

caseload



Arizona Department of Health Services

Bureau of Nutrition and Physical Activity

HANDS WIC System

Detailed Functional Design Document

End of Day (EOD)

June 16, 2015

Version 1.4

END OF DAY (EOD)	1
REVISION HISTORY	4
1 END OF DAY OVERVIEW	5
1.1 End Of Day System Diagram	5
1.2 End of Day Process Flow	5
1.3 Reviewing EOD Report/Log Files	7
2 FOOD BENEFIT ISSUANCE PROCESSING	12
2.1 Stale Dating of Food Benefits	12
2.2 Obligation of Food Benefits	12
2.3 Creation of Issuance File for the Financial Institution	12
3 TRANSMISSION OF FILES BETWEEN STATE AND FINANCIAL INSTITUTION	14
3.1 Issuance File	14
3.2 Peer Group Average and Not to Exceed	14
3.3 Vendor List	14
3.4 Farmers Market Coupons	14
3.5 Retrieve Food Benefit Payments	14
4 FOOD BENEFIT REDEMPTION PROCESSING	15
4.1 Rejected Food Benefits	15
4.2 Paid Food Benefits	15
4.3 Missing Issuance	15
5 AUTOMATED PARTICIPATION TERMINATION AND CHANGE PROCESSING	16
5.1 Participant Termination	16
5.2 Category changes	16
5.3 Category Synchronization	17
6 APPOINTMENT SCHEDULE PROCESSING	18
6.1 Assign Missing Nutritional Education Topics to Clients	18
6.2 Update Checkout Time	18
6.3 Set Appointment Status to Attended	18
6.4 Set Appointment Status to Missed	18
7 CASELOAD DETERMINATION PROCESSING	19
7.1 Determine Active Certifications	19

7.2	Determine Clients with Active Certifications that Received Food Benefits	20
7.3	Rebuild Caseload Details	21
7.4	Rebuild Caseloads.....	31
8	FINANCIAL SUMMARY PROCESSING	32
8.1	Building Food Benefit Obligation Summary	32
8.2	Building Food Benefit Outlays	32
8.3	Building Divisibility Records	32
8.4	Building Food Benefit Rebate Details	33
8.5	Building Food Benefit Peer Group Averages	34
9	STATE CLIENTS PROCESSING	37
10	UPDATE BUDGET FACTORS.....	38
10.1	Monthly Redemption Percentages:	38
10.2	Actual redemption Participation:	38
10.3	Actual Food Package Inflation Factor:	38
10.4	Actual Pre-Rebate Food Package Cost:	38
10.5	Actual Post-Rebate Food Package Cost:.....	38
10.6	Actual Rebate per Client:.....	39
10.7	Actual Redemption Factor:	39
11	REPLENISH FOOD INSTRUMENT INVENTORIES	40
12	LOAD LOCAL AGENCY & CLINIC FOOD BENEFIT AVERAGES	41
13	CLEANUP LOGS	42

REVISION HISTORY

The chart below indicates revisions made to this document:

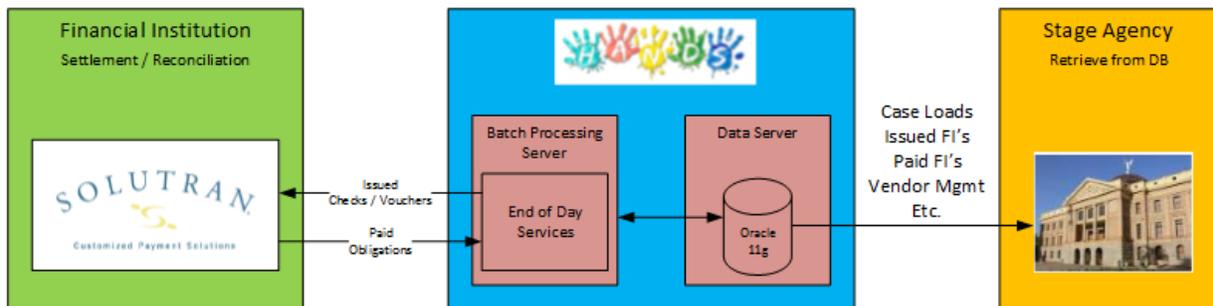
<u>Version</u>	<u>Name</u>	<u>Brief Description of Change</u>	<u>Published</u>
1.0	Steven Kirchner	Initial Draft	10/6/2014
1.1	Steven Kirchner	Caseload and FI Inventories Revision	2/24/2015
1.2	Chris Walker	Content Edits	3/5/2015
1.3	Chris Walker	Section 8.5, Peer Group Averages Calculation details updated	6/5/2015
1.4	Chris Walker	Update section 7.2 to support the 'T' Void Reason Code	6/16/2015

1 END OF DAY OVERVIEW

The End of Day (EOD) process is designed to be run nightly after clinic operations have ceased for the day. It is imperative that all clinic operations are stopped. EOD is broken up into several sections of work to be done. Each of these sections will be described in detail.

- Food Benefit Issuance Processing
- Transmission of Files between State and Financial Institution
- Food Benefit Redemption Processing
- Automated Participation Termination and Change Processing
- Appointment Scheduler Processing
- Caseload Determination Processing
- Financial Summary Processing
- State Clients Processing
- Update Budget Factors
- Replenish Food Instrument Inventories
- Load Local Agency & Clinic Food Benefit Averages
- Cleanup logs

1.1 End Of Day System Diagram



1.2 End of Day Process Flow

The End of Day (EOD) batch processing routines are executed once per day at a time of day chosen by the state agency. Batch processing consists of a series of sub-routines that follow an order of operation coordinated by the main windows batch file (HNDZ_EOD.BAT) responsible for controlling the entire EOD batch process work flow. The following table details the EOD sub-routines and the order of operation.

