feel.

To maintain a consistent look and feel across all of your applications, you can setup your provisioned Student Self Service environment be branded with your institution's primary color and logo. See [Manage Self-Service Branding](#).



# 2 Configuration

## Overview of Configuration

Product configuration is how you customize Student Financial Aid (SFA) to the needs of your institution. Once your configuration is defined and uploaded, SFA can process student financial aid based on your custom business rules.

While some configuration is maintained directly using dedicated setup pages in the Financial Aid System (FAS) UI, most configuration is maintained in a series of comma-separated values (CSV) files that can be loaded into the system using the Configuration Administration page in the FAS UI. Oracle provides a standardized Baseline Configuration set of CSV files based on best practices. These files can be used as delivered or further customized to your institution's specific processes, goals, and objectives.

## Configuration Paradigms

There are two configuration paradigms: Reference Table Lookup Values and Groovy Scripts.

These are the two configuration paradigms:

- Reference Table Lookup Values: A simple lookup, similar in theory to a key/value pair.
- Groovy Scripts: A dynamically-typed, Java-syntax scripting language used to evaluate complex expressions and drive systemic automation. Groovy runs in its own virtual machine, has its own sandbox, and uses the same syntax as Java.

All of the dedicated setup pages in the FAS UI use the Reference Table Lookup Values paradigm. Examples are updating your school's Federal School Code, OPEIDs, or COD Routing ID.

Configuration Workbooks use both the Reference Table Lookup Values and Groovy Scripts paradigms to accommodate more complex setups that require a series of decisions to, for example, request Documents, evaluate Document Acceptance Criteria, submit ISIR Corrections, clear ISIR C-Codes, verify an ISIR, or evaluate Disbursement Criteria.

Another example, for Student Financial Planning only, is configuring your school's Cost of Attendance (COA). This workbook setup includes updating Reference Table Lookup values including Program Type, Enrollment Status, and Modality, and using Groovy Scripts to calculate more granular COA values that may be based on the student's ISIR grade level, permanent address, and specific program details.

Here's a more detailed example of Configuration Workbook setup regarding the management of an 2021-2022 ISIR C-Code:

1. Groovy scripts are configured in the ISIR Verification Codes Configuration Workbook to determine which Required Documents to request and the Auto Code Clearing logic to clear the Verification requirement using this logic:

If Student Financial Aid (SFA) receives an ISIR with C-Code 171, the system evaluates the student's ISIR and determines that the student is selected for Independent Verification, Verification Tracking Group V1 (Standard), their Tax Return is Already Completed, and the student's IRS Request Flag has a value of 02 which means IRS data for the student was transferred from the IRS and was not changed by the user prior to submission of an application or correction.

2. Based on this configuration, SFA requests the Independent Verification Worksheet and a Rollover Statement documents if the Student IRS Untaxed Portions of IRA Distributions and Pensions Data Field Flag has a value equal to:
  - o 1 = IRS data was transferred and not changed by user OR
  - o 2 = IRS data was transferred and changed by user to a different value OR
  - o 3 = IRS Request flag is equal to 02, 03, 04, or 07, and field has been corrected on this transaction OR
  - o 4 = IRS Request flag is equal to 02, 03, 04, or 07, and field has been corrected on a previous transaction.
3. Upon receipt of the Requested Documents, SFA compares the document metadata against the ISIR metadata. If the values match, SFA clears the C-Code and completes the Valid ISIR Process, but if the values do not match, SFA automatically submits ISIR Corrections using the document meta-data using the ISIR Discrepancy Rules configuration.

## Implementing Configuration Workbooks

The majority of the configuration is maintained in Configuration Workbook Comma-Separated Value (CSV) files that are updated and uploaded to the system .

### Overview of Baseline Configuration

Oracle provides a Baseline Configuration of populated Configuration Workbooks to provide a strong starting point for your configuration based on common U.S. Department of Education (ED) guidelines and customer historical ISIR data.

If you elect not to implement any institution-specific configuration, installing the Baseline Configuration allows you to process financial aid in compliance with the ED's regulations for the majority of students. Oracle updates the Baseline Configuration when regulations and product features are updated. A full set of updated Configuration Workbooks is delivered with Release Notes for every product release. To access product release downloads, see Doc ID 2613016.1 on [My Oracle Support](#).

### Baseline Configuration

Baseline Configuration is designed to automatically request documents, clear ISIR Comment Codes (C-Codes), package, award, disburse, and return funds based on U.S. Department of Education (ED) guidelines.

These workbooks require updates in order to implement the provided baseline configurations for Student Financial Aid. Items that require updates for Student Financial Planning (SFP) only are noted:

- School Workbook
  - o Federal School Code
  - o OPEIDs
  - o COD Routing ID (**SFP only**)
- Cost of Attendance (COA) Workbook (**SFP only**)
  - o All fields/columns

## Extensibility and Change Management of Configuration Workbooks

You may modify the delivered Baseline Configuration as much or as little as desired to meet your school's business needs. However, if you modify the configuration workbooks, you are responsible for maintaining the modifications when the delivered configuration is updated due to annual regulatory updates or new features.

Oracle does not warranty institutional-specific configuration because Oracle cannot guarantee it is compliant or that it produces desired results.

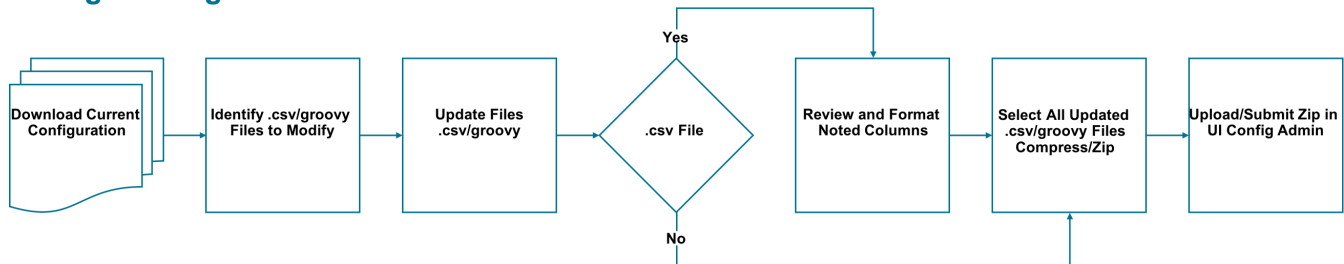
You can take a snapshot of each Baseline Configuration release to release manage configuration just like you release manage code. Anytime the Baseline Configuration is updated by Oracle, the changes or enhancements are documented in the release notes. If anything catastrophic goes wrong, you can always rollback the configuration to the prior configuration release. Anytime you update the configuration to meet your schools specific needs, you would save a ZIP file, for example, ConfigRelease\_%, of the configuration files and maintain a set of release notes for your school associated with the name of your ZIP file.

For more information, see [Configuration Workbooks](#).

## Deploy a Configuration Update

You can manage and deploy configuration with a simple user-friendly Configuration Administration option available in the FAS UI.

### Manage Configuration Process Flow



To deploy a Configuration Workbook update, you download the latest Baseline Configuration to your environment, modify CSV and Groovy files as you desire, and then save and upload the updates through the Load Configuration page in the FAS UI. For details about updating your configuration, see [Configuration Administration](#).

