HHS Unified Financial Management System

UFMS PMIS Interface Technical Design

Global

Prepared in collaboration with:

U.S. Department of Health and Human Services

Office of Program Management and Systems Policy
200 Independence Avenue, SW
Washington, DC 20201

Version 11.0
<01/14/2008>
PMOnline AFT# 3284

Prepared and Submitted By:

1676 International Drive
McLean, VA 22102

For Official Use Only

Document Revision History

Introduction and Functional Design Sections

Date	Version	Author	Level of Review	Summary of Comments
N/A				

Data Mapping Section

Date	Version	Author	Level of Review	Summary of Comments
N/A				

Technical Design Section

Date	Version	Author	Level of Review	Summary of Change
01/08/2007	1.0	Mahendra Dalvi	Document Creation	Initial version
02/05/2007	2.0	Scott Granieri	QA Review	Performed QA Review of **v0.1** of this design document against the UFMS Quality Assurance Common Document Review Criteria Worksheet, v1.4 (PMOnline AFT# 2615). Document Review Worksheet was submitted back to author and contains all QA updates made to **<Updated Document Version 0.2**.
02/13/2007	3.0	Mahendra Dalvi	Document Modification	Modified the document based on Betty's feedback and based on Technical Walkthrough. 1. Taken care of abnormal shutdown (see Figure 1). 2. Checking record count and total amount from trailer record with individual transaction record count and amount respectively for GL Interface (see Figure 2) 3. Added Data Mapping Section(see 3.1) 4. Assumptions changed to define batch and batch processing (see 4.3.2 – assumption # 4) 5. Removed PROCESS_DATE parameter for all the extracts (see section 4.6) 6. Change table names and Extract File names as per HHS naming conventions (see database design) 7. Describe Acknowledgement process in detail (See 4.6.4) 8. Added GPRA Code column and Logic for all three extracts.

Date	Version	Author	Level of Review	Summary of Change
02/20/2007	4.0	Mahendra Dalvi	Document Modification	Document modified based on the technical discussion dated 02/15/2007 1. Added Error Codes (4.8.1) under Error Handling Section 4.8. 2. Added process statuses 'S' for SUSPEND and 'U' for UFMS error records. 3. Explain UFMS (U) Error and Suspend (S) status under Section 4.4.4 (GL Program Logic)
03/01/2007	5.0	Mahendra Dalvi	Document Modifications	Changes added based on the discussion with Solomon. 1. Added column FUND_CANCEL_DATE for all three extracts. 2. Columns UNIT_PRICE and QUANTITY modified from not null to null for invoice extract 3. Removed column FDA_CONTRACT_NUMBER from PO Extract. 4. Added columns FDA_AGENCY_TAG_NUMBER and FDA_TAG_NUMBER to PO and Invoice Extract. 5. Change table structure and extract structure to accommodate fund cancel date. 6. Header record has been removed from GL Inbound File format. Columns Organization code and datetime from header record has been moved to trailer record.
04/05/2007	6.0	Mahendra Dalvi	Document Modifications	Changes made based on the discussion with Solomon 1. Removed GPRA_CODE column and added CONCATENATED_SEGMENTS column in all extract Programs. 2. Removed column FILE_NAME from the acknowledgement File Format. 3. Position of the RECORD_STATUS column has been changed and placed before ERROR_CODE column in the acknowledgement file format. 4. Added column REFERENCE4 in GL File format.
04/19/2007	7.0	Mahendra Dalvi	Document Modifications	Added comments to all the extract columns for Purchase Order, Receipts and Invoice. Please see section 4.5.1, 4.5.2, 4.5.3 Added approvals in the Appendix
05/03/2007	8.0	Mahendra Dalvi	Document Modifications	Added column TASK_ID to the section 4.4.2 i.e. General Ledger File Format
06/15/2007	9.0	Mahendra Dalvi	Document modifications	Increased the column length from VARCHAR2 (150) to VARCHAR2 (240) for comments and description columns in all extract staging table. **Please see new change in red color text.**
07/15/2007	10.0	Mahendra Dalvi	Document modifications	Added condition for profile option HHS_PMIS_START_DATE which looks for PMIS start date (Go Live Date for each opdiv). No change required for Sunflower. **Please see new changes in Blue color text.**
01/14/2008	11.0	Mahendra Dalvi	Document modifications	Added "RECEIPT_REQUIRED_FLAG" column at the bottom of the file layout for Purchase Order extract. Change mentioned under section 4.5.1 Table 14 in violet color. Change Request # 44383

Table of Contents

List of Figures

List of Tables

1.0. INTRODUCTION

The Property Management Information System (PMIS), also referred to as Sunflower, will be utilized by the Department of Health and Human Services (HHS) to provide a single standardized financial management system for personal property. The Sunflower system will provide tracking of accountable personal property, financial management of capital personal property throughout an asset's useful life, asset reporting, periodic depreciation calculations, asset Addition and removal. The following HHS systems will integrate with the Sunflower Asset Management system: UFMS (United Financial Management System).

This document addresses the interface between the Sunflower system and UFMS. Purchase orders, receiving and invoicing will be created in or interfaced to UFMS from other procurement systems.

Sunflower shall receive purchase order, receiving information and invoice information for personal property within the 31000 series of object class. The purchase order shall be sent as a skeletal record to Sunflower except for those credit card transactions for which no postings are made to Purchasing while the receiving information shall be sent once receiving is performed in UFMS. The PMIS interface shall extract the purchase order and receiving information from UFMS Purchasing and the invoice payment information from UFMS Payables. The interface is a two-way interface from Sunflower to UFMS. Three separate extract flat files will be sent from UFMS to Sunflower for the purchase orders, receiving and invoice payment; asset journal entries including depreciation shall be sent back to UFMS General Ledger from Sunflower.

All HHS Operating Divisions (OPDIVs) participated in the development of the global To-Be baseline business processes and the UFMS PMIS interface will be developed taking into account the business rules defined. Food and Drug Administration (FDA), Center for Disease Control and Prevention (CDC), Program Support Center (PSC) and Indian Health Service (IHS) will be deployed as part of the UFMS Release 4.0 scheduled for go-live in October 2007.

In this document, we identify business rules that drive how HHS property related activities would be interfaced, processed, stored and reported on in Sunflower. Specifically, this document addresses:

1) The design and functionality of the UFMS PMIS interface.
2) The technical requirements to process outbound (Purchase Order, Receiving and Invoice) extract of property data to Sunflower.
3) The technical requirements to process inbound interface of asset journal entries for capitalized property to UFMS.
4) The technical requirements for the UFMS PMIS interface.

Note: The operation of the Sunflower system will not be covered in this document. Only aspects that relate to the operation of the UFMS PMIS interface will be discussed.

Table 1: Derived Requirements

Rule Number	Derived Requirement	Business Scenario	Requirement ID (from Requisite Pro)
	The obligation and receipt information should be interfaced to the Sunflower system	4.1.1	
	The PMIS interface shall only extract personal property purchased using the 31XXX series of object class. This includes non-cap items procured in the 319XX object class series.	4.3.2	

Rule Number	Derived Requirement	Business Scenario	Requirement ID (from Requisite Pro)
	Shipping and Installation costs that are to be capitalized shall be entered on the requisition/purchase order using a capitalized object class in the 22403 and/or 25240.	4.3.2	
	The depreciation expense shall be generated in Sunflower using a current month convention and interfaced to UFMS.	4.4	

1.1. Major Features

1.2. Definitions

N/A

1.3. Assumptions

N/A

1.3.1. Business Assumptions

N/A

1.3.2. Design Assumptions

N/A

2.0. FUNCTIONAL DESIGN

N/A. Functional Design components are documented in a separate Functional Design document.

2.1. System Flows

N/A

2.2. Business Processes and Scenarios

N/A

2.2.1. Process

N/A

2.3. Accounting Impacts

N/A

2.3.1. Accounting for Transactions Type 1

Table 2: Accounting Transactions

Event	SGL Account Description	Debit	Credit	Fund	Budget FY	Budget Activity Plan	Org	Object Class	SGL Account	Admin Control	Project	Future 1	Future 2

2.4. User Procedures

N/A

2.5. Business Logic

N/A

2.6. Performance Considerations

N/A

2.7. Error Handling

N/A

Table 3: Error Handling

Action	Response

3.0. DATA MAPPING

3.1. Data Mapping

Table 4: Data Mapping

Asset journal entries including depreciation charges shall be sent to UFMS on a monthly basis. The PMIS inbound file shall have a source of 'SUNFLOWER' and the following journal categories: Addition, Adjustment, Reclass, Transfer, Retirement, and Depreciation. The asset journal entries shall be created in the UFMS General Ledger via the general ledger interface import process

	Source System (Sunflower)	Oracle System (UFMS)				
Ref #	**Field Name**	**Oracle Table**	**Column**	**Column Datatype**	**Required**	**Comment**
1	Set of Books Name	GL Interface	SET_OF_BOOKS_ID	NUMBER (15)	Y	Sunflower shall send the HHS Agency code (CDC, FDA, PSC, IHS)
2	Currency Code	GL Interface	CURRENCY_CODE	VARCHAR2 (15)	Y	USD
3	Source Name	GL Interface	USER_JE_SOURCE_NAME	VARCHAR2 (25)	Y	Value = 'SUNFLOWER'
4	Category Name	GL Interface	USER_JE_CATEGORY_NAME	VARCHAR2 (25)	Y	ADDITION, ADJUSTMENT, TRANSFER, RETIREMENT, RECLASS, DEPRECIATION
5	CAN	GL Interface		VARCHAR2 (25)	Y	PMIS Interface shall use the accounting values from Sunflower to derive the UFMS BACS.
6	Fiscal Year	GL Interface	SEGMENT2	VARCHAR2 (25)	Y	DERIVE UFMS BACS
7	Object Class	GL Interface	SEGMENT5	VARCHAR2 (25)	Y	DERIVE UFMS BACS
8	USSGL Account	GL Interface	SEGMENT6	VARCHAR2 (25)	Y	DERIVE UFMS BACS
9	Debit Amount	GL Interface	ENTERED_DR	NUMBER	Y	
10	Credit Amount	GL Interface	ENTERED_CR	NUMBER	Y	
11	Period Name	GL Interface	PERIOD_NAME	VARCHAR2 (15)	Y	Format must be 'MON-YY-FY-YY' for UFMS.

	Source System (Sunflower)	Oracle System (UFMS)				
Ref #	Field Name	Oracle Table	Column	Column Datatype	Required	Comment
12	Transaction Date	GL Interface	TRANSACTION_DATE	DATE	Y	
13	Batch Name	GL Interface	REFERENCE1	VARCHAR2 (100)	Y	Batch Name (JE Source + JE Category + Period Name)
14	Batch Description	GL Interface	REFERENCE2	VARCHAR2 (240)	Y	Batch Description
15	Journal Name	GL Interface	REFERENCE4	VARCHAR2 (100)	Y	
16	Journal Description	GL Interface	REFERENCE5	VARCHAR2 (100)	Y	
17	Line Number	GL Interface	JE_LINE_NUM	NUMBER (15)	Y	
18	Line Description	GL Interface	REFERENCE10	VARCHAR2 (100)	N	
19	HHS Eliminations Code	GL Interface	ATTRIBUTE1	VARCHAR2 (150)	N	
20	FACTS Trading Partner Code	GL Interface	ATTRIBUTE4	VARCHAR2 (150)	N	
21	Transaction Date	GL Interface	ATTRIBUTE10	VARCHAR2 (150)	Y	Same mapping for Ref # 12. Transaction Date shall be populated in the DFF as well.
22	GPRA Code	N/A	N/A	N/A		
23	Accounting Point	N/A	N/A	N/A		
24	Task ID	N/A	N/A	N/A		

4.0. TECHNICAL DESIGN

4.1. Technical Overview

This document describes the technical design of the UFMS PMIS interfaces to/from Sunflower. It describes the technical components to be developed to meet these requirements. The technical design components are divided into the following sections of this document.

1. Technical Flow for the PMIS outbound interfaces
   - Describes the flow of data among different objects in Oracle UFMS.
2. Technical Flow for the PMIS inbound interface
   - Describes the steps in the technical flow diagram
3. Extract Steps
   - Describes the custom database tables and dependencies for the PMIS interfaces.
4. Assumptions
   - Describes the technical assumptions that the PMIS interfaces are based on and the scope of the current deliverable
5. Application programs definition that will be defined to process the PMIS inbound and outbound interface. These include
   - Obligation Extract Program that creates the obligation data file and will be sent to sunflower.
   - Receipt Extract Program that creates the receiving data file and will be sent to sunflower.
   - Invoice Extract Program that creates the invoice data file for 2-way match obligations and will be sent to sunflower.
   - Custom General Ledger Interface Program that validates the journal transactions received from Sunflower.
6. Database objects created for PMIS Interfaces
   - Tables
   - Indexes
7. File Naming Convention and PMIS Interfaces

4.1.1 Technical Flow Diagram and Extract Steps

Figure 1: PMIS System Overview

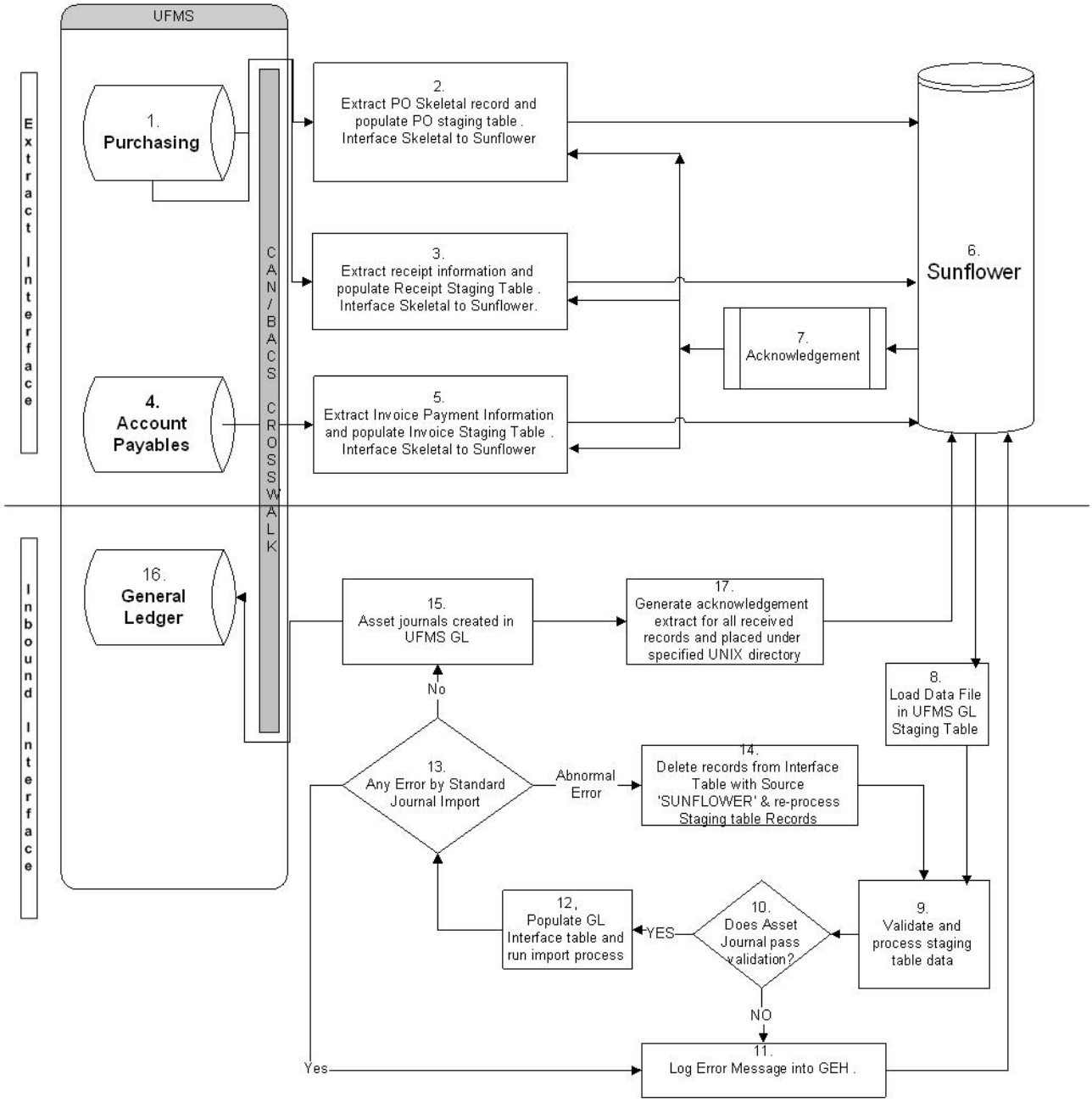


Table 5: PMIS Overview Flow Diagram Process (Figure 1)

Process Flow No	Process Step Description
1.	This interface process starts with Purchase Order Module. Obligations (Purchase Order) will be resides in UFMS Purchasing.
2.	All PO extract data columns will be selected from UFMS purchasing and will be populated into PO Staging Table. Generate PO skeletal from PMIS PO extract (Staging) table and will be placed in specified UNIX folder. The same skeletal will be sent to Sunflower.
3.	All Receipt extract data columns will be selected from UFMS purchasing and will be populated into Receipts Staging Table. Generate Receipts skeletal from PMIS Receipts extract (Staging) table and will be placed in specified UNIX folder. The same skeletal will be sent to Sunflower.
4.	Invoice interface process starts with Payables Module. Invoice information resides in UFMS Payables module.
5.	All Invoice (2-way) extract data columns will be selected from UFMS Payables and will be populated into Invoices Staging Table. Generate Invoices skeletal from PMIS Invoices extract (Staging) table and will be placed in specified UNIX folder. The same skeletal will be sent to Sunflower.
6.	Sunflower will receive and process PO/Receipt/Invoice extract sent by UFMS.
7.	Once processing complete at Sunflower, acknowledgement will be sent back to UFMS for every record available in the extract.
8.	Sunflower will create an Asset Journal skeletal and will be sent to UFMS.
9.	PMIS custom PL/SQL procedure will load and validate the staging table journal data based on the validations.
10.	If any UFMS error encounter, go to step 11. If any data error encounter, Only error records in the journal will be updated with status as ERROR and successful validated records in the same journal batch will be updated with status as ‘CHILD ERROR’. If no errors found in the journal batch, then process status will be updated with status as PROCESSED.
11.	If any UFMS configuration error encounter, whole journal batch will be rejected and only error records will be logged in GEH with proper error codes (Refer Error Handling for error codes). Only error records in the journal will be updated with status as UFMS ERROR and successful validated records in the same journal batch will be updated with status as ‘CHILD ERROR’. If no errors found in the journal batch, then process status will be updated with status as PROCESSED.
12.	All PROCESSED records will be loaded from PMIS GL Staging table to Oracle Standard Interface table. Once records get loaded, run oracle standard ‘Journal Import’ Program.
13.	If any abnormal error occurs go to step 14. If no abnormal errors then, check if there are any validation errors by standard GL Import program, If yes, then corresponding records in the staging table will be updated, as ‘UFMS ERROR’ and error records will be logged using GEH.
14.	Delete all records from standard interface table that has source ‘SUNFLOWER’ and period_name is current month period. Update record_status with status as NEW in staging

Process Flow No	Process Step Description
	table for the record that has period_name current month period. Restart the program from custom validation process onwards.
15.	If no error found, then asset journal records will be inserted to UFMS oracle standard General Ledger tables.
16.	Batches /Journals will be created in Oracle General Ledger of UFMS.
17.	Acknowledgement for all PROCESSED, ERROR, CHILD ERROR, UFMS ERROR records will be placed in specified UNIX directory.

Figure 2: General Ledger Inbound Interface Flow Chart (From Sunflower to UFMS)

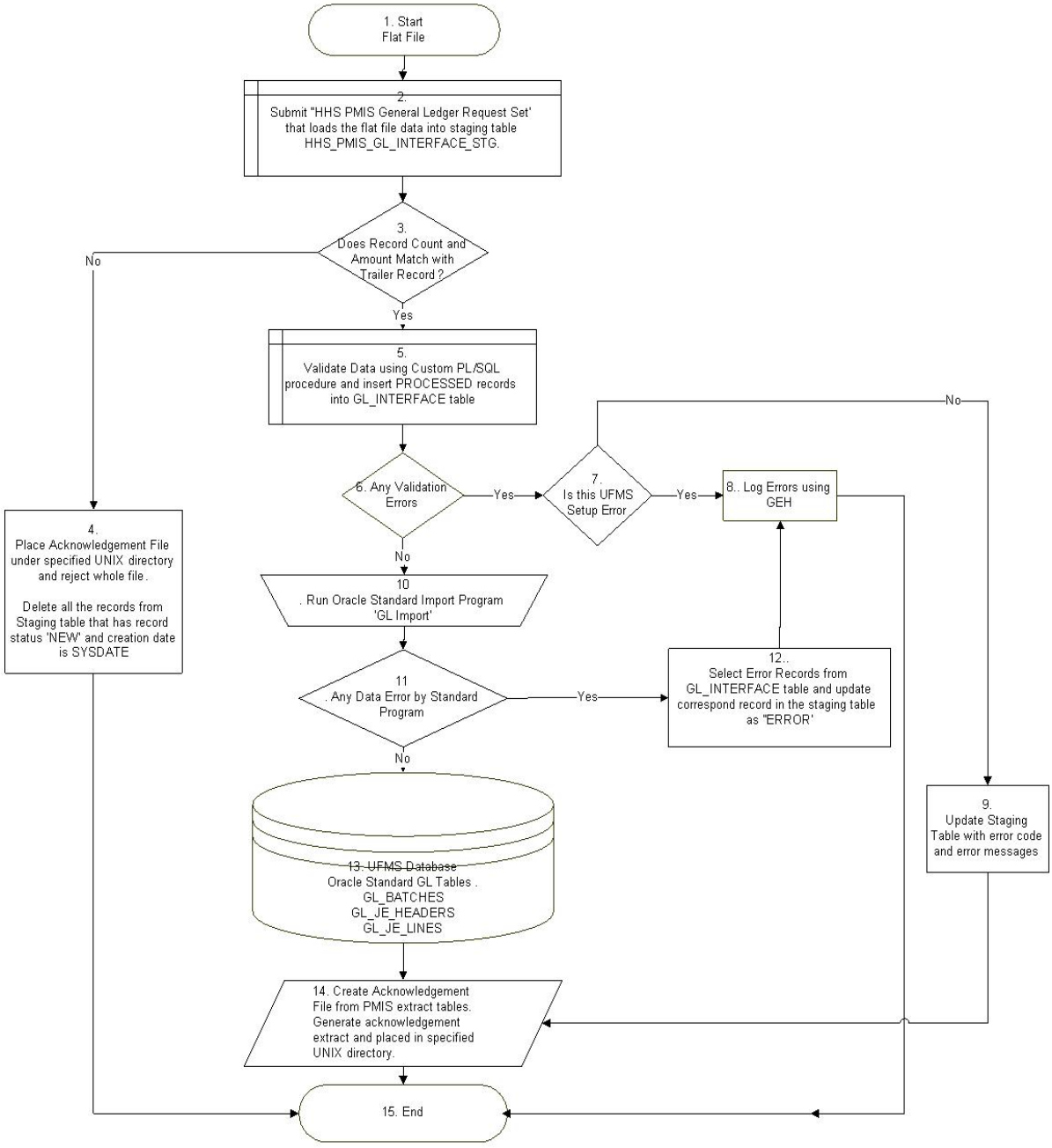


Table 6: General Ledger Inbound Interface Flow Diagram Process (Figure 2)

Process Flow No	Process Step Description
1.	This interface process starts with GL flat File in UFMS.
2.	As a part of inbound interface, Custom Request will be developed and registered as HHS PMIS General Ledger Request Set This Request Set calls SQL*loader script that loads the journal data into staging table HHS_PMIS_GL_INTERFACE_STG, header and trailer record into control table HHS_GLOBAL_INTERFACE_CTL_TBL. Custom PL/SQL package HHS_PMIS_GL_PK will be developed that select, validate the staging table data and insert the successful validated records into Oracle standard interface table GL_INTERFACE and corresponding records will be marked as 'PROCESSED' in staging table.
3.	Check if total journal count matches with record count in trailer record and total journal amount matches amount in trailer record.
4.	If step 3 fails, Acknowledgement will be placed in specified UNIX folder with error code RECORD_COUNT_MISMATCH and with proper error description. Delete all the records from General Ledger Staging table that has status NEW and period name is current period.
5.	If step 3 passes, then start validation for the records loaded in PMIS GL Staging table using custom PL/SQL package.
6.	If any errors during the custom PL/SQL validation? Go to step 7 else go to step 10.
7.	If any UFMS configuration errors, then go to step 8.
8.	All UFMS configuration and setup errors will be logged into GEH and whole journal batch will be rejected. Staging table column record_status will be updated with status as UFMS ERROR for error record and successful records in the same batch will be updated with status CHILD ERROR. After configuration correction, these records will be considered for next run. If these records are suspended using GEH, then all suspend records will be archived.
9.	If errors are not configuration errors, then whole journal batch will be rejected. Also update staging table columns record_status, error_ code and error_message with status ERROR, proper error code (see error Handling Section) and error message respectively for error records and all successful validated records in the batch will be updated with status as CHILD ERROR. If the whole journal batch is validated successfully, then update record_status column in staging table with status as 'PROCESSED' Insert all PROCESSED records from staging table to standard GL_INTERFACE table.
10.	Oracle Standard Concurrent Program **'Journal Import'** will be executed to import the PROCESSED transactions from step 9.
11.	If any error occurs, go to step 12 else go to step 13.

Process Flow No	Process Step Description
12.	If any error, records will be resided in GL_INTERFACE table (Standard Functionality). Select those records from GL_INTERFACE table and update corresponding record as 'ERROR' in the staging table for the corresponding journal batch and journal line number. Simultaneously log all the errors into GEH with the error message received from standard Import Program.
13.	If no errors found during standard Journal Import program, then successful validated records will be loaded in the Oracle Standard General Ledger interface table. Data will be inserted in the following standard table GL_JE_BATCHES GL_JE_HEADERS GL_JE_LINES
14.	Finally all records (ERROR, CHILD ERROR, UFMS ERROR and PROCESSED) will be selected from staging table. Acknowledgement file will be generated and placed in specified UNIX directory. In the next run, subsequent file will have only error batch that were error out in the previous run, load the file into PMIS staging table and update old ERROR record for same batch with status as SUSPEND and archive this records into archive table.
15.	The process Ends.