Order	Calling Module / Script	Processes	Script Name	LA	HANDS	Bank
Start	WINDOWS Scheduler	Runs the End of Day batch processing controller (HNDZ_EOD.BAT) at the configured time of day. The HNDZ_EOD.BAT controls the entire central EOD process.	HNDZ_EOD.BAT			

Order	Calling Module / Script	Processes	Script Name	LA	HANDS	Bank
1.0	HNDZ_EOD.BAT	Build issuance data files and send to the financial institution (FI)	HEOD_ISSUE.SQL			
1.1	HEOD_ISSUE.SQL	Issuance data preparation	EODADM.HANDS_EOD_ISSUE.EOD_MAIN			
1.1.1	EODADM.HANDS_EOD_ISSUE.EOD_MAIN	Void food instruments that are past their stale date	STALE_DATE_FIS		x	
1.1.2	EODADM.HANDS_EOD_ISSUE.EOD_MAIN	Compile direct payment and revalidated FI information	OBLIGATE_FIS		x	
1.1.3	EODADM.HANDS_EOD_ISSUE.EOD_MAIN	Executes INSERT_INTO_BANK_REPORTS	BUILD_CHECK_ISSUANCE		x	
1.1.4	EODADM.HANDS_EOD_ISSUE.EOD_MAIN	Insert into I_BANK_REPORTS	INSERT_INTO_BANK_REPORTS		x	
1.2	HEOD_ISSUE.SQL	Create bank reports data file	ISSUE_FSMC.SQL		x	
1.3	HEOD_ISSUE.SQL	Create WIC farmers market coupons file	ISSUE_FMFSMC.SQL		x	
1.4	HEOD_ISSUE.SQL	Create WIC farmers market checks file	FARMER_FSMC.SQL		x	
1.5	HEOD_ISSUE.SQL	Executes VENDOR_FSMC.BAT	VENDOR_FSMC.BAT		x	
1.5.1	VENDOR_FSMC.BAT	Create vendor data file	VENDOR_FSMC.SQL		x	
2.0	HNDZ_EOD.BAT	Execute file transfers	HEOD_TRANSFER_FILES.BAT			
2.1	HEOD_TRANSFER_FILES.BAT	Transmit the issuance text files to the bank	FTPTRANS.EXE			x
2.2	HEOD_TRANSFER_FILES.BAT	Load the redemption data received from the bank	IMPORT_FILE.BAT			x
2.3	HEOD_TRANSFER_FILES.BAT	Load the farmers market data received from the bank	LOAD_FM_FILE.BAT			x
2.4	HEOD_TRANSFER_FILES.BAT	Send the Vendor Peer Group Averages Data to the bank	PGA_FSMC.SQL			x
2.5	HEOD_TRANSFER_FILES.BAT	Transmits the peer group averages data to the bank	FTPPGA.EXE			x
2.6	HEOD_TRANSFER_FILES.BAT	Create state's peer group averages data	OUTBOUND_PGA_FSMC.SQL		x	
2.7	HEOD_TRANSFER_FILES.BAT	Creates state's vendor data	OUTBOUND_VENDOR_FSMC.SQL		x	
3.0	HNDZ_EOD.BAT	End of day completion steps	HEOD_POST.SQL			
3.1	HEOD_POST.SQL	End of day completion steps	EODADM.HANDS_EOD_POST.EOD_MAIN			
3.1.1	EODADM.HANDS_EOD_POST.EOD_MAIN	Create vendor payment records	POST_BANK_DATA_EOD		x	
3.1.2	EODADM.HANDS_EOD_POST.EOD_MAIN	Create consolidated record of all farmers' market coupons issuance and redemption information.	POST_FARMERS_COUPONS		x	
3.1.3	EODADM.HANDS_EOD_POST.EOD_MAIN	Participant termination	EOD_TERMINATE	x	x	
3.1.4	EODADM.HANDS_EOD_POST.EOD_MAIN	Category changes	EOD_CATEGORY_UPDATES	x	x	
3.1.5	EODADM.HANDS_EOD_POST.EOD_MAIN	Category synchronization	EOD_CATEGORY_SYNCS		x	
3.1.6	EODADM.HANDS_EOD_POST.EOD_MAIN	Rebuild Obligations	REBUILD_OBLIGATIONS		x	
3.1.7	EODADM.HANDS_EOD_POST.EOD_MAIN	Rebuild Outlays	REBUILD_OUTLAYS		x	
3.1.8	EODADM.HANDS_EOD_POST.EOD_MAIN	Rebuild Rebate Details	REBUILD_REBATE_DETAILS		x	
3.1.9	EODADM.HANDS_EOD_POST.EOD_MAIN	Store FI Divisibility Records	EOD_FI_DIVISIBILITY		x	
3.1.10	EODADM.HANDS_EOD_POST.EOD_MAIN	Rebuild Active Certifications	REBUILD_ACTIVE_CERTS		x	
3.1.11	EODADM.HANDS_EOD_POST.EOD_MAIN	Rebuild Client Part Base	REBUILD_CLIENT_PART_BASE		x	
3.1.12	EODADM.HANDS_EOD_POST.EOD_MAIN	Rebuild Caseload Details	REBUILD_CASELOAD_DETAILS		x	
3.1.13	EODADM.HANDS_EOD_POST.EOD_MAIN	Rebuild Caseloads	REBUILD_CASELOADS		x	