## Configuration in the FAS User Interface

There is some configuration that is maintained outside of the Configuration Workbooks in FAS (administrative) UI.

Configuration outside of the Configuration Workbooks includes whether QC Holds are enabled or disabled, Fund Allocation Periods and the amounts in each period, Award, Fund, and COD Closeout dates, User Roles and assigning

specific General Permissions and Document Permissions for each Role, and the password policy. These FAS UI screens require a Role with the Administration permission enabled. For more information, see [UI Configuration](#).

# 3 Integration

## Integration Overview

Integrations empower the exchange of information between two or more systems.

Integration messages contain information or the payload, which is being transmitted between Student Financial Aid and at least one school system. Student Financial Aid uses XML for its integration payload. Any external system may integrate with Student Financial Aid, as long as a consistent language is maintained; this language is defined in the XML Schema Definition (XSD) file(s). Student Financial Aid leverages its own integration messages to communicate between the Student Portal and itself.

Messages are sent to and from Student Financial Aid through the Message Processing Gateway (MPG), which contains built-in error handling and message retry mechanisms to handle exception cases. Integration messages may be sent in real-time or aggregated and sent in a batch at pre-determined intervals.

## Integration Guiding Principles

Student Financial Aid has a REST-Based Interface and uses two primary message types, event based and asynchronous.

Student Financial Aid Primary Message Types:

- Event-Based: Triggered at the time certain events occur
- Asynchronous Request/Reply: Triggered when a system makes a request for the information

Messages are Directional and are considered to be either Inbound or Outbound. Inbound messages are transmitted to Student Financial Aid from an external system. Outbound messages are created by Oracle Fusion Cloud Student Financial Aid Cloud Service and are intended to be consumed by an external system or the Student Portal.

Every message contains a Globally Unique Identifier (GUID) in the MessageId content to identify it as a unique message. Each message is validated before the data in the message is persisted to database. If an invalid message is received, an error is logged to the database and the message is discarded.

A message is considered invalid if:

- It does not match the XML schema.
- It has at least one field that is not compliant with the business validation rule(s) specified for that field.

## XML Schema Definition

The latest XSD files can be found in the API Update in the Release Notes. See Doc ID 2613016.1 on [My Oracle Support](#). You may find downloading the latest API assets beneficial as you work through the implementation process.

## Testing Your MPG and VUG Credentials

SFA APIs for MPG and the SAIG File Repository for VUG are usually accessed using integration software, not a website through an Internet browser like you would access the Administration (FAS) and Student Self-Service UIs. You can test your credentials to the MPG and VUG using the following procedure to trigger the ping service for each.

For MPG:

1. Identify the API URL, username, and password provided in the SFA Welcome Letter from Oracle with the subject "Welcome to Oracle Student Financial Aid Cloud".
2. In your browser, enter the URL into the address space and add `"/info/ping"` to the end:
  - a. Non-production example: `https://<username>@sfp.ocs.oc-test.com/<customer>-test/mpg/info/ping`.
  - b. Production example: `https://<username>@sfp.ocs.oraclecloud.com/<customer>/mpg/info/ping`.
3. Most modern browsers will open a dialog asking for a user ID and password. When it does, enter the appropriate username and password.
4. An example of a response for the MPG ping would be: "SFA MPG 19.12.2.0-SNAPSHOT (build b631f1d\_202001071806)".

For VUG:

1. Identify the SAIG File Repository URL, username, and password provided in the SFA welcome letter from Oracle with the subject "Welcome to Oracle Student Financial Aid Cloud".
2. In your browser, enter the URL into the address space and add `"/info/ping"` to the end:
  - a. Non-production example: `https://<username>@sfp.ocs.oc-test.com/<customer>-test/vug/info/ping`.
  - b. Production example: `https://<username>@sfp.ocs.oraclecloud.com/<customer>/vug/info/ping`.
3. Most modern browsers will open a dialog asking for a user ID and password. When it does, enter the appropriate username and password.
4. An example of a response for the VUG ping would be: "SFA USDE Gateway 19.12.2.0-SNAPSHOT (build 0f835a6\_202001071805)".

## Message Processing Gateway

Messages and files from the institution and student are sent to the Message Processing Gateway (MPG) application.

Outbound event messages are pulled from the MPG in first in first out (FIFO) sequence. Once a message is pulled from the MPG, it disappears from the outbound queue.

Messages and files from the institution and/or student must be saved with an `.xml` extension, as they contain xml payloads.

### Post a File to the MPG

This is an example using Mac OS X's built-in `curl` functionality. Any other web service integration tool you are comfortable with may be leveraged (for example *Postman*), as long as the username, password, file, and web service endpoint is maintained.

Example Command

```
curl -kv -X POST -u 'user:password' --key fasapi https://sfp.ocs.oraclecloud.com/customer/mpg/channels/in.ev/publish -H 'Vocado-Message-Class: FasStudentInitiationEvent' -H 'Content-Type: application/vocado-
```



```
message+xml" --data-binary "@/Users/Username/Desktop/student/FasStudentInitiationEvent.xml" -H 'Vocado-Message-Id: 11111111-1111-1111-1111-111111111111'
```

## Pull a File from the MPG

This is an example using Mac OS X's built-in *curl* functionality. Any other web service integration tool you are comfortable with may be leveraged (for example *Postman*), as long as the username, password, file, and web service endpoint is maintained.

Example Command to Pull a message based on it's Message ID

```
curl -k -X POST -u user:password https://sfp.ocs.oraclecloud.com/mpg/info/store/messages?messageId=11111111-1111-1111-1111-111111111111
```

Example Command to Pull a message based on First-In-First-Out

```
curl -X POST -u user:password -H 'Accept: application/vocado-message+xml' https://sfp.ocs.oraclecloud.com/mpg/channels/out.ev/consume -H 'Vocado-Pull-Id:11111111-1111-1111-1111-111111111111'
```

Create unique Message-Ids using the [Online GUID Generator](#).

## Vocado US Department of Education Gateway

Messages and Files sent from the US Department of Education (ED) via the Student Aid Internet Gateway (SAIG) are sent to the Vocado US Department of Education Gateway (VUG) application.

Institutional Student Information Reports (ISIRs) and National Student Loan Data System (NSLDS) financial aid history information files are flat files that must be saved with a .dat extension. Files from Common Origination and Disbursements (COD) are xml payloads and must be saved with a .xml extension.

**Note:** Student Financial Aid (SFA) is updated whenever ED makes changes to any record file layouts.

## SAIG Setup

For information regarding SAIG setup, see:

- [SAIG Administration](#).
- Student Financial Planning: Check list for New Award Year / Aid Year for OCI (Doc ID 2709030.1) on [My Oracle Support](#).

