



## Sample Detailed Design Document

# FFEL GA APPLICATION EXTENSION TECHNICAL DESIGN



## Federal Student Aid FMS Phase II

Author:	Brian McCann
Creation Date:	August 3, 2000
Last Updated:	September 1, 2000
Version:	Final



## Document Control

---

### Change Record

Date	Author	Version	Change Reference
03-Aug-99	Brian McCann	Draft 1a	No Previous Document
19-Aug-00	Adil Shaikh	Draft 1b	Incorporated changes from initial review, added text on error messages, appendices, database schema section.
01-Sep-00	Brian McCann	Final	Incorporated final comments
28-Oct-02	Robert Morse	Draft 2.0	Change SFA to FSA

#### Note To Holders:

If you receive an electronic copy of this document and print it out, please write your name on the equivalent of the cover page, for document control purposes.

If you receive a hard copy of this document, please write your name on the front cover, for document control purposes.



# Contents

Document Control .....	2
Technical Overview .....	4
Assumptions .....	4
Module List .....	4
Global Programs .....	7
Database Schemas .....	7
<b>FFEL Account Mapping - Form Logic</b> .....	11
Navigation Logic .....	12
Table and View Usage .....	12
Zone and Field Summary .....	12
Special Logic .....	14
<b>FSA GA Monthly Financial Report - Form Logic</b> .....	15
Navigation Logic .....	19
Table and View Usage .....	19
Zone and Field Summary .....	20
Special Logic .....	21
<b>FSA GA Maintenance Screen - Form Logic</b> .....	23
Navigation Logic .....	23
Table and View Usage .....	24
Zone and Field Summary .....	24
<b>FSALIB Global Package - Program Logic</b> .....	25
Table and View Usage .....	25
Program Logic (pseudo code) .....	26
SQL Statements .....	28
Validation Logic .....	30
<b>FSA FFEL GA Form 2000 AP/GL Load Interface - Concurrent Program Logic</b> .....	30
Calling Arguments .....	31
Log Output .....	31
Table and View Usage .....	32
Program Logic (pseudo code) .....	33
SQL Statements .....	34
Validation Logic .....	34
Incompatibility .....	35
Other Considerations .....	35
<b>FSA FFEL GA Form 2000 Monthly Data File Load - Concurrent Program Logic</b> .....	35
Calling Arguments .....	36
Log Output .....	36
Table and View Usage .....	37
Program Logic (pseudo code) .....	38
SQL Statements .....	38



**Sample Detailed Design Document**

Validation Logic .....39  
 Incompatibility .....40  
 Other Considerations.....40

**Technical Overview**

This document defines the technical components required to implement the Federal Family Education Loan Program - Guaranty Agency (FFEL GA) Application Extensions. This Technical Design Document complements the Functional Design Document for [FMS PHASE II FFEL GA Application Extensions](#) and the set combined should be considered to be the complete detailed design.

**Assumptions**

The following assumptions are necessary and relevant to the FSA FMS FFEL GA Extensions solution.

All incoming files received from GA's and NSLDS will have unique file names so that a clear audit trail may be maintained.

Output Treasury Files will be sent to the ED CFO office for routing to the Department of Treasury

The FSA FMS system will accept a single Monthly form for every Fiscal Month / Year. This assumptions still allows for two or more Monthly forms to be submitted within a one month time frame, but those forms must represent different Fiscal Months / Years.

**Module List**

**Forms**

[THE FFEL GA Application Extensions in FMS PHASE II](#) includes the following forms:

<b>FFEL Account Mapping</b>	This form allows the user to maintain account code combinations associated to data cells for the Monthly, Monthly/Quarterly, and Annual Financial Reports. In addition, this form can be used to maintain account code combinations for the AMF and LPIF fees as well as GAPS entries associated to LEAPP processing.
<b>GA Monthly Financial Report</b>	This form allows the GA users to create, modify, or delete monthly financial reporting information for submission to ED Financial Partners. The same form can be used by Financial Partners to accept or reject the report



Sample Detailed Design Document

	submitted.
<b>GA Monthly/Quarterly Financial Report</b>	This form allows the GA users to create, modify, or delete monthly/quarterly financial reporting information for submission to ED Financial Partners. The same form can be used by Financial Partners to accept or reject the report submitted.
<b>GA Annual Report</b>	This form allows the GA users to create, modify, or delete annual financial reporting information for submission to ED Financial Partners. The same form can be used by Financial Partners to accept or reject the report submitted.
<b>FSA GA Maintenance Screen</b>	The FSA GA Maintenance Screen allows the user to maintain GA Agreement Dates and specified GA parameters. The maintained dates are used for calculations within the Statement of Account report.

**Reports**

THE FFEL GA Application Extensions in FMS PHASE II includes the following reports:

<b>FSA FFEL Account Mapping Report</b>	This report allows users to view account code combinations that were 'active' for an extension as of a particular date.
<b>FSA FFEL GA Monthly Detail Report</b>	This report allows the user to write monthly financial information entered via the Form 2000 Monthly Form to an output file as a report.
<b>FSA FFEL GA Monthly/Quarterly Detail Report</b>	This report allows the user to write monthly/quarterly financial information entered via the Form 2000 Monthly/Quarterly Form to an output file as a report.
<b>FSA FFEL GA Annual Detail Report</b>	This report allows the user to write annual financial information entered via the Form 2000 Annual Form to an output file as a report.
<b>FSA FFEL GA Statement Of Account Report</b>	The report summarizes the Guaranty Agency's activity and provides information regarding its expected payment.
<b>FSA FFEL NSLDS AMF Fee Report</b>	This report provides information about the GA's AMF payments. It provides the AMF



Sample Detailed Design Document

	information for a FY (user provided parameter) like the fourth quarter actual amount, Fourth quarter adjusted amount (if spending cap is exceeded) , the net difference between the two and the projected first three quarter amounts per GA.
--	---

**Concurrent Programs**

THE FFEL GA Application Extensions in FMS PHASE II includes the following concurrent programs:

<b>FSA FFEL GA Form 2000 AP/GL Load Interface</b>	This program is responsible for creating invoices in AP and journal entries in GL for Form 2000 monthly, quarterly, and annual data submitted by each Guaranty Agency (GA). The program loads the information from the Form 2000 tables into the Oracle Applications Open Interface tables.
<b>FSA FFEL GA Form 2000 Monthly Data File Load</b>	This program is responsible for loading the Form 2000 Monthly financial data from a data file into the Form 2000 Monthly Reports table. GAs can use data files as an alternate method for submitting monthly reports.
<b>FSA FFEL GA Form 2000 Monthly/Quarterly Data File Load</b>	This program is responsible for loading the Form 2000 Monthly/Quarterly financial data from a data file into the Form 2000 Monthly/Quarterly Reports table. GAs can use data files as an alternate method for submitting monthly/quarterly reports.
<b>FSA FFEL GA Form 2000 Annual Data File Load</b>	This program is responsible for loading the Form 2000 Annual financial data from a data file into the Form 2000 Annual Reports table. GAs can use data files as an alternate method for submitting annual reports.
<b>FSA FFEL GA NSLDS LPIF Fee Load Interface</b>	This program is responsible for loading LPIF fee amounts associated to each GA from a data file into the AP open interface tables in order to create invoices.



Sample Detailed Design Document

<b>FSA FFEL GA NSLDS AMF Fee Load Interface</b>	This program is responsible for loading AMF fee amounts associated to each GA from a data file into the AP open interface tables in order to create invoices.
<b>FSA FFEL GA Populate Statement of Account Data</b>	This program is responsible for generating the Statement of Account (SOA) calculations and populating the SOA table which is the basis for the report referenced in the reports section above.