Order	Calling Module / Script	Processes	Script Name	LA	HANDS	Bank
3.1.14	EODADM.HANDS_EOD_POST.EOD_MAIN	Move data to Caseload Details	MOVE_DATA_TO_CASELOAD_DETAILS		x	
3.1.15	EODADM.HANDS_EOD_POST.EOD_MAIN	Move data to Caseloads	MOVE_DATA_TO_CASELOADS		x	
3.1.16	EODADM.HANDS_EOD_POST.EOD_MAIN	Rebuild Caseloads CSFP	REBUILD_CASELOADS_CSFP		x	
3.1.17	EODADM.HANDS_EOD_POST.EOD_MAIN	Move data to Caseload CSFP	MOVE_DATA_TO_CASELOADS_CSFP		x	
3.1.18	EODADM.HANDS_EOD_POST.EOD_MAIN	Rebuild Peer Group Averages	REBUILD_PEER_GROUP_AVGS		x	
3.1.19	EODADM.HANDS_EOD_POST.EOD_MAIN	Update State Clients	UPD_STATE_CLIENTS		x	
3.1.20	EODADM.HANDS_EOD_POST.EOD_MAIN	Update appointments	UPDATE_APPT_TABLES	x	x	
3.1.21	EODADM.HANDS_EOD_POST.EOD_MAIN	Update Budgets	UPD_BUDGETS		x	

1.3 Reviewing EOD Report/Log Files

1.3.1 Log Files

The end of day HANDS files can be accessed via the Windows Explorer and entering [\\ppveodwv00\c\\$\eod](\\ppveodwv00\c$\eod) to log into the HANDS EOD server.

Note – Refer to section 1.2, End of Day Process Flow for details regarding the process flow.

1. The current paid log files located under the 'eod\database name\data\load' directory must be checked to confirm that all records are successfully loaded into the table AIM.E_BANK_DATAS. The example below shows the paid log file located at 'eod\hndnav\data\load'. This log file displays 1734 rows successfully loaded. The 1 Row not loaded is the Total Record Count Row and this should be ignored.

The ONLY row that would show any errors is the one below. These counts should be included in the daily HANDS error message email to show that the data was successfully loaded.

"0 Rows not loaded due to data errors."

Example of a PAID LOG FILE WITH NO ERRORS

pdnt0105.log:

Record 1735: Discarded - failed all WHEN clauses.

Table "AIM"."E_BANK_DATAS":

1734 Rows successfully loaded.

0 Rows not loaded due to data errors.

1 Row not loaded because all WHEN clauses were failed.

0 Rows not loaded because all fields were null.

Table "AIM"."E_BANK_DATAS":

0 Rows successfully loaded.

0 Rows not loaded due to data errors.
1735 Rows not loaded because all WHEN clauses were failed.
0 Rows not loaded because all fields were null.

Example of a PAID LOG FILE WITH ERRORS

pdnt0102.log:

Record 768: Rejected - Error on table "AIM"."E_BANK_DATAS", column
CC_CLIENT_ID.
ORA-01438: value larger than specified precision allowed for this column

MAXIMUM ERROR COUNT EXCEEDED - Above statistics reflect partial run.

Table "AIM"."E_BANK_DATAS":

676 Rows successfully loaded.

92 Rows not loaded due to data errors.

0 Rows not loaded because all WHEN clauses were failed.

0 Rows not loaded because all fields were null.

Table "AIM"."E_BANK_DATAS":

0 Rows successfully loaded.

0 Rows not loaded due to data errors.

768 Rows not loaded because all WHEN clauses were failed.

0 Rows not loaded because all fields were null.

All processed files can be found in the 'eod\database name\data\processed' directory. The data files naming convention provides indication of the file data type based on the prefix of the file name. The file data name prefixes and subsequent data types include:

- **is** – The issued files
- **pd** – The paid files
- **pga** – The peer group average files
- **rpt** – The auditing reports
- **ven** – The vendor files.

The log files are located at 'eod\database name\eod_history'.

Log Files that should be checked daily for existence and reviewed for errors events:

2. Click on the date to review that the following log files are present in the directory:*Fsmc_ftp.log –Confirm that there are no error messages. An exception is the Farmer's market load file which currently only pertains to HANDS and Guam.

3. *Pga_fsmc_ftp.log –Confirm that there are no error messages.

Log Files that should be checked daily for existence “Only”:

4. Heod_issue_sql.log – displays the start/finish dates and times for each executed procedure. Also shows the initial eod_controls dates.
5. Heod_post_sql.log - displays the start/finish dates and times for each executed procedure. Also shows the updated eod_controls dates.
6. Issue.log – displays the start/finish date and times for the heod_issue.sql procedure.
7. Post.log - displays the start/finish date and times for the heod_post.sql procedure.
8. Eod_date.log – logs the date and time eod_controls were updated
9. Transfer.log - displays the start/finish date and times for the transferred files. This is a good resource if you find error messages in the fsmc_ftp.log.

Note – Error conditions discovered upon reviewing the Log Files should be consolidated into a single report and emailed to the WIC Hands support team:

- WIC Help Desk – wicservicedesk@azdhs.gov
- Steve Kirchner – skirchner@cma.com
- Joe Surls – jsurls@cma.com (cc)
- Fenris Daniel – fdaniel@cma.com (cc)
- Beth Graves – bgraves@cma.com (cc)

1.3.2 EOD Error Reports

A report is generated and emailed to the HANDS help desk for review for each Partner’s database as well as HANDS.