## Post an SAIG File to the VUG

This is an example using Mac OS X's built-in *curl* functionality. Any other web service integration tool you are comfortable with may be used, for example *Postman*, as long as the username, password, file, and web service endpoint is maintained.

```
curl -ki -X POST -u 'user:password' --key fasapi -F "file=@/Users/Username/Desktop/student/idap20op-a.dat" https://sfp.ocs.oraclecloud.com/customer/vug/in/publish
```

## Pull an SAIG File from the VUG

This is an example using Mac OS X's built-in *curl* functionality. Any other web service integration tool you are comfortable with may be used, for example *Postman*, as long as the username, password, file, and web service endpoint is maintained.

```
curl -X POST -u user:password https://sfp.ocs.oraclecloud.com/vug/out/consume?limit=1
```

## SAIG File Types: Naming Conventions

It is imperative during the testing and demo phases of implementation that mock SAIG files adhere to the naming conventions found in FSA Software and Manuals Message Class File Table on the U.S. Department of Education website. File types relevant to **SFP only** are indicated in the Description.

File Name	Direction	Description
FAHEXTOP	Inbound	<b>(SFP only)</b> Financial Aid History File
IDAPXXOP	Inbound	Daily Electronic Application ISIRs
CRPNXXOP	Inbound	<b>(SFP only)</b> Master Promissory Note
CRECMYOP	Inbound	<b>(SFP only)</b> Entrance Counseling Response

**Note:** If an NSLDS File load is successful using POSTMAN, but the NSLDS data isn't displayed on the FAS Administration UI, review the content of the file and validate that it meets all specifications outlined in the NSLDS record layouts from ED.

## Integration Messages

There are over 160 integration messages that are delivered with Student Financial Aid (SFA).

Delivered integration messages include messages sent to and received from the U.S. Department of Education (ED): SAIG, CPS, NSLDS, COD, and so on using the Vocado U.S Department of Education Gateway (VUG).

Additional integration messages are sent to and received between SFA and your school using the Message Processing Gateway (MPG).

**Note:** Messages that are applicable to Student Financial Planning only are noted in the Description column.

## Core Messages

There are a few core messages that institutions are required to create if they are not a current Campus Solutions customer.

Message	Source	Description	Type	Direction
FasStudentInitiationEvent	Contact Records Management (CRM)  Student Information System (SIS)	Creates the Student Record.	Event	Inbound
FasSafiSyncEvent	Student Information System (SIS)  Student Accounting System (SAS)	<b>(SFP only)</b> Referred to as the: Student Academic and Financial Information Message or SAFI Message.  Contains information related to the student's academic progress and financial account.  To identify the final academic year to be considered for proration, SFP uses the last academic year received in the Student Academic and Financial Information (SAFI) message. To ensure correct Direct Loan calculations, all academic years up to the end of the student's program must be sent in the SAFI Message.	Event	Inbound
FasStudentLoaAndBreaksIn	Student Information System (SIS)	<b>(SFP only)</b> Provides information about breaks in attendance from the academic program.	Event	Inbound
FasDisbursementInformation	Student Accounting System (SAS)  General Ledger (GL)	<b>(SFP only)</b> Updates the school's accounting system and reconciles the system based on the student's general ledger information.	Event & Asynchronous Request/Reply	Outbound
FasReturnToLenderEvent	Student Accounting System (SAS)  General Ledger (GL)	<b>(SFP only)</b> Indicates there is a Title IV payment due to the lender from accounting.	Event & Asynchronous Request/Reply	Outbound
FasCreditBalanceInformation	Student Accounting System (SAS)  General Ledger (GL)	<b>(SFP only)</b> Indicates a credit balance exists so accounting can provide money to the student.	Event	Outbound
FasSapAcademicInformation	Student Information System (SIS)	<b>(SFP only)</b> Shares the student's satisfactory academic progress data.	Event & Asynchronous Request/Reply	Outbound

## Frequently Used Integration Tools

You may find the following tools helpful during the integration process.

- Text Editor
- Command Line Tool
- XML Validation Tool
- Application Event Error Report

### Application Event Error Report

The Application Event Error Report helps you to troubleshoot message validation errors.

## Create New Student Self Service User Accounts

You can create student user accounts using the User Creation API REST-based messages. In addition to the Student Self Service User Interface user-create functionality, this API can be used to script bulk creation of student self service users.

Complete these steps:

### 1. Set Up Connectivity and Authentication

- a. **Authenticate** using an *admin* account username and password:

```
curl -v --cookie-jar ./cookie.txt https://admin-username:admin-  
password@sfp.ocs.oc-test.com/customerName/portal/fas
```

- b. Perform an initial GET request to acquire an **XSRFtoken**:

```
curl -v --cookie ./cookie.txt --cookie-jar cookie.txt https://sfp.ocs.oc-  
test.com/customerName/portal/
```

Steps 1a and 1b are only needed **once per session**. These steps create connectivity data in *cookie.txt* (SESSION and XSFR tokens) that is used to create one or more students repeating step 2.

### 2. Create a Student User

- a. Create Student:

```
curl -v -k \  
  -H "Content-Type: application/json" \  
  -H "X-XSRF-TOKEN: 60d9e1ee-ba14-4a95-9a0d-5de007ce2c02" \  
  --cookie ./cookie.txt \  
  https://sfp.ocs.oc-test.com/customerName/portal/api/v1/students -d @- <<  
EOF  
{  
  "firstName": "Test0001",  
  "lastName": "Student",  
  "externalId": "0001"
```

```
    }  
EOF
```

The **X-XSRF-TOKEN** value in this command must be set from the **cookies.txt** generated during **Step 1b** of the session.

```
# Netscape HTTP Cookie File
```

```
#HttpOnly_localhost FALSE /portal FALSE 0 JSESSIONID 92860D187E6B23D0E09802AE388A476A
```

```
localhost FALSE /portal FALSE 0 XSRF-TOKEN 60d9e1ee-ba14-4a95-9a0d-5de007ce2c02
```

**b. Create User for Student:**

```
curl -v -k \  
  -H "Content-Type: application/json" \  
  -H "X-XSRF-TOKEN: 60d9e1ee-ba14-4a95-9a0d-5de007ce2c02" \  
  --cookie ./cookie.txt \  
  https://sfp.ocs.oc-test.com/customerName/portal/api/v1/users -d @- << EOF  
  {  
    "username": "test0001",  
    "password": "0001",  
    "firstName": "Test0001",  
    "lastName": "Student",  
    "email": "test0001.Student@customerName.com",  
    "phone": "(800) 555-1212",  
    "externalId": "0001",  
    "student": "https://sfp.ocs.oc-test.com/customerName/portal/api/v1/  
students/100"  
  }  
EOF
```

The **X-XSRF-TOKEN** value in this command **must be set** from the **cookies.txt** generated during **Step 1b** of the session.

```
# Netscape HTTP Cookie File
```

```
#HttpOnly_localhost FALSE /portal FALSE 0 JSESSIONID 92860D187E6B23D0E09802AE388A476A
```

```
localhost FALSE /portal FALSE 0 XSRF-TOKEN 60d9e1ee-ba14-4a95-9a0d-5de007ce2c02
```

The **student** data element value is a unique url identifying the student and **must be set** from the **Location** header of the HTTP response output of **Step 2a** execution.

```
< HTTP/1.1 201
```

```
...
```

```
< X-Frame-Options: SAMEORIGIN
```

< Last-Modified: Wed, 24 Oct 2018 19:05:40 GMT

< Location: <https://sfp.ocs.oc-test.com/customerName/portal/api/v1/students/100>

Steps 2a and 2b can be repeated to create multiple students per session.