Global Programs

FSA FMS currently includes one global package (FSALIB\_GLOBAL\_PKG) that incorporates generic routines that can be used not only within PHASE II but in future phases as well. The package includes routines that are responsible for processes such as inserting into the AP interface tables, inserting into the GL interface table, getting a vendor’s information needed for invoice creation, submitting concurrent requests, etc.

One main reason for creating a global package was so that there would be minimal duplication of code across the various extensions (in PHASE II and future phases). It also provides the capability of having error checks incorporated in one central location rather than incorporating validation in all of the modules that need it.



Database Schemas

The FFEL GA extensions within FSA FMS utilize the FFELGA and FSALIB database schemas. The FFELGA schema has ownership of all database objects (i.e. tables, views, sequences, synonyms, packages, etc.) that serve as the basis for the development of the forms, reports, and concurrent programs. Objects created within the FFELGA schema are listed below:

OBJECT NAME	TYPE	PURPOSE
FFELGA_ACCOUNT_SEGMENTS	TABLE	Table used to store the account code combinations associated to various extension modules. The Form 2000 Account Mapping Form is the extension that modifies information in this table. See Appendix A for table definition.
FFELGA_ANNUAL_ITEMS	TABLE	Table used to store the Form 2000 Annual Financial Reports Itemized Line Item information. The Form 2000 Annual Reports Form (the Itemized Line Items Screen) and the Annual Data File Load program are extensions that create and modify information within this table. See Appendix A for table definition.
FFELGA_ANNUAL_ITEMS_TEMP	TABLE	Table used to temporarily store the Annual Line Item Data File information where it can be validated prior to being written to the FFELGA_ANNUAL_ITEMS table. See



Sample Detailed Design Document

		Appendix A for table definition.
FFELGA_ANNUAL_REPORTS	TABLE	Table used to store the Form 2000 Annual Financial Reports data. The Form 2000 Annual Reports Form and the Annual Data File Load program are extensions that create and modify information within this table. See Appendix A for table definition.
FFELGA_ANNUAL_TEMP	TABLE	Table used to temporarily store the Annual Data File information where it can be validated prior to being written to the FFELGA_ANNUAL_REPORTS table. See Appendix A for table definition.
FFELGA_GAMAINTS	TABLE	Table used to store GA Maintenance information. The GA Maintenance Form is used to create and/or modify data within this table. See Appendix A for table definition.
FFELGA_MONQUART_REPORTS	TABLE	Table used to store the Form 2000 Monthly/Quarterly Financial Reports data. The Form 2000 Monthly/Quarterly Reports Form and the Monthly/Quarterly Data File Load program are extensions that create and modify information within this table. See Appendix A for table definition.
FFELGA_MONTHLY_REPORTS	TABLE	Table used to store the Form 2000 Monthly Financial Reports data. The Form 2000 Monthly Reports Form and the Monthly Data File Load program are extensions that create and modify information within this table. See Appendix A for table definition.
FFELGA_MON_TEMP	TABLE	Table used to temporarily store the Monthly Data File information where it can be validated prior to being written to the FFELGA_MONTHLY_REPORTS table. See Appendix A for table definition.
FFELGA_NSLDS_AMF_FEES	TABLE	Table to store historical data associated to the NSLDS AMF fee process. The NSLDS AMF fees program extension creates and modifies information in this table. See Appendix A for table definition.
FFELGA_NSLDS_LPIF_FEES	TABLE	Table to store historical data associated to the NSLDS LPIF fee process. The NSLDS LPIF fees program extension creates and modifies information in this table. See Appendix A for table definition.
FFELGA_NSLDS_TEMP	TABLE	Table used to temporarily store the NSLDS AMF and LPIF Fee Data File information where it can be validated prior to being written to the FFELGA_NSLDS_FEES table. See Appendix A for table definition.
FFELGA_QUART_TEMP	TABLE	Table used to temporarily store the Monthly/Quarterly Data File information where it can be validated prior to being written to the FFELGA_MONQUART_REPORTS table.



Sample Detailed Design Document

		See Appendix A for table definition.
FFELGA_SOA_REPORTS	TABLE	Table to store generated Statements of Account. The Populate Statement of Account Data program extension creates and modifies information in this table. See Appendix A for table definition.
FFELGA_ACCOUNT_SEGMENTS_V	VIEW	A view object based on the FFELGA_ACCOUNT_SEGMENTS table used for query purposes. See Appendix A for view definition.
FFELGA_ANNUAL_ITEMS_V	VIEW	A view object based on the FFELGA_ANNUAL_ITEMS table used for query purposes. See Appendix A for view definition.
FFELGA_ANNUAL_REPORTS_V	VIEW	A view object based on the FFELGA_ANNUAL_REPORTS table used for query purposes. See Appendix A for view definition.
FFELGA_GAMAINTS_V	VIEW	A view object based on the FFELGA_GAMAINTS table used for query purposes. See Appendix A for view definition.
FFELGA_MONQUART_REPORTS_V	VIEW	A view object based on the FFELGA_MONQUART_REPORTS table used for query purposes. See Appendix A for view definition.
FFELGA_MONTHLY_REPORTS_V	VIEW	A view object based on the FFELGA_MONTHLY_REPORTS table used for query purposes. See Appendix A for view definition.
FFELGA_ANNUAL_FILE_LOAD_PKG	PACKAGE	Package containing procedures and functions associated with the Annual Data File Load program extension. See Appendix B for Package Definition.
FFELGA_AP_GL_LOAD_PKG	PACKAGE	Package containing procedures and functions associated with the AP/GL Load program extension. See Appendix B for Package Definition.
FFELGA_MONTHLY_FILE_LOAD_PKG	PACKAGE	Package containing procedures and functions associated with the Monthly Data File Load program extension. See Appendix B for Package Definition.
FFELGA_NSLDS_TO_AP_PKG	PACKAGE	Package containing procedures and functions associated with the LPIF Fee Load program extension. See Appendix B for Pkg Def.
NSLDS_AMF_FEES_PKG	PACKAGE	Package containing procedures and functions associated with the AMF Fee Load program extension. See Appendix B for Package Definition.
FFELGA_QUART_FILE_LOAD_PKG	PACKAGE	Package containing procedures and functions associated with the Monthly/Quarterly Data File Load program extension. See Appendix B for Package Definition.
FFELGA_SOA	PROCEDURE	Procedure responsible for performing Statement of Account calculations and populating the FFELGA_SOA_REPORTS table. See Appendix B for Procedure Definition.
FFELGA_ACCOUNT_SEGMENTS_S	SEQUENCE	Sequence used to generate unique values



Sample Detailed Design Document

		for the primary key of the FFELGA_ACCOUNT_SEGMENTS table. See Appendix A for sequence definition.
FFELGA_AR_ITEMS_S	SEQUENCE	Sequence used to generate unique values for the primary key of the FFELGA_ANNUAL_ITEMS table. See Appendix A for sequence definition.
FFELGA_AR_S	SEQUENCE	Sequence used to generate unique values for the primary key of the FFELGA_ANNUAL_REPORTS table. See Appendix A for sequence definition.
FFELGA_GA_MAINT_S	SEQUENCE	Sequence used to generate unique values for the primary key of the FFELGA_GAMAINTS table. See Appendix A for sequence definition.
FFELGA_MR_S	SEQUENCE	Sequence used to generate unique values for the primary key of the FFELGA_MONTHLY_REPORTS table. See Appendix A for sequence definition.
FFELGA_NSLDS_AMF_S	SEQUENCE	Sequence used to generate unique values for the primary key of the FFELGA_NSLDS_AMF_FEES table. See Appendix A for sequence definition.
FFELGA_NSLDS_LPIF_S	SEQUENCE	Sequence used to generate unique values for the primary key of the FFELGA_NSLDS_LPIF_FEES table. See Appendix A for sequence definition.
FFELGA_NSLDS_BATCH_S	SEQUENCE	Sequence used to generate unique numbers so that batch names being populated into the AP module are uniquely identified. See Appendix A for sequence definition.
FFELGA_QU_S	SEQUENCE	Sequence used to generate unique values for the primary key of the FFELGA_MONQUART_REPORTS table. See Appendix A for sequence definition.
FFELGA_SOA_S	SEQUENCE	Sequence used to generate unique values for the primary key of the FFELGA_SOA_REPORTS table. See Appendix A for sequence definition.