Figure 3: Obligations/Receipts/Invoices Extract Technical Flow Chart (From UFMS to Sunflower)

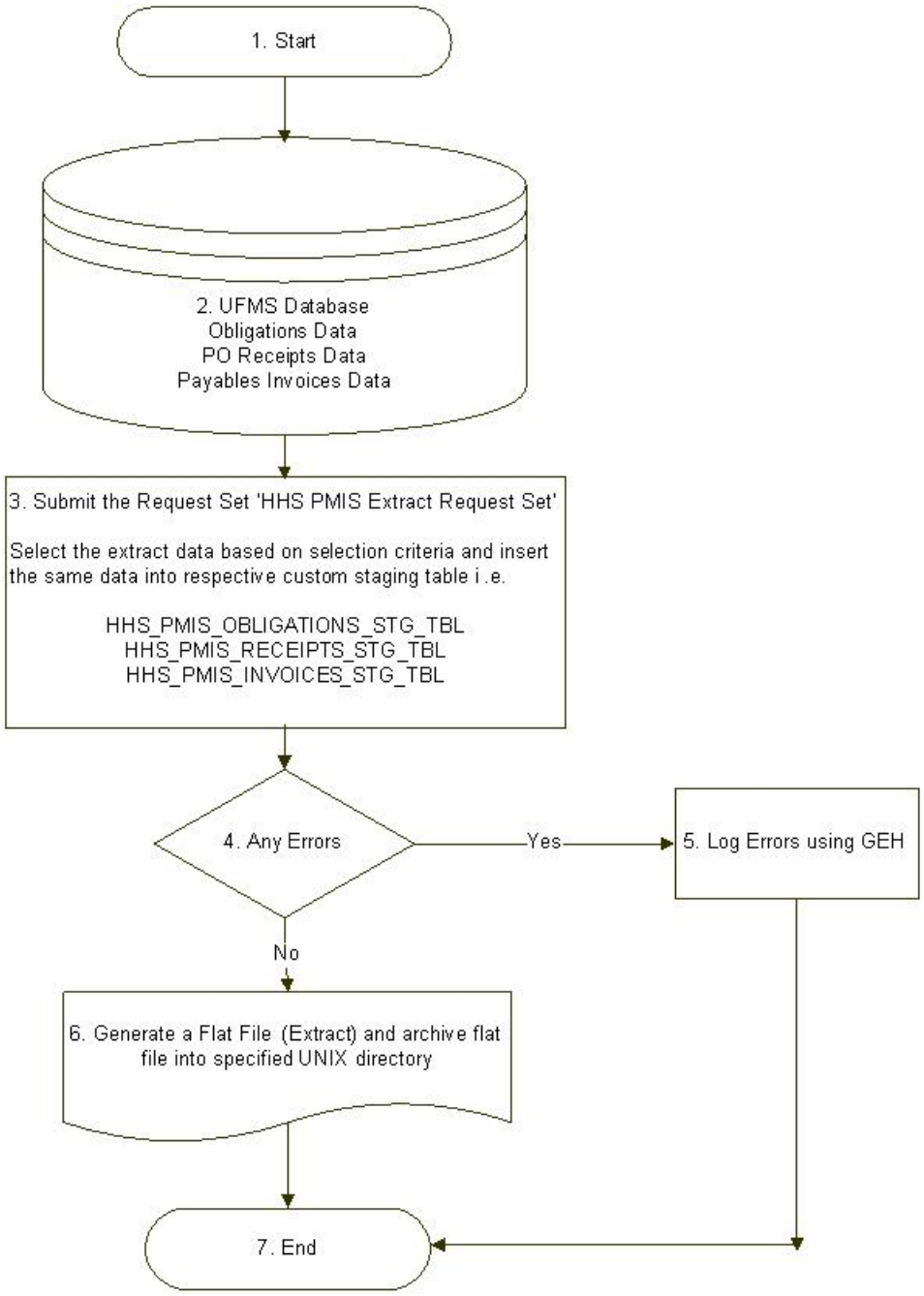


Table 7: Obligations/Receipts/Invoices Extract Flow Diagram Process (Figure 3)

Process Flow No	Process Step Description
1.	This interface process starts in UFMS.
2.	Obligations/Receipts/Invoices data resides in the Oracle Database of UFMS.
3.	As part of this outbound interface, Custom Request Set will be developed and registered as **HHS PMIS Extract Request Set'** that calls a customized PL/SQL package HHS_PMIS_PO_EXT_PK for obligations, HHS_PMIS_RCT_EXT_PK for Receipts and HHS_PMIS_INV_EXT_PK for invoices that select the records based on selection criteria, validate and insert the records into the staging table HHS_PMIS_OBLIGATIONS_STG_TBL, HHS_PMIS_RECEIPTS_STG_TBL, HHS_PMIS_INVOICES_STG_TBL respectively. This Program will have two parameter i.e. SET_OF_BOOKS_ID, ORG_ID.
4.	If any error occurs the go to step 5 else go to step 6.
5.	If any error occurs during the process, all errors will be logged into GEH.
6.	The program will finally generate and archive a copy of the extract file in to UFMS specified location (i.e.: $OS_ARCHIVE_TOP\ PMIS\out) if completed successfully.
7.	The process Ends

Figure 4: Sunflower to/from UFMS Acknowledgement Flow Chart

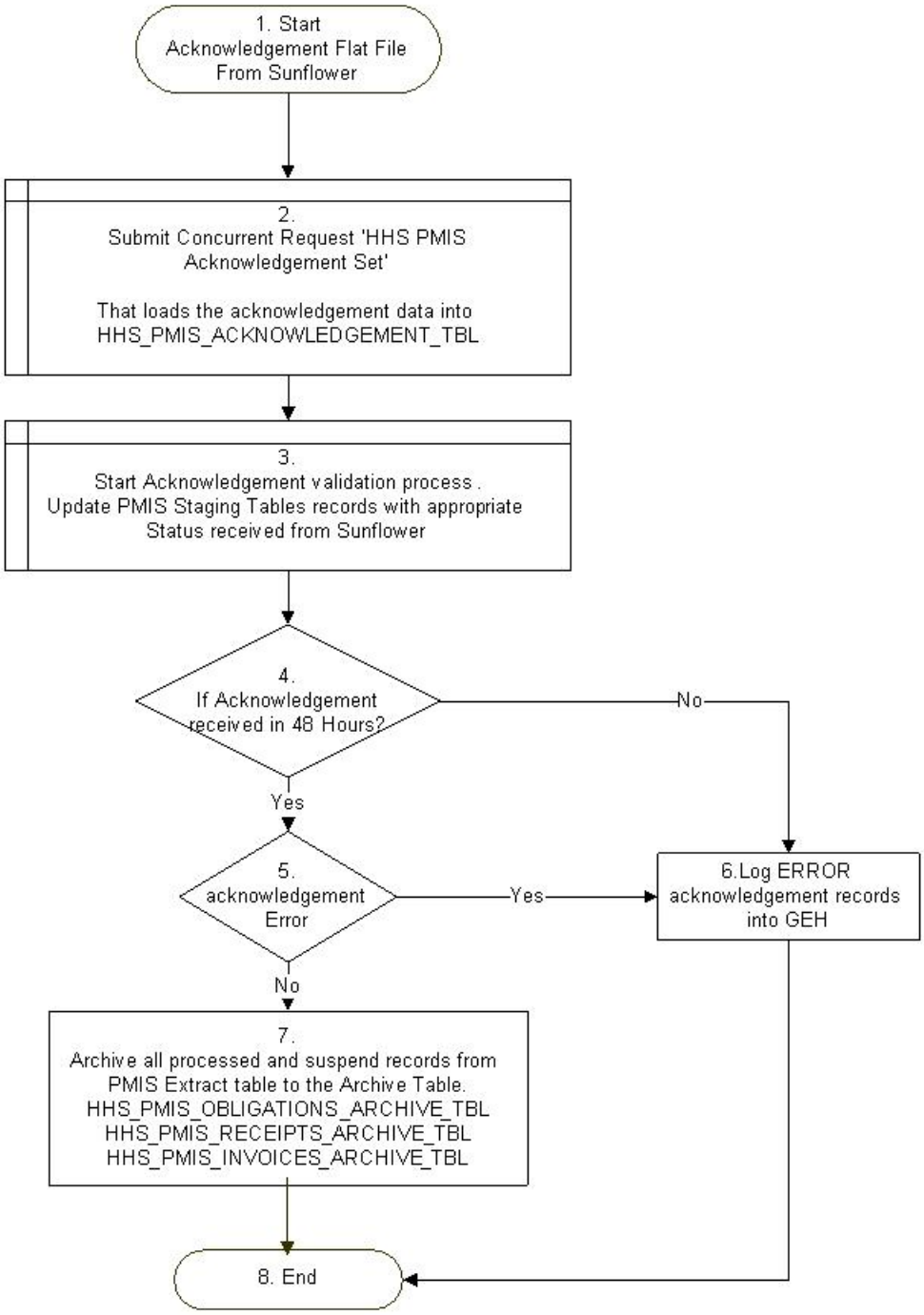


Table 8: PMIS-Sunflower Acknowledgement Flow Diagram Process (Figure 4)

Process Flow No	Process Step Description
1	This interface process starts in UFMS.
2	As part of this acknowledgement process, Custom Request Set will be developed and registered as **HHS PMIS Acknowledgement Request Set'** This Request Set calls SQL*loader script that loads the acknowledgment flat file data into staging table HHS_PMIS_ACKNOWLEDGEMNT_TBL
3.	Second program in the request set is a custom PL/SQL package, that will select the records from acknowledgement table and update the correspond records in the following staging table with status appropriate status available in the acknowledgement table. HHS_PMIS_OBLIGATIONS_STG_TBL HHS_PMIS_RECEIPTS_STG_TBL HHS_PMIS_INVOICES_STG_TBL
4.	If there is no acknowledgement file received within 48 hours, then log a message into GEH.
5	If any error acknowledgements, then go to step 6
6.	Update staging table column sunflower_status with appropriate status and logged ERROR records into GEH.
7.	In the next run, old error records that have sunflower status ERROR in the staging table will be updated with status as SUSPEND. Finally all processed and suspend records for Obligations, Receipts and Invoices staging table data will be archived into respective archive table mentioned below. HHS_PMIS_OBLIGATIONS_ARCHIVE_TBL HHS_PMIS_RECEIPTS_ARCHIVE_TBL HHS_PMIS_INVOICES_ARCHIVE_TBL
8.	The Process Ends.

4.2 Approach

4.2.1 Staging Tables

Table 9: Extract and Inbound Interface Tables

Table Name	Interface	Description
HHS_PMIS_OBLIGATIONS_STG_TBL	Extract	This staging table is use to store all the Obligations information before sending to Sunflower.
HHS_PMIS_RECEIPTS_STG_TBL	Extract	This staging table is use to store all the PO Receipts information before sending to Sunflower.
HHS_PMIS_INVOICES_STG_TBL	Extract	This staging table is use to store all the Invoices information before sending to Sunflower.
HHS_PMIS_ACKNOWLEDGEMENT_TBL	Extract	This table is use to report the entire acknowledgement received from Sunflower during the process at their end.
HHS_INTERFACE_CONTROL_TBL	Inbound	This table is use to store all header and trailer record information from the acknowledgement file receive from Sunflower.
HHS_PMIS_GL_INTERFACE_STG_TBL	Inbound	This custom staging table is use to store the journal ledger asset transactions received from Sunflower that will be processed and interfaced to UFMS General Ledger.
HHS_PMIS_OBLIGATIONS_ARCHIVE_TBL	Inbound	This staging table is use to archive all the Obligations data information from staging table.
HHS_PMIS_RECEIPTS_ARCHIVE_TBL	Inbound	This staging table is use to archive all the PO Receipts data from staging table.
HHS_PMIS_INVOICES_ARCHIVE_TBL	Inbound	This staging table is use to archive all the Invoices from data staging table.

4.2.2 Ordering of Tables

The PMIS interface tables will be used in the appropriate order as mentioned in section 4.2.1

4.2.3 Dependencies

1. The interface processes will be dependent on the setups specified in the functional document and configuration document.
2. UNIX directories on UFMS server have been created with valid read write and delete permissions.
3. Setup Descriptive flex fields on Obligation, Receipt and Invoice header table to capture PMIS Extract processing date.

4. Setup profile option HHS_PMIS_START_DATE at Responsibility level so that extract logic looks for PMIS start Date (Go-Live date for each opdiv). Date format should be DD-MON-YYYY.

4.3 Assumptions

4.3.1 General Assumptions

1. The PMIS interface is a two-way interface between Sunflower and UFMS, a success or failure notice will be sent by the interface back to Sunflower for each asset journal transaction sent to UFMS
2. The inbound data will be structured in a consistent manner for all Items and will confirm to the agreed HHS standards
3. The PMIS Interface processes are discussed in a separate document.
4. UNIX will have all READ. WRITE and EXECUTE permissions.
5. The GEH (General Error Handling) setups and configuration should be defined in the configuration document and is not discussed as part of this document.
6. "UFMS GEH Technical Design2 [1]. 1 Phase III" document can be referred for technical design details of Global Error Handling.
7. Flat will have one header and one trailer record associated.
8. All records being processed by the interface are assumed to be in USD currency.
9. File Transfer between Production Box and Feeder System will be done separately by UFMS drop box.
10. Sunflower will have access to UFMS drop box to get and put the file onto the server.

4.3.2 General Ledger Interface Assumptions

1. Separate property books shall be setup in Sunflower for each HHS OPDIV and journal entries shall be interfaced to UFMS by OPDIV.
2. Asset journal entries shall be created in UFMS General Ledger with a category type of Addition, Adjustment, Reclass, Transfer, Retirement and Depreciation
3. Asset journal entries shall be created in UFMS General Ledger with a source type of 'SUNFLOWER'.
4. Sunflower will send one data file for each opdiv. Each data file contains multiple batches based on category (Refer Assumption 2). Batch Name contains Source -Category for each opdiv.
5. If one line of an asset journal batch fails import to UFMS General Ledger then Sunflower shall be required to resend the failed journal batch
6. The PMIS interface shall send the CAN, Object Class, Fiscal Year, USSGL Account and HHS T-code and task_id (For HIS if applicable) to Sunflower as the accounting information. Sunflower shall send the CAN, Object Class, Fiscal Year and appropriate USSGL Account to UFMS for asset journal entries.
7. Calendar Period is open in General Ledger Module.

8. The PMIS interface shall pass all incoming asset journal entry accounting information through the CAN/BACS crosswalk and shall replace the value in the Account segment with the USSGL Account value from Sunflower.
9. Subsequent file that will be received from Sunflower will have only error batch records with same period.
10. Depreciation transactions will be rolled up by CAN, Object class, Fiscal Year, USSGL provided CAN is procurement related CAN.

4.3.3 Obligations/Receipts/Invoices Extract Assumptions

1. All the business setup, Descriptive Flex fields (DFF) and Custom Lookups defined in functional and configuration document will be completed before deployment of this document.
2. Obligation and Invoice header Global_attribute4 column will be setup to capture PMIS extract date. For Receipt, line level attribute4 column will be setup to capture PMIS extract date.
3. Profile option HHS_PMIS_START_DATE is setup at Responsibility level. PMIS start date value provided for all PMIS Responsibilities for each opdiv (IHS, FDA, PSC, and CDC). Once PMIS Start Date is defined, it can not be changed and if changed, we may loose the asset transactions. Date format for this profile option is DD-MON-YYYY.
4. Code Combination will be derived from HHS CANBACS XWALK Table based on CAN, Fiscal Year and Object Class.
5. If no data is fetched for the selection criteria, data file will not be created
6. Capitalized shipping and installation costs shall only be extracted to Sunflower if one or more lines on the purchase order have an object class value in the 31XXX series.
7. The PMIS shall extract shipping and installation costs only if there are created on the obligation using the capitalized object class value within the 22403 and 25240 object class series.
8. Obligation extract will fetch 2 way and 3-way match Obligations that are in OPEN and APPROVED state.
9. Only approved and reserved personal property purchase orders shall be sent to Sunflower as purchase order skeletal records.
10. If the purchase order is modified after the extract has been run, the PMIS interface shall send whole obligation information to Sunflower in the next run.
11. Receipt information will be extracted for the obligations that have already sent to Sunflower.
12. Receipt extract will be processed irrespective of whether fully or partially received.
13. In case of Invoice extract for PSC, Purchase Order details will not be available.
14. All invoice payment records for credit card purchases (2-way matched obligations) shall be extracted from UFMS Payables
15. The PMIS Interface shall not extract invoice information for 3-way matched obligations
16. Cancelled invoice lines shall not be interfaced to Sunflower.
17. For IHS, if the CAN used on the obligating document in UFMS is project related, the PMIS interface shall send the task Id to Sunflower and shall expect the same back from Sunflower when asset journal entries are being created in UFMS.

18. In the event of transmission error occurs, new extract will not be created. Instead request O&M to pull the extract from PMIS archive directory.

4.4 Inbound Program Logic (From Sunflower to UFMS)

4.4.1 General Ledger Interface Description

The PMIS inbound interface shall receive asset journal entries from Sunflower and interface them to the UFMS General Ledger by HHS OPDIV. The journal entries shall be created monthly in UFMS. The outbound PMIS interface shall process the following statuses of personal property payments:

(1) New and validated asset journal entry transactions from Sunflower

(2) The following asset journal categories shall be interfaced from Sunflower: Addition, Adjustment, Reclass, Transfer, Retirement and Depreciation.

(3) The source for the asset journal records shall be 'SUNFLOWER'

(4) The PMIS – UFMS Inbound interface shall receive an extract of periodic journal entries from Sunflower and interface them directly into the UFMS General Ledger as unposted journals.

(5) Sunflower shall send the debit and credit entries.

(6) The monthly depreciation expense shall be run automatically in Sunflower and interfaced to UFMS.

4.4.2 General Ledger Interface File Format

Fields in the flat file are pipe ('|') delimited. The following file name convention has been followed

File Name: - HHS_PMIS_CDC_GL_YYYYMMDDHHMISS.dat

HHS_PMIS_FDA_GL_YYYYMMDDHHMISS.dat

HHS_PMIS_PSC_GL_YYYYMMDDHHMISS.dat

HHS_PMIS_IHS_GL_YYYYMMDDHHMISS.dat

Table 10: General Ledger File Format

The following inbound Journal Asset transaction format will be received from Sunflower.

Field Name	Data type	Required/ Optional	Comments
Record Identifier	VARCHAR2(2)	R	Constant 'TX'
Record ID	NUMBER	R	Sequence for GL transaction in the flat file.
Set of Books Name	VARCHAR2(3)	R	Value Should be in ('CDC','FDA','PSC','IHS')
Currency Code	VARCHAR2(3)	R	Default Currency should be 'USD'
User JE Category Name	VARCHAR2(25)	R	Should be in Addition, Adjustment, Reclass, Transfer, Retirement and Depreciation

User JE Source Name	VARCHAR2(25)	R	Default Source should be 'SUNFLOWER'
Fiscal Year	VARCHAR2(150)	R	
Object Class	VARCHAR2(150)	R	
CAN	VARCHAR2(150)	R	
Account	VARCHAR2(150)	R	
Entered Debit	NUMBER	O	Required if Entered Credit Column is NULL
Entered Credit	NUMBER	O	Required if Entered Debit Column is NULL
Period Name	VARCHAR2(25)	R	Format MMM-CurrentYear-FY-FiscalYear E.g. DEC-06-FY-07
Transaction Date	DATE	R	Format DD-MON-YYYY E.g. 12-FEB-2007
Batch Name	VARCHAR2(150)	O	PMIS Asset Batch For <Category Name> - <Period Name>
Batch Description	VARCHAR2(150)	O	PMIS Asset Batch For <Category Name>
Journal Name	VARCHAR2(150)	O	PMIS Asset Journal for <Category Name>
Journal Description	VARCHAR2(150)	O	PMIS Asset Journal for <Category Name> and <Period>
JE Line Number	NUMBER	O	Journal Ledger Line Number
Line Description	VARCHAR2(150)	O	Document Number:<Document Number>
HHS Elimination Code	VARCHAR2(150)	O	If vendor is Federal
FACTS Trading Partner Code	VARCHAR2(150)	O	If vendor is Federal
GPRA code	VARCHAR2(3)	O	
Accounting Point	VARCHAR2(3)	O	Required only for 'IHS'
Task ID	NUMBER	O	Required only for 'IHS'

The following is the Trailer Record Format

Field Name	Data type	Comments
Record Identifier	VARCHAR2 (2)	Value will be 'TT'
Organization Code	VARCHAR2 (3)	Value will be 'CDC','FDA','PSC','IHS'
Total Number of journal batches in the extract	NUMBER	
Total Number of lines in the extract	NUMBER	
Total journal Amount for the extract	NUMBER	Addition of total debits + total credits
Batch Date/SYSDATE	NUMBER	Format (YYYYMMDDHH24MISS) E.g. 20070212103245

4.4.3 Table and View Usage

Table 11: Table Usage

Table Name	Select	Insert	Update	Delete

HHS_GLOBAL_INTERFACE_CTL_TBL	X	X	X	
HHS_PMIS_GL_INTERFACE_STG_TBL	X	X	X	
GL_INTERFACE	X	X		
HHS_PMIS_ACKNOWLEDGEMENT_TBL	X	X	X	

4.4.4 GL Interface Program Logic

1. The SQL*loader (HHS_PMIS_GL_LOAD) and PL/SQL package (HHS_PMIS_GL_PK) will be developed to load and validate the data respectively.
2. Run 'HHS PMIS General Ledger Request Set' that loads Sunflower Asset Transactions from flat file into the PMIS General Ledger Staging table (HHS_PMIS_GL_INTERFACE_STG_TBL) with status as 'NEW' and a trailer record will be loaded into the control table (HHS_GLOBAL_INTERFACE_CTL_TBL) with batch count, transaction count, total amount.

- **Pre Validation Steps**

a) Load transaction from flat file into staging table HHS_PMIS_GL_INTERFACE_STG_TBL.

b) If the data file is subsequent data file from previous run and contains only error batch records that were error out in previous run, then compare the same batch with record status ERROR that were error out in previous run and NEW records for current loaded records. If there are duplicate records for the same batch then update record_status in the staging table from ERROR TO SUSPEND.

c) Archive all suspended and processed records that have record_status as SUSPEND and PROCESSED respectively from previous run.

d) Check whether total number of asset transactions with status as NEW and dollar amount from PMIS staging table equals count of transaction, total amount respectively from control stable.

e) If above validation fails, the whole flat file records will be rejected and acknowledgement will be placed under respective UNIX directory that contains following error code and error message and flat file name. This acknowledgment will have a same acknowledgement format with following error code and error message.

Table 12: Pre- Validations

Error Code	Error Message	Action
RECORD_COUNT_DOLLAR__ MISMACH	Total Number of Journal Entries doesn't match with the count in Trailer Record	Cancel whole Flat File records

f) If above validation passes, perform the below mentioned validation steps for the records that as record_status as NEW, UFMS ERROR and CHILD ERROR. Default validation is, below mentioned columns should be available in the flat file

- **Validation Steps**

Table 13: Validations

Flat File Column	Validation	Table Name	Action if validation Fail
Set of Books Name	Should be exists in the table and derive set_of_books_id	GL_SETS_OF_BOOK	Reject full journal batch
Transaction Date	Should be open date in GL Calendar	GL_PERIOD_STATUSES	Reject full journal batch
Object Class	Should be exists in the table and use to derive Code Combination	HHS_CANBACS_XWALK	Reject full journal batch
Fiscal Year	Should be exists in the table and use to derive Code Combination	HHS_CANBACS_XWALK	Reject full journal batch
CAN	Should be exists in the table and use to derive Code Combination	HHS_CANBACS_XWALK	Reject full journal batch
Actual Flag	Defaulted to 'A'		Reject full journal batch
User JE Category Name	Should be available in the table	GL_JE_CATEGORIES	Reject full journal batch
User JE Source Name	Source will be defaulted to 'SUNFLOWER'and exist in the table	GL_JE_SOURCES	Reject full journal batch
Currency Code	Currency Code will be defaulted to 'USD'and exist in the table	FND_CURRENCIES	Reject full journal batch
Entered Debit	Should be available if Entered Credit is zero		Reject full journal batch
Entered Credit	Should be available if Entered Debit is zero		Reject full journal batch

a. If any of the transaction in the batch gets error out due to configuration at UFMS, whole batch will be error out and only error record will be logged into GEH. Update record_status column in the staging table with ERROR whereas sunflower_status column will be updated as UFMS ERROR for error records and if there are successful validated records within the same batch then update record_status column in staging table with status as 'CHILD ERROR'. Also update error code and error messages in the staging table based on above error types. If whole journal batch passes the validation, record_status column will be updated with status as PROCESSED.

b. If any there are any data errors, Update record_status column in the staging table with ERROR for error records and if there are successful validated records within the same batch then update record_status column in staging table with status as 'CHILD ERROR'. Also update error code and error messages in the staging table based on above error types. If whole journal batch passes the validation, record_status column will be updated with status as PROCESSED.

c. If above validations passes successfully, then PROCESSED journal transactions will be loaded into the oracle standard interface table (GL_INTERFACE).

d. Finally Oracle Standard Interface program 'Journal Import' will be executed that will load the successfully validated journal transactions from standard interface table into the standard base tables and journal batch will be created in UFMS Journal Ledger.

e. If any error occurs by standard program, update record_status from PROCESSED to UFMS ERROR in PMIS GL staging table for corresponding journal name and journal line number and errors will be logged into GEH.

f. If there are any abnormal errors, delete journal transactions from GL_INTERFACE table for the source as 'SUNFLOWER', period_name is current period name. Also update staging table records with record_status as 'NEW for the source as 'SUNFLOWER', period_name is current period name. Restart the process from validation program onwards.

g. Finally select PROCESSED, ERROR, CHILD ERROR and UFMS ERROR records from staging table along with RECORD ID, ERROR SOURCE, HHS Agency, Journal Batch, Journal Batch Line Number, Error Code and Error Description. In case of PROCESSED records, error code will be zero (0) and error description will be blank (NULL). Create an extract (acknowledgement) for all records and placed in the specified UNIX directory.