## Integration of SAML 2.0 and SSO

Student Financial Aid (SFA) uses SAML 2.0 SP-initiated SSO exchange to provide access (authentication and authorization) for client users into the SFA System.

### Access Student Financial Aid Via SSO and SAML

If your faculty and staff are logging into other web-applications through your enabled SAML 2.0 Identity Provider, you already know the identity of these users, so it only makes sense to use this information to log them into the Student Financial Aid Cloud Service. Security Assertion Markup Language (SAML) enables a Single Sign-On (SSO) exchange and provides your institutions users access (authentication and authorization) to the Student Financial Aid System.

Student Financial Aid uses the SAML 2.0 model where the Service Provider (SP) is defined as Student Financial Aid (SFA) and the Identity Provider (IDP) is defined as your Institution.

The Authentication Response SAML Assertion is sent by the IDP to the SP and contains necessary data about the user and the user authorization. The following attributes must be included in the SAML Assertion XML document:

- FirstName
- LastName
- EmailAddress
- Roles

**Note:** A unique identifier (usually the user login) should be used for the NameID.

**Note:** A username created in SFA can't be the same as the username in the IDP that you are using for SSO. In other words, the same account using the same set of identifiers can't exist in a single environment in both local and external form. But, a single environment can contain both local and external accounts with different identifiers for a unique user.

### Access Student Self-Service Via SAML 2.0

The Student Financial Aid Student Self-Service supports Security Assertion Markup Language (SAML) based authentication to the user interface.

To enable institutions to invoke Single Sign On (SSO), Student Self-Service has a URL that can be exposed. When a user attempts to access the user interface, the user is redirected to the IDP system configured in Student Self-Service. The user is then authenticated into the IDP and redirected to the Student Self-Service with the proper authorization. Users entering the Self Service UI via SAML, will not have to create separate SFA Self Service accounts, and will have access to all functions based on the role requested in their SAML assertion.

SAML assertion should contain these attributes, in addition to the (required) NameID. For information on managing SAML Attribute names, see *Manage SAML Attribute Names*:

- FirstName (required)
- LastName (required)
- EmailAddress (optional)
- PhoneNumber (optional)
- Student ID:
  - Role: Student and Parent (required)
  - Role: Admin (optional)
- Roles (required): Value can be any defined Role. For information on defining Roles, see *Manage Role Permissions*.

SAML/SSO integration allows the student or parent to access the Student Financial Aid Student Self-Service in the context of the student. The student's profile screen populates based on the values received in the SAML assertion message. It isn't necessary to create local student or parent accounts when accessing the system via SAML.

When external users access Student Self-Service via SAML this user profile information is displayed upon successful authentication:

- First Name
- Last Name
- Student ID
- Email Address (if available)

## Authentication Scenarios

In the authentication functional flow the Service Provider (SP) is defined as Student Financial Aid (SFA) and the Identity Provider (IDP) is defined as your Institution.

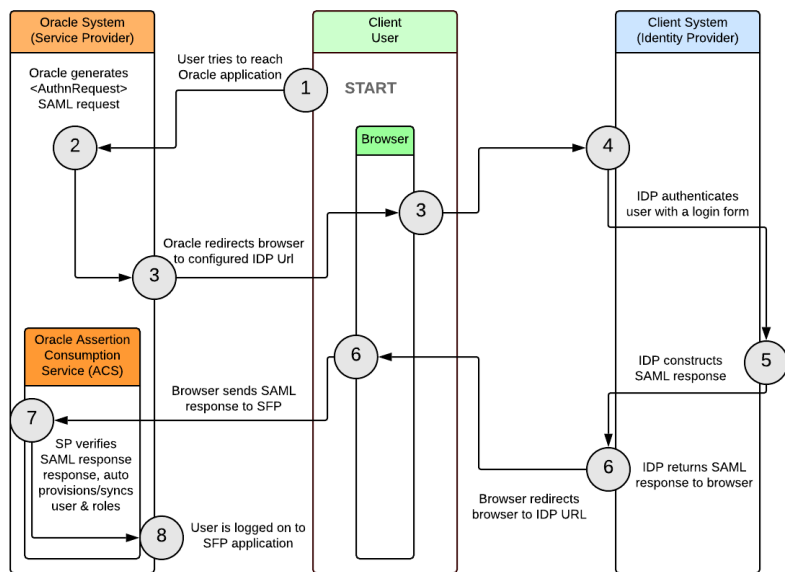
There are three authentication scenarios users may encounter:

- No SP Session and No IDP Session
- No SP Session but a Valid IDP Session
- Valid SP Session

### Login Scenario 1: No SP Session and No IDP Session

1. User tries to reach the Oracle application (SP).
2. SP generates the SAML request (e.g. AuthnRequest).
3. SP redirects the browser to the configured IDP URL.
  - a. Institution's Browser redirects the browser to the IDP URL.
4. IDP authenticates user by presenting login form.
5. IDP constructs SAML response.
6. IDP returns SAML response to institution's browser.
  - a. Institution's browser sends SAML response to SFA.
7. SFA Assertion Consumption Service (ACS) verifies SAML response, auto provisions and auto syncs user & roles.
8. User is logged into the SFA application.

**Login Scenario 1: No SP Session No IDP Session**

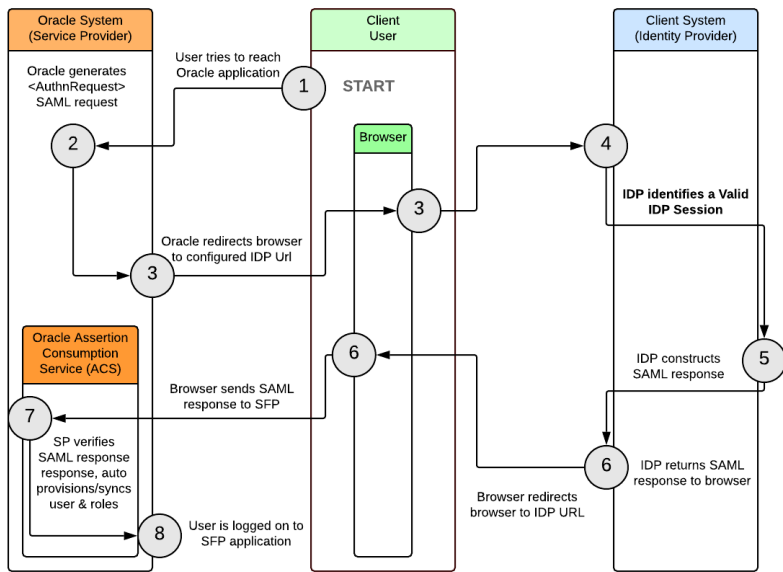


**Login Scenario 2: No SP Session but a Valid IDP Session**

1. User tries to reach the Oracle application (SP).
2. SP generates the SAML request (e.g. AuthnRequest).
3. SP redirects the browser to the configured IDP URL.
  - a. Institution's Browser redirects the browser to the IDP URL.
4. IDP identifies a valid session.
5. IDP constructs SAML response.
6. IDP returns SAML response to institution's browser.
  - a. Institution's browser sends SAML response to SFA.
7. SFA Assertion Consumption Service (ACS) verifies SAML response, auto provisions and auto syncs user & roles.
8. User is logged into the SFA application.