This report runs the following for the previous two days:

1. All distinct error message located in the HANDS_EOD_LOGS table. This includes the following fields: procedure name, good proc date, date created, and error message.
2. All Issued Checks NOT Uploaded to i_food_instruments. This includes the following fields: serial number and issue date.
3. All Paid Checks NOT Uploaded to i_food_instruments. This includes the following fields: serial number, process date, and redemption amount.

Note – These EOD Error Reports are automatically emailed to the HANDS Help Desk Support team.

This report displays HANDS EOD Errors based on a start/end date range.

Report Date: 11/17/2014 10:49:39 AM



American Samoa EOD Status

From: 11/8/2014 To: 11/17/2014

Database: HND\$AM
 User: WIC\isleckmannf
 Page 1 of 113

Errors in HANDS_EOD_LOGS Table

Procedure Name	Good Proc. Date	Date Created	Error Message
FINSIH-REBUILD_CASELOADS_C SFP	11/1/2014	11/16/2014	
INSERT INTO BANK REPORTS		11/12/2014	ORA-00001: unique constraint (AIM.IBR_PK) violated
REBUILD_OBLIGATIONS	10/1/2014	11/11/2014	ORA-02292: Integrity constraint (AIM.FOL_FO_FK) violated - child record found
REBUILD_OUTLAYS	10/1/2014	11/12/2014	ORA-02292: Integrity constraint (AIM.FRD_FOL_FK) violated - child record found
REBUILD_REBATE_DET A ILS	9/1/2014	11/13/2014	ORA-02291: Integrity constraint (AIM.FRD_FOL_FK) violated - parent key not found

Issued Checks NOT Uploaded to I_food_instruments

Serial Number	Issue Date
0001967094	14-NOV-14
0001967095	14-NOV-14
0001967096	14-NOV-14
0001967097	14-NOV-14

Paid Checks NOT Uploaded to I_food_instruments

Serial Number	Process Date	Redemption Amount
28920	13-NOV-14	\$41.68
28921	13-NOV-14	\$41.68
28922	13-NOV-14	\$62.52
29000	13-NOV-14	\$42.50
29001	13-NOV-14	\$49.90
29002	13-NOV-14	\$4.00
29003	13-NOV-14	\$4.00
29013	13-NOV-14	\$41.70
29014	13-NOV-14	\$41.70
29015	13-NOV-14	\$62.55

1.3.3 Deletion of Processed Paid Files

The redemption files from the bank are stored on the EOD server for 12 months, then archived to disk before being deleted. All paid information received from the bank is stored in the data exactly as it arrived from the bank for verification and reconciliation tasks.

1.3.4 EOD Job Scheduler

Batch windows files are utilized for the creation of the bank text files. The HNDZ_EOD.bat file in each database directory are called by a scheduled job located on the EOD Server. The scheduled jobs can be accessed by clicking on the 'Server Manager' icon, then clicking on 'Task Scheduler' in the Tools dropdown.

The current schedule is listed below and may change at any time:

- EOD_HNDZ Everyday 9:00 PM
- EOD_HNDCNM Sunday – Thursday 3:20 AM

- | | | |
|---------------|-------------------|---------|
| 3. EOD_HNDGU | Sunday – Thursday | 3:00 AM |
| 4. EOD_HNDSAM | Monday – Friday | 1:00 AM |
| 5. EOD_HNDNAV | Sunday – Thursday | 8:45 PM |

Note: The QA and UAT databases do not execute the bank financial processing.

- | | | |
|---------------|-----------------|---------|
| 6. EOD_TSTHAZ | Sunday – Friday | 8:30 PM |
| 7. EOD_UATHAZ | Monday – Friday | 8:45 PM |

A weekly Guam Not to Exceed File is created by the 'RUN_NTE_EXTRACT.BAT' file which resides at 'c:\eod\hndgu\NTE_email'. This file automatically generates an email message which is currently sent to skirchner@cma.com. This email must be forwarded to all those listed every Thursday. Below is the scheduled job which executes this file:

- | | | |
|-------------------|----------------|---------|
| 8. GUAM_NTE_EMAIL | Every Thursday | 8:00 AM |
|-------------------|----------------|---------|

2 FOOD BENEFIT ISSUANCE PROCESSING

Food benefit processing is broken out into several procedures.

- Stale Dating of Food Benefits
- Obligation of Food Benefits
- Creation of Issuance File for the Financial Institution

2.1 Stale Dating of Food Benefits

The first thing EOD does is to stale date any food benefit that has not been redeemed or voided before the stale date of the food benefit is reached. The stale date for each food benefit is calculated at the time of issuance, it is 67 days after the first date to use. Stale dated food benefits will be sent to the bank as voided records.

2.2 Obligation of Food Benefits

EOD then obligates any food instruments that are not redeemed or voided that do not currently have an obligated amount. The obligation amount is based on a three month average of redeemed food benefit amounts. There is a hierarchical scheme to the process of determining the obligation amount.

- **Level One** – Verify if 10 or more food benefits have been cashed in the prior three months for the same FI Type, Category, Priority, and Clinic. If yes then use the three month average as the obligation amount, if not go to level 2.
- **Level Two** – Verify if 10 or more food benefits have been cashed in the prior three months for the same FI Type, Category and Priority. If yes use the three month average as the obligation amount, if not the go to level three.
- **Level Three** – Verify if 10 or more food benefits have been cashed in the prior three months for the same FI Type and Category. If yes then use the three month average as the obligation amount, if not then go to level four.
- **Level Four** – Verify if 10 or more food benefits have been cashed in the prior three months for the same FI Type. If yes then use the three month average as the obligation amount, if not then go to level five.
- **Level Five** – Uses the Annual Food Benefit amount from the Annual Factors table to determine the obligation amount.