4.5 Outbound File Format

4.5.1 Obligation (Purchase Order) File Format

All Fields in the flat file are pipe ('|') delimited. The following file name convention has been followed:

File Name: - HHS_PMIS_CDC_PO_YYYYMMDDHHMISS.dat
HHS_PMIS_FDA_PO_YYYYMMDDHHMISS.dat
HHS_PMIS_PSC_PO_YYYYMMDDHHMISS.dat
HHS_PMIS_IHS_PO_YYYYMMDDHHMISS.dat

Table 14: Obligation File Format

The following is the Header Record Format:

Field Name	Data type	Comments
Record Identifier	VARCHAR2 (2)	Value will be 'TH'
Organization Code	VARCHAR2 (3)	Value will be 'CDC','FDA','PSC','IHS'
Batch Date/SYSDATE	DATETIME	Format (YYYYMMDDHH24MISS) E.g. 20070212103245

The following is the Transaction Record Format:

Field Name	Data type	Optional/ Required	Comments
Record Identifier	VARCHAR2 (2)	R	Value will be 'TX'
Record ID	NUMBER	R	Sequence for each PO# in the file.

Document Number	VARCHAR2 (50)	R	Obligation Number
Line Number	VARCHAR2 (15)	R	Obligation Line Number
Document Date		R	Purchase Order Approval Date
Issuing Office	VARCHAR2 (3)	R	Value will be in 'FDA','CDC','PSC','HIS'
Revision Number	NUMBER	R	Purchase Order Revision Number
Revised Date	DATE	O	Date of Revision
Description	VARCHAR2 (240)	O	
Document Type	VARCHAR2 (25)	R	Should be 'STANDARD'
Interface Source	VARCHAR2 (150)	O	Source of the Purchase Order
Vendor Name	VARCHAR2 (150)	R	
Vendor Site Code	VARCHAR2 (25)	R	Vendor Site Code
Address Line 1	VARCHAR2 (25)	R	Ship To Address Line 1
Address Line 2	VARCHAR2 (25)	O	Ship To Address Line 2
City	VARCHAR2 (25)	R	Ship To City
State	VARCHAR2 (25)	R	Ship To State
Postal Code	VARCHAR2 (25)	R	Ship to Postal Code
Telephone Number	VARCHAR2 (15)	O	
Document Cancel Flag	VARCHAR2 (1)	O	Purchase Order Cancellation Flag
Line Type	VARCHAR2 (10)	R	PO Line Type E.g. AMOUNT, QUANTITY
Line Description	VARCHAR2 (240)	O	
Category Description	VARCHAR2 (240)	R	
Unit of Measure	VARCHAR2 (25)	R	Unit of Measure Code E.g. Each
Line Cancel Flag	VARCHAR2 (1)	O	Line Cancellation Flag
Line Cancel Date	DATE	O	Purchase Order Line Cancellation Date
Shipment Number	NUMBER	R	Purchase Order Shipment Line Number
Shipment Cancel Flag	VARCHAR2 (1)	O	Shipment Cancel Flag
Shipment Cancel Date	DATE	O	Shipment Line Cancellation Date
Match Option	VARCHAR2 (1)	R	Match Option Ex. 2-Way, 3-Way
Distribution Number	NUMBER	R	Purchase Order Distribution Line Number
CAN	VARCHAR2 (10)	R	CAN Number
Fiscal Year	VARCHAR2 (4)	R	Fiscal Year
Object Class	VARCHAR2 (6)	R	Object Class
Account	VARCHAR2 (7)	R	SGL Account
Concatenated Segments	VARCHAR2 (150)	R	
Unit Price	NUMBER	R	Purchase Order Line Unit Price
Quantity	NUMBER	R	Purchase Order Quantity at Distribution Level
Quantity Received	NUMBER	O	Purchase Order Received Quantity
Quantity Billed	NUMBER	O	Purchase Order Billed Quantity
Quantity Cancelled	NUMBER	O	Purchase Order Cancelled Quantity
Requisition Number	VARCHAR2 (50)	O	Requisition Number
Requisition Line Number	NUMBER	O	Requisition Line Number

Project Number	VARCHAR2 (25)	O	Project Number if CAN is Project Related
Project Name	VARCHAR2 (240)		Project Name if CAN is Project Related
Task ID	NUMBER	O	Task ID if CAN is Task Related
Task Number	VARCHAR2 (25)	O	Task Number if CAN is Task Related
Task Name	VARCHAR2 (240)	O	Task Name if CAN is Task Related
Expenditure type	VARCHAR2 (30)	O	Expenditure Type
Expenditure Organization	NUMBER	O	Expenditure Organization
Expenditure Item Date	DATE	O	Expenditure Item Date
PSC Contract Number	VARCHAR2 (50)	O	
FDA Tag Number	VARCHAR2 (50)	O	Applicable only for 'FDA'. Attribute10 on PO Distributions
FDA Agency Tag Number	VARCHAR2 (150)	O	Applicable only for 'FDA'. Attribute11 on PO Distributions
Accounting Point	VARCHAR2 (3)	O	Required only for 'HIS'
Fund Cancel Date	DATE	O	Fund Cancel Date
Receipt Required Flag	CHAR(1)	R	Possible values are 'Y' and 'N'

The following is the Trailer Record Format for:

Field Name	Data type	Comments
Record Identifier	VARCHAR2 (2)	Value will be 'TT'
Total Number of Obligations in the extract	NUMBER	
Total Number of Obligations lines in the extract	NUMBER	
Total Obligation Amount for the extract	NUMBER	Total Absolute value (Unit Price * Quantity)

4.5.2 Receipts File Format

All Fields in the flat file are pipe ('|') delimited. The following file name convention has been followed:

File Name: - HHS_PMIS_CDC_RCT_YYYYMMDDHHMISS.dat

HHS_PMIS_FDA_RCT_YYYYMMDDHHMISS.dat

HHS_PMIS_PSC_RCT_YYYYMMDDHHMISS.dat

HHS_PMIS_IHS_RCT_YYYYMMDDHHMISS.dat

Table 15: Receipt File Format

The following is the Header Record Format:

Field Name	Data type	Comments

Field Name	Data type	Comments
Record Identifier	VARCHAR2 (2)	Value will be 'TH'
Organization Code	VARCHAR2 (3)	Value will be 'CDC','FDA','PSC','IHS'
Batch Date/SYSDATE	DATETIME	Format (YYYYMMDDHH24MISS) E.g. 20070212103245

The following is the Transaction Record Format:

Field Name	Data type	Required/ Optional	Comments
Record Identifier	VARCHAR2(2)	R	Value will be 'TX'
Record ID	NUMBER	R	Sequence for Receipt # available in the flat file.
Receipt Number	VARCHAR2(30)	R	PO Receipt Number
Receipt Date	DATE	R	Transaction Date
Comments	VARCHAR2(240)	O	
Receipt Line Number	NUMBER	R	This is Receipt Line Number
Receipt Quantity	NUMBER	R	Quantity for each Line
PO Unit Price	NUMBER	R	PO unit price
Item Description	VARCHAR2(240)	O	
Shipment Line Status	VARCHAR2(25)	R	Values will be ('FULLY RECEIVED', 'PARTIALLY RECEIVED','EXPECTED')
Line Comments	VARCHAR2(240)	O	
Transaction Type	VARCHAR2(25)	R	Value will be 'RECEIVE'
Unit of Measure	VARCHAR2(25)	R	Unit of Measure Code E.g. Each
Currency Code	VARCHAR2(3)	R	Receipt Currency Code
Quantity Shipped	NUMBER	O	Quantity Shipped
Quantity Received	NUMBER	O	Quantity Received
PO Number	VARCHAR2 (25)	R	PO Number associated with the Receipt
PO Revision Number	NUMBER	R	Purchase Order Revision Number
PO Line Number	NUMBER	R	PO Line Number associated with the Receipt
PO Shipment Number	NUMBER	R	PO Shipment Number associated with the Receipt
PO Distribution Number	NUMBER	R	PO Distribution Number associated with the Receipt
Vendor Name	VARCHAR2 (150)	R	Vendor Name associated with PO
Vendor Site Code	VARCHAR2 (25)	R	Vendor Site Code associated with PO
CAN	VARCHAR2 (25)	R	CAN Number
Fiscal Year	VARCHAR2 (4)	R	Fiscal Year
Object Class	VARCHAR2 (5)	R	Object Class
USSGL Account	VARCHAR2 (7)	R	USSGL Account Number
Concatenated Segments	VARCHAR2 (150)	R	
HHS Agency Code	VARCHAR2 (3)	R	Value will be 'CDC','FDA','PSC','IHS'
Accounting Point	VARCHAR2 (3)	O	Required only for 'IHS'

Fund Cancel Date	DATE	O	Fund Cancel Date

The following is the Trailer Record Format:

Field Name	Data type	Comments
Record Identifier	VARCHAR2 (2)	Value will be 'TT'
Total Number of Receipts in the extract	NUMBER	
Total Number of Receipt lines in the extract	NUMBER	
Total Receipt Amount for the extract	NUMBER	Total Absolute value (PO Unit Price * Receipt Quantity)

4.5.3 Invoices File Format

Fields in the flat file are pipe ('|') delimited. The following file name convention has been followed

File Name: - HHS_PMIS_CDC_INV_YYYYMMDDHHMISS.dat

HHS_PMIS_FDA_INV_YYYYMMDDHHMISS.dat

HHS_PMIS_PSC_INV_YYYYMMDDHHMISS.dat

HHS_PMIS_IHS_INV_YYYYMMDDHHMISS.dat

Table 16: Invoice File Format

The following is the Header Record Format

Field Name	Data type	Comments
Record Identifier	VARCHAR2 (2)	Value will be 'TH'
Organization Code	VARCHAR2 (3)	Value will be 'CDC','FDA','PSC','HIS'
Batch Date/SYSDATE	DATETIME	Format (YYYYMMDDHH24MISS) E.g. 20070212103245

The following is the Transaction Record Format:

Field Name	Data type	Required/ Optional	Comments
Record Identifier	VARCHAR2 (2)	R	Value will be 'TX'
Record ID	NUMBER	R	Sequence for Invoice # available in the flat file.
Invoice Number	VARCHAR2 (50)	R	Invoice Number
Invoice Line number	NUMBER	R	Invoice Distribution Number
Invoice Date	DATE	R	Date of invoice created
Invoice Currency Code	VARCHAR2 (3)	R	Invoice currency code
Invoice Source	VARCHAR2 (50)	R	Source of the Invoice

PO Number	VARCHAR2 (50)	O	PO Number (Not Applicable for PSC)
PO Line Number	NUMBER	O	PO Line Number (Not Applicable for PSC)
Vendor Name	VARCHAR2 (150)	O	Name of the Vendor
Vendor Site Code	VARCHAR2 (25)	O	Site Name of the Vendor
Invoice Type	VARCHAR2 (25)	R	Values will be ('PREPAYMENT', 'STANDARD', 'INTEREST','ADJUSTMENT' etc.)
Payment Status Flag	VARCHAR2 (1)	R	Values will be ('Y','N','P')
Interface Source	VARCHAR2 (150)	O	Attribute1 in Invoice headers.
Asset Tracking Flag	VARCHAR2 (1)	O	Values will be ('Y','N')
Invoice Description	VARCHAR2 (240)	O	
Quantity Invoiced	NUMBER	O	Invoice Quantity
Unit Price	NUMBER	O	Unit Price
CAN	VARCHAR2 (15)	R	CAN
Fiscal Year	VARCHAR2 (4)	R	Fiscal Year
Object Class	VARCHAR2 (5)	R	Object Class
USSGL Account	VARCHAR2 (7)	R	USSGL Account
Concatenated Segments	VARCHAR2 (150)	R	
Payment Amount	NUMBER	R	This is a payment amount.
Payment number	NUMBER	R	Payment Line Number
Check Number	VARCHAR2 (50)	R	Payment Check Number
Payment Date	DATE	R	Check Date or Payment Date
Project Number	VARCHAR2 (25)	O	Project Number if CAN is Project Related
Project Name	VARCHAR2 (240)	O	Project Name if CAN is Project Related
Task Id	NUMBER	O	Task Id if CAN is Task Related
Task Number	VARCHAR2 (25)	O	Task Number if CAN is Task Related
Task Name	VARCHAR2 (240)	O	Task Name if CAN is Task Related
Expenditure Item Date	DATE	O	Only applicable for Project Related CAN
Expenditure Org ID	NUMBER	O	Only applicable for Project Related CAN
Expenditure Type	VARCHAR2 (50)	O	Only applicable for Project Related CAN
HHS Agency	VARCHAR2 (4)	R	
FDA Tag Number	VARCHAR2 (150)	O	Applicable only for 'FDA'. Attribute10 on Invoice Distributions
FDA Agency Tag Number	VARCHAR2(150)	O	Applicable only for 'FDA'. Attribute11 on Invoice Distributions
MACCS Document Key	VARCHAR2 (150)	O	Applicable only for 'PSC'. Attribute13 on Invoice Distributions
Accounting Point	VARCHAR2 (3)	O	Required only for 'IHS'
Fund Cancel Date	DATE	O	Fund Cancel Date

The following is the Trailer Record Format:

Field Name	Data type	Comments
Record Identifier	VARCHAR2 (2)	Value will be 'TT'
Total Number of Invoices in the extract	NUMBER	

Field Name	Data type	Comments
Total Number of Invoice lines in the extract	NUMBER	
Total Invoice Amount for the extract	NUMBER	Sum of Payment Amount

4.6 Extract Program Logic (From UFMS to Sunflower)

4.6.1 Obligations Extract Logic

1. A PL/SQL package will be developed for this extract (HHS_PMIS_PO_EXT_PK) and execute 'HHS PMIS Extract Request Set' with following parameters

Table 17: Obligations Extract - Parameter

Parameter Name	Default Value	Required	Display	Comments
ORG_ID	FND_PROFILE.value ('ORG_ID')	Y	N	HHS Agency Value.
SET_OF_BOOK_ID	FND_PROFILE.value (GL_SET_OF_BKS_ID)	Y	N	Set of Books ID for HHS Agency

2. This Request Set calls the above package that select all the data elements mentioned in the Table 8.0 based on the following criteria and insert into PMIS obligation extract table.
   a) Obligations are APPROVED and RESERVED.
   b) If one or more lines on the obligation have an object class value in the 31XXX series.
   c) If obligation lines are created using object class value within 22403 and 25240
   d) Obligations are modified and new revision number is greater than old revision number that has already sent to Sunflower.
3. Select all the records from PMIS Obligation extract table and generate a flat file in the specified UNIX directory with one header and trailer record that contains HHS Agency, Obligation count, Obligation Line Count and total obligations amount.
4. If any error occurs during the extract, errors will be logged into GEH. If no errors found, then extract will be generated in PMIS archive folder.
5. Once file gets created, Global_attribute4 column on the Obligation header will be updated with current process date (PMIS Extract Date) and will be considered for next run.
6. As a final step the program will copy the data file from PMIS archive directory to the specified UNIX location ($OS_INTERFACE_TOP/PMIS/out) on successful completion of extract program.
7. If no data selected, extract will not be created.

4.6.2 Receipts Extract Logic

1. A PL/SQL package will be developed for this extract (HHS_PMIS_RCT_EXT_PK) and execute 'HHS PMIS Extract Request Set' with following parameters

Table 18: Receipts Extract - Parameter

Parameter Name	Default Value	Required	Display	Comments
ORG_ID	FND_PROFILE.value ('ORG_ID')	Y	N	HHS Agency Value.
SET_OF_BOOK_ID	FND_PROFILE.value (GL_SET_OF_BKS_ID)	Y	N	Set of Books ID for HHS Agency

2. This Request Set calls the above package that select all the data elements mentioned in the Table 9.0 based on the following criteria and insert into PMIS Receipts extract table.
   g) Obligation is already extracted by Obligation extract (Global_attribute4 column should have PMIS Process Date)
   h) Receipt record is new (attribute4 doesn't have PMIS Process Date).
3. Select all the records from PMIS Receipt extract table and generate a flat file in the specified UNIX directory with one header and trailer record that contains HHS Agency, Receipt count, Receipt Line Count and total receipt amount.
4. If any error encounters, they will be logged into GEH. If no errors found, then extract will be generated in PMIS archive folder.
5. Once file gets created, attribute4 column on the Receipt line will be updated with current process date (PMIS Extract Date) and will be considered for next run.
6. As a final step the program will copy the data file from PMIS archive directory to the specified UNIX location ($OS_INTERFACE_TOP/PMIS/out) on successful completion of extract program.
7. If no data selected, extract will not be created.

4.6.3 Invoices Extract Logic

1. A PL/SQL package will be developed for this extract (HHS_PMIS_INV_EXT_PK) and execute 'HHS PMIS Extract Request Set' with following parameters

Table 19: Invoices Extract - Parameter

Parameter Name	Default Value	Required	Display	Comments
ORG_ID	FND_PROFILE.value ('ORG_ID')	Y	N	HHS Agency Value.

SET_OF_BOOK_ID	FND_PROFILE.value (GL_SET_OF_BKS_ID)	Y	N	Set of Books ID for HHS Agency

2. This Request Set calls the above package that select all the data elements mentioned in the Table 10.0 based on the following criteria and insert into PMIS Invoices extract table.
   a) Invoices are APPROVED and VALIDATED.
   b) Invoice record is new. (Global_attribute4 doesn't have PMIS Process Date).
   c) If one or more lines on the invoice have an object class value in the 31XXX series.
   e) If invoice lines are created using objects class value within 22403 and 25240.
3. Select all the records from PMIS Invoice extract table and generate a flat file in the specified UNIX directory with one header and trailer record that contains HHS Agency, Invoice count, Invoice Line Count and total invoices amount.
4. If any error encounters, they will be logged into GEH. If no errors found, then extract will be generated in PMIS archive folder.
5. Once file gets created, Global_attribute4 column on the Invoice header will be updated with current process date (PMIS Extract Date) and will be considered for next run.
6. As a final step the program will copy the data file from PMIS archive directory to the specified UNIX location ($OS_INTERFACE_TOP/PMIS/out) on successful completion of extract program.
7. If no data selected, extract will not be created.

4.6.4 Acknowledgement Process

This acknowledgement process receives the acknowledgement file from Sunflower for the extract that has been already sent to Sunflower.

Acknowledgement File Format

Fields in the flat file are pipe ('|') delimited. The following file name convention has been followed

File Name: HHS_PMIS_CDC_ACK_*YYYYMMDDHHMISS.dat
HHS_PMIS_FDA_ACK_*YYYYMMDDHHMISS.dat
HHS_PMIS_PSC_ACK_*YYYYMMDDHHMISS.dat
HHS_PMIS_IHS_ACK_*YYYYMMDDHHMISS.dat
(* indicates acknowledgement file for PO, Receipt and Invoice)

Table 20: Acknowledgement File Format

The following inbound Journal Asset transaction format will be received from Sunflower.

Column Name	Data Format	Required /Optional	Default Value
Record ID	NUMBER	R	Zero (0) in case of Record Count Mismatch Error else actual record_id available in the file.
Interface Source	VARCHAR2(25)	R	For outbound ('PO','RCT','INV'), For inbound 'GL'
HHS Agency	VARCHAR2(3)	R	Value will be ('CDC','PSC','FDA','IHS')
Document Number	VARCHAR2(50)	R – If record count and dollar amount Matches	PO # or Receipt # or Invoice # based for extract and Journal Batch Name for GL
Document Line Number	NUMBER	R – If record count and dollar amount Matches	PO Line# or Receipt Line# or Invoice Line# based for extract and journal Line number for GL
Accounting Point	VARCHAR2(10)	O	Required only for 'IHS' and required if record count and dollar amount Matches
Record Status	CHAR(1)	R	Valid Sunflower Statuses are 'P','E','C','S'. (Refer section 4.8, table 30 for error statuses) Valid UFMS Status for GL Interface are 'P','E','C','U'
Error Code	VARCHAR2(15)	R	
Error Description	VARCHAR2(250)	O	

Acknowledgement Extract Logic (From Sunflower to UFMS)

1. The SQL*loader (HHS_PMIS_ACK_LOAD) and PL/SQL package (HHS_PMIS_ACK_PK) will be developed to load and validate the data respectively.
2. Run 'HHS PMIS Extract Request Set' that loads the acknowledgement transactions received from Sunflower into PMIS acknowledgement table and perform the following process by using custom PL/SQL package.
3. Acknowledgement file contains a record identifier (RECORD_ID), Document Number and Document Line Number, Error code, Error Description and record Status.
4. Select all the records along with above columns from PMIS acknowledgement table and perform the following process for each record in the staging table.
   a) If Error Code is other than zero (that means ERROR), then update record_status column in staging table from PROCESSED to ERROR and simultaneously update column sunflower_status to ERROR. Also update the error code and error message in columns ERROR_CODE and ERROR_MESSAGE of staging table respectively for corresponding RECORD_ID.
   b) If there are any SUSPEND records in the acknowledgement, then update sunflower_stautus with status as SUSPEND.
   c) Log all error acknowledgements that contain error code and error messages into GEH.

d) If error code is zero (0) it means the record is validated successfully and it is PROCESSED at Sunflower. Select all PROCESSED records from acknowledgement table and update sunflower_status column in the staging table with status as, PROCESSED.

5. If UFMS doesn't receive the acknowledgement file in 48 hours, then log a message into GEH. The process to find out 48 hours period is, once the acknowledgement process is completed (step 4), then select sunflower_status column. If the value for this column is NULL i.e. acknowledgement file has not been received, then log a message into GEH.
6. Finally archives all PROCESSED and SUSPEND records from staging table to respective archive table and error records will be considered for next extract (outbound) run.
7. If no action has taken on error records, then these records will be considered for next extract (Obligation/Receipts/Invoices) run and will be selected along with new records.
8. Upon correction of these error obligations records in UFMS, Revision number will be changed for modified obligations.
9. Select new created records along with the modified records (mentioned in step 7) and update record_status and sunflower_status from ERROR to SUSPEND for old error records in staging table and archive these records into archive table in the next acknowledgement process.

Acknowledgement Extract Logic (From UFMS to Sunflower)

File Name: - HHS_PMIS_CDC_ACK_GL_YYYYMMDDHHMISS.dat
HHS_PMIS_FDA_ACK_GL_YYYYMMDDHHMISS.dat
HHS_PMIS_PSC_ACK_GL_YYYYMMDDHHMISS.dat
HHS_PMIS_IHS_ACK_GL_YYYYMMDDHHMISS.dat

1. Once the Journal Import Process completes, select acknowledgement extract columns (see Tale 20, Acknowledgement File Format Section) from PMIS GL staging table.
2. Generate a flat file with above file naming conventions and placed in specified UNIX directory.

4.6.5 Table and View Usage

Table 21: Extract Program – Table/View Usage

Table Name	SELECT	INSERT	UPDATE	DELETE
HHS_PMIS_OBLIGATIONS_STG_TBL	X	X	X	
HHS_PMIS_RECEIPTS_STG_TBL	X	X	X	
HHS_PMIS_INVOICES_STG_TBL	X	X	X	
HHS_GLOBAL_INTERFACE_CTL_TBL	X	X	X	
HHS_PMIS_OBLIGATIONS_ARCHIVE_TBL		X		

Table Name	SELECT	INSERT	UPDATE	DELETE
HHS_PMIS_RECEIPTS_ARCHIVE_TBL		X		
HHS_PMIS_INVOICES_ARCHIVE_TBL		X		
PO_HEADERS_ALL	X		X	
PO_LINES_ALL	X			
PO_LINES_LOCATIONS_ALL	X			
PO_DISTRIBUTIONS_ALL	X			
RCV_SHIPMENT_HEADERS	X		X	
RCV_SHIPMENT_LINES	X			
RCV_TRANSACTIONS	X			
AP_INVOICES_ALL	X		X	
AP_INVOICE_DISTRIBUTIONS_ALL	X			
AP_INVOICE_PAYMENTS_ALL	X			
HHS_PMIS_ACKNOWLEDGEMENT_TBL	X	X		

4.7 Database Design

4.7.1 Obligations (Purchase Order) Extract Table

Table 22: Obligations Table Design

HHS_PMIS_OBLIGATIONS_STG_TBL

HHS_PMIS_OBLIGATIONS_STG_TBL is a custom staging table is use to store the Obligations Information before sending to Sunflower. All the custom validations and updates to the records are performed on this table.