The FSALIB database schema currently owns the global package (FSALIB\_GLOBAL\_PKG) described above. The primary reason in separating the global package from other objects is because the global package will be utilized by other extensions (i.e. LEAPP, ED CFO) as well. Below is a list of objects within the FSALIB schema:

OBJECT NAME	TYPE	PURPOSE
FSALIB_GLOBAL_PKG	PACKAGE	Global package consisting of generic routines that are used by the various extensions. See Appendix B for Package definition.



**FFEL Account Mapping - Form Logic**

FILE NAME : FSAACCT.fmx

SFA Accounting Segments

Action Edit Query Go Folder Special Help

SUBSYSTEM Monthly

Item No	Fund	Cat	BFY	Account	Organiztn	Lim	Object Class	Acty	CFDA	CY	Sec	Source Code	Cost Code	Loan Grant	Dest
MR1_A_O	0230X01	B	01	901011	ENNC	BB7	44000	959	032	CY	0	GA	GAPA	00	GL
MR1_A_P	0230X01	B	01	218001	ENNC	BB7	44000	959	032	CY	0	GA	GAPA	00	AP
MR1_B_P	0230X01	B	01	218001	ENNC	BB7	44000	959	032	CY	0	GA	GAPA	00	AP
MR1_C_P	0230X01	B	01	218001	ENNC	BB7	44000	959	032	CY	0	GA	GAPA	00	AP
MR1_D_P	0230X01	B	01	218001	ENNC	BB7	44000	959	032	CY	0	GA	GAPA	00	AP
MR1_E_P	0230X01	B	01	218001	ENNC	BB7	44000	959	032	CY	0	GA	GAPA	00	AP
MR1_F_P	0230X01	B	01	218001	ENNC	BB7	44000	959	032	CY	0	GA	GAPA	00	AP
MR1_G_P	0230X01	B	01	218001	ENNC	BB7	44000	959	032	CY	0	GA	GAPA	00	AP
MR1_T	00000FY	0	BY	000000	000000000	000	00000	000	000	CY	0	00	0000	00	GL
MR2_I	0230X01	B	01	218001	ENNC	BB7	44000	959	032	CY	0	GA	GAPA	00	AP

Accounts

Transaction Code

Debit	Credit
<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>

The FFEL Account Mapping screen is used by the FSA CFO to manage the Guaranty Agency’s general ledger accounts by mapping Form 2000 entries to Account Codes.

The FFEL Account Mapping Form allows a user to create or update Account Code segments. The user may enter or select from a List of Values every Account Code segment except for Institution. The List of Values will display the codes and their description.

The Destination Code field controls whether the Form 2000 entry should go to ‘AP’ or ‘GL.’ There are also destination codes of ‘GLREV’ and ‘N/A’. The ‘GLREV’ specifies whether there is a reversal entry made in GL or not. Reversal entries are made for those items where balances are reported. Therefore, a previous time period balance is reversed and the new balance is entered reflecting the change. The ‘N/A’ destination code specifies that although there is an account mapping available for a particular item, the item’s value is not being populated into the Oracle Applications. Upon the user entering or selecting the Transaction Code from a List of Values, the corresponding debit and credit accounts will be displayed.



Sample Detailed Design Document

The Account Mapping Form defaults to query mode upon initiation. Using the 'SUBSYSTEM' list box, the user can query the database for all account code combinations relating to Monthly, Quarterly, Annual, NSLDS, or GAPS. In addition, users can query all account code combinations by selecting "All" as the SUBSYSTEM Code.

The user may update the Account Code and Transaction Code at any time. When an update is made, an audit trail is maintained. The Form logic updates the 'ENDDATE' column with the current date deactivates the original record. This way, FSA users can view which codes were effective at a particular time by generating the Form 2000 Account Mapping Report described in a subsequent section.

### Navigation Logic

#### Entry Mode

##### START

Log into Oracle Financials

From the Navigator screen, select 'FFEL Account Mapping'

User selects a 'SUBSYSTEM' (default = 'Monthly')

System queries all entries currently in the system related to the subsystem selected.

Cursor moves to field 1('Item Number').

User modifies data on the form.

User saves record.

##### END

### Table and View Usage

Table Name	Select	Insert	Update	Delete	Base Table
FFELGA ACCOUNT SEGM	X				X
FFELGA ACCOUNT SEGM		X	X		
GL USSGL ACCOUNT PAI	X				X
FND_FLEX_VALUES	X				
FND_FLEX_VALUE_SETS	X				

### Zone and Field Summary

#### Zone 1

Base Table: FFELGA\_ACCOUNT\_SEGMENTS\_V

Where/Order By:

TO\_NUMBER(SUBSTR(ASEG\_MR\_CODE,3,INSTR(ASEG\_MR\_CODE,'\_',1,1)-3)), ASEG\_MR\_CODE



Sample Detailed Design Document

Field/Column	Type	Rqd ?	Default	Validation	QuickPick
Item No	Char	Yes		None	
Fund	Char	Yes		Fund Account Segment Value Set	LOV
Category	Char	Yes		Category Account Segment Value Set	LOV
Budget Fiscal Year	Char	Yes		Budget Fiscal Year Account Segment Value Set	LOV
Account	Char	Yes		Account Segment Value Set	LOV
Organization	Char	Yes		Organization Account Segment Value Set	LOV
Limitation	Char	Yes		Limitation Account Segment Value Set	LOV
Object Class	Char	Yes		Object Class Account Segment Value Set	LOV
Activity	Char	Yes		Activity Account Segment Value Set	LOV
CFDA	Char	Yes		CFDA Account Segment Value Set	LOV
Cohort Year	Char	Yes		Cohort Year Account Segment Value Set	LOV
Sector	Char	Yes		Sector Account Segment Value Set	LOV
Source Code	Char	Yes		Source Code Account Segment Value Set	LOV
Cost Code	Char	Yes		Cost Code	LOV



Sample Detailed Design Document

Field/Column	Type	Rqd?	Default	Validation	QuickPick
		s		Account Segment Value Set	
Loan Grant	Char	Yes		Loan Grant Account Segment Value Set	LOV
Destination	Char	Yes			Drop Down List: AP, GL, GLREV

**Zone 2**

Base Table: GL\_USSGL\_ACCOUNT\_PAIRS

Where/Order By:

Field	Type	Rqd?	Default	Validation	QuickPick
FFELGA_ACCOUNT_SEGMENTS_V.ASEG_TRANSACTION_CODE	Char	No		GL_USSGL_ACCOUNT_PAIRS.USSGL_TRANSACTION_CODE	LOV
Debit	Char	No		None – Read Only	
Credit	Char	No		None – Read Only	

**Special Logic**

**Zone 2**

The GL\_USSGL\_ACCOUNT\_PAIRS base table block within the Account Mapping form is a read-only block. The Transaction Code field in Zone 2 is from the FFELGA\_ACCOUNT\_SEGMENTS\_V base table block and a relationship associated to the USSGL\_TRANSACTION\_CODE field from the GL\_USSGL\_ACCOUNT\_PAIRS block.

The Debit and Credit fields in Zone 2 are read-only fields and are populated based on the transaction code chosen. If modifications need to be made to the Debit and Credit account pairs, the GL Transaction Codes form provided as standard functionality within Oracle Applications needs to be used.