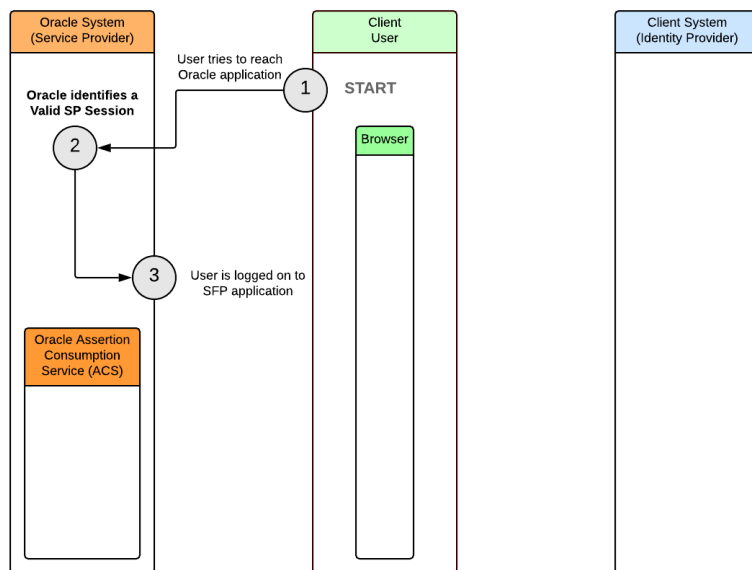
**Login Scenario 2: No SP Session Valid IDP Session**



**Login Scenario 3: Valid SP Session**

1. User tries to reach the Oracle application (SP).
2. SP identifies a valid session.
3. User is logged into the SFA application.

**Login Scenario 3: Valid SP Session**



## Enable SSO-SAML 2.0

Once your environment(s) have been provisioned and you are ready to setup Single Sign On you need to log a Service Request (SR) for *each* environment (e.g. Test, Stage, Production) and request IT setup the Financial Aid System (FAS) authentication redirect message. Because each Identity Provider and Service Provider need to agree upon the configuration for SAML, both ends need to have the exact configuration for the SAML authentication to work. You must also attach the IDP Metadata XML document to each SR. You may use a different IDP for accessing the FAS than you use for accessing Self Service.

### Steps to Enable SSO-SAML 2.0

1. Ensure the target environment has been provisioned and is ready to use.
2. Go to *My Oracle Support* and create an SR titled Enable SSO-SAML 2.0 (environment) for each provisioned target environment.
3. Include the IDP Metadata XML file (or link) in the SR.
4. Include the IDP entityID (or link) in the SR
5. Once Oracle Operations completes the SR, you will be able to download the SP/SFA metadata.
6. Update the environment\_name in the following non-production/production links to download the SP/SFA metadata.
  - a. Non-Production Link: [https://sfp.ocs.oc-test.com/environment\\_name/vm-ui/saml/metadata](https://sfp.ocs.oc-test.com/environment_name/vm-ui/saml/metadata)
  - b. Production Link: [https://sfp.ocs.oraclecloud.com/environment\\_name/vm-ui/saml/metadata](https://sfp.ocs.oraclecloud.com/environment_name/vm-ui/saml/metadata)

**Note:** The IDP Metadata XML document is required for each environment as the attributes may be different from one environment to another.

## Provisioning Scenarios

For users to be properly provisioned and synced, ideally the user and every role that can be mapped to a user should exist in Student Financial Aid System. But at a minimum, each role needs to exist to ensure unintended roles are not assigned to a user. To learn more about user and/or roles management, see:

*User Management.*

*Create A New Role.*

The following table provides common scenarios and their expected behaviors based on your school's user and roles setup in Student Financial Aid (SFA).

Scenario	Expected Behavior
User and all roles that should be mapped to the user exist in SFA.	<ul style="list-style-type: none"> <li>User is retrieved from the SFA user record.</li> <li>User is mapped to appropriate roles in SFA.</li> <li>User attributes and user-role assignments are synced in SFA.</li> <li>User session is initiated with SFA.</li> </ul>
User exists in SFA, but none or only some of the roles that should be mapped to the user exist in SFA.	<ul style="list-style-type: none"> <li>User is retrieved from the SFA user record</li> <li>User is mapped to all existing roles in SFA.</li> <li>User attributes and user-role assignments are synced in SFA.</li> <li>User session is initiated with SFA.</li> </ul>
User does not exist SFA, but all roles that should be mapped to the user exist in SFA.	<ul style="list-style-type: none"> <li>User lookup by userId.</li> <li>User not found, but user record is created.</li> <li>User is mapped to the appropriate roles in SFA.</li> <li>User session is initiated.</li> </ul>
User does not exist in SFA and none or only some of the roles that should be mapped to the user exist in SFA.	<ul style="list-style-type: none"> <li>User lookup by userId.</li> <li>User not found, but user record is created.</li> <li>User is mapped to all existing roles in SFA.</li> <li>User session is initiated.</li> </ul>
SAML assertion invalid/malformed.	<ul style="list-style-type: none"> <li>User is denied access.</li> <li>An integration error message is returned and displayed to the user.</li> </ul>

### Note:

- This use case does not support the use of Transient Identifiers in *Nameld* due to auto provisioning and syncing of the user.
- SFA Internal Users are managed under a different namespace than the SAML Provisioned Users.

## Session and Session Logout Management

When users are logged into Student Financial Aid they are engaged in an active local application session. The local application session is invalidated upon a system logout. A logout is triggered either explicitly by the user or implicitly by a session timeout.

To change your session timeout setting, log a service request with Oracle Support.

## Sample SAML XML Messages

You can download SFA Sample SAML XML Documents from Oracle Student Financial Aid Cloud Service: Integration Management 2514682.1 on [My Oracle Support](#).

# Integration with Oracle PeopleSoft Campus Solutions

Oracle PeopleSoft Campus Solutions (CS) customers can opt to integrate with the Student Financial Aid (SFA) products as an alternative to using CS Financial Aid.

SFA supports student financial aid programs in the United States including ISIR Processing, and, in SFP only, Standard Academic Year (SAY), Borrower Based Academic Year (BBAY), and Non-Term (NT) financial aid. SFP can also be used globally, but doesn't support financial aid localizations for Great Britain or Canada.

The integration between CS and SFA is an exchange of data in the form of sending and receiving integration messages between the two systems. For more information, see these resources for [PeopleSoft Campus Solutions in Oracle Help Center](#):

- [Integrating with Oracle Student Financial Planning Cloud Service in Oracle Campus Solutions 9.2: Campus Community Fundamentals](#).
- [Mapping Financial Transactions between CS Student Financials and Student Financial Planning \(SFP\) in Setting Up Item Types and Item Type Groups in Oracle Campus Solutions 9.2: Student Financials](#).
- [SFP Data Mapped to CS in Oracle Student Financial Planning Cloud Service: Integration Management 2514682.1 on My Oracle Support](#).

## Batch Jobs and TD Client Communications

Here is information regarding batch jobs and TD Client communications.