2.3 Creation of Issuance File for the Financial Institution

EOD generates a list of Food Benefits that have been issued or voided since the last time EOD was executed. The process creates a fixed width flat file with the following fields.

- **Serial Number** – Unique Identifier of the clients food benefits
- **Client Id** – Unique identifier of the WIC Client

- **First Date To Use** – The first calendar day that the food benefit can be used
- **Last Date To Use** – The last calendar day that the food benefit can be used for purchase
- **FI Type** – Identifier of the food items the client is allowed to purchase
- **Org Code** – Identifier of which clinic the client is associated with
- **Issue Date** – The calendar date when the food benefit was issued
- **Issue Method** – How the Food Benefit was issued. (Check, EBT, Replacement, ...)
- **Maximum Amount** – The food benefit cannot exceed this amount for this Food Benefit

3 TRANSMISSION OF FILES BETWEEN STATE AND FINANCIAL INSTITUTION

All files transferred to the financial institution are performed over the secure File Transfer Protocol (sFTP). The files transferred to the financial institution include:

- Issuance File – Daily except Sunday
- Peer Group Average and Not to Exceed File – Daily except Sunday
- Not to Exceed File – Weekly, Thursday
- Vendor List – Daily except Sunday
- Farmers Market Coupons – Daily except Sunday
- Retrieve Food Benefits Coupons – Daily Except Sunday

3.1 Issuance File

The issuance file contains the list of food benefits issued and voided for the given processing date. The Issuance File is sent daily except Sunday.

3.2 Peer Group Average and Not to Exceed

The peer group average contains weekly food benefit redemption averages for every food benefit type that has been cashed by a vendor in the defined peer groups. Peer group averages are calculated weekly during Thursday's EOD processing cycle. The peer group averages are used to generate the Not To Exceed file. The Not to Exceed file is sent daily except Sunday.

3.3 Vendor List

The Vendor List file contains a list of all vendors in the system and their status (authorized or unauthorized). The Vendor List file is sent daily except Sunday.

3.4 Farmers Market Coupons

The Farmers Market Coupons file contains all issuance information for the Farmers Market program for the day being processed. The information contained in the file is the same as the regular check issuance file. The Farmers Market Coupons file is sent daily except Sunday.

3.5 Retrieve Food Benefit Payments

After the issuance files have been transferred the daily paid file from the bank is downloaded from the financial institution and placed on the End of Day (EOD) server. The paid file's name is based on the State Program and date of the file (i.e. PDAN1001.dat -> Paid file for Arizona WIC for the day of November 10th). AN-Arizona, GU-Guam, AM-American Samoa, NM-CNMI, NN-Navajo Nation.

4 FOOD BENEFIT REDEMPTION PROCESSING

The paid file received from the bank is loaded into the E_BANK_DATAS table via the Oracle SQLLOADER application. Once the paid file is loaded, a PL/SQL job runs to copy the redemption data to a master paid table (E_BANK_DATAS_MASTER) to keep a historical record of the redemption data loaded from the financial institution. Redemption records fall into two basic categories within the HANDS system, paid food benefits or rejected food benefits.

4.1 Rejected Food Benefits

If a food benefit has been rejected by the bank the process creates a record in the I_REJECT_REASONS table for the food benefit and updates the I_FOOD_INSTRUMENTS table with the rejection date and requested amount along with changing the disposition code to "5".

4.2 Paid Food Benefits

If a food benefit is paid (redeemed) by the bank then the process updates the I_FOOD_INSTRUMENTS table with redemption amount, cleared date, vendor id and peer group id along with changing the disposition code to "4".

4.3 Missing Issuance

If a food benefit is paid by the bank, but there is no record of the issuance in the HANDS system, a skeleton record is created to ensure the payment is recorded in the system. The skeleton record contains the minimum information required:

- Serial Number Cleared Date
- Process Date
- Redemption Amount
- Issuance Month
- FI Type (system defined default value)
- Vendor ID
- Peer Group
- Date Created
- Date Modified

Note – All of the information in the skeleton record either comes from the paid record or is populated with a default value.

Reason for Missing Issuance Record – Given that HANDS consists of a single centralized database the most plausible cause for a missing issuance record is a misread MICR number on the check.

5 AUTOMATED PARTICIPATION TERMINATION AND CHANGE PROCESSING

5.1 Participant Termination

During the End of Day process automatic terminations¹ are performed on participants no longer eligible to participate in the State's WIC program as follows:

- The WIC participant is postpartum (not breastfeeding) and today's date is 180 days after the actual delivery date.
- The CSF participant is postpartum (not breastfeeding) and today's date is the last day of the calendar month of the Actual Delivery date +12 calendar months.
- The WIC participant is breastfeeding and today's date is 365 days after the actual delivery date.
- The CSF participant is breastfeeding and today's date is the last day of the calendar month of the Actual Delivery date + 12 calendar months.
- The WIC participant is 5 years of age and today's date is the last day of the calendar month in which the participant turned 5 years of age.
- The CSF participant is 6 years of age and today's date is the last day of the calendar month in which the participant turned 6 years of age.
- The WIC or CSF participant's current certification period has expired and there is not a new certification period.

The Client's records are updated to Inactive when:

- The termination date is not null in both the client and certification record.
- The termination date is not null in the certification record and the latest termination date < current date – 7 days.