Name	Format	Optional/ Required	Data Source / Default Value	Validation
RECORD IDENTIFIER	VARCHAR2 (2)	R	Constant	Constant 'TX'
RECORD ID	NUMBER	R	Select from Sequence	Select HHS_PMIS_PO_S.nextval from Dual
PO_HEADER_ID	NUMBER	R		PO_HEADERS_ALL.po_header_id.
PO_NUMBER	VARCHAR2 (50)	R		PO_HEADERS_ALL.segment1
LINE_NUM	NUMBER	R		PO_LINES_ALL.line_num
REVISION_NUM	NUMBER	O		PO_HEADERS_ALL.revision_num
REVISIED_DATE	DATE	O		PO_HEADERS_ALL.revisied_date
PO_DATE	DATE	R		PO_HEADERS_ALL..approved_date
PO_UPDATE_DATE	DATE	R		PO_HEADERS_ALL.last_update_date
INTERFACE_SOURCE	VARCHAR2 (150)	O		PO_HEADERS_ALL.attribute1
COMMENTS	VARCHAR2 (240)	O	PO Description	PO_HEADERS_ALL.comments
ENABLED_FLAG	CHAR (1)	R		PO_HEADERS_ALL.enabled_flag
TYPE_LOOKUP_CODE	VARCHAR2 (25)	R	Document Type (Default = STANDARD)	PO_HEADERS_ALL.type_lookup_code

Name	Format	Optional/ Required	Data Source / Default Value	Validation
VENDOR_ID	NUMBER	R		PO_HEADERS_ALL.vendor_id
VENDOR_NUMBER	VARCHAR2 (50)	R		PO_VENDORS.segment1
VENDOR_NAME	VARCHAR2 (150)	R		PO_VENDORS.vendor_name
VENDOR_SITE_ID	NUMBER	R		PO_HEADERS_ALL.vendor_site_id
VENDOR_SITE_CODE	VARCHAR2(50)	R		PO_VENDOR_SITES_ALL.vendor_site_code
SHIP_TO_LOCATION_ID	NUMBER	R		PO_HEADERS_ALL.ship_to_location_id
ADDRESS_LINE_1	VARCHAR2(150)	R		HR_LOCATIONS.address_line_1
ADDRESS_LINE_2	VARCHAR2(150)	O		HR_LOCATIONS.address_line_2
CITY	VARCHAR2(50)	R		HR_LOCATIONS.city
STATE	VARCHAR2(50)	R		HR_LOCATIONS.state
POSTAL_CODE	VARCHAR2(15)	R		HR_LOCATIONS.postal_code
TELEPHONE_NUMBER_1	VARCHAR2(15)	O		HR_LOCATIONS.telephone_number_1
ORG_ID	NUMBER	R		PO_HEADERS_ALL.org_id
HHS_AGENCY	VARCHAR2(3)	R	Issuing Office	ORG_ORGANIZATION_DEFINTIONS.organization_code
CURRENCY_CODE	VARCHAR2(3)	R		PO_HEADERS_ALL.currency_code
PSC_CONTRACT_NUMBER	VARCHAR2(50)	O		PO_HEADERS_ALL.attribute3
FDA_CONTRACT_NUMBER	VARCHAR2(50)	O		PO_HEADERS_ALL.attribute14
AUTHORIZATION_STATUS	VARCHAR2(25)	R		PO_HEADERS_ALL.authorization_status
APPROVED_FLAG	VARCHAR2(1)	O		PO_HEADERS_ALL.approved_flag
APPROVED_DATE	DATE	O		PO_HEADERS_ALL.approved_date
PO_CANCEL_FLAG	CHAR(1)	O	Document Cancel flag	PO_HEADERS_ALL.cancel_flag
PO_LINE_ID	NUMBER	R		PO_LINES_ALL.po_line_id
LINE_TYPE_ID	NUMBER	R		PO_LINES_ALL.line_type_id
LINE_TYPE	VARCHAR2(25)	R		PO_LINE_TYPES_ALL.line_type
CATEGORY_ID	NUMBER	R		PO_LINE_TYPES_ALL.category_id

Name	Format	Optional/ Required	Data Source / Default Value	Validation
CATEGORY	VARCHAR2(100)	R		MTL_CATEGORIES.segment1
CATEGORY_DESCR	VARCHAR2(240)	R		MTL_CATEGORIES.description
LINE_DESCRIPTION	VARCHAR2(240)	O		PO_LINES_ALL.description
UNIT_OF_MEASURE	VARCHAR2(50)	R		PO_LINES_ALL.unit_of_measure_lookup_code
UNIT_PRICE	NUMBER	R		PO_LINES_ALL.unit_price
QUANTITY	NUMBER	R		PO_LINES_ALL.quantity
LINE_CANCEL_FLAG	CHAR(1)	O		PO_LINES_ALL.line_cancel_flag
LINE_CANCEL_DATE	DATE	O		PO_LINES_ALL.line_cancel_date
LINE_CANCEL_REASON	VARCHAR2(240)	O		PO_LINES_ALL.line_cancel_reason
LINE_LOCATION_ID	NUMBER	R		PO_LINE_LOCATIONS_ALL.line_location_id
SHIPMENT_UPDATE_DATE	DATE	R		PO_LINE_LOCATIONS_ALL.last_update_date
SHIPMENT_QUANTITY	NUMBER	R		PO_LINE_LOCATIONS_ALL.shipment_quantity
QUANTITY_RECEIVED	NUMBER	O		PO_LINE_LOCATIONS_ALL.quantity_received
QUANTITY_CANCELLED	NUMBER	O		PO_LINE_LOCATIONS_ALL.quantity_cancelled
QUANTITY_BILLED	NUMBER	O		PO_LINE_LOCATIONS_ALL.quantity_billed
SHIPMENT_CANCEL_FLAG	CHAR(1)	O		PO_LINE_LOCATIONS_ALL.shipment_cancel_flag
SHIPMENT_CANCEL_DATE	DATE	O		PO_LINE_LOCATIONS_ALL.shipment_cancel_date
SHIPMENT_CANCEL_REASON	VARCHAR2 (240)	O		PO_LINE_LOCATIONS_ALL.shipment_cancel_reason
RECEIPT_REQUIRED_FLAG	CHAR(1)	R		PO_LINE_LOCATIONS_ALL.receipt_required_flag
MATCH_OPTION	CHAR(1)	R		PO_LINE_LOCATIONS_ALL.match_option
SHIPMENT_NUM	NUMBER	R		PO_LINE_LOCATIONS_ALL.shipment_number
PO_DISTRIBUTION_ID	NUMBER	R		PO_DISTRIBUTIONS_ALL.distribution_id
DISTRIBUTION_UPDATE_DATE	DATE	R		PO_DISTRIBUTIONS_ALL.last_update_date
SET_OF_BOOKS_ID	NUMBER	R		PO_DISTRIBUTIONS_ALL.set_of_books_id
CODE_COMBINATION_ID	NUMBER	R		PO_DISTRIBUTIONS_ALL.code_combination_id

Name	Format	Optional/ Required	Data Source / Default Value	Validation
CONCATENATED_SEGMENTS	VARCHAR2(500)	O		GL_CODE_COMBINATIONS_KFV.concatenated_segments
CAN	VARCHAR2(150)	R		concatenated_segments = SUBSTR (GL_CODE_COMBINATIONS.concatenated_segments,1,INSTR (GL_CODE_COMBINATIONS.concatenated_segments,'.',1,4)) || '00000.000000'|| SUBSTR (GL_CODE_COMBINATIONS.concatenated_segments,INSTR (GL_CODE_COMBINATIONS.concatenated_segments,'.',1,6)) join CANBAX XWALK History table and CANBAX XWALK table and pass po_creation_date
FISCAL_YEAR	VARCHAR2(100)	R		GL_CODE_COMBINATIONS.segment2
OBJECT_CLASS	VARCHAR2(100)	R	Between 31001 and 31999	GL_CODE_COMBINATIONS.segment5
ACCOUNT	VARCHAR2(100)	R	Not in 183XXX series	GL_CODE_COMBINATIONS.segment6
REQ_DISTRIBUTION_ID	NUMBER	O		PO_DISTRIBUTION_ALL.req_distribution_id
REQUISITION_NUMBER	VARCHAR2(50)	O	Derive if REQ_DISTRIBUTION_ID is not null	PO_REQUISITION_HEADERS_ALL.segment1
REQUISITION_LINE_NUM	NUMBER	O	Derive if REQ_DISTRIBUTION_ID is not null	PO_REQUISITION_LINES_ALL.line_num
DISTRIBUTION_QTY	NUMBER	R		PO_DISTRIBUTIONS_ALL.quantity_orderd
DISTRIBUTION_CANCEL_QTY	NUMBER	O		PO_DISTRIBUTIONS_ALL.quantity_cancelled
DISTRIBUTION_NUM	NUMBER	R		PO_DISTRIBUTIONS_ALL.distribution_num
DISTRIBUTION_CANCEL_FLAG	CHAR	O		PO_DISTRIBUTIONS_ALL..distribution_cancel_flag
DISTRIBUTION_CANCEL_DATE	DATE	O		PO_DISTRIBUTIONS_ALL..distribution_cancel_date
FDA_TAG_NUMBER	VARCHAR2(150)	O		PO_DISTRIBUTIONS_ALL.attribute10
FDA_AGENCY_TAG_NUMBER	VARCHAR2(150)	O		PO_DISTRIBUTIONS_ALL.attribute11
PROJECT_ID	NUMBER	O		PO_DISTRIBUTIONS_ALL.project_id
PROJECT_NUMBER	VARCHAR2(50)	O	Derive if PROJECT_ID is not null	PA_PROJECTS_ALL.segment1
PROJECT_NAME	VARCHAR2(150)	O	Derive if PROJECT_ID is not null	Derive Project Name PA_PROJECTS_ALL.name

Name	Format	Optional/ Required	Data Source / Default Value	Validation
TASK_ID	NUMBER	O		PO_DISTRIBUTIONS_ALL.task_id
TASK_NUMBER	VARCHAR2(50)	O	Derive if TASK_ID is not null	Derive Task Number PA_TASKS.task_number
TASK_NAME	VARCHAR2(150)	O	Derive if TASK_ID is not null	Derive Task Name PA_TASKS_ALL.task_name
EXPENDITURE_TYPE	VARCHAR2(50)	O		PO_DISTRIBUTIONS_ALL.expenditure_type
EXPENDITURE_ORGANIZATION_ID	NUMBER	O		PO_DISTRIBUTIONS_ALL.expenditure_org_id
EXPENDITURE_ITEM_DATE	DATE	O		PO_DISTRIBUTIONS_ALL.expenditure_item_date
USSGL_TRN_CODE	VARCHAR2(10)	O		PO_DISTRIBUTIONS_ALL.ussgl_transaction_code
GPRA_CODE	VARCHAR2(10)	O		Select segment3 from GL_CODE_COMBINATIONS. For PSC, Last 3 digits from Segment3 For CDC, select segment3 from bacs, join this table to FND_FLEX_VALUES and find out the GPRA code in DFF. HIS, Same process as IHS For FDA, Not Defined Yet.
ACCOUNTING_POINT	VARCHAR2(4)	O		ORG_ORGANIZATION_DEFINITIONS.organization_id based on ship_to_organization_id from PO_LINE_LOCATIONS_ALL
FUND_CANCEL_DATE	DATE	O		Select fund_cancel_date from FV_FUND_PARAMETERS where fund_value = segment1 from GL_CODE_COMBINATIONS
RECORD_STATUS	VARCHAR2(15)	R		For new run then ‘NEW’, for successfully extracted ‘PROCESSED’, for error records ‘ERROR’
SUNFLOWER_STATUS	VARCHAR2(15)	O		For successful records ‘PROCESSED’, for error records ‘ERROR’
ERROR_CODE	VARCHAR2(25)	O		
ERROR_DESCRIPTION	VARCHAR2(250)	O		
REQUEST_ID	NUMBER	O		
FILE_NAME	VARCHAR2(100)	O		
CREATION_DATE	DATE	R	Standard WHO Column	Current System Date

Name	Format	Optional/ Required	Data Source / Default Value	Validation
CREATED_BY	NUMBER	R	Standard WHO Column	Select FND_GLOBAL.user_id from Dual
LAST_UPDATED_BY	NUMBER	R	Standard WHO Column	Select FND_GLOBAL.user_id from Dual
LAST_UPDATE_DATE	DATE	R	Standard WHO Column	Current System Date
LAST_UPDATE_LOGIN	NUMBER	R	Standard WHO Column	Select FND_GLOBAL.login_id from Dual

4.7.2 Receipts Extract Table

Table 23: Receipts Table Design

HHS_PMIS_RECEIPTS_STG_TBL

HHS_PMIS_RECEIPTS_STG_TBL is a custom staging table is use to store the Receipts Information before sending to Sunflower. All the custom validations and updates to the records are performed on this table.

Field Name	Data Type	Required/ Optional	Data Source/ Default Value	Validations
RECORD_IDENTIFIER	VARCHAR2(2)	R	Constant	Constant = 'TX'
RECORD_ID	NUMBER	R	Select from Sequence	Select HHS_PMIS_PO_S.nextval from Dual
SHIPMENT_HEADER_ID	NUMBER	R		RCV_SHIPMENT_LINES.shipment_header_id
SHIPMENT_LINE_ID	NUMBER	R		RCV_SHIPMENT_LINES.shipment_line_id
PO_HEADER_ID	NUMBER	R		RCV_SHIPMENT_LINES.po_header_id
PO_NUMBER	VARCHAR2(50)	R		PO_HEADERS_ALL.segment1
PO_REVISION_NUM	NUMBER	R		PO_HEADERS_ALL.revision_num
PO_LINE_ID	NUMBER	R		RCV_SHIPMENT_LINES.po_line_id
PO_LINE_NUM	NUMBER	R		PO_LINES_ALL.line_num
SHIP_TO_ORG_ID	NUMBER	R		RCV_SHIPMENT_HEADERS.ship_to_org_id

CATEGORY_ID	NUMBER	R		RCV_SHIPMENT_LINES.category_id
LINE_LOCATION_ID	NUMBER	R		RCV_SHIPMENT_LINES.po_line_location_id
SHIPMENT_NUM	NUMBER	R		PO_LINE_LOCATIONS_ALL.shipment_num
SET_OF_BOOKS_ID	NUMBER	R		GL_SETS_OF_BOOKS.set_of_books_id
ITEM_ID	NUMBER	O		RCV_SHIPMENT_LINES.item_id
CHARGE_ACCOUNT_ID	NUMBER	R		RCV_SHIPMENT_LINES.charge_account_id
PO_DISTRIBUTION_ID	NUMBER	R		RCV_SHIPMENT_LINES.po_distribution_id
CAN	VARCHAR2(25)	R	Derive if PO_DISTRIBUTION_ID is not null	concatenated_segments = SUBSTR (GL_CODE_COMBINATIONS.concatenated_segments,1,INSTR (GL_CODE_COMBINATIONS.concatenated_segments,'.',1,4)) || '00000.000000'|| SUBSTR (GL_CODE_COMBINATIONS.concatenated_segments,INSTR (GL_CODE_COMBINATIONS.concatenated_segments,'.',1,6)) join CANBAX XWALK History table and CANBAX XWALK table and pass receipt_creation_date
FISCAL_YEAR	VARCHAR2(4)	R	Derive if PO_DISTRIBUTION_ID is not null	GL_CODE_COMBINATIONS.segment2
OBJECT_CLASS	VARCHAR(5)	R	Derive if PO_DISTRIBUTION_ID is not null	GL_CODE_COMBINATIONS.segment5
ACCOUNT	VARCHAR2(6)	R	Derive if PO_DISTRIBUTION_ID is not null	GL_CODE_COMBINATIONS.segment6
USSGL_ACCOUNT	VARCHAR(7)	R		
CONCATENATED_SEGMENTS	VARCHAR2(150)	R		GL_CODE_COMBINATIONS_KFV.concatenated_segments
DISTRIBUTION_NUM	NUMBER	R		PO_DISTRIBUTIONS_ALL.distribution_num
RECEIPT_NUM	VARCHAR2(30)	R		RCV_SHIPMENT_HEADERS.receipt_num
RECEIPT_DATE	DATE	R		RCV_TRNSACTIONS.transaction_date

COMMENTS	VARCHAR2(240)	O		RCV_SHIPMENT_HEADERS.comments
RECEIPT_LINE_NUM	NUMBER	R		RCV_SHIPMENT_LINES.line_num
RECEIPT_QUANTITY	NUMBER	R		RCV_TRANSACTIONS.quantity
UNIT_PRICE	NUMBER	R		RCV_TRANSACTIONS.po_unit_price
ITEM_DESCRIPTION	VARCHAR2(240)	O		RCV_SHIPMENT_LINES.item_description
SHIPMENT_LINE_STATUS	VARCHAR2(25)	R		RCV_SHIPMENT_LINES.shipment_line_status_code
LINE_COMMENTS	VARCHAR2(240)	O		RCV_SHIPMENT_LINES.comments
TRANSACTION_TYPE	VARCHAR2(25)	R		RCV_TRANACTIONS.transaction_type
UNIT_OF_MEASURE	VARCHAR2(25)	R		RCV_TRANSACTIONS.uom_code
CURRENCY_CODE	VARCHAR2(3)	R		RCV_TRANSACTIONS.currency_code
QUANTITY_SHIPPED	NUMBER	O		RCV_TRANSACTIONS.quantity_shipped
QUANTITY_RECEIVED	NUMBER	O		RCV_TRANSACTIONS.quantity_received
VENDOR_ID	NUMBER	R		RCV_TRANSACTIONS.vendor_id
VENDOR_NUMBER	VARCHAR2(25)	R		PO_VENDORS.segment1
VENDOR_NAME	VARCHAR2(150)	R		PO_VENDORS.vendor_name
VENDOR_SITE_ID	NUMBER	R		RCV_TRANSACTIONS.vendor_site_id
VENDOR_SITE_CODE	VARCHAR2(50)	R		PO_VENDOR_SITES.vendo_site_code
ORG_ID	NUMBER	R		Select FND_PROFILE.value ('ORG_ID') from Dual
HHS_AGENCY_CODE	VARCHAR2(3)	R		ORG_ORGANIZATION_DEFINITION.organization_code
GPRA_CODE	VARCHAR2(10)	O		**Select segment3 from GL_CODE_COMBINATIONS.** **For PSC,** Last 3 digits from Segment3 For CDC, select segment3 from bacs, join this table to FND_FLEX_VALUES and find out the GPRA code in DFF. HIS, Same process as CDC For FDA, Not Defined Yet.
ACCOUNTING_POINT	VARCHAR2(4)	O	Applicable for IHS only	ORG_ORGANIZATION_DEFINITIONS.organization_id based on ship_to_organization_id from PO_LINE_LOCATIONS_ALL
FUND_CANCE_DATE	DATE	O		Select fund_cancel_date from FV_FUND_PARAMETERS where fund_value = segment1 from GL_CODE_COMBINATIONS
RECORD_STATUS	VARCHAR2(15)	R	Default = 'NEW'	For new run then 'NEW', for successfully extracted 'PROCESSED', for error records 'ERROR'
SUNFLOWER_STATUS	VARCHAR2(15)	O		For successful records 'PROCESSED', for error records 'ERROR'

ERROR_CODE	VARCHAR2(25)	O		
ERROR_MESSAGE	VARCHAR2(250)	O		
REQUEST_ID	NUMBER	O		
CREATION_DATE	DATE	R	Standard WHO Columns	Current System Date
CREATED_BY	NUMBER	R	Standard WHO Columns	Select FND_GLOBAL.user_id FROM dual
LAST_UDPATED_BY	NUMBER	R	Standard WHO Columns	Select FND_GLOBAL.user_id FROM dual
LAST_UPDATE_DATE	DATE	R	Standard WHO Columns	Current System Date
LAST_UPDATE_LOGIN	NUMBER	R	Standard WHO Columns	Select FND_GLOBAL.login_id FROM dual

4.7.3 Invoices Extract Table

Table 24: Invoices Table Design

HHS_PMIS_INVOICES_STG_TBL

HHS_PMIS_INVOICES_STG_TBL is a custom staging table is use to store the Invoices Information before sending to Sunflower. All the custom validations and updates to the records are performed on this table.

Field Name	Data type	Required/ Optional	Data Source / Default Value	Comments
RECORD IDENTIFIER	VARCHAR2(2)	R	Constant	Value will be 'TX'
RECORD ID	NUMBER	R	Select from Sequence	Select HHS_PMIS_INV_S.nextval from Dual
INVOICE_ID	NUMBER	R		AP_INVOICES_ALL.invoice_id
INVOICE_NUM	VARCHAR2(50)	R		AP_INVOICES_ALL.invoice_num
INVOICE_DATE	DATE	R		AP_INVOICES_ALL.invoice_date
CURRENCY_CODE	VARCHAR2(3)	R		AP_INVOICES_ALL.invoice_currency
INVOICE_SOURCE	VARCHAR2(50)	R		AP_INVOICES_ALL.source
INVOICE_TYPE	VARCHAR2(25)	R		AP_INVOICES_ALL.invoice_type_lookup_code

SET_OF_BOOKS_ID	NUMBER	R		AP_INVOICES_ALL.set_of_books_id
VENDOR_ID	NUMBER	R		AP_INVOICES_ALL.vendor_id
VENDOR_NUMBER	VARCHAR2(50)	R		PO_VENDORS.vendor_name
VENDOR_NAME	VARCHAR2(250)	R		PO_VENDORS.vendor_name
VENDOR_SITE_ID	NUMBER	R		AP_INVOICES_ALL.vendor_site_id
VENDOR_SITE_CODE	VARCHAR2(50)	R		PO_VENDOR_SITES.vendor_site_code
PAYMENT_STATUS_FLAG	VARCHAR2(1)	R		AP_INVOICES_ALL.payment_status_flag
ATTRIBUTE1	VARCHAR2(150)	O		AP_INVOICES_ALL.attribute1
DISTRIBUTION_LINE_NUM	NUMBER	R		AP_INVOICE_DISTRIBUTIONS_ALL.distribution_line_num
ASSET_TRACKING_FLAG	VARCHAR2(1)	O		AP_INVOICE_DISTRIBUTIONS_ALL.asset_tracking_flag
CODE_COMBINATION_ID	NUMBER	R		AP_INVOICE_DISTRIBUTIONS_ALL.dist_code_combination_id
OBJECT_CLASS	VARCHAR2(150)	R		GL_CODE_COMBINATIONS.segment5
FISCAL_YEAR	VARCHAR2(150)	R		GL_CODE_COMBINATIONS.segment2
ACCOUNT	VARCHAR2(150)	R		GL_CODE_COMBINATIONS.segment6
CAN	VARCHAR2(150)	R		concatenated_segments = SUBSTR (GL_CODE_COMBINATIONS.concatenated_segments,1,INSTR (GL_CODE_COMBINATIONS.concatenated_segments,'.',1,4)) || '00000.000000'|| SUBSTR (GL_CODE_COMBINATIONS.concatenated_segments,INSTR (GL_CODE_COMBINATIONS.concatenated_segments,'.',1,6)) join CANBAX XWALK History table and CANBAX XWALK table and pass invoice_creation_date
CONCATENATED_SEGMENTS	VARCHAR2(150)	R		GL_CODE_COMBINATIONS_KFV.concatenated_segments
DESCRIPTION	VARCHAR2(240)	O		AP_INVOICE_DISTRIBUTIONS_ALL.description
QUANTITY_INVOICED	NUMBER	O		AP_INVOICE_DISTRIBUTIONS_ALL.quantity_invoiced
UNIT_PRICE	NUMBER	R		AP_INVOICE_DISTRIBUTIONS_ALL.unit_price
FDA_TAG_NUMBER	VARCHAR2(150)	O	Applicable only for FDA	AP_INVOICE_DISTRIBUTIONS_ALL.attribute10
FDA_AGENCY_TAG_NUMBER	VARCHAR2(150)	O	Applicable only for FDA	AP_INVOICE_DISTRIBUTIONS_ALL.attribute11
MACCS_DOC_KEY	VARCHAR2(150)	O	Applicable only for PSC	AP_INVOICE_DISTRIBUTIONS_ALL.attribute13
PROJECT_ID	NUMBER	O		AP_INVOICE_DISTRIBUTIONS_ALL.project_id

PROJECT_NUMBER	VARCHAR2(25)	O	Derive if PRJECT_ID is not null	PA_PROJECTS_ALL.segment1
PROJECT_NAME	VARCHAR2(50)	O	Derive if PRJECT_ID is not null	PA_PROJECTS_ALL.name
TASK_ID	NUMBER	O		AP_INVOICE_DISTRIBUTIONS_ALL.task_id
TASK_NUMBER	VARCHAR2(50)	O	Derive if TASK_ID is not null	PA_TASKS.task_number
TASK_NAME	VARCHAR2(50)	O	Derive if TASK_ID is not null	PA_TASKS.task_name
EXPENDITURE_ITEM_DATE	DATE	O		AP_INVOICE_DISTRIBUTIONS_ALL.expenditure_item_date
EXPENDITURE_ORG_ID	NUMBER	O		AP_INVOICE_DISTRIBUTIONS_ALL.expenditure_org_id
EXPENDITURE_TYPE	VARCHAR2(50)	O		AP_INVOICE_DISTRIBUTIONS_ALL.expenditure_type
USSGL_ACCOUNT	VARCHAR2(7)	O		AP_INVOICE_DISTRIBUTIONS_ALL.ussgl_transaction_code
APPROVAL_STATUS	VARCHAR2(150)	R		AP_INVOICES_ALL.approval_status
ORG_ID	NUMBER	R		AP_INVOICES_ALL.org_id
HHS_AGENCY	VARCHAR2(3)	R		ORG_ORGANIZATION_DEFINITION.organization_code
INVOICE_PAYMENT_ID	NUMBER	R		AP_INVOICE_PAYMENTS_ALL.invoice_payment_id
PAYMENT_AMOUNT	NUMBER	R		AP_INVOICE_PAYMENTS_ALL.amount
PAYMENT_NUMBER	NUMBER	R		AP_INVOICE_PAYMENTS_ALL.payment_number
CHECK_ID	NUMBER	R		AP_CHECKS_ALL.check_id
CHECK_DATE	DATE	R		AP_CHECKS_ALL.check_date
CHECK_NUMBER	VARCHAR2(50)	R		AP_CHECKS_ALL.check_number
PO_DISTRIBUTION_ID	NUMBER	O		AP_INVOICE_DISTRIBUTIONS_ALL.po_distribution_id
PO_NUMBER	VARCHAR2(50)	O	Applicable only for CDC, FDA, HIS	PO_HEADERS_ALL.segment1
PO_LINE_NUMBER	NUMBER	O	Applicable only for CDC, FDA, HIS	PO_LINES_ALL.line_num
SHIPMENT_LINE_NUMBER	NUMBER	O	Applicable only for CDC, FDA, HIS	PO_LINE_LOCATIONS_ALL.line_location_id
PO_DISTRIBUTION_NUMBER	NUMBER	O	Applicable only for CDC, FDA, HIS	PO_DISTRIBUTIONS_ALL.distribution_line_num
INVOICE_DISTRIBUTION_ID	NUMBER	R		AP_INVOICE_DISTRIBUTIONS_ALL.invoice_distribution_id
GPRA_CODE	VARCHAR2(10)	O		Select segment3 from GL_CODE_COMBINATIONS. For PSC, Last 3 digits from Segment3 For CDC, select segment3 from bacs, join this table to FND_FLEX_VALUES and find out the GPRA code in DFF.

				HIS, Same process as CDC For FDA, Not Defined Yet.
ACCOUNTING_POINT	VARCHAR2(4)	O	Applicable for IHS Only	ORG_ORGANIZATION_DEFINITIONS.organization_id based on ship_to_organization_id from PO_LINE_LOCATIONS_ALL
FUND_CANCEL_DATE	DATE	O		Select fund_cancel_date from FV_FUND_PARAMETERS where fund_value = segment1 from GL_CODE_COMBINATIONS
RECORD_STATUS	VARCHAR2(15)	R	Default = 'NEW'	For new run then 'NEW', for successfully extracted 'PROCESSED', for error records 'ERROR'
SUNFLOWER_STATUS	VARCHAR2(15)	O		For successful records 'PROCESSED', for error records 'ERROR'
ERROR_CODE	VARCHAR2(25)	O		
ERROR_MESSAGE	VARCHAR2(250)	O		
REQUEST_ID	NUMBER	O		
CREATION_DATE	DATE	R	Standard WHO Columns	Current System Date
CREATED_BY	NUMBER	R	Standard WHO Columns	Select FND_GLOBAL.user_id FROM dual
LAST_UPDATED_BY	NUMBER	R	Standard WHO Columns	Select FND_GLOBAL.user_id FROM dual
LAST_UPDATE_DATE	DATE	R	Standard WHO Columns	Current System Date
LAST_UPDATE_LOGIN	NUMBER	R	Standard WHO Columns	Select FND_GLOBAL.login_id FROM dual

4.7.4 Acknowledgement Extract Table

Table 25: Acknowledgement Table Design

HHS_PMIS_ACKNOWLEDGEMENT_TBL

The following HHS_PMIS_ERROR_ACK_TBL table shows the mapping of error acknowledgement table fields to the error acknowledgement flat file that will be received from Sunflower. Following table is used for only outbound interfaces.