**Field: TRANSACTION CODE**

Once the Transaction Code field is populated by the user, the Enter key or the Tab key must be pressed in order to update the Debit and Credit account pairs. If this is not done, the user will not see the Debit and Credit pairs associated to the new transaction code until the form is re-queried.



**FSA GA Monthly Financial Report - Form Logic**

FILE NAME: FSAMON.fmx

Item No	Category Detail	Amount DueTo/(From) Guarantor	Principal Amount	Interest Amount	Other Amount
MR-1	Claims Paid	.00			
MR-1-A	Defaults-Net		20.00		0.00
MR-1-B	Exempt/Lender of Last Resort		0.00		
MR-1-C	Death/Disability		0.00		
MR-1-D	Closed School/False Certificat		0.00		
MR-1-E	Bankruptcy		0.00		
MR-1-F	Unpaid Refunds		0.00		
MR-1-G	Discharges		0.00		
MR-2	Borrower Payment Return (Cl	.00	0.00	0.00	0.00
MR-3	Status Changes	.00			

(GA VIEW)

**Entry Mode - GA**

The FSA GA Monthly Financial Report Form allows the GA to enter, submit, and review the monthly GA financial information corresponding to entries MR1 though MR23 on Form 2000.

Assuming that the GA user will be entering a new Monthly Financial Report, the screen begins in Data Entry Mode and is filled with the following default values: the user's own GA Code and Name, a Status of "New", and "0.00" for all MR fields. The user may enter or select the desired month and year from a list of values that are based on defined value sets by clicking on the List icon located in the toolbar.

The GA Code for a particular user is defaulted based on the user's association to a GA. Each GA within Oracle Applications is set up as a supplier. When GA users are created, the supplier ID for the respective GA is associated to the user's account thereby creating the link. The GA information is



Sample Detailed Design Document

stored within the PO\_VENDORS, PO\_VENDOR\_SITES\_ALL, and PO\_VENDOR\_CONTACTS table. The user\_ids of the GA users are stored within the FND\_USER table. These tables are part of the standard Oracle Applications schema.

The GA user then enters the dollar amounts for enterable, non-calculated MR fields. The Amount Due To/From Guarantor MR fields are non-enterable fields display-only fields and are calculated automatically based upon the calculations rules described in the FFEL GA Extensions Functional Design Document.

The user may either save the financial report for further processing by pressing the Save icon in the toolbar, or submit it to the ED Financial Partners for acceptance by pressing the Submit button. When the record is simply saved, the Status is automatically updated to "In Process." However, when the Submit button is pressed, the financial report is automatically saved and the Status is updated to "Submitted." All information associated to the monthly financial reports is saved into the FFELGA\_MONTHLY\_REPORTS table.

At the time that the user saves the financial report for further processing or submits it for acceptance, reasonability edits are performed against the dollar amounts using the following reasonability rules:

- Before submitting, a check is performed to ensure there is at least one non-zero dollar amount,
- MR-1 through MR-23 : + or -5% of corresponding amount for the same month for the previous year and,
- MR-1-A Other: >= MR-1-A Principal for same form

If the user is saving the financial report for further processing, a warning message will be displayed if a reasonability rule is violated. If the user is attempting to submit the report, a confirmation message is displayed asking whether to continue with the submission. These messages are described in the "Special Logic" section below.

Once submitted, the Financial Report becomes read-only. The GA user may not update or delete it. Furthermore, the Submit button becomes disabled. Also, a notification email via an Oracle Application Alert is sent to the ED Financial Partners as well as the GA user who submitted the Financial Report. The contents of the message will include GA Code, Fiscal Month and Fiscal Year.

The GA User may retrieve any previously saved financial report for his/her GA. The GA user may search on any combination of Status, Month, Year, ED Comment, GA Comment, and MR fields and may use wildcards. The reports retrieved based on the query will be sorted by Year and Month.

Although the user can retrieve all of his GA's financial reports, he/she may only update those with a status of "In Process" or "Rejected." Reports that have a status of "Submitted" or "Accepted" cannot be updated. Also, only reports with a status of "In Process" may be deleted should the user wish to start over.

If a report is rejected, the GA user may make the appropriate changes, provide a comment in the GA Comments field then, and re-submit the report to the ED Financial Partners. The Status is automatically updated back to



Sample Detailed Design Document

“Submitted.” An email notification via an Oracle Application Alert is sent to the ED Financial Partners and the GA user once a re-submit occurs.

**SFA GA MONTHLY FINANCIAL REPORT**

Action Edit Query Go Folder Special Help

GA Code  Status  For Fiscal Month of  /

GA Name

ED Comments

GA Comments

Item No	Category Detail	Amount DueTo/(From) Guarantor	Principal Amount	Interest Amount	Other Amount
<b>MR-1</b>	<b>Claims Paid</b>	<b>330.00</b>			
MR-1-A	Defaults-Net		110.00		110.00
MR-1-B	Exempt/Lender of Last Resort		110.00		
MR-1-C	Death/Disability		110.00		
MR-1-D	Closed School/False Certificat		0.00		
MR-1-E	Bankruptcy		0.00		
MR-1-F	Unpaid Refunds		0.00		
MR-1-G	Discharges		0.00		
MR-2	Borrower Payment Return (Cl	.00	0.00	0.00	0.00
MR-3	Status Changes	.00			



**(FINANCIAL PARTNER VIEW)**

**SFA GA MONTHLY FINANCIAL REPORT**

Action Edit Query Go Folder Special Help

GA Code  Status  Source  For Fiscal Month of  /

GA Name

ED Comments

GA Comments

Item No	Category Detail	Amount DueTo/(From) Guarantor	Principal Amount	Interest Amount	Other Amount
MR-1	Claims Paid	.00			
MR-1-A	Defaults-Net		20.00		0.00
MR-1-B	Exempt/Lender of Last Resort		0.00		
MR-1-C	Death/Disability		0.00		
MR-1-D	Closed School/False Certificat		0.00		
MR-1-E	Bankruptcy		0.00		
MR-1-F	Unpaid Refunds		0.00		
MR-1-G	Discharges		0.00		
MR-2	Borrower Payment Return (Cl	.00	0.00	0.00	0.00
MR-3	Status Changes	.00			

**Entry Mode - ED Financial Partner**

The ED Financial Partner users also use the Monthly Financial Report Form. The form allows the ED Financial Partner users to either accept or reject the reports submitted by GAs based on reasonability edits and operational procedures. When the ED Financial Partners accept the submission from a GA, transactions are created in the Oracle Financial Accounts Payable (AP) and General Ledger (GL) modules by the Form 2000 AP/GL Load Interface.

The ED Financial Partner users can also use this form to simply review the monthly financial reports submitted by GAs. However, the ED users are only capable of viewing data. They have no ability to modify it with the exception of the ED Comments field. In addition, only records that have a status of "Submitted," "Rejected," or "Accepted" may be viewed.

The ED Comments and GA Comments fields are primarily used to exchange information and are up to 4000 characters long. If the financial report is rejected (by pressing the Rejected button) by the ED Financial Partner user, then an ED Comment must be entered explaining the reason for rejection. For the GA user, the "ED Comments" field is read-only.

If the report is accepted (by pressing the Accept button), the same reasonability edits as stated above for GA users are performed. If a reasonability rule is violated, a confirmation message is displayed



Sample Detailed Design Document

asking to continue with acceptance. When 'OK' is selected, the Status is automatically updated to "Accepted" and the Accept and Reject buttons are disabled. In addition, the AP and GL Load program is executed in order to create AP and GL entries into the respective Oracle Application modules. At this point, the ED Comments field can no longer be updated. **Note:** an email is not sent to the GA notifying them that the Financial Report has been accepted. The GAs will be notified of acceptance when the invoice is approved for payment in Oracle Payables.