5.2 Category changes

This script performs automatic category changes for clients based upon their certification category codes, current date and birth date.

The following steps are performed when a category change is performed:

1. The existing active certification is terminated.
A new active certification is created with the certification start date of the current date + 1, the certification end date of the terminated record. The certification start date of the previous certification is carried over so that the respective "data_dates" point to the prior certification so that the complete_record check passes when printing checks for the client later on.
2. Inserts records into the following tables:

¹ For all Participant records terminated all non-certification future appointments for the participant are deleted for the End of Day process.

- c_infant_child_medicals
 - c_w_medicals
 - c_bloodwork_data
 - c_i_c_healths
 - c_w_healths
 - c_health_risk_factors
3. Transfers the prior active food package/s over and then updates the food packages end date with birthday - 1.
 4. Updates the category code from EN/PN to 'PN+' if the infant is an IPN+.

The following table outlines updates that take place on the client certification record (does not apply to clients classified in the checkout clinic status):

Old code	Months between current month and birth date	New code
IEN, IPN, IPN+, IFF	12	C1
C1	24	C2
C2	36	C3
C3	48	C4
C4	60	C5

5.3 Category Synchronization

The Client record's category code is synced with the certification record's category code. In the case of a future certification, the client's category code maintains the current certification category until the current certification period ends. When the current certification period ends, the end of day process will change the client's category code to equal the new certification's category code.

6 APPOINTMENT SCHEDULE PROCESSING

The Appointment Schedule Processing performs several actions:

- Assign Missing Nutritional Education Topics to Clients
- Update Checkout Time
- Set Appointment Status to Attend
- Set Appointment Status to Missed

6.1 Assign Missing Nutritional Education Topics to Clients

For each active group scheduled for today or in the past with a group client status as “Scheduled”, ensure that the care plan record of each client that attended the group topic is updated to reflect client’s attendance.

6.2 Update Checkout Time

Update all checkout times where the appointment dates are in the past and the status code is “Scheduled” and the checkout time of the appointment is not already set.

6.3 Set Appointment Status to Attended

Update all appointment status to “Attended” where either of these conditions exists:

1. The appointment date is in the past with a status code of “Scheduled”, and the checkout time is blank.
2. The appointment date is in the past with a status code of “Scheduled”, and the checkout time is NOT blank, then set appointment status to “Attended” for ALL appointments.

6.4 Set Appointment Status to Missed

Update all Appointment Status to “Missed” when the following conditions exist:

1. The appointment date is in the past and the appointment does not exist in the appointment Check-in table.
2. The appointment’s start date and time is in the past and has a status code of “Scheduled” along with an active status.

7 CASELOAD DETERMINATION PROCESSING

The Caseload Determination Process performs several steps in order to determine the State's enrollment and participation details for a given month. The steps taken include:

- Determine Active Certifications
- Determine Clients with Active Certifications that Received Food Benefits
- Rebuild Caseload Details
- Rebuild Caseloads

Important Notes on Caseload Determination

Clinic Note

A client is counted in the last clinic that serviced the client. If a client was transferred in during a certification period the client is counted in the last receiving clinic for the month being processed. Prior months the client will be counted in the former clinic.

Category Note

The client category used for caseload is based on the category of the current certification of the month being processed. If a client changes category during the month the new category will be used as the client category for the month.

For certification periods covering a participant's transition from infancy into childhood, the State agency must provide the participant with an infant food package and report her as an infant participant on the FNS-798 through the calendar month or rolling month during which she reaches one year old. Beginning with the following month, the State agency must change the category of the participant from infant to child, provide her with a child food package, and report her as a child participant on the FNS-798.

7.1 Determine Active Certifications

Determined by finding all certification records (Client Id, Certification Start Date and Actual Delivery Date) where the certification start date is less than the last day of the month being processed and the least of the certification end date or certification termination date is greater than the first day of the month being processed. Any client in this set of records is determined to be enrolled in the system for the month being processed.

Delete records for month being processed

```
DELETE FROM C_CERTIFICATIONS_ACTIVE
WHERE CERT_ACTIVE_DATE = V_IN_DATE;

COMMIT;
```

Insert Active Certification into the temp table

```
INSERT INTO C_CERTIFICATIONS_ACTIVE F (CERT_ACTIVE_DATE,
F.cc_CLIENT_ID,
F.CERT_START_DATE,
f.ACTUAL_DELIVERY_DATE)

SELECT CERT_ACTIVE_DATE,
```

```

CC_Client_id,
cert_start_date,
(SELECT   CERT1.ACTUAL_DELIVERY_DATE
 FROM     AIM.c_certifications cert1
 WHERE    cert1.cc_client_id = cert.cc_client_id
 AND CERT1.CERT_START_DATE = CERT.CERT_START_DATE)
ACTUAL_DELIVERY_DATE
FROM      ( SELECT   V_IN_DATE CERT_ACTIVE_DATE,
                   CERT.CC_Client_id,
                   MAX (CERT.CERT_start_Date) cert_start_date
           FROM     AIM.c_certifications cert
           WHERE    CERT.CERT_START_DATE BETWEEN ADD_MONTHS (
                                                       LAST_DAY (
                                                       V_IN_DATE
                                                       ),
                                                       -18
                                                       )
           AND     LAST_DAY (
                   V_IN_DATE
                   )
           AND (LEAST (
                   cert_end_Date,
                   NVL (cert.termination_date,
                       cert_end_date)
                   ) > LAST_DAY (ADD_MONTHS (V_IN_DATE, -1)))
           GROUP BY CERT.CC_Client_id) cert;

```

7.2 Determine Clients with Active Certifications that Received Food Benefits

Determined by finding all food benefit issuance records where the issuance month is the same as the month being processed. The issuance date of the food benefit must be between the first date of the month and the last day of the month being processed. Any client in this set of records is determined to be participating in the program. Additional processing is done for special case Category types which will be discussed in the Rebuild Caseload Details section. The Food Benefits received includes the clinic id where the food benefit was issued. This clinic is used when assigning caseload to a clinic/agency.