Column Name	Data Format	Required /Optional	Date Source / Default Value	Validation

Column Name	Data Format	Required /Optional	Date Source / Default Value	Validation
RECORD_ID	NUMBER	R		
ERROR_SOURCE	VARCHAR2 (25)	R		'PO' if PO Interface, RCT if Receipt Interface, INV for Invoice Interface
DATA FILE NAME	VARCHAR2(100)	O		
HHS_AGENCY	VARCHAR2 (3)	R		Should be in 'CDC','FDA','PSC','IHS'
DOCUMENT_NUMBER	VARCHAR2 (50)	R		For Outbound, PO Number, Receipt Number or Invoice Number based on extract and for Inbound, General Ledger Batch Name
DOCUMENT_LINE_NUM	NUMBER	R		For Outbound, PO Line Number, Receipt Line Number or Invoice Line Number based on extract and for inbound, Journal Line Number
ACCOUNTING_POINT	VARCHAR2(4)	O		
ERROR_CODE	VARCHAR2 (15)	R		
ERROR_DESCRIPTION	VARCHAR2 (250)	O		
REQUEST_ID	NUMBER	O		
RECORD_STATUS	CHAR(1)	R		
PROCESS_STATUS	CHAR(1)	R		For new run then 'NEW', for successfully extracted 'PROCESSED'.
CREATED_BY	NUMBER	R		Select FND_GOLBAL.user_id from dual
CREATION_DATE	DATE	R	Default = SYSDATE	Current System Date
LAST_UPDATED_BY	NUMBER	R		Select FND_GOLBAL.user_id from dual
LAST_UPDATE_DATE	DATE	R	Default = SYSDATE	Current System Date
LAST_UPDATE_LOGIN	NUMBER	R		Select FND_GOLBAL.login_id from dual

4.7.5 General Ledger Staging Table

Table 26: General Ledger Table Design

HHS_PMIS_GL_INTERFACE_STG_TBL

The following HHS_PMIS_GL_INTERFACE_STG_TBL is a replica of Oracle Standard GL_INTERFACE table along with some extra columns requires for the validation purpose. This table shows the mapping of General Ledger open interface table fields to the flat file. For inbound interfaces, the Table and Column identify the open interface table columns and the validation column identifies corresponding validation rules and source of data.

Interface Table Column	Data Format	Required (R) /Optional (O) /	Source Of Data / Default Value / Derived Value	Flat File column	Validation
RECORD_IDENTIFIER	VARCHAR2 (2)	R	Flat File	Record Identifier	Constant = 'TX'
RECORD_ID	NUMBER	R	Derived value by sequence	Record ID	
STATUS	VARCHAR2 (50)	R	Default Value		Default = 'NEW'
SET_OF_BOOKS_NAME	VARCHAR2 (3)	R	Flat File	Set of books Name	
SET_OF_BOOKS_ID	NUMBER	R	Derived Value		Should be exists in GL_SETS_OF_BOOKS.set_of_books_id
ACCOUNTING_DATE	DATE	R	Default Value		Accounting_date=Transaction_date
CURRENCY_CODE	VARCHAR2 (15)	R	Default Value	Currency Code	Default = 'USD'
DATE_CREATED	DATE	R	Standard WHO column		SYSDATE
CREATED_BY	NUMBER	R	Standard WHO column		Select FND_GOLBAL.user_id from dual
ACTUAL_FLAG	CHAR (1)	R	Default Value		Should be 'A'
USER_JE_CATEGORY_NAME	VARCHAR2 (25)	R		Category Name	Should be exists in GL_JE_CATEGORIES.je_category_name
USER_JE_SOURCE_NAME	VARCHAR2(25)	R		Source Name	Should be exists in GL_JE_SOURCES.je_source_name

Interface Table Column	Data Format	Required (R) /Optional (O) /	Source Of Data / Default Value / Derived Value	Flat File column	Validation
CURRENCY_CONVERSION_DATE	DATE	O			
ENCUMBRANCE_TYPE_ID	NUMBER	O			
BUDGET_VERSION_ID	NUMBER	O			
USER_CURRENCY_CONVERSION_TYPE	VARCHAR2 (30)	O			
CURRENCY_CONVERSION_RATE	NUMBER	O			
AVERAGE_JOURNAL_FLAG	VARCHAR2 (1)	O			
ORIGINATING_BAL_SEG_VALUE	VARCHAR2(25)	O			
SEGMENT1	VARCHAR2(25)	O			
SEGMENT2	VARCHAR2(25)	R	Flat File	Fiscal Year	Should be exists in HHS_CBX_XWALK_TBL.segment2
SEGMENT3	VARCHAR2(25)	O			
SEGMENT4	VARCHAR2(25)	O			
SEGMENT5	VARCHAR2(25)	R	Flat File	Object Class	Should be exists in HHS_CBX_XWALK_TBL.segment5
SEGMENT6	VARCHAR2(25)	R	Flat File	Account	Should be exists in HHS_CBX_XWALK_TBL.segment6
SEGMENT7	VARCHAR2(25)	O			
SEGMENT8	VARCHAR2(25)	O			
SEGMENT9	VARCHAR2(25)	O			

Interface Table Column	Data Format	Required (R) /Optional (O) /	Source Of Data / Default Value / Derived Value	Flat File column	Validation
SEGMENT10	VARCHAR2(25)	O			
SEGMENT11	VARCHAR2(25)	O			
SEGMENT12	VARCHAR2(25)	O			
SEGMENT13	VARCHAR2(25)	O			
SEGMENT14	VARCHAR2(25)	O			
SEGMENT15	VARCHAR2(25)	O			
SEGMENT16	VARCHAR2(25)	O			
SEGMENT17	VARCHAR2(25)	O			
SEGMENT18	VARCHAR2(25)	O			
SEGMENT19	VARCHAR2(25)	O			
SEGMENT20	VARCHAR2(25)	O			
SEGMENT21	VARCHAR2(25)	O			
SEGMENT22	VARCHAR2(25)	O			
SEGMENT23	VARCHAR2(25)	O			
SEGMENT24	VARCHAR2(25)	O			
SEGMENT25	VARCHAR2(25)	O			
SEGMENT26	VARCHAR2(25)	O			
SEGMENT27	VARCHAR2(25)	O			
SEGMENT28	VARCHAR2(25)	O			
SEGMENT29	VARCHAR2(25)	O			
SEGMENT30	VARCHAR2(25)	O			

Interface Table Column	Data Format	Required (R) /Optional (O) /	Source Of Data / Default Value / Derived Value	Flat File column	Validation
ENTERED_DR	NUMBER	O	Flat File	Entered Debit	Required if Entered Credit amount is null
ENTERED_CR	NUMBER	O	Flat File	Entered Credit	Required if Entered Debit amount is null
ACCOUNTED_DR	NUMBER	O			
ACCOUNTED_CR	NUMBER	O			
TRANSACTION_DATE	DATE	R	Flat File	Transaction Date	Should not be NULL
REFERENCE1	VARCHAR2 (100)	O	Flat File	Batch Name	Should be (Source Name + Category Name + Period Name)
REFERENCE2	VARCHAR2 (240)	O	Flat file	Batch Description	
REFERENCE3	VARCHAR2 (100)	O			
REFERENCE4	VARCHAR2 (100)	O			
REFERENCE5	VARCHAR2 (240)	O	Flat File	Document Reference Number	Should be PO Number
REFERENCE6	VARCHAR2 (100)	O			
REFERENCE7	VARCHAR2 (100)	O			
REFERENCE8	VARCHAR2 (100)	O			
REFERENCE9	VARCHAR2 (100)	O			
REFERENCE10	VARCHAR2 (240)	O	Flat File	Line Description	
REFERENCE11	VARCHAR2 (100)	O			
REFERENCE12	VARCHAR2 (100)	O			
REFERENCE13	VARCHAR2 (100)	O			
REFERENCE14	VARCHAR2 (100)	O			
REFERENCE15	VARCHAR2 (100)	O			

Interface Table Column	Data Format	Required (R) /Optional (O) /	Source Of Data / Default Value / Derived Value	Flat File column	Validation
REFERENCE16	VARCHAR2 (100)	O			
REFERENCE17	VARCHAR2 (100)	O			
REFERENCE18	VARCHAR2 (100)	O			
REFERENCE19	VARCHAR2 (100)	O			
REFERENCE20	VARCHAR2 (240)	O			
REFERENCE21	VARCHAR2 (240)	O			
REFERENCE22	VARCHAR2 (240)	O			
REFERENCE23	VARCHAR2 (240)	O			
REFERENCE24	VARCHAR2 (240)	O			
REFERENCE25	VARCHAR2 (240)	O			
REFERENCE26	VARCHAR2 (240)	O			
REFERENCE27	VARCHAR2 (240)	O			
REFERENCE28	VARCHAR2 (240)	O			
REFERENCE29	VARCHAR2 (240)	O			
REFERENCE30	VARCHAR2 (240)	O			
JE_BATCH_ID	NUMBER	O			
PERIOD_NAME	VARCHAR2 (15)	O			
JE_HEADER_ID	NUMBER	O			
JE_LINE_NUM	NUMBER	O			
CHART_OF_ACCOUNTS_ID	NUMBER	O			

Interface Table Column	Data Format	Required (R) /Optional (O) /	Source Of Data / Default Value / Derived Value	Flat File column	Validation
FUNCTIONAL_CURRENCY_CODE	VARCHAR2(15)	O			
CODE_COMBINATION_ID	NUMBER	R	Derived Value		Derived code_combination_id from HHS_CBX_XWALK_TBL by passing Fiscal Year, Object Class and Account as a parameter.
DATE_CREATED_IN_GL	DATE	O			
WARNING_CODE	VARCHAR2	O			
STATUS_DESCRIPTION	VARCHAR2(240)	O			
STAT_AMOUNT	NUMBER	O			
GROUP_ID	NUMBER (15)	O			
REQUEST_ID	NUMBER (15)	O			
SUBLEDGER_DOC_SEUENCE_ID	NUMBER	O			
SUBLEDGER_DOC_SEQUENCE_VALUE	NUMBER	O			
ATTRIBUTE1	VARCHAR2(150)	O	Flat File	HHS Elimination Code	Should 'HHS Elimination Code' if Vendor is Federal
ATTRIBUTE2	VARCHAR2 (150)	O			
GL_SL_LINK_ID	NUMBER	O			
GL_SL_LINK_TABLE	VARCHAR2(30)	O			
ATTRIBUTE3	VARCHAR2(150)	O			
ATTRIBUTE4	VARCHAR2(150)	O	Flat File	FACTS Trading Partner Code	Should 'HHS Elimination Code' if Vendor is Federal

Interface Table Column	Data Format	Required (R) /Optional (O) /	Source Of Data / Default Value / Derived Value	Flat File column	Validation
ATTRIBUTE5	VARCHAR2(150)	O			
ATTRIBUTE6	VARCHAR2(150)	O			
ATTRIBUTE7	VARCHAR2(150)	O			
ATTRIBUTE8	VARCHAR2(150)	O			
ATTRIBUTE9	VARCHAR2(150)	O			
ATTRIBUTE10	VARCHAR2(150)	O	Flat File	Attribute10	Attribute10 =Transaction Date
ATTRIBUTE11	VARCHAR2(150)	O			
ATTRIBUTE12	VARCHAR2(150)	O			
ATTRIBUTE13	VARCHAR2(150)	O			
ATTRIBUTE14	VARCHAR2(150)	O			
ATTRIBUTE15	VARCHAR2(150)	O			
ATTRIBUTE16	VARCHAR2(150)	O			
ATTRIBUTE17	VARCHAR2(150)	O			
ATTRIBUTE18	VARCHAR2(150)	O			
ATTRIBUTE19	VARCHAR2(150)	O			
ATTRIBUTE20	VARCHAR2(150)	O			
CONTEXT	VARCHAR2(150)	O			
CONTEXT2	VARCHAR2(150)	O			
INVOICE_DATE	DATE	O			
TAX_CODE	VARCHAR2(15)	O			
INVOICE_IDENTIFIER	VARCHAR2 (20)	O			

Interface Table Column	Data Format	Required (R) /Optional (O) /	Source Of Data / Default Value / Derived Value	Flat File column	Validation
INVOICE_AMOUNT	NUMBER	O			
CONTEXT3	VARCHAR2(150)	O			
USSGL_TRANSACTION_CODE	VARCHAR2 (30)	O			
DESCR_FLEX_ERROR_MESSAGE	VARCHAR2 (240)	O			
JGZZ_RECON_REF	VARCHAR2 (240)	O			
REFERENCE_DATE	DATE	O			
CAN	Varchar2 (25)	R	Flat File	CAN	concatenated_segments = SUBSTR (GL_CODE_COMBINATIONS.concatenated_segments,1,INSTR (GL_CODE_COMBINATIONS.concatenated_segments,'.',1,4)) || '00000.000000'|| SUBSTR (GL_CODE_COMBINATIONS.concatenated_segments,INSTR (GL_CODE_COMBINATIONS.concatenated_segments,'.',1,6)) join CANBAX XWALK History table and CANBAX XWALK table and pass journal transaction date
RECORD_STATUS	VARCHAR2 (15)	R	Constant		For new run then 'NEW', for successfully extracted 'PROCESSED', for error records 'ERROR'
ERROR_CODE	VARCHAR2(25)	O			
ERROR_MESSAGE	VARCHAR2(250)	O			
REQUEST_ID	NUMBER	O			
GPRA_CODE	VARCHAR2(10)	O			
ACCOUNTING_POINT	VARCHAR2(3)	O			Mandatory for IHS only

4.7.6 Control Staging Table

Table 27: Control Table Design

HHS_GLOBAL_INTERFACE_CTL_TBL

The following HHS_GLOBAL_INTERFACE_CTL_TBL table shows the mapping of acknowledgement table fields to the acknowledgement flat file that will be received from Sunflower. Following table is used for only outbound interfaces.

Column Name	Data Format	Required /Optional	Date Source / Default Value	Validation
RECORD_ID	NUMBER	R	Sequence	Select HHS_GLOBAL_INTERFACE_S.nextval From Dual
RECORD_IDENTIFIER	VARCHAR2 (2)	R		Value should be 'TH' or 'TT'
INTERFACE_NAME	VARCHAR2(150)	R	Constant	'PMIS INTERFACE'
REQUEST_ID	NUMBER	R	Derived Value	
FILE_NAME	VARCHAR2(200)	O		Current File Name
HHS AGENCY CODE	VARCHAR2 (3)	O	Flat File	Value should be in ('CDC','FDA','PSC','IHS')
DATE_TIME	NUMBER	O	Flat File	Check NOT NULL
LOCATION	NUMBER	O	Flat File	
HEADER_COUNT	NUMBER	O	Flat File	Check NOT NULL
LINE_COUNT	NUMBER	O	Flat File	Check NOT NULL
TOTAL_AMOUNT	NUMBER	O	Flat File	Check NOT NULL
MATCH_FLAG	CHAR (1)	O		
ATTRIBUTE1	VARCHAR2 (150)	O		
ATTRIBUTE2	VARCHAR2 (150)	O		
ATTRIBUTE3	VARCHAR2 (150)	O		
ATTRIBUTE4	VARCHAR2 (150)	O		
ATTRIBUTE5	VARCHAR2 (150)	O		

Column Name	Data Format	Required /Optional	Date Source / Default Value	Validation
CREATED_BY	NUMBER	R		Select FND_GOLBAL.user_id from dual
CREATION_DATE	DATE	R	Default = SYSDATE	Current System Date
LAST_UPDATE_DATE	DATE	R	Default = SYSDATE	
LAST_UPDATED_BY	NUMBER	R		Select FND_GOLBAL.user_id from dual
LAST_UPDATE_LOGIN	NUMBER	R		Select FND_GOLBAL.login_id from dual

4.7.7 Custom Sequences

Table 28: Error Handling

Sequence Name	Used By	Comments
HHS_PMIS_PO_S	PO Extract	Generate RECORD_ID based on HHS_PMIS_PO_S.nextval
HHS_PMIS_RCT_S	Receipt Extract	Generate RECORD_ID based on HHS_PMIS_RCT_S.nextval
HHS_PMIS_INV_S	Invoice Extract	Generate RECORD_ID based on HHS_PMIS_INV_S.nextval
HHS_PMIS_GL_S	GL Inbound Interface	Generate RECORD_ID based on HHS_PMIS_GL_S.nextval
HHS_GLOBAL_INTERFACE_S	Control Table	Generate RECORD_ID based on HHS_GLOBAL_INTERFACE_S.nextval

4.8 Error Handling

Error handling works in conjunction with operations and maintenance branch of the UFMS project. This interface requires monitoring in UFMS by operations and maintenance for the timely resolution of interface errors.

This interface program will handle the following errors:

Table 29: Error Handling

Error	Response
PL/SQL -- Errors	This error records / messages will be captured and logged using the Global Error Handler (GEH) program The program will stop if any PL/SQL Error occurred. Program has to be resubmitted manually.
Unable to Write to the UNIX Out Directory	Check whether the directory is created with proper permissions and resubmit manually.
No records found for PMIS Extract	Program has to be resubmitted manually.

Table 30: UFMS Process Statuses

Process Status	Short Code	Meaning
NEW	N	The records are new and processing is done on this records.
PROCESSED	P	The records are validated processed successfully.
ERROR	E	The records are error out.
CHILD ERROR	C	If batch contains 5 lines and if one line error out then, error line will be marked as 'ERROR' and other success lines will be marked as 'CHILD ERROR'
SUSPEND	S	User can Suspend an Error record using GEH. Once the record is suspended, it will be moved to archive table.
UFMS ERROR	U	These records are error records due to UFMS configuration or setup.

4.8.1 Error Codes for General Ledger Interface

Table 31: Error Codes

Error Code	Description	Action	Transaction	Staging Table Update for

		Should be Taken By	Status	Journal Batch Processing
SOB_ERROR	1. Set of Books Name is not valid. Valid values are (CDC, FDA,PSC and IHS) 2. Set of Books Name is not setup in UFMS	Sunflower	Reject Full Batch	Acknowledgement File will be created with proper error code and error message and will be placed in specified UNIX directory. Full journal batch will be rejected.
TRN_DATE_NOT_OPENED	1. Transaction date format is not in DD-MON-YYYY. 2.Transaction Date is Not Open in UFMS General Ledger	UFMS	Reject Full Batch	Error records will be updated with status ERROR in record_status column and remaining all success records will be updated with status CHILD ERROR.
CCID_BACS_ERROR	Bacs is not valid in UFMS	UFMS	Reject Full Batch	Error records will be updated with status ERROR in record_status column and remaining all success records will be updated with status CHILD ERROR.
OBJECT_CLASS_ERROR	Given Object Class is not valid.	Sunflower	Reject Full Batch	Acknowledgement File will be created with proper error code and error message and will be placed in specified UNIX directory. Full journal batch will be rejected.
FISCAL_YEAR_ERROR	Given Fiscal Year is not valid in UFMS	Sunflower	Reject Full Batch	Acknowledgement File will be created with proper error code and error message and will be placed in specified UNIX directory. Full journal batch will be rejected.
CAN_ERROR	Given CAN is not valid in UFMS	Sunflower	Reject Full Batch	Acknowledgement File will be created with proper error code and error message and will be placed in specified UNIX directory. Full journal batch will be rejected.
ACTUAL_FLAG_ERROR	Actual Flag should be ‘A’ only.	Sunflower	Reject Full Batch	Acknowledgement File will be created with proper error code and error message and will be placed in specified UNIX directory. Full journal batch will be rejected.

JE_CATEGORY_ERROR	1. Category Name is not setup in UFMS. 2. Category Name is not in UFMS. (Valid Categories are Addition, Adjustment, Reclass, Transfer, Retirement and Depreciation	UFMS/ Sunflower	Reject Full Batch	Error records will be updated with status ERROR in record_status column and remaining all success records will be updated with status CHILD ERROR.
JE_SOURCE_ERROR	1. JE Source is not setup in UFMS. 2. JE Source is not SUNFLOWER	UFMS/ Sunflower	Reject Full Batch	Error records will be updated with status ERROR in record_status column and remaining all success records will be updated with status CHILD ERROR.
CURRENCY_ERROR	1. Currency Code USD is not defined UFMS. 2. Currency code is not valid (Valid value is 'USD')	UFMS /Sunflower	Reject Full Batch	Error records will be updated with status ERROR in record_status column and remaining all success records will be updated with status CHILD ERROR.
DEBIT_AMOUNT_ERROR	Provide Debit Amount if Credit amount is NULL	Sunflower	Reject Full Batch	Acknowledgement File will be created with proper error code and error message and will be placed in specified UNIX directory. Full journal batch will be rejected.
CREDIT_AMOUNT_ERR0R	Provide Credit Amount if Debit amount is NULL	Sunflower	Reject Full Batch	Acknowledgement File will be created with proper error code and error message and will be placed in specified UNIX directory. Full journal batch will be rejected.
0	Processed Record		Process Full Batch	Successfully validated records in the batch will be updated with status PROCESSED

4.8.2 Restart Strategy

In case of abnormal termination of concurrent request, the program needs to be resubmitted.

4.8.3 Crash Recovery

Should a serious failure occur, using the standard Oracle database functionality, processed data will be rolled back and the program can be re-run.

4.8.4 Incompatibility

This module contains no incompatibilities.

4.8.5 Performance Consideration

None

5.0. ISSUES

Table 32: Issues

Issue ID	Description	Status (Open/Closed)	Responsible	Resolution Date	Comments
1	GPRA Code for FDA can be mapped to Cost Center segment, however the mapping doesn't exist.	Closed	Solomon Iyamu	03/25/2007	Removed GPRA Code from all extract
2.	What happens if UFMS GL period is closed. How will PMIS interface import the current month transactions?	Open	Solomon Iyamu		
3.	Sunflower sends the transactions on the last day of the month. The transactions need to be created in GL prior to closing GL period for the month.	Open	Rich Bonick		
4.	How will sunflower know a correct can (Refer 4.3.2, assumption#10)	Open	Solomon Iyamu		

6.0. MEETINGS AND WORKING SESSIONS

Table 33: Meetings

Date	Description	Attendees	Meeting Highlights
11/21/2006	Working Group Meeting	Barry Koitz, Solomon Iyamu, Mahendra Dalvi, Liz Lane	Accounting Treatment
12/21/2006	Technical Discussion	Jaimie Wilson, Rich Bonick, Mahendra Dalvi	Acknowledgement Handling
01/05/2006	Working Group meeting	Barry Koitz, Solomon Iyamu, Jaimie Wilson, Mahendra Dalvi, Rich Bonick, Chris Martin	HIS Specific Requirement
01/11/2006	Working group meeting	Barry Koitz, Solomon Iyamu, Jaimie Wilson, Mahendra Dalvi, Rich Bonick	Error Handling
01/16/2006	Working group meeting	Barry Koitz, Solomon Iyamu, Jaimie Wilson, Mahendra Dalvi, Rich Bonick	Error Handling
01/02/2006	Working group meeting	Barry Koitz, Solomon Iyamu, Mahendra Dalvi, Jaimie Wilson, Bonick Rich	File Transfer and Networking
02/13/2007	Technical Walkthrough	Barry Koitz, Solomon Iyamu, Jamie Wilson, Betty Psillos, Mahendra Dalvi, Narayana Ayyala, Ramamohan Maadhavan, Rich Bonick	Technical Walkthrough
02/16/2007	Technical Group Discussion	Barry Koitz, Solomon Iyamu, Jamie Wilson, Betty Psillos, Mahendra Dalvi, Narayana Ayyala, Ramamohan Maadhavan, Rich Bonick, Rajesh Khatri	Technical Walkthrough - Acknowledgement process and journal batch processing
02/20/2007	Technical Group Discussion	Barry Koitz, Solomon Iyamu, Jamie Wilson, Betty Psillos, Mahendra Dalvi, Ramamohan Maadhavan, Rich Bonick, Rajesh Khatri	Technical Walkthrough - Continued Acknowledgement process and journal batch processing
03/25/2007	Technical discussion	Solomon Iyamu, Jaimie Wilson, Mahendra Dalvi	Technical changes

Appendix A: Baseline Requirements

Table 34: Baseline Requirements

Requirement Source	Requirement ID	Description
Baseline Requirement	The system shall provide the ability to identify & extract financial transaction changes to feeder system(s).	Baseline requirement 5015

Appendix B: Installation Procedure

<Document the procedures for installing the interface components. >

Register Program Executables

<List and describe the program executables registered in the Oracle Applications. >

Register Concurrent Program

<List and describe the concurrent programs registered in the Oracle Applications. >

Define Program Incompatibility

<List and describe the concurrent program incompatibilities. >

Implementation

From the Oracle Applications

<Document the implementation steps performed within the Oracle Applications. >

From the Unix Prompt

<Document the implementation steps performed at the operating system level. >

Appendix C: Data Transfer Methods

- Server to server files transfer.
- FI system provided with a UNIX account on UFMS server.
- Secure FTP used to send/retrieve files from UFMS.

Note: Approved UFMS Data Transfer Methods can be found in the UFMS Interface Data Transfer Policies document – PMOnline AFT# 3101.

Appendix D: Application Configuration Steps

All interfaces are run as standard request submission (SRS) concurrent programs. The following information defines the concurrent program executable and program. Any fields not explicitly listed in the table should retain their default values.