As stated above, Financial Partner users can use the Monthly Reports Form to view all financial reports that have been submitted by the various GAs. By default, the Monthly Financial Report Form queries all applicable reports submitted from the GAs. Similar to GA users, ED Financial Partner users may search on any combination of GA Code, Status, Month, Year, ED Comments, GA Comments, and MR fields and may use wildcards.



Navigation Logic

**Entry Mode – GA**

**START**

Log into Oracle Financials

From the Navigator screen, select 'GA Monthly Report'

User selects the Month and Year for which the report is going to be created.

User enters or modifies data on the form.

User commits transaction.

**END**

**Entry Mode – ED Financial Partner**

**START**

Log into Oracle Financials

From the Navigator screen, select 'GA Monthly Report'

User accepts or rejects financial report. If rejected, user must enter comments explaining why the rejection has occurred.

**END**



Table and View Usage

Table Name	Select	Insert	Update	Delete	Base Table
FFELGA_MONTHLY_REPOR	X	X	X	X	X
PO_VENDORS	X				
PO_VENDOR_SITES_ALL	X				



Sample Detailed Design Document

Table Name	Select	Insert	Update	Delete	Base Table
PO_VENDOR_CONTACTS	X				
FND_USER	X				

Zone and Field Summary

**Zone 1**

Base Table: FFELGA\_MONTHLY\_REPORTS\_V

Where/Order By: GA Code, Month, Year

Field/Column	Type	Rqd?	Default	Validation	QuickPick
GA Code	Char	Yes	**	PO_VENDOR_SITES_ALL.ATTRIBUTE2	**Defaulted for GA users and LOV for ED Financial Partner users
Status	Char	Yes	New	None – auto updated by process	
Fiscal Month	Number	Yes		No future Month/Year combination	Month Value Set
Fiscal Year	Number	Yes		No future Month/Year combination	Year Value Set
GA Name	Char	Yes		Auto populated based on the GA Code	
ED Comments	Char	No		Must be populated if Financial Report Rejected	
GA Comments	Char	No			
MR1-MR23 Amount Due, Principal Amount, Interest Amount, Other Amount	Number	No	0.00	All fields cannot be 0.00. Amount Due fields auto calculated.	



---

## Special Logic

The FSA GA Monthly Financial Report Form includes several messages that are displayed at various stages of the process. Below is a breakdown of all of the messages that a user of the Monthly Form may see:

### ***Month and Year Message***

If a GA user starts entering data for a financial report without entering a month and year, the following message will appear on the screen:

Please enter a month and year.

This message basically prevents a GA user from accidentally providing wrong information by reminding them to enter a month and year combination prior to entering the financial report data.

### ***Reasonability Edit Message***

If the GA user attempts to save a financial report, the form will check to see if data exists for the same month in the previous year in order to perform reasonability edits against. If no data is found, the GA user will see the following message:

There is no data for the same month of the previous year so the reasonability edits cannot be performed for this month.

The message above will appear every time a user tries to save the record and there indeed is no data for the same month in the previous year.

### ***Submission Confirmation Message***

Once the GA user is ready to submit the report to ED Financial Partners and they press the "Submit" button, the following confirmation message will appear on the screen:

Are you sure you wish to submit the report to FSA?

The GA user may either choose "Yes" or "No." If "Yes" is chosen, the record is saved and the status is automatically changed to "Submitted." However, if "No" is chosen, the user is simply returned to the monthly form.



### ***Report Exists Message***

If a GA user attempts to enter a duplicate report (i.e. same month/year), the following message will appear on the screen:

A document already exists for the current year and month.  
There should only be one document for each month.  
Please make corresponding changes.

When the user clicks the “OK” button, the month and year fields are blanked out and the cursor is placed in the month field.

### ***ED Comments Required Message***

When a Financial Partner user attempts to reject a report without providing a reason within the ED Comments field, the following message appears on the screen:

Please fill the comment into the ED Comments field. It is required if you decide to reject this report.

Once the “OK” button is clicked, the user may enter comments and press the “Reject” button again.

### ***AP/GL Load Confirmation Message***

When a Financial Partner user accepts a report, reasonability edits are performed again. If there was no data for the same month in the previous year, the Financial Partner user sees the same message (Reasonability Edit Message) as seen by the GA user mentioned above.

However, the Financial Partner also sees the following message:

Please note the AP/GL Load Processes are going to start.  
You can view the status by going to Help and then select View My Request.

This message basically is providing information that the AP/GL Load Interface process is about to be submitted into the concurrent manager. A detailed description of this process is contained within this document below.



**FSA GA Maintenance Screen - Form Logic**

FILE NAME: FSAMAIN.fmx

Expected Quarterly Report Submission	
<input checked="" type="radio"/> Quarterly	<input type="radio"/> Monthly

  

GA Date Maintenance	
Basic Agreement Date	<input type="text"/>
Rehabilitation Agreement Date	<input type="text"/>
Plus Agreement Date	<input type="text"/>
IRS Agreement Date	<input type="text"/>
SLS Agreement Date	<input type="text"/>
Eligibility PFY Agreement Date	<input type="text"/>
Garnishment Agreement Date	<input type="text"/>
Repurchase Agreement Date	<input type="text"/>
Supplemental Reinsurance Agreement Date	<input type="text"/>

The FSA GA Maintenance Screen allows the user to maintain GA Agreement Dates and specified GA parameters. The screen begins in data entry mode and no default values are provided. The ED Financial Partners may insert or update any Date field, but will not be able to delete any records. The user may enter or select the GA Code from a List of Values. The GA Name is automatically displayed based upon the GA code. The format for the Date fields will follow the standard format for Oracle Applications (i.e. DD-MON-YYYY). All Date fields are optional. The user may only enter one set of dates per GA. The user can retrieve the data by searching on any combination of GA Code and Date fields. The records retrieved are sorted by GA Code.

---

## Navigation Logic

### **Entry Mode**

#### **START**

Log into Oracle Financials

From the Navigator screen, select 'GA Maintenance'

User enters or modifies data

User commits transaction



**END**

### Table and View Usage

Table Name	Select	Insert	Update	Delete	Base Table
FFELGA_GAMAINTS	X	X	X		X
PO_VENDORS	X				
PO_VENDOR_SITES_ALL	X				

### Zone and Field Summary

#### Zone 1

Base Table: FFELGA\_GAMAINTS\_V

Where/Order By: GA\_CODE

Field/Column	Type	Rqd?	Default	Validation	QuickPick
GA Code	Char	Yes		GA Code Value Set	LOV
GA Name	Char	No		None – Read Only. Populated based on GA Code	
Expected Quarterly Report Submission	Char	No	Q	Radio Button. Valid Values : Monthly (M) or Quarterly (Q)	
Basic Agreement Date	Date	No			
Rehabilitation Agreement Date	Date	No			
Plus Agreement Date	Date	No			
IRS Agreement Date	Date	No			
SLS Agreement Date	Date	No			
Eligibility PFY Agreement Date	Date	No			



Sample Detailed Design Document

Field/Column	Type	Rqd?	Default	Validation	QuickPick
Garnishment Agreement Date	Date	No			
Repurchase Agreement Date	Date	No			
Supplemental Reinsurance Agreement Date	Date	No			

**FSALIB Global Package - Program Logic**

The FSALIB Global Package (FSALIB\_GLOBAL\_PKG) contains a set of generic routines that are referenced throughout the various extensions described below. An attempt was made to identify as many procedures and functions that can be used in multiple extensions and those that were identified were incorporated into the global package.

These generic procedures and functions include processes such as creating records into the AP interface and GL interface tables, writing to the concurrent request output and log files, retrieving the relevant set of books id from GL, retrieving account codes combinations for the various extensions, etc. The PL/SQL code for all of the routines within the FSALIB\_GLOBAL\_PKG is provided in Appendix B.

The advantage of having a generic global package is that most of the validation and error checking can be coded for in one central location without having to duplicate the effort in each application. In addition, the global package also provides us with the opportunity to only make minimal modifications in case changes are required.