Delete records for month being processed

```

DELETE FROM C_CLIENT_PARTICIPATION_BASE
 WHERE     ISSUE_MONTH = V_IN_DATE;

COMMIT;

```

Insert records into table for clients who received food benefits for the month being processed

```

INSERT INTO C_CLIENT_PARTICIPATION_BASE (ISSUE_MONTH,
                                         CC_CLIENT_ID,
                                         FI_COUNT,
                                         FIS_RECEIVED,
                                         participation_date,
                                         ou_seq_id)

SELECT   issue_month,
         cc_client_id,
         1 fi_count,
         'Y' FIS_RECEIVED,
         participation_date,

```

```

        (SELECT      MAX (ou_Seq_id)
          FROM      i_food_instruments ifil
          WHERE     ifil.cc_client_id = gg.cc_client_id
                  AND ifil.issue date = gg.participation_date
                  AND ifil.date_created = dateC)
        ou_Seq_id
FROM      ( SELECT  TRUNC (ISSUE_MONTH, 'MM') ISSUE_MONTH,
                  CC_CLIENT_ID,
                  MAX (TRUNC (issue_date)) participation_date,
                  MAX (date_created) dateC
          FROM      AIM.I_FOOD_INSTRUMENTS IFI
          WHERE     TRUNC (ISSUE_MONTH, 'MM') = V_IN_DATE
                  AND ( (IFI.IDIS_DISPOSITION_CODE IN
                        ('2', '4', '5', '6'))
                      OR (IFI.IDIS_DISPOSITION_CODE IN ('3')
                          AND IFI.IVR_VOID_REASON_CODE IN
                              ('B', 'E', 'L', 'T', 'Z'))
                      OR (IFI.IDIS_DISPOSITION_CODE IN ('3')
                          AND TRUNC (void_date) >=
                              TRUNC (first_date_to_use)
                          AND ivr_void_reason_code IN
                              ('D', 'H', 'J', 'T')))
          GROUP BY TRUNC (ISSUE_MONTH, 'MM'), CC_CLIENT_ID) gg;

```

Insert records into table for clients who received food box distributions for the month being processed

```

INSERT INTO eodadm.C_CLIENT_PARTICIPATION_BASE (ISSUE_MONTH,
                                                CC_CLIENT_ID,
                                                FI_COUNT,
                                                FIS_RECEIVED,
                                                participation_date,
                                                ou_Seq_id)

SELECT  TRUNC (FF.PICK_UP_DATE, 'mm'),
        cc1_cc_client_id,
        1 fi_count,
        'Y' fis_received,
        TRUNC (FF.PICK_UP_DATE, 'mm'),
        ou_Seq_id
FROM    C_FOOD_BOX_DISTs ff
WHERE   TRUNC (FF.PICK_UP_DATE, 'mm') = V_IN_DATE
GROUP BY cc1_cc_Client_id,
         TRUNC (FF.PICK_UP_DATE, 'mm'),
         ou_Seq_id;

```

7.3 Rebuild Caseload Details

Caseload Type Details contains the enrollment and participation status of all clients with an active certification and/or food benefits pending for future months. The Caseload Summary table is rebuilt every night during End of Day processing for the three month period starting one month prior to the month being processed.

Delete records for the month being processed

```

DELETE FROM f_caseload_type_Details_new FCD
WHERE      FCD.FBU_FFY_MONTH = V_FFY_MONTH
          AND FCD.FBU_FFF_FFY = V_FFY_YEAR;

```

```

COMMIT;

```

BEGIN

Insert records for caseload type details into temporary processing table

```

INSERT INTO f_caseload_type_Details_new fcd (FCD.ADJ_ELIGIBLE_FLAG,
                                             FCD.CAT_CATEGORY_CODE,
                                             FCD.CC_CLIENT_ID,

FCD.CEG_ETHNIC_GROUP_CODE,

                                             FCD.CERTIFIED_FLAG,
                                             FCD.CP2_ID,
                                             FCD.CREATED_BY,
                                             FCD.DATE_CREATED,
                                             FCD.DATE_MODIFIED,
                                             FCD.FAMILY_SIZE,
                                             FCD.FBU_FFY_FFY,
                                             FCD.FBU_FFY_MONTH,
                                             FCD.FPL_ID,
                                             FCD.INCOME,
                                             FCD.LANG_LANGUAGE_CODE,
                                             FCD.MIGRANT_FLAG,
                                             FCD.MODIFIED_BY,
                                             FCD.OU_SEQ_ID,
                                             FCD.PARTICIPANT_FLAG,
                                             FCD.PLUS_FLAG,
                                             FCD.PROGRAM,
                                             FCD.REFUGEE_FLAG,
                                             FCD.STATE_FUND_FLAG,
                                             FCD.CERT_START_DATE,
                                             ACTUAL_DELIVERY_DATE,
                                             ISSUE_MONTH)

SELECT CFEU.ADJ_ELIGIBLE_FLAG,
       CERT.CAT_CATEGORY_CODE,
       ff.cc_client_id,
       NVL (CC.CEG_ETHNIC_GROUP_CODE, 105)
         CEG_ETHNIC_GROUP_CODE,
       -- set to 38- other is null
       'Y' CERTIFIED_FLAG,
       CERT.CP2_ID,
       'EODADM' Created_by,
       TRUNC (NVL (pbase.participation_date, V_IN_DATE))
         Date_created,
       TRUNC (NVL (pbase.participation_date, V_IN_DATE))
         date_modified,
       CFEU.FAMILY_SIZE,
       fdate.ffy_year,
       fdate.ffy_month,
       HANDS_EOD_POST.get_fpl_id (cfeu.FAMILY_SIZE,
                                 CC.PROGRAM,
                                 CFEU.INCOME,
                                 LAST_DAY (cal_date))
         fpl_id,
       CFEU.INCOME,
       NVL (CFEU.LANG_LANGUAGE_CODE, 1),
       CFEU.MIGRANT_FLAG,
       'EODADM' Modified_by,
       NVL (pbase.OU_SEQ_ID, NVL (cfeu.ou_seq_id, 1)) OU_SEQ_ID,
       -- Set to state agency if null
       DECODE (pbase.fis_received, 'Y', 1, 0) PARTICIPANT_FLAG,
       CERT.PLUS_FLAG,
       NVL (cert.PROGRAM, 'W'),