Table 35: Executable and Concurrent Program

Concurrent Executable and Concurrent Program for Obligations Extract

Define Concurrent Program Executable		
Field Name	**Field Value**	**Comments**
Executable	HHS_PMIS_PO_EXTRACT	Name of Executable
Short Name	HHS_PMIS_PO_EXTRACT	Short name of Executable
Application	HHS Global Applications	Application Name
Description	HHS PMIS Purchase Order Extract	HDL Payroll to GL Load
Execution Method	PL/SQL Stored Procedure	Method to Execute
Execution File Name	HHS_PMIS_PO_EXT_PK.main	Execution File Name
Define Concurrent Program		
Field	**Field Value**	**Comments**
Program Name	HHS PMIS Purchase Order Extract	
Short Name	HHS_PMIS_PO_EXTRACT	
Description	HHS PMIS Purchase Order Extract	
Executable Name	HHS_PMIS_PO_EXTRACT	
Method	PL/SQL Stored Procedure	
Application Name	HHS Global Applications	
Enabled	Y	
Use SRS	Y	
NLS Complaint	Y	
Format	Text	
Save	Y	
Print	Y	

Define Parameters		
Field Name	**Field Value**	**Comments**
Parameter Name	ORG_ID	Parameter Name
Description	HHS Agency	Description
Value Set	HHS_PMIS_ORG_VS	
Default Type	PROFILE	
Default Value	FND_PROFILE.value (‘ORG_ID’)	

Mandatory	Y	
Display	N	
Prompt	HHS Agency for Purchase Extract	
Define Parameters		
Field Name	**Field Value**	**Comments**
Parameter Name	SET_OF_BOOKS_ID	Parameter Name
Description	HHS Set of Books	Description
Value Set	HHS_PMIS_SOB_VS	
Default Type		
Default Value		
Mandatory	Y	
Display	N	
Prompt	HHS Set of Book	

Concurrent Executable and Concurrent Program for Receipts Extract

Define Concurrent Program Executable		
Field Name	**Field Value**	**Comments**
Executable	HHS_PMIS_RECEIPT_EXTRACT	Name of Executable
Short Name	HHS_PMIS_ RECEIPT _EXTRACT	Short name of Executable
Application	HHS Global Applications	Application Name
Description	HHS PMIS Receipts Extract	HDL Payroll to GL Load
Execution Method	PL/SQL Stored Procedure	Method to Execute
Execution File Name	HHS_PMIS_RCT_EXT_PK.main	Execution File Name
Define Concurrent Program		
Field	**Field Value**	**Comments**
Program Name	HHS PMIS Receipt Extract	
Short Name	HHS_PMIS_ RECEIPT _EXTRACT	
Description	HHS PMIS Receipt Extract	
Executable Name	HHS_PMIS_RECEIPT_EXTRACT	
Method	PL/SQL Stored Procedure	
Application Name	HHS Global Applications	
Enabled	Y	
Use SRS	Y	
NLS Complaint	Y	
Format	Text	
Save	Y	

Print	Y	
Define Parameters		
Field Name	**Field Value**	**Comments**
Parameter Name	ORG_ID	Parameter Name
Description	HHS Agency	Description
Value Set	HHS_PMIS_ORG_VS	
Default Type	PROFILE	
Default Value	FND_PROFILE.value ('ORG_ID')	
Mandatory	Y	
Display	N	
Prompt	HHS Agency for Purchase Extract	
Define Parameters		
Field Name	**Field Value**	**Comments**
Parameter Name	SET_OF_BOOKS_ID	Parameter Name
Description	HHS Set of Books	Description
Value Set	HHS_PMIS_SOB_VS	
Default Type		
Default Value		
Mandatory	Y	
Display	N	
Prompt	HHS Set of Book	

Concurrent Executable and Concurrent Program for Invoice Extract

Define Concurrent Program Executable		
Field Name	**Field Value**	**Comments**
Executable	HHS_PMIS_INVOICE_EXTRACT	Name of Executable
Short Name	HHS_PMIS_ INVOICE _EXTRACT	Short name of Executable
Application	HHS Global Applications	Application Name
Description	HHS PMIS Invoice Extract	HDL Payroll to GL Load
Execution Method	PL/SQL Stored Procedure	Method to Execute
Execution File Name	HHS_PMIS_PO_INV_PK.main	Execution File Name
Define Concurrent Program		
Field	**Field Value**	**Comments**
Program Name	HHS PMIS Invoice Extract	
Short Name	HHS_PMIS_ INVOICE _EXTRACT	
Description	HHS PMIS Invoice Extract	
Executable Name	HHS_PMIS_ INVOICE _EXTRACT	

Method	PL/SQL Stored Procedure	
Application Name	HHS Global Applications	
Enabled	Y	
Use SRS	Y	
NLS Complaint	Y	
Format	Text	
Save	Y	
Print	Y	

Define Parameters		
Field Name	**Field Value**	**Comments**
Parameter Name	ORG_ID	Parameter Name
Description	HHS Agency	Description
Value Set	HHS_PMIS_ORG_VS	
Default Type	PROFILE	
Default Value	FND_PROFILE.value ('ORG_ID')	
Mandatory	Y	
Display	N	
Prompt	HHS Agency for Purchase Extract	

Define Parameters		
Field Name	**Field Value**	**Comments**
Parameter Name	SET_OF_BOOKS_ID	Parameter Name
Description	HHS Set of Books	Description
Value Set	HHS_PMIS_SOB_VS	
Default Type		
Default Value		
Mandatory	Y	
Display	N	
Prompt	HHS Set of Book	

Concurrent Executable and Concurrent Program for GL Transaction Load

Define Concurrent Program Executable		
Field Name	**Field Value**	**Comments**
Executable	HHS_PMIS_GL_LOAD	Name of Executable
Short Name	HHS_PMIS_GL_LOAD	Short name of Executable
Application	HHS Global Application	Application Name
Description	HHS PMIS General Ledger Load	

Execution Method	SQL*Loader	Method to Execute
Execution File Name	HHS_PMIS_GL_LOAD	Name of Execution File Name
Define Concurrent Program		
Field	**Field Value**	**Comments**
Program Name	HHS PMIS GL Asset Transaction Load	
Short Name	HHS_PMIS_GL_LOAD	
Description	HHS PMIS GL Asset Transaction Load	
Executable Name	HHS_PMIS_GL_LOAD	
Method	SQL*Loader	
Application Name	HHS Global Application	
Enabled	Y	
Use SRS	Y	
NLS Complaint	Y	
Format	Text	
Save	Y	
Print	Y	
Define Parameters		
Field Name	**Field Value**	**Comments**
Parameter Name	HHS PMIS GL File Name	Parameter Name
Description	HHS PMIS GL File Name	Description
Value Set	30Character	
Default Type	Constant	
Default Value	$OS_INTERFACE_TOP/PMIS/in/HHS_PMIS_GL_*.dat	
Mandatory	Y	
Display	Y	
Prompt	HHS PMIS GL File Name	

Concurrent Executable and Concurrent Program for GL Import

Define Concurrent Program Executable		
Field Name	**Field Value**	**Comments**
Executable	HHS_PMIS_GL_IMPORT	Name of Executable
Short Name	HHS_PMIS_GL_IMPORT	Short name of Executable
Application	HHS Global Application	Application Name
Description	HHS PMIS General Ledger Import	
Execution Method	PL/SQL Stored Procedure	Method to Execute

Execution File Name	HHS_PMIS_GL_PK.main	Name of Execution File Name
Define Concurrent Program		
Field	**Field Value**	**Comments**
Program Name	HHS PMIS GL Asset Transaction Import	
Short Name	HHS_PMIS_GL_IMPORT	
Description	HHS PMIS GL Asset Transaction Import	
Executable Name	HHS_PMIS_GL_IMPORT	
Method	PL/SQL Stored Procedure	
Application Name	HHS Global Application	
Enabled	Y	
Use SRS	Y	
NLS Complaint	Y	
Format	Text	
Save	Y	
Print	Y	

Concurrent Executable and Concurrent Program for Acknowledgement Load

Define Concurrent Program Executable		
Field Name	**Field Value**	**Comments**
Executable	HHS_PMIS_ ACKNOWLDGEMENT_LOAD	Name of Executable
Short Name	HHS_PMIS_ ACKNOWLDGEMENT_LOAD	Short name of Executable
Application	HHS Global Application	Application Name
Description	HHS PMIS Acknowledgement Load	
Execution Method	SQL*Loader	Method to Execute
Execution File Name	HHS_PMIS_ ACKNOWLDGEMENT_LOAD	Execution File Name
Define Concurrent Program		
Field	**Field Value**	**Comments**
Program Name	HHS PMIS Acknowledgement Load	
Short Name	HHS_PMIS_ ACKNOWLDGEMENT_LOAD	
Description	HHS PMIS Acknowledgement Load	
Executable Name	HHS_PMIS_ ACKNOWLDGEMENT_LOAD	
Method	SQL*Loader	
Application Name	HHS Global Application	
Enabled	Y	
Use SRS	Y	
NLS Complaint	Y	

Format	Text	
Save	Y	
Print	Y	
Define Parameters		
Field Name	**Field Value**	**Comments**
Parameter Name	P_ ACKNOWLDGEMENT_FILE_NAME	Parameter Name
Description	HHS PMIS Acknowledgement File Name	Description
Value Set	240Character	
Default Type	Constant	
Default Value	$OS_INTERFACE_TOP/PMIS/in/HHS_PMIS_PO*.dat	
Mandatory	Y	
Display	Y	
Prompt	HHS PMIS Acknowledgement File Name	

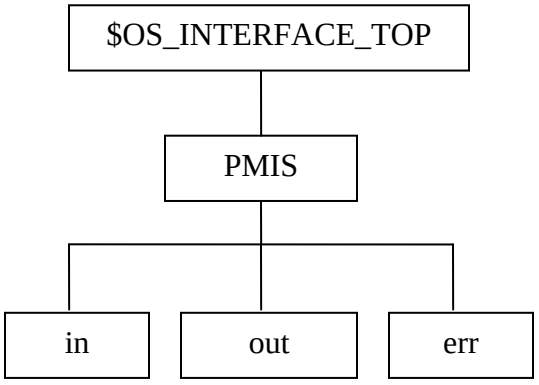
Appendix E: UNIX Directory Structure

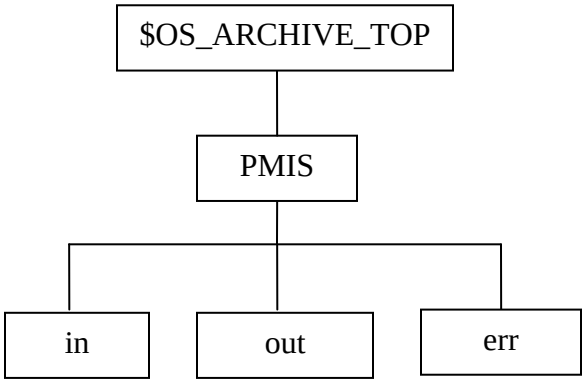
UNIX directory structures shall be created as follows for storing the PMIS inbound and outbound Files.

Create Sub Directories **in**, **out** and **err** under $**OS_INTERFACE_TOP/PMIS**

Table 36: UNIX Directory

NO	Unix Parent Directory Name	Unix Sub Directory	In/out/err	Database Directory Name	Read/Write/Delete
	Interface TOP Directories				
1	$OS_INTERFACE_TOP	PMIS	In	HHS_PMIS_IN_DIR	Read/Write/Delete
			out	HHS_PMIS_OUT_DIR	Read/Write/Delete
			err	HHS_PMIS_ERR_DIR	Read/Write/Delete
	Archive TOP Directories				
1	$OS_ARCHIVE_TOP	PMIS	In	HHS_PMIS_ARCHIVE_IN_DIR	Read/Write/Delete
			out	HHS_PMIS_ARCHIVE_OUT_DIR	Read/Write/Delete
			err	HHS_PMIS_ARCHIVE_ERR_DIR	Read/Write/Delete





Appendix F: Deliverables

Following individual objects will be delivered as a part of PMIS (Outbound and Interface) interface.

Table 37: Deliverables

File Name	Description	Target Location
HHS_PMIS_OBLIGATIONS_STG_TBL.sql	Script to create the staging table for Obligation Extract	$OS_INTERFACE_TOP/admin/sql
HHS_PMIS_RECEIPTS_STG_TBL.sql	Script to create the staging table for Receipts Extract	$OS_INTERFACE_TOP/admin/sql
HHS_PMIS_INVOICES_STG_TBL.sql	Script to create the staging table for Invoices Extract	$OS_INTERFACE_TOP/admin/sql
HHS_GLOBAL_INTERFACE_CTL_TBL.sql	Script to create global control table for inbound interface	$OS_INTERFACE_TOP/admin/sql
HHS_PMIS_GL_INTERFACE_STG_TBL.sql	Script to create the staging table for GL Interface	$OS_INTERFACE_TOP/admin/sql
HHS_PMIS_ACKNOWLEDGEMENT_TBL.sql	Script to create the staging table for Acknowledgement	$OS_INTERFACE_TOP/admin/sql
HHS_PMIS_GL_LOAD.ctl	Control file to load GL Asset transactions	$OS_INTERFACE_TOP/bin
HHS_PMIS_ACKNOWLEDGEMENT_LOAD.ctl	Control file to load Ack transactions	$OS_INTERFACE_TOP/bin
HHS_PMIS_PO_EXT_PK.sql	HHS Purchase Order Extract PL/SQL Package	$OS_INTERFACE_TOP/admin/sql
HHS_PMIS_RCT_EXT_PK.sql	HHS Receipt Extract PL/SQL Package	$OS_INTERFACE_TOP/admin/sql
HHS_PMIS_INV_EXT_PK.sql	HHS Invoice Extract PL/SQL Package	$OS_INTERFACE_TOP/admin/sql
HHS_PMIS_GL_PK	HHS GL Interface PL/SQL Package	$OS_INTERFACE_TOP/admin/sql

Appendix G: UFMS OPDIV Approvals

"IHS PMIS Functiona
Doc Approval.doc"

"FDA PMIS
Functional Doc Appro

"PSC PMIS Functiona
Doc Approval.doc"

"CDC PMIS
Functional Doc Appro

PMIS Error Handling Process

Deloitte.

Change Record

Date	Author(s)	Version	Change Reference
26-MAR-09	PMIS Team	V1.0	PMIS Error Handling

Reviewers

Date	Name	Position
26-Mar-09	Ravi Thota	UFMS O&M Global Technical Lead

1 Background

The PMIS General Ledger Inbound data file contains PMIS Asset Journal transactions sent by PMIS system to UFMS. The asset journals are sent for the following journal categories: Addition, Adjustment, Reclassification, Transfer, Retirement, and Depreciation. These transactions are created in the UFMS General Ledger module via a two step process.

Step 1: HHS PMIS General Ledger Loader Program

Step 2: HHS PMIS General Ledger Interface Program

During execution of each step, there is an associated error handling mechanism that maintains data integrity and identifies potential error situations.

2 Step 1: HHS PMIS General Ledger Loader Program

This process is a SQL*Loader program that loads the PMIS inbound file data along with the file's header and trailer record into UFMS staging tables.

2.1 File Validation

The program logic has a built in file validation to confirm the following:

1. Inbound data file is new and not a duplicate or previously processed file.
2. Total asset journal count in staging table = Record count in trailer record
3. Total asset journal amount in staging table = Amount in trailer record

If the file is previously processed or the above counts do not match, a corresponding error acknowledgement is sent to PMIS. The data in the staging tables is cleared in such an event.

2.1.1 Error Handling

Error Condition - Abnormal system or power failure during this program run

Mitigation - Upon resumption of system services, previously processed data is rolled back from the staging tables and the inbound file is reprocessed.

2.2 Staging Tables – Data Validation

Validation Check	Description
Transaction Date	Confirm that transactions are in an open period
CAN	CAN is valid / active

The following process statuses are associated with transactions in the PMIS inbound staging table:

Process Status	Short Code	Meaning
NEW	N	The transactions are new and processing is done on these records.
PROCESSED	P	The transactions are validated and processed successfully.
ERROR	E	The transactions are in error.
CHILD ERROR	C	If batch contains 5 lines and if one line is in error, then error line will be marked as 'ERROR' and remaining lines will be marked as 'CHILD ERROR'
SUSPEND	S	User can Suspend an Error transaction using GEH. Once the transaction is suspended, it will be moved to archive table.
UFMS ERROR	U	These transactions are in error due to UFMS configuration or setup.

2.2.1 Error Handling

Error Condition #1 – Abnormal program termination while execution of data validation process.

Mitigation – All data validations are rolled back. When the program is resubmitted all transactions are revalidated. All database commits are performed at the end of data validation process.

Error Condition #2 – Asset transactions are associated with their appropriate asset category in the inbound data file. As per the above file and data validations, if one transaction within an asset category is in error, the entire asset category is marked for error. A subsequent error acknowledgement is sent to PMIS system notifying them of the error condition.

Mitigation – PMIS system sends a new inbound transaction data file only for the errored asset category. UFMS processes these transactions, and if there are no errors, then it creates asset journals in UFMS.

2.2.2 Archiving

All processed and suspended transactions are moved to the PMIS archive table. The corresponding transactions are deleted from the PMIS staging table.

3 Step 2: HHS PMIS General Ledger Interface Program

3.1 GL Interface Load

After completion of Step 1 above, all transactions processed successfully are loaded into the GL Interface table.

3.1.1 Error Handling

Error Condition – Abnormal program termination when PMIS staging table transactions are being loaded into the GL Interface table

Mitigation – All transactions are rolled back. When the program is resubmitted, PMIS staging table transactions are re-loaded into the GL Interface table.

3.2 Journal Import

The journal import program processes the PMIS transactions loaded into the GL Interface table and creates GL Journal batches, headers and lines. This is a standard Oracle Applications program and is executed as part of PMIS transaction processing.

3.2.1 Error Handling

Error Condition – Journal Import process validates transactions for several error conditions. These error conditions are shown in the report output of Journal Import concurrent program.

Mitigation – Users determine the nature and source of the above error conditions and take appropriate measures to correct them. After resolution of the error conditions, the Journal Import program can be restarted to process the errored transactions.

PMIS UFMS Outbound Interface Process

Version 0.2
February 19, 2009

Prepared and Submitted By:

Metrix Technologies, Inc.

For Official Use Only

Document Revision History

Introduction and Functional Design Sections

Date	Version	Author	Level of Review	Summary of Comments
10/18/07	0.0	Rajesh Khatri	Initial Draft	Initial Document Creation.
10/30/2007	0.1	Rich Bonick	QA Review	
2/19/2009	0.2	Rich Bonick	Fianl Document	

Table of Contents

List of Figures

List of Tables

1.0. INTRODUCTION

The Property Management Information System (PMIS), also referred to as Sunflower, will be utilized by the Department of Health and Human Services (HHS) to provide a single standardized management system for personal property. The Sunflower system will provide tracking of accountable personal property, financial management of capital personal property throughout an asset's useful life, asset reporting, periodic depreciation calculations, asset additions and removal. The following HHS systems will integrate with the Sunflower Asset Management system: UFMS (United Financial Management System).

This document addresses the interface between the Sunflower system and UFMS. Purchase orders, receiving, and invoicing records will be created in or interfaced to UFMS from other procurement systems.

Sunflower shall receive purchase order, receiving information and invoice information for personal property within the 31000 series of object class. This includes both capitalized and non-capitalized personal property. The purchase order shall be sent as a skeletal record to Sunflower except for those credit card transactions for which no postings are made to Purchasing. The receiving information shall be sent once receiving is performed in UFMS. The PMIS interface shall extract the purchase order and receiving information from UFMS Purchasing and the invoice payment information from UFMS Payables. The interface is a two-way interface from Sunflower to UFMS. Three separate extract flat files will be sent from UFMS to Sunflower for the purchase orders, receiving records and invoice payment records; asset journal entries including depreciation shall be sent back to UFMS General Ledger from Sunflower.

All HHS Operating Divisions (OPDIVs) except CMS and NIH participated in the development of the global To-Be baseline business processes and the UFMS PMIS interface will be developed taking into account the business rules defined. Food and Drug Administration (FDA), Center for Disease Control and Prevention (CDC), Program Support Center (PSC) and Indian Health Service (IHS) will be deployed as part of the UFMS Release 4.0 scheduled for go-live in October 2007.

In this document, we identify business rules that drive how HHS property related activities would be interfaced, processed, stored and reported on in Sunflower. Specifically, this document addresses:

1) The design and functionality of the UFMS PMIS interface.
2) The business scenarios for property management.
3) The outbound extract of property data to Sunflower.
4) The technical requirements for the UFMS PMIS interface.
5) The operational process for identifying and correcting errors.

In this document PMIS outbound means the file received by Sunflower, PMIS inbound means the file received by UFMS.

1.1. Process flow, high level

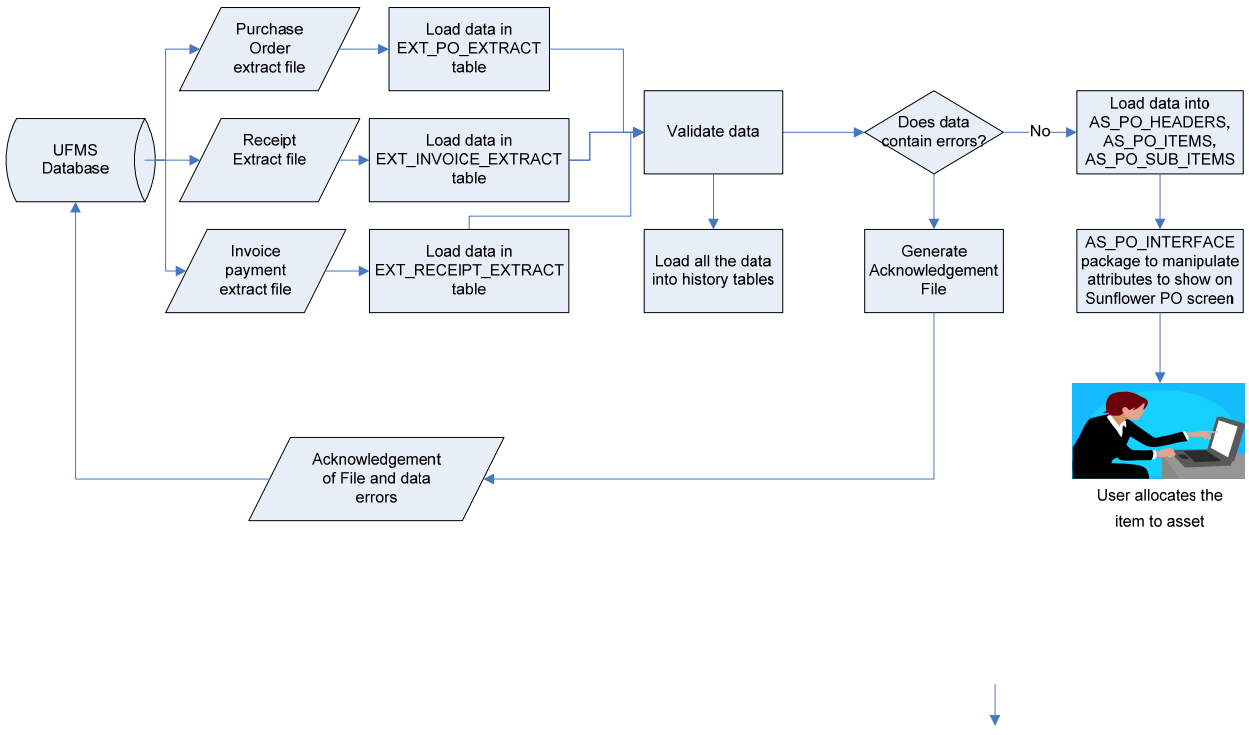
UFMS Outbound interface flow

1. PO, RCT, INV files are retrieved from the UFMS outbox at every morning at 10 am.

2. The files are checked for file errors and data errors.
3. Acknowledgement file generated :
    a. If there are any file errors, they are listed in the acknowledgment file
    b. If there are any data error at the file line level, they are listed in the acknowledgment file
    c. If no errors are identified, the acknowledgement file will report all lines processed successfully.
4. Upon file processing completion, an email is sent to the OPDIV which contains the status for each file received, loaded, and errors related to the OPDIV.

1.2. Process flow in PMIS

Figure 1: PMIS Interface Data flow, from PMIS point of view



Steps:

1. Purchase Orders, Receipts, and Invoice payments are sent in three independent files.
2. Each file shall is loaded into a corresponding staging table in PMIS
3. The data from all three files are validated
4. An acknowledgement file and any identified errors is sent back to UFMS
5. Validated data is then be loaded into Sunflower API tables to display on the Sunflower PO screen
6. Users then manually creates assets in Sunflower

1.3. Error handling in PMIS

For every file type sent from UFMS to PMIS, an acknowledgement file generated and sent back to UFMS. The acknowledgement file will indicate at the record level whether each record within the file successfully processed in PMIS. This shall enable the extract to be regenerated for the records that were not processed in PMIS.

Table 1: Error Handling

Action	Response
FTP errors	Error records and messages will be captured and logged using the Global Error Handler (GEH) program. Check the error messages and correct the records in UFMS as needed.
Data flat file errors	Error records and messages will be captured and logged using the Global Error Handler (GEH) program. Each OPDIV shall check the error messages and correct their records as needed. If PMIS sends back a negative acknowledgement for any record, the obligating document shall be re-generated again on the next run of the extract. Error records shall remain on the staging table.
Interface transactions from staging table to Sunflower	If the extract fails from the staging table, the records can be reprocessed to PMIS on the next run on the PMIS Interface.

1.3.1. Error code groups in PMIS:

Error codes are divided into following logical groups:

Table 2: Error Code Logical Groups

Error Code Range	Error Type
0	No error
10001 – 19999	File errors
20001 – 29999	Data Validation errors
30001 – 39999	Business Rule errors

For every error there will be a responsible party. When the file is received it will be checked for all the errors related to that type of file.

1.3.2. In PMIS, Actions to be Performed by Error Types

- No error: Record id is sent with error code 0 in the acknowledgement file.
- File errors: Acknowledgement file is sent with the correct error code, no data is saved in the PMIS staging tables.
- Data errors: In the acknowledgement file, the record with the error marked with E, other records related to the same document will be marked with C (child error). Remaining records with the record status “P” will be loaded into PMIS.
- Business Rule errors: In the acknowledgement file, the records with business errors are marked with P (validated, PMIS error) due to the fact that UFMS does not have the capability to correct procurement record logic errors. In this case, OPDIVs are notified via email. OPDIV must fix these types of errors in the source procurement system which will result in UFMS resending the document record with corrections.

1.3.3. In PMIS, Actions to be Performed for records with no error

Only the records with the record status “P” are loaded into Sunflower, all other type of records which fail, should be corrected by either UFMS or OPDIV and be resent in a subsequent file.