Student Financial Aid triggers Batch Jobs at scheduled times or based on specific events. For more information, see Batch Jobs and TD Client Communications, Document 2859959.1, on [My Oracle Support](#).

## Frequently Asked Integration Questions

Here are some questions frequently asked about integration. Information that relates to Student Financial Planning (SFP) only is indicated:

**Q:** Can we request NSLDS file from U.S. Department of Education (ED) from a Test SFP system? **(SFP only)**

**A:** No, the test environment does not communicate with ED via the SAIG mailbox.

**Q:** What is the web service end point for VUG where the SAIG files would be loaded into?

**A:** The test SAIG file repository URL is <https://sfp.ocs.oc-test.com/customer-test/vug>

**Q:** What are the commands to Post SAIG (for example, ISIR, NSLDS **(SFP only)**, COD **(SFP only)**) files and files from the institution or student?

**A:** An example curl command that demonstrates how to load files into the SAIG file repository can be found here: [Vocado US Department of Education Gateway](#). An example curl command that demonstrates how to load files into MPG application can be found here: [Message Processing Gateway](#). If you need the credentials, contact your service administrator.

**Q:** How many consumers can we connect to the MPG?

**A:** You can connect multiple consumers, but please note messages are pulled from the MPG in first in first out sequence and once a message is pulled from the MPG it disappears from the outbound queue. Therefore, for consistency, we recommend connecting only one consumer for event-based outbound messages.

**Q:** How do I set up an external Document Management System (DMS)?

**A:** To set up an external DMS instead of using the delivered DMS feature in Student Self Service, log a service request providing the base URL of your DMS. For example, if you access your DMS using a URL like [schooldms.com/docid123](#) where *docid123* is the document's unique ID, then your DMS's base URL would be [schooldms.com](#). Usually if you are using an external DMS, you are also using an external Student Self-Service solution.



# 4 Conversion

## Overview of Data Conversion

The U.S. Department of Education (ED) rigidly defines the exchange requirements for Financial Aid data. This enables Student Financial Aid (SFA) to use a delivered standardized set of code to quickly migrate data rather than requiring custom code for each school.

Oracle is focused on ensuring financial aid continuity for all students migrating from a legacy system to SFA.

Data conversion typically has a four phase lifecycle, exploration, planning, testing, and execution.

- **Exploration:** Review data requirements, foster understanding of data sources, and devise conversion strategy to minimize custom development for extracting data from legacy system(s).
- **Planning:** Establish cut-over plan and method; devise and refine strategy and time line for development, testing, and execution with consideration for student lifecycle and institutional requirements; for example, academic terms and end-of-life support for legacy systems.
- **Testing:** Perform testing and end-to-end dress rehearsals iterations to gain confidence in the data conversion process.
- **Execution:** Migrate students to SFA in accordance to the cut-over plan.

## Conversion Methods

Student Financial Aid can support converting students at the beginning of a new academic year.

### Academic Year Conversions

Convert students to Student Financial Aid at the beginning of a new academic year to minimize the need for legacy data extraction. This method does not Require disbursement or award data from the legacy system, simplifying the legacy extraction process. However, depending on student lifecycles, this method may take longer to migrate all students from the legacy system to Student Financial Aid.

### Big Bang Conversions

“Big Bang” conversions are a type of Academic Year conversion where all students are migrated from the school’s legacy system to SFA at one time. Using this method of conversion depends on timing, institutional requirements, and customer preference. This method works better for schools with traditional semester-based calendars.

## Components of Data Conversion

The tables below illustrate the file type it's associated naming convention, a description of the file, and the conversion method that could be used for the file type.

## SAIG Pre-Load Process

In order to continue student financial aid, legacy data from ED may be required.

Since SFA acts as the SAIG intermediary for an institution upon go-live, the conversion tool suite has a mechanism to load historical SAIG prior to go-live to maintain financial aid continuity for all students, even if the students are not immediately cutover to the system.

At the time of cutover, students are reunited with their appropriate and holistic set of Student Aid Internet Gateway (SAIG) data necessary to seamlessly continue financial aid, whether the data was received prior to go-live or during the pilot phase.

SAID File Types for **(SFP only)** are indicated.

## SAIG File Types and Naming Convention

File Type	Naming Convention
Institutional Student Information Record (ISIR)	idsaXXop-UUID.yyy
Financial Aid History (FAH) <b>(SFP only)</b>	fahextop-UUID.yyy
Common Origination and Disbursement (COD)  File Types are: crpn, crec, crsp, crcs, crco, crac, crat, crib <b>(SFP only)</b>	<File Type>XXop-UUID.yyy

Where:

- “XX” is the last two digits of the Award Year; except for COD File Type crec which is not Award Year-stripped, but always crecmy-UUID.yyy.
- UUID is the SFA-generated Universally Unique Identifier (UUID).
- “yyy” is initially “dat”, and then a sequence of numbers: “001”, “002”, “003” and so on.

For file name descriptions, see the U.S. Department of Education website.

## Executing Conversion

In Student Financial Aid (SFA), data is converted by sending the relevant payloads to the applicable web service interfaces: the Message Processing Gateway (MPG) and the Vocado USDE Gateway (VUG).

Conversion is conducted in iterative steps starting with the institution identifying the location and type of legacy data that needs to be migrated to SFA and ends with Oracle validating the data migrated to SFA successfully. Upon



completion of the last step, the SFA support team transitions maintenance and further execution of conversion to the institution's support resources.

Executing conversion includes the following steps:

- Identify Conversion Data
- Cleanse Legacy Data
- Extract Legacy Data
- Validate Legacy Data
- Transform Legacy Data
- Load Legacy Data into SFA

## Identify Conversion Data

Oracle is focused on ensuring financial aid continuity for all students migrating from a legacy system to SFA rather than persistence of historical student data.

As a best practice, it is recommended to only convert data necessary to continue a student's going-forward financial aid, without converting historical academic years for two reasons:

- Once financial aid is established for a prior academic year in the legacy system, it is generally unadvisable for SFA to attempt to re-state historical packages, disbursements, or returns.
- Not having the data in SFA inherently prevents accidental user errors in editing historical data.

Instead of converting historical data to SFA, it is recommended that historical data be exposed by means of a data warehouse solution, reporting engine, and so on in order to make the information available to users or auditors when necessary. As a byproduct, the implementation process is typically shorter and less costly.

**Note:** SFA does not support Financial Aid Award Years prior to 2012-2013. Although it is possible from a technology perspective to bring over all historical data, the level of effort and cost may be prohibitive to build the tools required to do so. Therefore, as a rule, it is recommended to focus on financial aid continuity rather than history, especially because closed out award years should be read-only.

## Cleanse Legacy Data

The school should make every effort to detect and correct corrupt or inaccurate records from a record set, table, or database. If the school identifies incomplete, incorrect, inaccurate or irrelevant parts of the data they should replace, modify, or delete that data prior to the execution of converting it to SFA.

## Extract Legacy Data

To lessen the burden on the institution and/or implementer as it relates to data extraction and data cleansing, SFA's standard data conversion tools leverage as much data directly from the U.S. Department of Education as possible such as ISIR rebuild files and NSLDS Financial Aid History.