Since the FSALIB\_GLOBAL\_PKG package only consists of routines that are referenced by other programs, there is no concurrent program registered to call any of the stored procedures directly.

**Table and View Usage**

Table Name	Select	Insert	Update	Delete
AP INVOICE LINES INTERFA		X		
AP INVOICES INTERFACE		X		
FFELGA ACCOUNT SEGMENT	X			
FFELGA ANNUAL REPORTS	X			
FFELGA GAMAIMTS	X			
FFELGA MONQUART REPOR	X			
FFELGA MONTHLY REPORTS	X			
FND APPLICATION	X			
FND CONCURRENT PROGRA	X			



Sample Detailed Design Document

Table Name	Select	Insert	Update	Delete
FND_CONCURRENT_PROGRA	X			
FND_CONCURRENT_REQUES	X			
GL_INTERFACE		X		
GL_JE_SOURCES	X			
GL_PERIOD_STATUSES_V	X			
GL_SETS_OF_BOOKS	X			
PO_VENDOR_SITES_ALL	X			

Program Logic (pseudo code)

**FUNCTION GET\_SET\_OF\_BOOKS\_ID RETURNS NUMBER**

BEGIN

    Get Set of Books ID based on “Program” flexfield being marked as ‘Yes.’

(see SQL statement 1).

END

**FUNCTION GET\_INVOICE\_ID RETURNS NUMBER**

BEGIN

    Get Next Invoice ID from Sequence (see SQL statement 2)

END

**FUNCTION GET\_INVOICE\_LINE\_ID RETURNS NUMBER**

BEGIN

    Get Next Invoice Line ID from Sequence (see SQL statement 3)

END

**FUNCTION GET\_AP\_GROUP\_ID RETURNS NUMBER**

BEGIN

    Get Next Group ID from Sequence (see SQL statement 4)

END

**FUNCTION GET\_JOURNAL\_IMPORT\_ID RETURNS NUMBER**

BEGIN

    Get Next Journal Import ID from Sequence (see SQL statement 5)

END

**FUNCTION GET\_GL\_CURRENCY\_CODE RETURNS VARCHAR2**

BEGIN

    IF Account Code (argument) starts with ‘9’ (i.e. 9000 series account) THEN  
         Currency Code = ‘STAT’

    ELSE

        Currency Code = ‘USD’

    ENDIF





Sample Detailed Design Document

into GL\_INTERFACE  
END

**PROCEDURE WRITE\_LOG**

BEGIN  
Using FND\_FILE (Standard Application Package), write to the Concurrent Request Log File  
END

**PROCEDURE WRITE\_OUT**

BEGIN  
Using FND\_FILE (Standard Application Package), write to the Concurrent Request Output File  
END

**PROCEDURE COMMIT\_TRANSACTION**

BEGIN  
IF Procedure is not called from Trigger THEN  
COMIT Record  
ENDIF  
END

---

SQL Statements

**1 - Data Selection**

```
SELECT      MIN (SET_OF_BOOKS_ID)
INTO        v_set_of_books_id
FROM        GL_SETS_OF_BOOKS
WHERE       ATTRIBUTE1 = 'YES'; (program flexfield)
```

**2 - Data Selection**

```
SELECT      AP_INVOICES_INTERFACE_S.NEXTVAL
INTO        v_invoice_id
FROM        DUAL;
```

**3 - Data Selection**

```
SELECT      AP_INVOICE_LINES_INTERFACE_S.NEXTVAL
INTO        v_invoice_line_id
FROM        DUAL;
```

**4 - Data Selection**

```
SELECT      AP_INTERFACE_GROUPS_S.NEXTVAL
INTO        v_group_id
FROM        DUAL;
```



Sample Detailed Design Document

**5 - Data Selection**

```
SELECT      GL_JOURNAL_IMPORT_S.NEXTVAL
INTO        v_journal_import_id
FROM        DUAL;
```

**6 - Data Selection**

```
SELECT      COUNT(1)
INTO        v_count
FROM        FND_CONCURRENT_REQUESTS FCR
           ,FND_CONCURRENT_PROGRAMS FCP
WHERE       FCR.CONCURRENT_PROGRAM_ID =
           FCP.CONCURRENT_PROGRAM_ID
AND        UPPER(FCP.CONCURRENT_PROGRAM_NAME) =
           UPPER(p_short_name)
AND        PHASE_CODE = 'P' /* pending */
AND        FCR.REQUESTED_BY = v_user_id
AND        NVL(FCR.ARGUMENT1,'X') = NVL(p_argument1,'X')
AND        NVL(FCR.ARGUMENT2,'X') = NVL(p_argument2,'X')
AND        NVL(FCR.ARGUMENT3,'X') = NVL(p_argument3,'X')
AND        NVL(FCR.ARGUMENT4,'X') = NVL(p_argument4,'X')
AND        NVL(FCR.ARGUMENT5,'X') = NVL(p_argument5,'X')
AND        NVL(FCR.ARGUMENT6,'X') = NVL(p_argument6,'X')
AND        NVL(FCR.ARGUMENT7,'X') = NVL(p_argument7,'X')
AND        NVL(FCR.ARGUMENT8,'X') = NVL(p_argument8,'X')
AND        NVL(FCR.ARGUMENT9,'X') = NVL(p_argument9,'X')
AND        NVL(FCR.ARGUMENT10,'X') = NVL(p_argument10,'X');
```

**7 - Data Selection**

```
SELECT      USER_CONCURRENT_PROGRAM_NAME
INTO        v_descr
FROM        FND_CONCURRENT_PROGRAMS_VL FCP
           ,FND_APPLICATION FA
WHERE       FA.APPLICATION_ID = FCP.APPLICATION_ID
AND        CONCURRENT_PROGRAM_NAME = p_short_name
AND        APPLICATION_SHORT_NAME = p_app_name;
```

**8 - Data Selection**

```
SELECT      VENDOR_ID
           ,VENDOR_SITE_ID
           ,TERMS_ID
           ,ATTRIBUTE1 INSTITUTION
INTO        p_vendor_id
           ,p_vendor_site_id
           ,p_terms_id
           ,p_institution
FROM        PO_VENDOR_SITES_ALL
WHERE       ATTRIBUTE2 = p_ga_code
AND        INACTIVE_DATE IS NULL
AND        PAY_SITE_FLAG = 'Y';
```



**Validation Logic**

**Error Conditions:**

The following errors are being handled within the FSALIB\_GLOBAL\_PKG package:

PROCEDURE / FUNCTION	ERROR CONDITION	PURPOSE
GET_VENDOR_INFO	NO DATA FOUND	This error is raised if vendor information is not found based on the GA Code that is passed into the procedure. A message is written to the log file
	OTHERS	This is a catchall exception. In case this exception is raised, the program will write the error message to the log file
FILE_EXISTS	OTHERS	This is a catchall exception. In case this exception is raised, the function will return a value of FALSE indicating that the data file name (passed in as a parameter) was not found.
SUBMIT_REQUEST	OTHERS	This is a catchall exception. In case this exception is raised, the function returns a value of 0 (zero) indicating that nothing was submitted to the concurrent manager.

**FSA FFEL GA Form 2000 AP/GL Load Interface - Concurrent Program Logic**

The FSA FFEL GA Form 2000 AP/GL Load Interface is responsible for creating invoices and journal entries into Oracle applications based on the financial information that is submitted by each GA. The financial information, once accepted by Financial Partners, is imported into the Oracle Applications Open Interface tables for AP and GL.

Once the Open Interface tables have been populated, the Payables Open Interface Import and the Journal Entry Import concurrent programs are executed in order to create non-approved invoices within AP and journal entries within GL. These two concurrent programs are provided in Oracle Applications and are not custom extensions.