1.3.4. PMIS Error Code table

Table 3: Error Codes

ERROR_CODE	ERROR_DESCRIPTION	RESPONSIBLE_PARTY
0	No Error	NONE
10001	File po_extract not found	UFMS
10002	File receipt_extract not found	UFMS
10003	File invoice_extract not found	UFMS
10004	File po_extract not complete	UFMS
10005	File receipt_extract not complete	UFMS
10006	File invoice_extract not complete	UFMS
10007	Column RECORD_IDENTIFIER does not contain TH, TX or TT	UFMS
10008	Dollar amount on the trailer does not match with the file total dollar amount	UFMS
20001	Column RECORD_ID is NULL	UFMS
20005	Column RECORD_ID has non numeric data	UFMS
20006	Column DOCUMENT_NUMBER length is 0 or more than 50	UFMS
20008	Column LINE_NUMBER length is 0 or more than 15	UFMS
20009	Column DOCUMENT_DATE is null	UFMS
20010	Column DOCUMENT_DATE has non date characters	UFMS
20011	Column REVISION_NUMBER has non numeric characters	UFMS
20012	Column REVISION_NUMBER is NULL	UFMS
20100	Column RECEIPT_NUMBER length is either 0 or more than 30	UFMS
20101	Column RECEIPT_DATE is NULL	UFMS
20102	Column COMMENTS is NULL	UFMS
20103	Column RECEIPT_LINE_NUMBER is NUMBER data type, data contains non-numeric characters	UFMS
20104	Column RECIPT_LINE_NUMBER is NULL	UFMS
20105	Column RECEIPT_QUANTITY is NUMBER data type, data contains non-numeric characters	UFMS
20106	Column RECEIPT_QUANTITY is NULL	UFMS
20107	Column PO_UNIT_PRICE is NUMBER data type, data contains non-numeric characters	UFMS
20108	Column PO_UNIT_PRICE is NULL	UFMS
20109	Column ITEM_DESCRIPTION length is more than 150	UFMS
20110	Column SHIPMENT_LINE_STATUS length is either 0 or more than 25	UFMS
20111	Column LINE_COMMENTS length is more than 150	UFMS
20112	Column TRANSACTION_TYPE length is either 0 or more than 25	UFMS
20113	Column UNIT_OF_MEASURE length is either 0 or more than 25	UFMS
20114	Column CURRENCY_CODE length is either 0 or more than 3	UFMS
20115	Column QUANTITY_SHIPPED is NUMBER data type, data contains non-numeric characters	UFMS
20116	Column QUANTITY_RECEIVED is NUMBER data type, data contains non-numeric characters	UFMS

ERROR_CODE	ERROR_DESCRIPTION	RESPONSIBLE_PARTY
20117	Column PO_NUMBER length is either 0 or more than 25	UFMS
20118	Column PO_REVISION_NUMBER is NUMBER data type, data contains non-numeric characters	UFMS
20119	Column PO_LINE_NUMBER is NUMBER data type, data contains non-numeric characters	UFMS
20120	Column PO_SHIPMENT_NUMBER is NUMBER data type, data contains non-numeric characters	UFMS
20121	Column PO_DISTRIBUTION_NUMBER is NUMBER data type, data contains non-numeric characters	UFMS
20122	Column VENDOR_NAME length is either 0 or more than 150	UFMS
20123	Column VENDOR_SITE_CODE length is either 0 or more than 25	UFMS
20124	Column CAN length is either 0 or more than 25	UFMS
20125	Column FISCAL_YEAR length is either 0 or more than 4	UFMS
20126	Column OBJECT_CLASS length is either 0 or more than 5	UFMS
20127	Column USSGL_ACCOUNT length is either 0 or more than 7	UFMS
20128	Column HHS_AGENCY_CODE length is either 0 or more than 3	UFMS
20129	Column GPRA_CODE length is more than 3	UFMS
20130	Column ACCOUNTING_POINT length is more than 3	UFMS
20131	PO_ID is mandatory(for agencies other than PSC), but could not found in the invoice_extract file	UFMS
20132	Record found in receipt_extract without a reference to PO	UFMS
30001	Asset cost is greater than 24999, object class code can not be 319XX	OPDIV
30002	Quantity in the Receipt record is more than the Ordered Quantity in the PO record	OPDIV
30003	Quantity in the Receipt record is less than the Created Asset Quantity	OPDIV
30004	Asset cost is less than 25,000, object class code can not be 31XXX	OPDIV

1.4. Outputs from PMIS

1.4.1. Acknowledgement file

An acknowledgement file is sent for every file received from UFMS, If there are any file errors, the errors are listed in the acknowledgment file at the file level. If there are data errors, the error is listed at the line level in the acknowledgment file. If there no errors are detected, the acknowledgement file will indicate the file was processed successfully.

File Naming convention:

HHS_PMIS_CDC_ACK_PO_YYYYMMDDHHMISS.dat

HHS_PMIS_FDA_ACK_RCT_YYYYMMDDHHMISS.dat

HHS_PMIS_IHS_ACK_INV_YYYYMMDDHHMISS.dat

Table 4: Logical Acknowledgement File Layout

Attribute Name	Attribute Description
RECORD_ID	RECORD_ID of the record from the incoming file
ACK_SOURCE	e.g. PO, RECEIPT, INVOICE
HHS_AGENCY	e.g. CDC, HIS, FDA etc.
DOCUMENT_NUMBER	In the case of PO file it is PO# In the case of Invoice file it is Invoice # In the case of Receipt file it is Receipt #
DOCUMENT_LINE_NUMBER	In the case of PO file it is PO Line# In the case of Invoice file it is Invoce Line# In the case of Receipt file it is Receipt Line#
ACCOUNTING_POINT	NIH, PHX (In case of IHS data, location to be extracted from the incoming file and put on the acknowledgment file
RECORD_STATUS	P: Processed E: Error C: Child Error
ERROR_CODE	The code that identifies the type of error. See the Error Handling section for specific types.
ERROR_DESCRIPTION	Logical Description of the error

Examples:

1. If there are multiple file errors, the acknowledgment file will look like this:

0|PO|IHS||||E|MULTIPLE_ERROR|HHS_PMIS_IHS_PO_20080807052124:Dollar amount on the trailer does not match with the file total dollar amount***154 Records did not load due to fatal error

2. If there is one file error in the file, the acknowledgment file will look like this:

0|PO|CDC||||E|10008|HHS_PMIS_CDC_PO_20070327134048:Dollar amount on the trailer does not match with the file total dollar amount

3. If there is no error in the received file, acknowledgment file will look like this:

e.g. File name: HHS_PMIS_CDC_ACK_PO_20090212120341.dat

38502|PO|CDC|MAC9020084729|1001||P|0|No Error

38503|PO|CDC|MAC9020076204|3001||P|0|No Error

38504|PO|CDC|MAC9020076204|2001||P|0|No Error

38505|PO|CDC|MAC9020078937|1001||P|0|No Error

38506|PO|CDC|MAC9020078242|1001||P|0|No Error

38507|PO|CDC|2002009F28896|100||P|0|No Error

38508|PO|CDC|2002009F28987|100||P|0|No Error

4. If there are some error are on one line, the acknowledgment file will look like this, here records with E means error record, records with C denotes child error, means these are for the same PO, hence did not load, records with P means processed successfully, no error in those records.

55255|PO|CDC|MAC7100354919|1003||E|20012|Column REVISION_NUMBER is NULL

55256|PO|CDC|MAC7100354919|2002||C|0|No Error

55257|PO|CDC|MAC7100354919|4001||C|0|No Error

55227|PO|CDC|MAC7100355072|4001||P|0|No Error

55314|PO|CDC|MAC7100355122|3002||P|0|No Error

55315|PO|CDC|MAC7100355122|3005||P|0|No Error

55316|PO|CDC|MAC7100355296|1002||P|0|No Error

55317|PO|CDC|MAC7100355296|4002||P|0|No Error

55319|PO|CDC|MAC7100355296|6002||P|0|No Error

55320|PO|CDC|MAC7100355296|11003||P|0|No Error

55318|PO|CDC|MAC7100355296|5002||P|0|No Error

55321|PO|CDC|MAC7100355437|1001||P|0|No Error

55322|PO|CDC|MAC7100355437|4001||P|0|No Error

5. If there are multiple data errors on one line, the acknowledgment file will look like this:

55255|PO|CDC|MAC7100354919|1003||E|MULTIPLE_ERROR|Column REVISION_NUMBER is NULL***Column RECEIPT_DATE is NULL

55256|PO|CDC|MAC7100354919|2002||C|0|No Error

55257|PO|CDC|MAC7100354919|4001||C|0|No Error

55227|PO|CDC|MAC7100355072|4001||P|0|No Error

55314|PO|CDC|MAC7100355122|3002||P|0|No Error

55315|PO|CDC|MAC7100355122|3005||P|0|No Error

55316|PO|CDC|MAC7100355296|1002||P|0|No Error

55317|PO|CDC|MAC7100355296|4002||P|0|No Error

55319|PO|CDC|MAC7100355296|6002||P|0|No Error

1.4.2. Email to all OPDIVs

An email is sent to the OPDIV with the status for the files received every day, any OPDIV can provide the email address to be used for this email.

Sample email:

```
Sender: AssetWeb@psc.hhs.gov

*****************************************
*****************************************
PO Interface load summary Batch Date 20090214
*****************************************
*****************************************

Batch Date: 20090214
--------------------
PO Files Received:
PO FILE_NAME|NO_OF_DOCUMENTS_IN_FILE|
HHS_PMIS_CDC_PO_20090214055321.dat|22|


PO Loaded into Interface:
DOCUMENTS_NUMBER
2002009F28967
2122009F29016
2542009M28999
MAC9010076634
MAC9010077400
MAC9010084059
MAC9010086526
MAC9020001585
MAC9020074928
MAC9020074993
MAC9020075362
MAC9020075495
MAC9020075529
MAC9020076204
MAC9020076618
MAC9020077418
MAC9020077822
MAC9020078242
MAC9020078283
MAC9020078937
MAC9020084661
MAC9020087979

-----------------------------------------------------
Receipt Files Received:
FILE_NAME|NO_OF_DOCUMENTS_IN_FILE|
HHS_PMIS_CDC_RCT_20090214060432.dat|5|

Receipts Loaded into Interface:
RECEIPT_DOCUMENT_NUMBER|PO_NUMBER|
```

```
2002008F28197-HSC0021309|2002008F28197|
2002009F28715-194727|2002009F28715|
2002009F28889-193587|2002009F28889|
2002009F28899-193592|2002009F28899|
2542009F28758-HSC0021309|2542009F28758|

-----------------------------------------------------
Invoice Files Received:
FILE_NAME|NO_OF_DOCUMENTS_IN_FILE|
HHS_PMIS_CDC_INV_20090214065539.dat|2

Invoice Loaded into Interface:
INVOICE_DOCUMENT_NUMBER|PO_NUMBER|
0000164493|2542009F28758|
INV029027|2002009M28669|

-----------------------------------------------------
PO Interface load summary Batch Date 20090214
```

1.4.3. Email to OPDIVs with Accounting Points:

IHS emails are generated on the basis of accounting point, and sent to the accounting point contacts.

Sample email:

Sender: AssetWeb@psc.hhs.gov

****
****
PO Interface info for BatchDate:20090218,Accounting point:46
****
****

Batch Date: 20090218

Receipts Loaded into Sunflower Interface for the Accounting point:46

RECEIPT_DOCUMENT_NUMBER|PO_NUMBER|
4004546I390910281P
4004561I390910313P
4004562I390910200P

Invoice Loaded into Sunflower Interface for the Accounting point:46

INVOICE_DOCUMENT_NUMBER|PO_NUMBER|
MXR2352I390910290P|
XD3DTD864I390910289P|

PO Interface info for BatchDate:20090218,Accounting point:46

1.4.4. Restart Strategy

If the interface fails, for any reason other than the errors noted in this document:

1. Metrix Technologies, Inc. will analyze the failure to determine the cause of the malfunction and take immediate corrective action as applicable.
2. Metrix will inform PSC, UFMS, and any affected OPDIV via email.
3. Metrix will provide an updated status within 8 hrs of the error.
4. If corrective actions are required on the part of UFMS, the appropriate O&M UFMS contact will be notified along with any necessary corrective action.
5. Once a resolution is in place, the interface program will be rerun to load the failed files.

PMIS UFMS Inbound Interface Process

Version 2.0
August 30, 2007

Prepared and Submitted By:

Metrix Technologies, Inc.

For Official Use Only

Document Revision History

Introduction and Functional Design Sections

Date	Version	Author	Level of Review	Summary of Comments
1/26/07	0.0	Harjeet Parmar	Initial Draft	Initial Document Creation.
8/29/07	1.0	Cindy Fang	Format and Edit	Edited for formatting and content
8/30/07	2.0	Richard Bonick	QA Review	All

Table of Contents

List of Figures

List of Tables

1.0. INTRODUCTION

The Property Management Information System (PMIS), also referred to as Sunflower, will be utilized by the Department of Health and Human Services (HHS) to provide a single standardized management system for personal property. The Sunflower system will provide tracking of accountable personal property, financial management of capital personal property throughout an asset's useful life, asset reporting, periodic depreciation calculations, asset additions and removal. The following HHS systems will integrate with the Sunflower Asset Management system: UFMS (United Financial Management System).

This document addresses the interface between the Sunflower system and UFMS. Purchase orders, receiving, and invoicing records will be created in or interfaced to UFMS from other procurement systems.

Sunflower shall receive purchase order, receiving information and invoice information for personal property within the 31000 series of object class. This includes both capitalized and non-capitalized personal property. The purchase order shall be sent as a skeletal record to Sunflower except for those credit card transactions for which no postings are made to Purchasing. The receiving information shall be sent once receiving is performed in UFMS. The PMIS interface shall extract the purchase order and receiving information from UFMS Purchasing and the invoice payment information from UFMS Payables. The interface is a two-way interface from Sunflower to UFMS. Three separate extract flat files will be sent from UFMS to Sunflower for the purchase orders, receiving records and invoice payment records; asset journal entries including depreciation shall be sent back to UFMS General Ledger from Sunflower.

All HHS Operating Divisions (OPDIVs) except CMS and NIH participated in the development of the global To-Be baseline business processes and the UFMS PMIS interface will be developed taking into account the business rules defined. Food and Drug Administration (FDA), Center for Disease Control and Prevention (CDC), Program Support Center (PSC) and Indian Health Service (IHS) will be deployed as part of the UFMS Release 4.0 scheduled for go-live in October 2007.

In this document, we identify business rules that drive how HHS property related activities would be interfaced, processed, stored and reported on in Sunflower. Specifically, this document addresses:

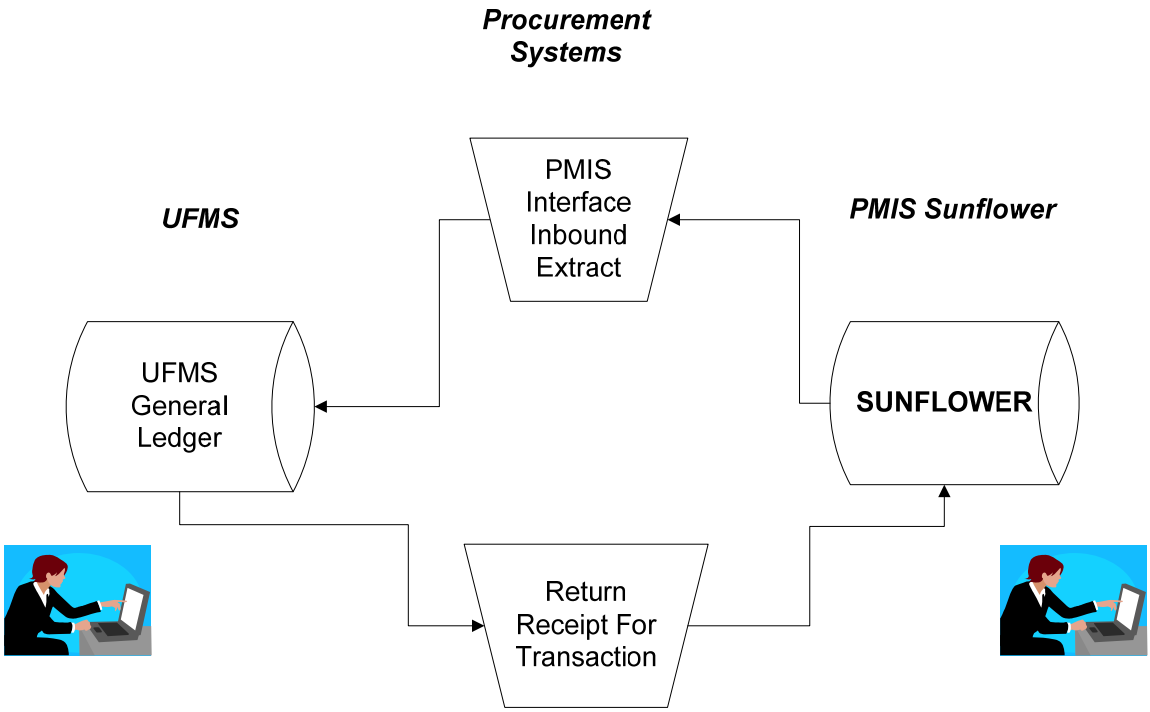
1) The design and functionality of the UFMS PMIS interface.
2) The error handling process for the Inbound UFMS PMIS interface.

In this document PMIS outbound means the file received by Sunflower, PMIS inbound means the file received by UFMS.

1.1. Major Features

The diagram below depicts, at a high level, the salient features of the global UFMS – PMIS interface, and how data will flow between the PMIS Sunflower application and the UFMS System.

Figure 1: UFMS-PMIS Interface



2.0. FUNCTIONAL DESIGN

2.1. Function Overview

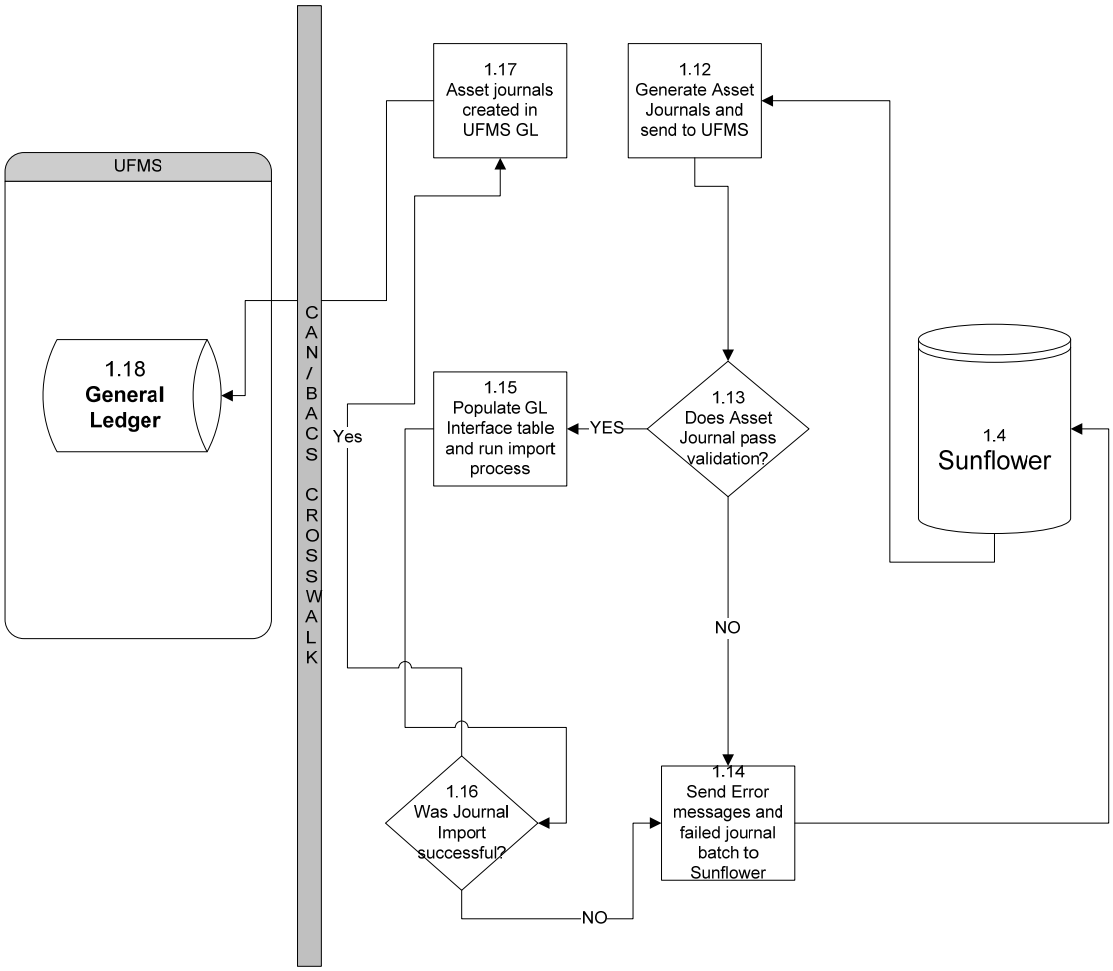
This function design outlines Sunflower's interface to the UFMS PMIS system. It describes the process flow based on the functional scenarios and requirements dictated by UFMS. The functional design components are divided into the following sections of this document.

1. Function System Flow for the PMIS inbound interface
   - Describes the general flow of data among different objects in Oracle UFMS.
2. Extract Methodology
   - Describes the custom database tables and dependencies for the PMIS interfaces.
3. Assumptions
   - Describes the technical assumptions that the PMIS interfaces are based on and the scope of the current deliverable
4. System objects created for PMIS Interfaces
   - Tables
   - PL/SQL Components
5. File Naming Convention and PMIS Interfaces
6. Error Handling

2.2. Functional System Flow

The following diagram depicts a detailed functional data flow of personal property activity from Sunflower to UFMS General Ledger. A detailed technical data flow (included processing steps and failure junctures) will be included as a part of the technical design document.

Figure 2: Functional Data Flow from PMIS to UFMS General Ledger



Steps:

1.12 At thee end of the period (monthly), Sunflower shall generate asset journals entries and send to UFMS.

1.13 The PMIS interface shall perform a pre-validation of the asset journal entries sent to ensure that the records can be populated in the GL interface table. All the required columns (CAN, Object class, Fiscal Year, USSGL Account, Amount, debit and credit accounts) shall be verified.

1.14 If an error exists when running the pre-validation, the PMIS Inbound interface shall send an error message to Sunflower including the failed journal batch. Sunflower shall resend a corrected batch to UFMS.

1.15 If the asset journal entry records pass pre-validation by the PMIS interface, the records shall be inserted into the GL interface table and the standard Oracle GL import process shall be run.

1.16 If journal import is successful, the asset journals shall be created in UFMS.

1.17 Asset journals are created in UFMS General Ledger as unposted journals. The following journals shall be sent from Sunflower: Addition, Adjustments, Reclassification, Transfers, Retirements, and Depreciation.

1.18 Prior to the UFMS General Ledger closing for the period, all unposted asset journals shall be posted. The trial balance can also be run to reflect the asset balances.

2.3. Extraction Methodology

At the end of every month, Sunflower will extract data from a repository where the occurrence, storage, and manipulation of data for that month has occurred. Regardless of number of transactions, at the end of every month, Sunflower is tasked to extract data, store that data for processing, and transmit all UFMS specified extractions.

The extraction of data will be isolated to the Sunflower repository. Any incomplete records will not be processed. For all records that are incomplete, the extraction process will intersect all data with the set of all valid data.

2.4. Assumptions

The following assumptions generalize to Sunflower and the UFMS PMIS General Ledger system. The assumptions are meant to comply with Sunflower's understanding of the needs articulated by UFMS as outlined in UFMS PMIS Interface Technical Design Document v.10.

2.4.1. General Assumptions

1. The PMIS interface is a two-way interface between Sunflower and UFMS, a success or failure notice will be sent by the interface back to Sunflower for each asset journal transaction sent to UFMS.
2. The inbound data will be structured in a consistent manner for all Items and will confirm to the agreed HHS standards.
3. The PMIS Interface processes are discussed in Section 2.2.
4. UNIX will have all READ. WRITE and EXECUTE permissions for the purpose of file transmission and file consumption.
5. UFMS GEH Technical is outlined starting in Section 4.
6. All transmitted files will have one header and one trailer record associated with them.
7. All records being processed by the interface are assumed to be in USD currency.
8. File Transfer between Production Box and Feeder System will be done separately by UFMS drop box. Once Sunflower has successfully transmitted data in the form of a flat file to the UFMS drop-box, Sunflower's inbound processing is terminated.
9. Sunflower will have access to UFMS drop box to get and put the file onto the server.

2.4.2. Interface Assumptions

1. Separate property books shall be setup in Sunflower for each HHS OPDIV and journal entries shall be interfaced to UFMS by OPDIV.
2. Asset journal entries shall be created in UFMS General Ledger with a category type of Addition, Adjustment, Reclass, Transfer, Retirement and Depreciation.

3. Asset journal entries shall be created in UFMS General Ledger with a source type of 'SUNFLOWER'.
4. Sunflower will send one data file for each Op-div. Each data file contains multiple batches based on category (Refer Assumption 2). Batch Name contains Source -Category for each Op-div.
5. If one line of an asset journal batch fails import to UFMS General Ledger then Sunflower shall be required to resend the failed journal batch.
6. The PMIS interface shall send the CAN, Object Class, Fiscal Year, USSGL Account and HHS T-code and task_id (For HIS if applicable) to Sunflower as the accounting information. Sunflower shall send the CAN, Object Class, Fiscal Year and appropriate USSGL Account to UFMS for asset journal entries. See Table One.
7. Sunflower assumes that the Calendar Period is open in the UFMS General Ledger Module.
8. Sunflower expects that once processing has terminated on the UFMS General Ledger system, Sunflower will have access to a receipt file with UFMS system's processing results for consumption.
9. Sunflower will expect a return receipt file, with the appropriate processing codes, from the UFMS system no later than 5 days of the initial Sunflower inbound broadcast to UFMS.

2.5. System Objects

Sunflower's processing is limited to certain mediums. In particular, an Oracle instance stores the Sunflower repository. All data processing is to be isolated in this single instance of Oracle. Once all data manipulation has occurred, processing will be moved to the file system that supports the Oracle Instance. The current implementation resides on Solaris 5.9. The file system is responsible for any distributed transmissions. In particular, an operating system script will be responsible for the transmission of data as it is accumulated and organized into a flat file by the database.

2.6. File Naming Conventions

The following details the file naming conventions used by Sunflower for the UFMS General Ledger inbound file.

Fields in the flat file are pipe ('|') delimited. The following file name convention is used:

File Name: - HHS_PMIS_CDC_GL_YYYYMMDDHHMISS.dat

HHS_PMIS_FDA_GL_YYYYMMDDHHMISS.dat

HHS_PMIS_PSC_GL_YYYYMMDDHHMISS.dat

HHS_PMIS_IHS_GL_YYYYMMDDHHMISS.dat

2.7. Error Handling

For every record contained in the end-month inbound file, from Sunflower to UFMS, a receipt of acknowledgement shall be returned to Sunflower providing a status every record constitutive of the inbound file. Should any records fail UFMS processing for reasons specified in the UFMS PMIS Interface Technical Design Document v.10, Sunflower will provide another full extract for the appropriate month.

Any failures internal to Sunflower or UFMS are idiosyncratic to that particular implementation. Sunflower, will maintain a database log of processing. This will include any errors occurrence during data extracting, processing and organization, file creation and transmission. Section 3 includes a restart strategy for any error conditions that falls within the purview of Sunflower's interface processing.