For a Campus Solutions (CS) installations, SFA can convert all data (in most common scenarios) without requiring any custom data extraction. All data comes directly from the ED's Electronic Data Exchange (EDE) Systems or through a delivered interface between CS and SFA.

For a non-CS installations, SFA's standard data conversion tools leverage as much data directly from ED as possible. This includes Institutional Student Information Report (ISIR) rebuild files from the Central Processing System (CPS),

Common Origination and Disbursement (COD) rebuild files, Pell Year-to-Date files, Master Promissory Notes (MPNs), Credit Decision files from COD, and Financial Aid History files from the National Student Loan Data System (NSLDS). Furthermore, the conversion tools also attempt to leverage as much integration XML payloads as possible, including the standard Student Initiation, LOA and Breaks in Attendance, and SAP Information event messages.

## Load Data

Loading data to SFA entails sending relevant payloads to the applicable web service interface: the Message Processing Gateway (MPG) or the Vocado USDE Gateway (VUG).

# Frequently Asked Conversion Questions

These are questions frequently asked about Data Conversion.

Q: Can we convert student aid for prior years?

A: For federal funding, SFA automatically leverages a student's NSLDS Financial Aid History, which is a complete historical view of the student's federal funding. For non-federal funds, this decision can be made at time of implementation; the system can support as many historical years as there is historical non-federal data. However, please see Identify Conversion Data for additional information regarding historical financial aid information.

Q: How does SFA handle continuing the Pell disbursement numbering from existing disbursements in the Pell award year?

A: SFA's message API includes a historical fund information message, allowing SFA to be informed of Pell disbursements from the prior academic year in situations where the same award year crosses over two academic years. This message may be leveraged during the data conversion process to ensure SFA is able to seamlessly continue financial aid processing post-conversion.

# 5 Implementing Oracle Analytics Server Reports

## Overview of Reporting

You can create and run reports on system data using Oracle Analytics Server (OAS).

Use the initial Admin account login credentials and URL sent to you to access your instance of OAS.

The default configuration consists of a local user Admin account. Admin accounts can provision new local accounts and control local user accounts and roles using OAS. For more information, see *Installing and Configuring Oracle Analytics Server*. For more SFA-specific guidance on creating users in OAS, see Oracle Student Financial Aid Cloud Service: How To Create A New User In Oracle Analytics Server (Doc ID 2789879.1) on *My Oracle Support (MOS)*.

**Note:** OAS is provisioned for each SFA instance with default Security Configuration in the XMLP server and the Analytics server. Don't update any of the default security permissions on these servers.

## Report Scheduling

You can schedule when to run reports using dynamic date values. For more information, see *Using Oracle Analytics Publisher in Oracle Analytics Server - Define the Schedule for a Job*.

## Reporting Data Models

Oracle Analytics Server (OAS) uses groups of data reporting views called Reporting Data Models. These are different from System Data Models and are used only for reporting.

When you log into OAS, you can view the list of delivered Reporting Data Models by navigating to the Data Models folder. Click **Catalog > Shared Folders > SFA Reports > Data Models**.

The Reporting Data Models in that folder are maintained and updated by Oracle. If you update any of these Reporting Data Models and save them to this folder, they will be overwritten when there is a new release of Student Financial Aid. Also, there is potential negative downstream impact to delivered reports. If you update or create Reporting Data Models, Oracle recommends that you also create a separate folder for them. For more information on creating Reporting Data Models, see *Designing and Publishing Pixel-Perfect Reports in Oracle Analytics Server - Components of a Data Model*.

Reporting Data Models map to one or more FASRPT reporting views and aren't tailored for one specific report. They are designed to support multiple reports and used when you are creating new reports or customizing delivered reports. You can find more information about Reporting Views in the latest release notes. See Oracle Fusion Cloud Student Financial Aid: Release Notes (Doc ID 2613016.1) on *My Oracle Support (MOS)*.

## Delivered Reports

Oracle delivers and maintains several Oracle Analytics Server (OAS) reports.

You can find these by navigating to the Product Reports folder. Click **Catalog > Shared Folders > SFA Reports > Product Reports**.

For a list of delivered reports, see Oracle Student Financial Aid: Delivered Oracle Analytics Server Reports (Doc ID 2797029.1) on *My Oracle Support (MOS)*.

If a report includes both Summary and Detail output, only a single run of the report is required. In this case, the Summary and Detail outputs are displayed on different tabs. In OAS, you must select **Apply** when you navigate to each tab to view results.

**Note:** If all parameters are left blank, the report results will include all data.

**Note:** If you do not select values for all of the Required parameters, the report won't return any results.

## User-Defined Reports

You can create your own reports from scratch or by modifying delivered reports. You can also share reports or have reports that are only visible to you by using the Shared Folders and My Folders feature of Oracle Analytics Server (OAS).

- Shared Folders reports can be seen by all users with the correct permissions.
- My Folders reports can only be seen by you.

To create a report, see *Designing and Publishing Pixel-Perfect Reports in Oracle Analytics Server - Create Reports*.

You can copy and paste reports into multiple folders in OAS. You can also download reports and upload them to different environments as needed.

# 6 Letters Using Oracle Analytics Server

## Overview of Letters

You can engage students about their Financial Aid application process, including U.S. Department of Education (ED) requirements, using the Letters feature integrated with Oracle Analytics Server (OAS).

Events in Student Financial Aid (SFA) can trigger a Letter. Letters are sent outbound to be used by an external system. From the external system, maintained by the school, the Letter can be turned into an email, text message, pdf, or other format to be communicated to the student.

Use the initial Admin account login credentials and URL sent to you to access your instance of OAS.

The default configuration consists of a local user Admin account. Admin accounts can provision new local accounts and control local user accounts and roles using OAS. For SFA-specific guidance on creating users in OAS, see Oracle Student Financial Planning Cloud Service: How To Create A New User In OAS (Doc ID 2789879.1) on My Oracle Support (MOS).

**Note:** OAS is provisioned for each SFA instance with default Security Configuration in the XMLP server and the Analytics server. Don't update any of the default security permissions on these servers.

## Letter Processing

Letters are initiated by triggering actions and events in Student Financial Aid (SFA).

Triggering actions can initiate the Letter process. The triggering actions initiate an integration event message which can be associated to a specific letter template using Configuration. In the *Letter Configuration Workbook*, you can configure and review the relationship between the integration event message and the Letter template in the Oracle Analytics Server (OAS). You can also configure an integration event message to trigger multiple letters from a single letter template. Letter generation can be in real-time, configured to a specific time interval, or generated at-will by user.

To view configuration updates, view the Product Release Notes: Doc ID 2613016.1 on *My Oracle Support* and select the latest release version. You can download Configuration Workbooks from any available release and from the user interface if your role has the Administration and Load Configuration permission.

These event messages can be used to trigger a letter.