The triggering event for submitting the AP/GL Load Interface program to the concurrent manager is the “Accept” button on the Monthly, Monthly/Quarterly, and Annual Forms. Once a report is accepted, the LOAD procedure within the FFELGA\_AP\_GL\_LOAD\_PKG package is executed in order to submit the concurrent request. Below is the list of arguments that are passed to the concurrent request program.



## Calling Arguments

The FSA FFEL GA Form 2000 AP/GL Load Interface program can be called from the Standard Report Submission form. However, the program is triggered to be called from the LOAD procedure embedded within the Form 2000 Monthly, Monthly/Quarterly, and Annual forms as described above. Below is the list of arguments that are passed:

Argument	Prompt	Value Set	Default Value
P_GA_CODE	GA Code	EDFSA_GA_CODE	None
P_MONTH	Month	FSA_FFEL_Month_Code_Value_Set	None
P_YEAR	Year	FSA_FFEL_Year_Code_Value_Set	None
P_REPORT_TYPE	Report Type	FSA_FFEL_Report_Code_Value_Set	None

Example of call from a form:

```
FFELGA_AP_GL_LOAD_PKG.LOAD(:GA_CODE, :F_MONTH, :F_YEAR, 'M');
```

## Log Output

.....1.....2.....3.....4.....5.....6.....7.....8

```
+-----+
FSA FFEL Extensions: Version : UNKNOWN - Development

Copyright (c) Oracle Corporation 1979, 1991. All rights reserved.

FSAAPGLLD module: Form 2000 AP/GL Load Interface
+-----+

Current system time is 20-JUL-2000 11:34:07

+-----+

+-----+
Start of log messages from Plsql program
+-----+
Starting AP GL Load
-----
. Submitting request for APXIIMPT
. Argument1 = Form 2000
. Argument2 = 497
```



Sample Detailed Design Document

```
. Argument3 = FFEL708M031999
. Argument7 = Y
. Request id = 1917
. Commit transaction
. Submitting request for GLLEZL
. Argument1 = 40
. Argument2 = 2
. Argument3 = N
. Argument6 = N
. Argument7 = N
. Request id = 1918
. Commit transaction
Import Complete
```

```
+-----+
End of log messages from PLSql program
+-----+
```

```
+-----+
```

Executing request completion options...

Printing output file.

```
Request ID : 1916
Number of copies : 0
Printer : TEST
```

Finished executing request completion options.

```
+-----+
Concurrent request completed successfully
Current system time is 20-JUL-2000 11:34:08
+-----+
```

## Table and View Usage

Table Name	Select	Insert	Update	Delete
AP INVOICE LINES INTERFA		X		
AP INVOICES INTERFACE		X		
FFELGA ACCOUNT SEGMENT	X			



Sample Detailed Design Document

Table Name	Select	Insert	Update	Delete
FFELGA ANNUAL REPORTS	X			
FFELGA GAMAINTS	X			
FFELGA MONQUART REPOR	X			
FFELGA MONTHLY REPORTS	X			
FND APPLICATION	X			
FND CONCURRENT PROGRA	X			
FND CONCURRENT PROGRA	X			
FND CONCURRENT REQUES	X			
GL INTERFACE		X		
GL JE SOURCES	X			
GL PERIOD STATUSES V	X			
GL SETS OF BOOKS	X			
PO VENDOR SITES ALL	X			
PO VENDORS	X			

**Program Logic (pseudo code)**

```

BEGIN
  Select rows (see SQL statement 1)
  IF Report Type = 'Monthly' THEN (see arguments)
    GET Invoice ID
    GET AP Group ID
    GET Report ID
  ENDIF
  Get Vendor Information
  Get Set of Books ID
  Get GL Category and Source

  FOR EACH row returned
    GET Line Item Amount
    Get Account Segments
    Get Transaction Code
    IF Item Destination = 'AP' THEN
      Add Line Item Amount to Total Invoice Amount
      Assign Line Number
      Assign Line Description
      Assign Accounting Distribution Code
      Get Invoice Line ID
      Insert into AP_INVOICE_LINES_INTERFACE
    ELSIF Item Destination in ('GL', 'GLREV') THEN
      Determine Reversal Flag (Yes if GLREV, No if GL)
      IF Reversal Flag = 'Yes' THEN
        Get Reversal Period
      ENDIF
      Assign Journal Entry Description
      Assign Journal Entry Reference
      Assign Journal Entry Line Description
      Determine Currency Code (STAT if Statistical Entry,

```



Sample Detailed Design Document

```

                                Otherwise USD)
                                Insert into GL_INTERFACE
                                ENDIF
                                ENDFOR
                                Check if Lines were created
                                IF Line Count > 0 THEN
                                    Assign Invoice Number
                                    Assign AP Batch Name
                                    Assign Invoice Description
                                    Insert into AP Invoices Interface
                                ENDIF
                                Assign Journal Import ID
                                Submit AP Request
                                Submit GL Request
                                END

```

SQL Statements

1 - Data Selection

```

SELECT      ASEG_MR_CODE
            ,ASEG_DESTINATION_CODE
            ,ASEG_TRANSACTION_CODE
FROM        FFELGA_ACCOUNT_SEGMENTS
WHERE       SYSDATE BETWEEN ASEG_START_DATETIME AND
            NVL(ASEG_END_DATETIME, '31-DEC-2099')
AND        ASEG_SUBSYSTEM_CODE = p_subsystem_code
ORDER BY   TO_NUMBER(SUBSTR(ASEG_MR_CODE,3,
            INSTR(ASEG_MR_CODE,'_',1,1)-3)), ASEG_MR_CODE;

```

Validation Logic

Error Conditions:

The following errors are being handled within the FFELGA\_AP\_GL\_LOAD\_PKG package:

PROCEDURE / FUNCTION	ERROR CONDITION	PURPOSE
GET_GL_SOURCE	NO DATA FOUND	This error is raised if GL Source information is not found based on the Source Name passed in as a parameter.
GET_GL_REVERSAL_PERIOD	NO DATA FOUND (inner block)	This exception is raised if no entry in the GA_MAINS table is found for the GA Code passed in as a parameter. A message is written to the log file.
	NO DATA FOUND (outer block)	This exception is raised if there is no reversal period found based on the calculated dates



Sample Detailed Design Document

		within this function. A message is written to the log file.
GET_REPORT_ID	NO DATA FOUND	This exception is raised if there is no report found based on the GA Code, Month, and Year parameters passed in.
LOAD_AP_GL_DATA	INVALID REPORT TYPE EX	This exception is raised if the report type parameter is something other than M (Monthly), Q (Quarterly), or A (Annual).

---

### Incompatibility

The AP/GL Load Interface program has been set up to be incompatible with itself. By doing so, it is guaranteed that two instances of this concurrent program will never run simultaneously thereby avoiding possible problems with multiple instances of the program working on the same set of data.

---

### Other Considerations

#### **Restart Strategy**

The AP/GL Load Interface program not only is a concurrent request itself, it also submits two additional standard concurrent requests provided by Oracle Applications. If the error occurs within the AP/GL Load Interface program, then the user needs to make corrections and re-submit the request. However, if the error occurs in the Payables Open Interface Invoice Import program, the user needs to analyze the error(s), make corrections, and then re-submit the request for Payables Open Interface Invoice Import only. The same holds true for the Journal Import request.

#### **Crash Recovery**

The AP/GL Load Interface program has been marked to restart itself incase of a database crash. If a database crash were to occur while in the middle of execution, the AP/GL Load program will re-start itself.

#### **FSA FFEL GA Form 2000 Monthly Data File Load - Concurrent Program Logic**

The FSA FFEL GA Form 2000 Monthly Data File Load is responsible for loading Monthly financial information that was submitted by a GA in a data file. GAs can submit monthly financial reports via a data file instead of typing it in using the Form 2000 Monthly form.