3.0. TECHNICAL DESIGN

3.1. Technical Overview

The PMIS inbound interface shall broadcast asset journal entries from the most current implementation of Sunflower to the UFMS General Ledger by HHS OPDIV. The journal entries shall be created monthly in UFMS. The inbound PMIS interface shall process the following statuses of personal property payments:

(1)New and validated asset journal entry transactions from Sunflower

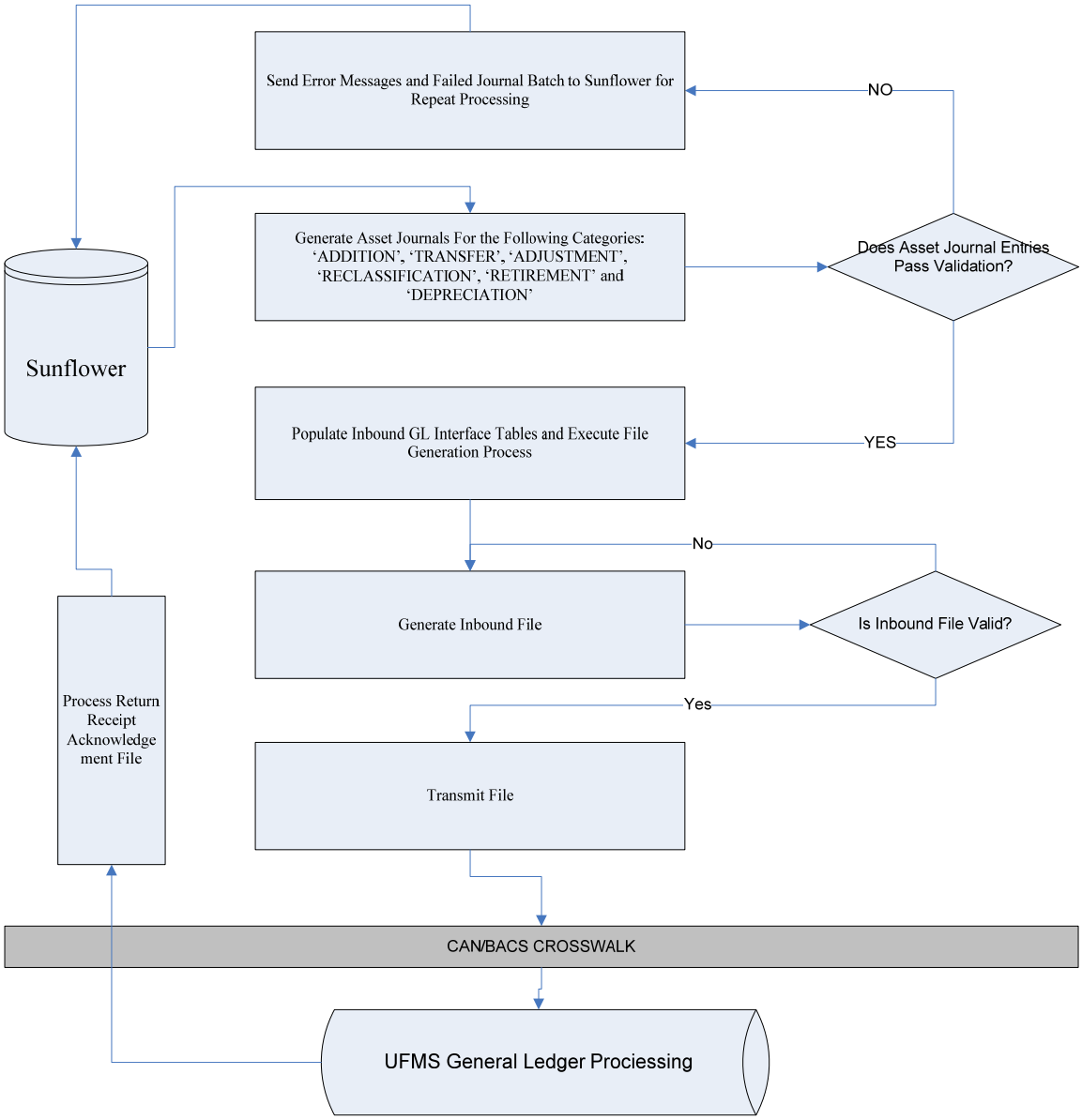
(2)The following asset journal categories shall be interfaced from Sunflower: Addition, Adjustments, Reclassification, Transfers, Retirements and Depreciation.

(3)The source for the asset journal records derives from Sunflower and will be transmitted to the UFMS General Ledger system as it is operated by HHS OPDIV.

3.1.1. Technical Flow

Figure 3 describes the technical information processing flow once it is determined by Sunflower that asset journal records are ready for transmission to the UFMS General Ledger system. In accord with the predetermined needs of the UFMS General Ledger system and Sunflower, processing will commence and complete at the end of every fiscal year month.

Figure 3: Technical Flow Diagram



3.1.2. Interface Steps

The following steps provide a supporting narrative for the technical functionality illustrated in Figure 3.

(1) At the closing of every month, an Oracle maintained job-scheduler initiates the processing of asset journal entries.

(2) Asset journal entries are generated for Addition, Adjustments, Reclassification, Transfers, Retirements and Depreciation.

(3) Validate on all necessary columns to ensure that they are populated. There is not semantic validation performed on the columns.

(4) Once validation (Step 3) is complete, all data columns are cascaded into their respective tables in preparation for file generation

(5) Generation of the UFMS inbound file commences. There are Oracle specific directories established which house the file upon generation. These Oracle specific directories are enumerated below in Section entitled Dependencies.

(6) There are minimal checks on the UFMS inbound file to ensure that the file is valid and ready for broadcast. Sunflower validates the file size to ensure the file is greater than 0 bytes in size.

(7) Based on system specific script, the UFMS inbound file is transmitted to the UFMS General Ledger system for processing.

(8) Sunflower expects confirmation of processing from the UFMS General Ledger system and updates the processing status of each record broadcast from the Sunflower system.

3.2. Approach

The following describes the interface steps and the validation overview.

3.2.1. Interface Tables

The following provides a list and description of the tables that are inserted into, extracted from, and used for data validation.

Table Name	Description
EXT_PMIS_UFMS_INBOUND_VALIDATE	This table contains all of the validation rules specific to processing asset journal entries generated for Addition, Adjustments, Reclassification, Transfers, Retirements and Depreciation.

3.2.2. Ordering of Tables

There is only one validation table used to maintain the rules used to minimally validate data before it is populated into the main processing table: EXT_PMIS_UFMS_INBOUND.Validation Overview

3.2.3. Dependencies

There are dependencies on which the successful processing of the UFMS General Ledger inbound file depends. These dependencies are catalogued below.

(1) The interface processing in Sunflower relies on successful addition, maintenance and evolution of asset entries in Sunflower.

(2) An Oracle scheduled job is responsible for the inception for the Sunflower interface processing. The job will be schedule once. The job will contain the

necessary parameterized information for repeat processing. The current requirement for kicking-off interface processing is month's end.

(3) The necessary directories are required for file creation. The current file system repositories are enumerated below:

DNS Name	File System Type	Oralce Directory Name	Directory Location
Bambi.psc.gov	Solaris 9	UFMS_INBOUND	/u04/home/sunflr/3800/appserver/ext/interfaces/ufms/inbound

(4) The system level script used to transmit the file to UFMS General Ledger. This process places the Sunflower generated file into the inbox folder for UFMS General Ledger processing.

(5) Sunflower expects a return receipt acknowledgement containing a mirrored set of records and their processing status. This data shall be included in a file broadcast from UFMS General Ledger system to Sunflower upon successful completing by the UFMS General Ledger system.

(6) Upon return receipt, Sunflower will update the processing status for all record broadcast and archive the result.

3.3. Assumptions

The necessary assumptions used while completing the interface are detailed below.

1. The interface processing in Sunflower relies on successful addition, maintenance and evolution of asset entries in Sunflower.
2. An Oracle scheduled job is responsible for the inception for the Sunflower interface processing. The job will be schedule once. The job will contain the necessary parameterized information for repeat processing. The current requirement for kicking-off interface processing is month's end.
3. The necessary directories are required for file creation. The current file system repositories are enumerated below:
4. The system level script used to transmit the file to UFMS General Ledger. This process places the Sunflower generated file into the inbox folder for UFMS General Ledger processing. Sunflower assumes that it will have access to write files to the UFMS General Ledger system.
5. Sunflower assumes that a return receipt acknowledgement containing a mirrored set of records and the processing status will be broadcast back to Sunflower upon successful UFMS General Ledger system processing.
6. Upon return receipt, Sunflower will update the processing status for all record broadcast and archive the result.

3.4. Inbound Program Logic

The inbound interface selects data from the Sunflower tables for entries generated for Addition, Adjustments, Reclassification, Transfers, Retirements and Depreciation.

3.4.1. Sunflower Interface Processing - Description

Below are the names and brief description of the program units used to selects data from the Sunflower tables for entries generated for Addition, Adjustments, Reclassification, Transfers, Retirements and Depreciation.

Program Unit	Datatype
EXT_PMIS_INBOUND_MANAGER	Manager unit responsible for ensuring that all aspects entry generation are processed.
EXT_PMIS_INBOUND_ADDITION	Unit responsible for generating all addition asset entries.
EXT_PMIS_INBOUND_ADJUSTMENTS	Unit responsible for generating all adjustment asset entries.
EXT_PMIS_INBOUND_DEPRECIATION	Unit responsible for generating all depreciation asset entries.
EXT_PMIS_INBOUND_RECLASS	Unit responsible for generating all reclassification asset entries.
EXT_PMIS_INBOUND_RETIREMENT	Unit responsible for generating all retirement asset entries.
EXT_PMIS_INBOUND_TRANSFERS	Unit responsible for generating all transfer asset entries.
EXT_PMIS_FILE_UTILITIES	Unit responsible for generating inbound file.

3.4.2. Inbound File Format

The following contains the format for the UFMS General Ledger inbound file.

Fields in the flat file are pipe ('|') delimited. The following file name convention is used:

File Name: - HHS_PMIS_CDC_GL_YYYYMMDDHHMISS.dat

HHS_PMIS_FDA_GL_YYYYMMDDHHMISS.dat

HHS_PMIS_PSC_GL_YYYYMMDDHHMISS.dat

HHS_PMIS_IHS_GL_YYYYMMDDHHMISS.dat

Table 2: General Ledger File Format

The following inbound Journal Asset transaction format will be received from Sunflower.

Field Name	Data type	Position	Required /Optional	Comments
Record Identifier	VARCHAR2(2)	1	R	Constant 'TX'
Record ID	NUMBER	2	R	Sequence for GL transaction in the flat file.
Set of Books Name	VARCHAR2(3)	3	R	Value Should be in ('CDC','FDA','PSC','IHS')

Currency Code	VARCHAR2(3)	4	R	Default Currency should be 'USD'
User JE Category Name	VARCHAR2(25)	5	R	Should be in Addition, Adjustment, Reclass, Transfer, Retirement and Depreciation
User JE Source Name	VARCHAR2(25)	6	R	Default Source should be 'SUNFLOWER'
Fiscal Year	VARCHAR2(150)	7	R	AS_FINANCIAL_ASSET_STATES.ATTRIBUTE_VALUE_1
Object Class	VARCHAR2(150)	8	R	AS_ASSET_STATES.ATTRIBUTE_VALUE_2
CAN	VARCHAR2(150)	9	R	AS_ASSET_EVENTS.ATTRIBUTE_VALUE_3
Account	VARCHAR2(150)	10	R	AS_JE_LINES.LEDGER_IDENTIFIER
Entered Debit	NUMBER	11	O	Required if Entered Credit Column is NULL
Entered Credit	NUMBER	12	O	Required if Entered Debit Column is NULL
Period Name	VARCHAR2(25)	13	R	Format MMM-CurrentYear-FY-FiscalYear E.g. DEC-06-FY-07
Transaction Date	DATE	14	R	Format DD-MON-YYYY E.g. 12-FEB-2007
Batch Name	VARCHAR2(150)	15	O	PMIS Asset Batch For <Category Name> - <Period Name>
Batch Description	VARCHAR2(150)	16	O	PMIS Asset Batch For <Category Name>
Journal Name	VARCHAR2(150)	17	O	PMIS Asset Journal for <Category Name>
Journal Description	VARCHAR2(150)	18	O	PMIS Asset Journal for <Category Name> and <Period>
JE Line Number	NUMBER	19	O	Journal Ledger Line Number
Line Description	VARCHAR2(150)	20	O	Document Number:<Document Number>
HHS Elimination Code	VARCHAR2(150)	21	O	If vendor is Federal
FACTS Trading Partner Code	VARCHAR2(150)	22	O	If vendor is Federal
GPRA code	VARCHAR2(3)	23	O	AS_ASSET_STATES.ATTRIBUTE_VALUE_16
Accounting Point	VARCHAR2(3)	24	O	Required only for 'IHS'
Task ID	NUMBER	25	O	Required only for 'IHS'

The following is the Trailer Record Format

Field Name	Data type	Comments
Record Identifier	VARCHAR2 (2)	Value will be 'TT'
Organization Code	VARCHAR2 (3)	Value will be 'CDC','FDA','PSC','IHS'
Total Number of journal batches in the extract	NUMBER	

Field Name	Data type	Comments
Total Number of lines in the extract	NUMBER	
Total journal Amount for the extract	NUMBER	Addition of total debits + total credits
Batch Date/SYSDATE	NUMBER	Format (YYYYMMDDHH24MISS) E.g. 20070212103245

3.4.3. Table and View Usage

Table Name	Description
EXT_PMIS_UFMS_INBOUND	This table serves to capture the processing of asset journal entries generated for Addition, Adjustments, Reclassification, Transfers, Retirements and Depreciation.
EXT_PMIS_UFMS_INBOUD_VALIDATE	This table contains all the validation rules specific to processing asset journal entries generated for Addition, Adjustments, Reclassification, Transfers, Retirements and Depreciation.
EXT_PMIS_UFMS_INBOUND_RECEIPT	This table temporarily contains data before it is archived. The purpose of this table is to update the 'Processed' column based on receipt data received from the UFMS General Ledger system.
EXT_PMIS_UFMS_INBOUND_HISTORY	This table is an archive of all previous data that has been broadcast to UFMS General Ledger system
EXT_PMIS_UFMS_INBOUND_LOG	This table maintains a detailed log of Sunflower's interface processing.

3.4.4. Program Logic

An Oracle schedule job is used to begin processing. Below is a sequential flow of the program units called.

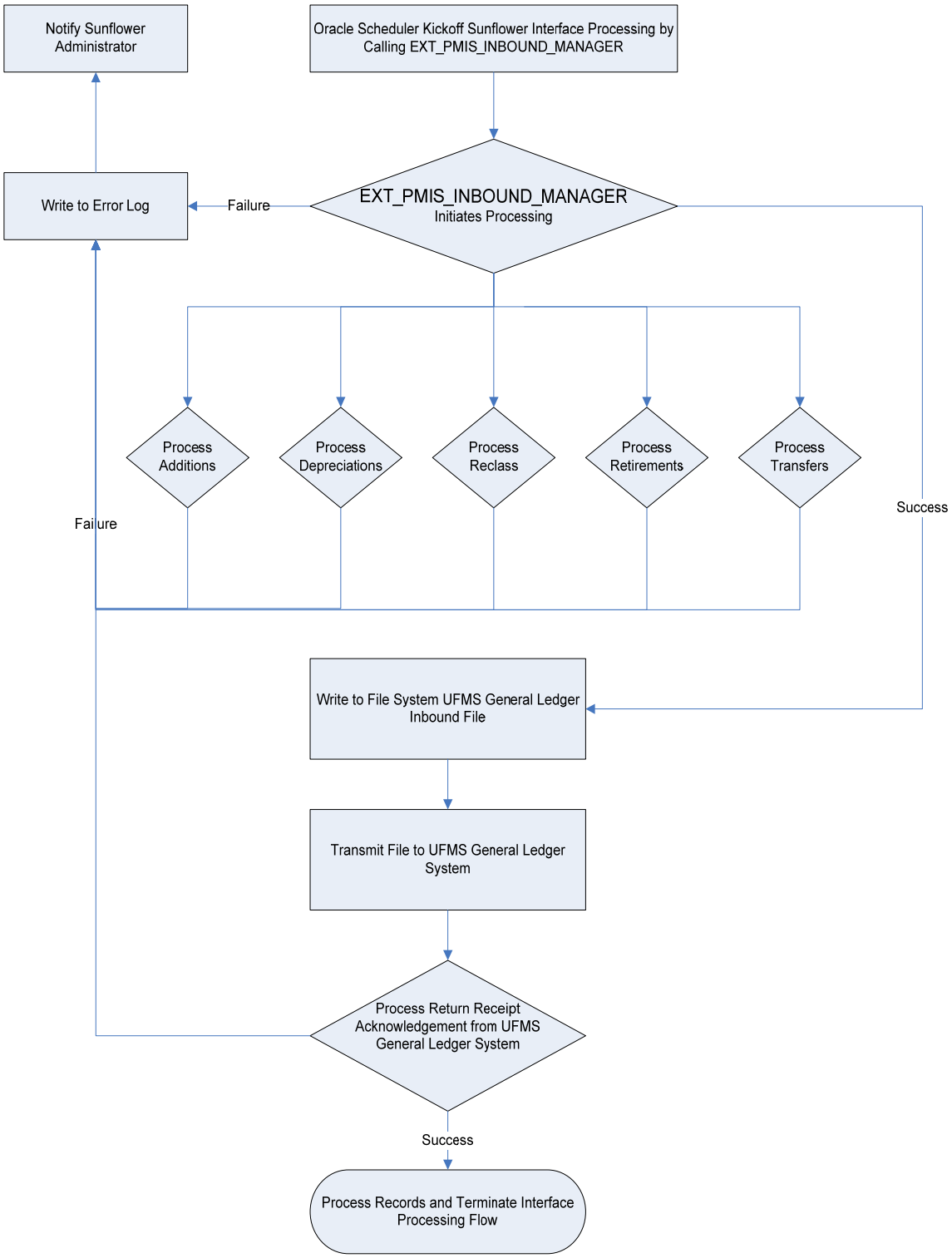
Program Unit	Datatype
EXT_PMIS_INBOUND_MANAGER	Manager unit responsible for ensuring that all aspects entry generation are processed.
EXT_PMIS_INBOUND_ADDITION	Unit responsible for generating all addition asset entries.
EXT_PMIS_INBOUND_ADJUSTMENTS	Unit responsible for generating all adjustment asset entries.
EXT_PMIS_INBOUND_DEPRECIATION	Unit responsible for generating all depreciation asset entries.
EXT_PMIS_INBOUND_RECLASS	Unit responsible for generating all reclassification asset entries.
EXT_PMIS_INBOUND_RETIREMENT	Unit responsible for generating all retirement asset entries.
EXT_PMIS_INBOUND_TRANSFERS	Unit responsible for generating all transfer asset entries.
EXT_PMIS_FILE_UTILITIES	After processing of asset entries is complete a file will

Program Unit	Datatype
	be generate to system specific directory.
EXT_PMIS_INBOUND_RECEIPT	Unit responsible for processing and validating UFMS General Ledger processing at the asset entry level.

1. EXT_PMIS_INBOUND_MANAGER manages processing for all program units.
2. The first programming unit called is EXT_PMIS_INBOUND_ADDITION. Upon selecting the relevant asset journal entries, the data is validated and inserted into the following table: EXT_PMIS_UFMS_INBOUND. If a failure occurs, notify the Sunflower system administrator.
3. The second unit called is EXT_PMIS_INBOUND_ ADJUSTMENTS. Upon selecting the relevant asset journal entries, the data is validated and inserted into the following table: EXT_PMIS_UFMS_INBOUND. If a failure occurs, notify the Sunflower system administrator.
4. The third unit called is EXT_PMIS_INBOUND_ DEPRECIATION. Upon selecting the relevant asset journal entries, the data is validated and inserted into the following table: EXT_PMIS_UFMS_INBOUND. If a failure occurs, notify the Sunflower system administrator.
5. The fourth unit called is EXT_PMIS_INBOUND_ DEPRECIATION. Upon selecting the relevant asset journal entries, the data is validated and inserted into the following table: EXT_PMIS_UFMS_INBOUND. If a failure occurs, notify the Sunflower system administrator.
6. The fifth unit called is EXT_PMIS_INBOUND_ RECLASS. Upon selecting the relevant asset journal entries, the data is validated and inserted into the following table: EXT_PMIS_UFMS_INBOUND. If a failure occurs, notify the Sunflower system administrator.
7. The sixth unit called is EXT_PMIS_INBOUND_ RETIREMENT. Upon selecting the relevant asset journal entries, the data is validated and inserted into the following table: EXT_PMIS_UFMS_INBOUND. If a failure occurs, notify the Sunflower system administrator.
8. The seventh unit called is EXT_PMIS_INBOUND_ TRANSFERS. Upon selecting the relevant asset journal entries, the data is validated and inserted into the following table: EXT_PMIS_UFMS_INBOUND. If a failure occurs, notify the Sunflower system administrator.
9. The eighth unit called is EXT_PMIS_FILE_UTILITIES. This code unit will ensure the generation of the UFMS General Ledger inbound file. The file will be placed in the Oracle pre-specified directory.
10. The ninth unit is an operating system level script that is responsible for transmitting the inbound file to the UFMS General Ledger system: EXT_PMIS_FILE_UTILITIES.
11. The tenth unit is responsible for checking for, and processing the return receipt acknowledgement file issued by the UFMS General Ledger system. If a failure occurs, notify the Sunflower system administrator.

Below is a flowchart of the Sunflower interface processing:

Figure 4: Sunflower Interface Processing



3.5. Database Design

There is only one temporary tables used for Sunflower's interface processing.

Table Name	Description
EXT_PMIS_UFMS_INBOUND_RECEIPT	This table temporarily contains data before it is archived. The purpose of this table is to update the 'Processed' column based on receipt data received from the UFMS General Ledger system.

3.6. Error Handling

Program failure may result during various stages of processing. The following supplies the reasons and the mitigation strategy for each failure. Additionally, the Sunflower interface processing maintains a logging strategy to capture processing: Both successful and any error transmission.

Table 3: Error Handling

Type of Error	Resolution
Oracle Job Scheduler Failure	If the Oracle Job Scheduler fails to kickoff the Sunflower interface process, Oracle will write to an Oracle specific error log detailing the reasons for failure.
INBOUND_MANAGER_ERROR	If the processing 'Manager', EXT_PMIS_INBOUND_MANAGER, detects a failure in processing, all error conditions and error descriptions will be captured in EXT_PMIS_UFMS_INBOUND_LOG. The 'Manager' will terminate processing and notify the Sunflower system administrator.
ADDITION_ERROR	If any time during the processing of 'Addition' assets an error occurs, the error will be captured, written to the log table, and propagated to EXT_PMIS_INBOUND_MANAGER. The EXT_PMIS_INBOUND_MANAGER will terminate processing and notify the Sunflower system administrator.
ADJUSTMENTS_ERROR	If any time during the processing of Adjustment assets an error occurs, the error will be captured, written to the log table, and propagated to EXT_PMIS_INBOUND_MANAGER. The EXT_PMIS_INBOUND_MANAGER will terminate processing and notify the Sunflower system administrator.
DEPRECIATION_ERROR	If any time during the processing of Depreciation assets an error occurs, the error will be captured, written to the log table, and propagated to EXT_PMIS_INBOUND_MANAGER. The EXT_PMIS_INBOUND_MANAGER will terminate

Type of Error	Resolution
	processing and notify the Sunflower system administrator.
RECLASS_ERROR	If any time during the processing of Reclassification assets an error occurs, the error will be captured, written to the log table, and propagated to EXT_PMIS_INBOUND_MANAGER. The EXT_PMIS_INBOUND_MANAGER will terminate processing and notify the Sunflower system administrator.
RETIREMENT_ERROR	If any time during the processing of Retirement assets an error occurs, the error will be captured, written to the log table, and propagated to EXT_PMIS_INBOUND_MANAGER. The EXT_PMIS_INBOUND_MANAGER will terminate processing and notify the Sunflower system administrator.
TRANSFERS_ERROR	If any time during the processing of Transfer assets an error occurs, the error will be captured, written to the log table, and propagated to EXT_PMIS_INBOUND_MANAGER. The EXT_PMIS_INBOUND_MANAGER will terminate processing and notify the Sunflower system administrator.
UTL_FILE_ERRORS	If an error occurs during the generation of the UFMS General Ledger inbound file, processing will terminate and the error will be captured, written to the log table, and propagated to EXT_PMIS_INBOUND_MANAGER. The EXT_PMIS_INBOUND_MANAGER will terminate processing and notify the Sunflower system administrator.
TRANSMISSION ERRORS	If an error occurs during transmission of the application, i.e., network failure, the Sunflower interface administrator is notified the next day.
RETURN_RECEIPT_ERRORS	If an error occurs during processing of return receipt acknowledgement file, the Sunflower interface administrator will be notified.

3.6.1. Restart Strategy

Should the Sunflower interface administrator be notified of any failures occurring during any stage of the processing, another ‘Manager’ process will be schedule. This entails processing all asset entries and regeneration of UFMS General Ledger inbound file for the appropriate month and resending the appropriate file(s).

In the event that the errors cannot be corrected by the closing of the UFMS accounting period, or in the case of system failure, the accounting period in question will be combined with the subsequent accounting period’s data.

3.6.2. Restart Procedures

During the execution of the PMIS UFMS Inbound file process, two error types have been identified. The below table provides details on how the O&M team will approach resolving interface problems as they arise. Note: due to the sensitive nature of individual contact

information, the appropriate contacts for each OPDIV will be maintained in a separate document.

Table 4: Error Handling Procedures

Type of Error	Resolution
File Generation Errors	All file generation errors result in an email being generated to the main Metrix O&M email address. Depending on the cause of the error, appropriate action is taken to correct the error and the file is regenerated for delivery to the UFMS drop box.
Financial data errors	If errors are detected during the processing of the file in UFMS, an acknowledgement file is generated and sent to PMIS. Each acknowledgement file is processed in PMIS against the original source data. If errors are detected, an email is sent to the central Metrix O&M email address. The types f errors usually detected in this process are as follows: ▪ Expired CAN, ▪ Incorrect CAN FY combination, ▪ Period not available for posting. For these types of errors, the appropriate OPDIV POC is contacted via email. The email will contain sufficient information for the OPDIV POC to correct the affected assets. Once Metrix has been notified that the corrections have been made in PMIS, Metrix will then regenerate the file and transmit to UFMS for processing. Of Note, these types of error usually require UFMS to clear the staging tables in the UFMS production instance. The OPDIV POC is responsible for opening a trouble ticket with the UFMS O&M team prior to re-processing the file.