**Note:** Event messages applicable to Student Financial Planning (SFP) only are noted.

- FasCreditBalanceInformationEvent (**SFP only**)
- FasDisbursementAcceptanceInformationEvent (**SFP only**)
- FasDisbursementAcceptanceInformationV2Event (**SFP only**)
- FasDisbursementCancelEvent (**SFP only**)
- FasDisbursementInformationEvent (**SFP only**)

- FasDocumentRejectedEvent
- FasDocumentRequestCanceledEvent
- FasDocumentRequestEvent
- FasFinancialPlanOutboundEvent **(SFP only)**
- FasGrantOverpaymentReferralEvent **(SFP only)**
- FasProfessionalJudgementProcessOutboundEvent
- FasReleaseStudentRecordHoldEvent
- FasReturnToLenderEvent **(SFP only)**
- FasSapAcademicInformationEvent **(SFP only)**
- FasSapAppealsProcessInformationEvent **(SFP only)**
- FasStudentAwardInformationEvent **(SFP only)**
- FasStudentAwardInformationV2Event **(SFP only)**
- FasStudentNotificationEvent
- FasStudentRecordHoldPlacedEvent **(SFP only)**
- FasValidIsirProcessStatusInformationEvent
- TeachEligibilityCheckFailed **(SFP only)**
- TitlelvEligibilityCheckFailed **(SFP only)**
- FasFinancialPlanOutboundV2Event **(SFP only)**

## Letter Templates and Delivered Letters

Oracle delivers Letter templates and Letters. You can also customize delivered Letters.

Oracle delivers Letter templates that contain lorem Ipsum text and dynamic fields that are populated by simple queries.

Your business or regulatory office should provide appropriate letter verbiage. Use the Groovy-based *Letter Guide* configuration to determine when to use the letter template to generate a letter.

Letters are generated automatically based on configuration. Once generated, they can also be printed from the Student Financial Aid (SFA) FAS UI. See *Student Letters*.

**Note:** Letters that are Student Financial Planning (SFP) related only are noted in the Description.

Letter Code	Description	Template Name	Triggering Event
108	Student Record Hold Letter	FasStudentRecordHoldPlacedReport	FasReleaseStudentRecordHoldEvent
110	Verification Letter	FasValidIsirProcessStatusInformationReport	FasValidIsirProcessStatusInformationEvent
200	<b>SFP only</b> Exit Interview	FasStudentNotificationReport	FasStudentNotificationEvent
310	<b>SFP only</b> Right to Cancel Letter	FasDisbursementInformationReport	FasDisbursementInformationEvent

Letter Code	Description	Template Name	Triggering Event
325	<b>SFP only</b> Late Disbursement Notification	FasDisbursementAcceptanceInform	FasDisbursementAcceptanceInformationEvent
520	<b>SFP only</b> Cancellation FA Notification	FasDisbursementCancelReport	FasDisbursementCancelEvent
610	<b>SFP only</b> SAP RG Notification Letter	FasSapAcademicInformationReport	FasSapAcademicInformationEvent
620	<b>SFP only</b> SAP FA Warning Letter	FasSapAcademicInformationReport	FasSapAcademicInformationEvent
625	<b>SFP only</b> SAP FA Disqualification Letter	FasSapAcademicInformationReport	FasSapAcademicInformationEvent
725	<b>SFP only</b> Post Withdrawal Notification	FasDisbursementAcceptanceInform	FasDisbursementAcceptanceInformationEvent
FN1	<b>SFP only</b> Award Letter	FasStudentAwardInformationReport	FasStudentAwardInformationEvent
P01	<b>SFP only</b> Provisional Aid Offer Letter	FasFinancialPlanOutboundReport	FasFinancialPlanOutboundV2Event

You can also update delivered Letters by following these steps:

1. Access Oracle Analytics Server (OAS).
2. Navigate to **Shared Folders > SFP Letters Samples**.
3. Click **Edit** on the Letter you want to update.
4. Click **Edit** under the Letter template to download the template.
5. Complete any updates directly on the downloaded template.
6. Click **Save**.
7. Upload to your SFP Letters Samples folder.

## Letter Batch Job

You can create multiple letters at once using the Letter batch process.

To prevent duplicate letters being sent to the student, Student Financial Aid (SFA) uses a batch job and a configurable delay between the first Letter Code triggering action and when the batch job runs. The purpose of delay is to reduce the number of triggers the Letter batch job has to process. The Letter batch job always uses the latest trigger for each Letter Code for each student. This ensures that a student receives only one letter for each Letter Code for each batch job.

For example, let's assume a Letter Code has a configured delay of **1 hour**. The triggering actions occur at the noted times in the Letter Batch Example table, and the Letter batch Job is scheduled to run at once a day at 11:00 PM.

### Letter Batch Example

Time of FAS Event Trigger	Used in Letter Batch Job
6:00 PM	No
8:00 PM	No

Time of FAS Event Trigger	Used in Letter Batch Job
9:30 PM	Yes
10:30 PM	No, due to configured 1 hour delay. Considered in next day's run.

When the Letter Batch job runs at 11:00 PM, only the trigger created at 9:30 PM is used to create the Letter since the 9:30 PM trigger event contains the latest information and supersedes previous triggers. The 10:30 PM trigger falls within the 1 hour delay and will be considered for the next day's run.

When the Letter batch job runs:

- The parameter values and updated data from the latest triggering event are used to populate its corresponding Letter.
- All Letters are generated and published to the Student Financial Aid user interface.

As delivered, the Letters Batch job is configured to run once nightly. Use the Oracle Analytics Server Batch Job Schedule Report to see the schedule. To change the Letter Batch job schedule, by filing a service request with Oracle Support.

The Letter batch job can be scheduled to run once or multiple times per day. The delay can be configured in minutes in the Letter Configuration Workbook. Letters that do not have a configured delay are generated in real-time. Authorized users can manually trigger the Letter batch job as needed.

## At-Will Letters

Authorized users can generate letters at-will from the Oracle Analytics Server (OAS) user interface.

At-Will Letters can only be generated from the OAS user interface. There is nowhere in the Student Financial Aid user interface to generate a letter at-will. At-Will Letters can be generated from letter templates that aren't tied to specific Integration Events. To generate letters At-Will, you must provide the correct parameter values from the Student Financial Aid database; otherwise, you may create an inaccurate or empty letter.

## Letters Components

There are three main components to Letters: Structure, Data Model, and Parameters.

The **Structure** of a Letter template is composed of a Header, Body, and Footer. The Header of all Letter templates displays your school's logo and the Footer of all Letter templates displays general information such as address, phone number, and email address. Most of the components of a Letter are placed in the detail band in the Body of a Letter template. Examples of components that are placed inside the detail band are:

- Dynamic Text Boxes used to display dynamic fields and static text.
- Tables used to list funds and their information: name, amount, and payment period.

The **Data Model** tied to your letter template is what drives and populates the dynamic fields within the letter.

The **Parameters** are external values that are passed to their corresponding Letter templates by Student Financial Aid or a user.



## Disabling Letters

You can disable a Letter.

To stop sending a Letter, you can disable it using the *Letters Configuration Workbook*:

1. Access the Letter Configuration Workbook from the Configuration Administration page of the SFA FAS UI. See *Configuration Administration*.
2. To disable a single letter or letter code, update Column D, DEFN\_ACTIVE to FALSE for the letter or letter code you want to disable.
3. To disable the triggering event, update Column A, MAPPING\_ACTIVE to FALSE for the letter or letter code you want to disable.
4. Upload the updated workbook.