Once a data file is provided, the Monthly Data File Load concurrent program will be triggered via a Unix shell script that will provide the file location and the name of the file submitted.

The Monthly Data File Load program, once triggered, first populates a temporary table (FFELGA\_MON\_TEMP) with the file data, then validates the data by applying the rules stated below:

There must be exactly one header and one trailer,



Sample Detailed Design Document

The record count stated in the header must equal to the total number of detail records within the file,

The dollar total stated in the trailer for the principal amount, interest amount, and other amount must equal to the sum of all detail record amounts.

If the file is a new submission (submission code in the header), then verify that there is no existing financial report for the same period.

If the file is a re-submit, then verify that a prior submission exists.

If any one validation fails, the data file is rejected. On the other hand, if the data file passes all of the validation rules, the data is populated into the FFELGA\_MONTHLY\_REPORTS table. The file data can now be viewed using the Form 2000 Monthly Reports form.

**Calling Arguments**

The FSA FFEL GA Form 2000 Monthly Data File Load program can be called from the Standard Report Submission form. However, the program will be triggered from a Unix shell script in order to automate file processing. Below is the list of arguments that are passed:

Argument	Prompt	Value Set	Default Value
P_FILE_LOC	File Location Directory	50 Chars	None
P_FILENAME	File Name	15 Chars	None

**Log Output**

```

.....1.....2.....3.....4.....5.....6.....7.....8
-----
+-----+
FSA FFEL GA Extensions: Version : UNKNOWN - Development

Copyright (c) Oracle Corporation 1979, 1991. All rights reserved.

FSAMONLD module: FSA FFEL GA Form 2000 Monthly Data File Load
+-----+

Current system time is 24-JUL-2000 15:52:25

+-----+

+-----+
Start of log messages from Plsql program

```



Sample Detailed Design Document

```

+-----+
. Extracting data from file to temp table...
Opening the data file....
Closing File...
. Extraction Complete...
. Validating data...
Trailer and header count verified...
. Validation Complete...
. Loading into Monthly table...
. Loading Complete...
. Deleting from temp table...
. Commit transaction
Monthly data file load Complete
+-----+
End of log messages from PlSql program
+-----+

```

Executing request completion options...

```

Printing output file.
      Request ID : 8749
      Number of copies : 0
      Printer : TEST

```

Finished executing request completion options.

```

+-----+
Concurrent request completed successfully
Current system time is 24-JUL-2000 15:52:26
+-----+

```

**Table and View Usage**

Table Name	Select	Insert	Update	Delete
FFELGA MONTHLY REPORTS	X	X	X	
FFELGA MON TEMP	X	X		X
FND APPLICATION	X			
FND CONCURRENT PROGRA	X			



Sample Detailed Design Document

Table Name	Select	Insert	Update	Delete
FND_CONCURRENT_PROGRA	X			
FND_CONCURRENT_REQUES	X			

**Program Logic (pseudo code)**

```

BEGIN
  Open Data File
  Extract information from data file into temp table
  Validate Data
    Verify Header and Trailer count
    Verify Record Total as specified in Header
    Verify Principal, Interest, Other Amount Totals
    IF New Submission, verify that no record currently exists
  IF Data is Valid THEN
    Select rows          (see SQL statement 1)
    FOR Each row returned
      Assign column values for Monthly Report
    ENDFOR
    Calculate Totals
    IF New Submission THEN
      Insert into FFELGA_MONTHLY_REPORTS
    ELSE
      Update FFELGA_MONTHLY_REPORTS
    ENDIF
  ELSE Reject File
  ENDIF
  Delete data from Temporary Table (FFELGA_MON_TEMP)
END
  
```

**SQL Statements**

**1 - Data Selection**

```

SELECT      SUBSTR(FILE_DATA, 1,3) GA_CODE
            ,SUBSTR(FILE_DATA, 4,10) ITEM_NUMBER
            ,TO_NUMBER(LTRIM(SUBSTR(FILE_DATA, 14,15),' 0'))
              PRINCIPAL_AMOUNT
            ,TO_NUMBER(LTRIM(SUBSTR(FILE_DATA, 29,15),' 0'))
              INTEREST_AMOUNT
            ,TO_NUMBER(LTRIM(SUBSTR(FILE_DATA, 44,15),' 0'))
              OTHER_AMOUNT
FROM        FFELGA_MON_TEMP
WHERE      TYPE = 'D';
  
```



Validation Logic

**Error Conditions:**

The following errors are being handled within the FFELGA\_MONTHLY\_FILE\_LOAD\_PKG package:

<b>PROCEDURE / FUNCTION</b>	<b>EXCEPTION / ERROR CONDITION</b>	<b>PURPOSE</b>
INSERT_INTO_FFELGA_MON_TEMP	NO DATA FOUND	This error is raised if no more data is found within the data file. A message is written to the log file.
	VALUE ERROR	This exception is raised if an attempt is made to insert inconsistent values into the temp table (data type inconsistencies). A message is written to the log file.
	OTHERS	This is a catchall exception. In case of any other errors, the error message is written to the log file.
VALIDATE_FFELGA_MON_TEMP	NO DATA FOUND (header check)	This exception is raised if there is no header record found within the temp table. A message is written to the log file and validation of the data file fails.
	OTHERS	This is a catchall exception. In case of any other errors, the error message is written to the log file and validation of the data file fails.
	HEADER RECORD TOTAL VALUE <> DETAIL RECORD COUNT	This error is raised if the record total specified within the header is not equal to the number of total detail records. A message is written to the log file and validation of the data file fails.
	SUM OF DETAIL RECORD AMOUNT TOTALS <> TRAILER TOTAL AMOUNTS	This error is raised if the sum of the detail record amounts (principal, interest, other) does not equal the total amounts stated within the trailer. A message is written to the log file and validation of the data file fails.
	SUBMISSION CODE = 'NEW' AND PREVIOUS REPORT EXISTS	This error is raised if a file is received with a submission code of 'N' (new) and there is a previous record for the same GA/month/year combination within the system. A message is written to the log file and validation of the data file fails.
	SUBMISSION CODE =	This error is raised if a file is



Sample Detailed Design Document

	"RESUBMIT" and NO PREVIOUS RECORD EXISTS	received with a submission code of 'R' (Resubmit) and there is no previous record for the same GA/month/year combination within the system. A message is written to the log file and validation of the data file fails.
GET_USER_ID	NO DATA FOUND (user name)	This error is raised if no header record is found with a user name. A message is written to the log file and processing is terminated.
	NO DATA FOUND (user id)	This error is raised if a user name was found but did not match any of the user names defined within the FSA FMS system. A message is written to the log file and processing is terminated.
INSERT_INTO_MONTHLY_REPORTS	NO DATA FOUND (extracting GA Code, Month, Year from Header)	This error is raised if there is no header record within the FFELGA_MON_TEMP table. A message is written to the log file and processing is terminated.
PROCESS_MONTHLY_FILE	VALIDATION CHECK	This error is raised if validation of the data file fails (see above errors). If validation fails, then messages are written to the log file and processing is terminated.
	DONE PROCESSING	This error is raised from other error handling exceptions throughout the program in case the data file load process needs to be terminated. Messages are written to the log file and records from the temporary table are deleted

---

## Incompatibility

The Monthly Data File Load program has been set up to be incompatible with itself. By doing so, it is guaranteed that two instances of this concurrent program will never run simultaneously thereby avoiding possible problems with multiple instances of the program working on the same set of data.

---

## Other Considerations

### **Restart Strategy**

The Monthly Data File Load program needs to be re-submitted manually after correcting any errors.



### ***Crash Recovery***

The Monthly Data File Load program has been marked to restart itself in case of a database crash. If a database crash were to occur while in the middle of execution, the Monthly Data File Load program will re-start itself.