```

```

        CFEU.REFUGEE_FLAG,
        'N' STATE_FUND_FLAG,
        FF.CERT_START_DATE,
        ff.ACTUAL_DELIVERY_DATE,
        FDATE.CAL_DATE
FROM    c_certifications_active ff,
        AIM.c_certifications cert,
        AIM.c_family_economic_units cfu,
        AIM.c_clients cc,
        AIM.f_ffy_dates fdate,
        c_client_participation_base pbase
WHERE   ff.cc_client_id = cert.cc_client_id
AND     ff.cert_start_date = CERT.CERT_START_DATE
AND     CFEU.FAMILY_ID(+) = CC.CFEU_FAMILY_ID
AND     cc.client_id = ff.cc_client_id
AND     FF.CERT_ACTIVE_DATE = fdate.cal_date
AND     fdate.cal_date = V_IN_DATE
AND     pbase.issue_month(+) = ff.CERT_ACTIVE_DATE
AND     pbase.cc_client_id(+) = ff.cc_client_id;

COMMIT;
EXCEPTION
WHEN OTHERS
THEN
    P_MESSAGE := SUBSTR (SQLERRM, 1, 225);
    ROLLBACK;

    INSERT INTO HANDS_EOD_LOG (PROCEDURE_NAME,
                              GOOD_PROC_DATE,
                              ERROR_MSG,
                              CREATED_BY)
    VALUES ('REBUILD_CASELAOD_DETAILS1',
            p_in_date,
            p_message,
            'HANDSEOD');

COMMIT;
END;

BEGIN

```

Insert records for Participating not Certified Clients

```

FOR REC3
IN ( SELECT  'N' ADJ_ELIG,
            IFI.CAT_CATEGORY_CODE,
            IFI.CC_CLIENT_ID,
            NVL (CC.CEG_ETHNIC_GROUP_CODE, 105) ETHNIC_GRP,
            'N' CERT_FLAG,
            IFI.CP2_ID,
            'EODADM' CREATED_BY,
            TRUNC (NVL (MAX (ifi.issue_date), V_IN_DATE))
              Date_created,
            TRUNC (NVL (MAX (ifi.issue_month), V_IN_DATE))
              date_modified,
            CFEU.FAMILY_SIZE,
            FD.FFY_YEAR,
            FD.FFY_MONTH,
            157 FPL_ID,
            CFEU.INCOME,
            NVL (CFEU.LANG_LANGUAGE_CODE, 1) LANG,
            CFEU.MIGRANT_FLAG,

```

```

        'EODADM' MODIFIED_BY,
        IFI.OU_SEQ_ID,
        '1' PART_FLAG,
        'N' PLUS_FLAG,
        CC.PROGRAM,
        CFEU.REFUGEE_FLAG,
        'N' STATE_FUND_FLAG,
        FD.CAL_DATE ISSUE_MONTH
FROM    AIM.i_food_instruments ifi,
        AIM.C_CLIENTS CC,
        AIM.F_FFY_DATES FD,
        AIM.C_FAMILY_ECONOMIC_UNITS CFEU
WHERE   ifi.cc_client_id IN
        (SELECT cc_client_id
         FROM   eodadm.c_client_participation_base pb
         WHERE  PB.ISSUE_MONTH = V_IN_DATE
         MINUS
         SELECT cc_client_id
         FROM   eodadm.c_certifications_active am
         WHERE  AM.CERT_ACTIVE_DATE = V_IN_DATE)
AND TRUNC (issue_month, 'MM') = FD.CAL_DATE
AND CC.CLIENT_ID = IFI.CC_CLIENT_ID
AND CFEU.FAMILY_ID = CC.CFEU_FAMILY_ID
AND FD.CAL_DATE = V_IN_DATE
GROUP BY IFI.CAT_CATEGORY_CODE,
         IFI.CC_CLIENT_ID,
         NVL (CC.CEG_ETHNIC_GROUP_CODE, 105),
         IFI.CP2_ID,
         CFEU.FAMILY_SIZE,
         FD.FFY_YEAR,
         FD.FFY_MONTH,
         CFEU.INCOME,
         NVL (CFEU.LANG_LANGUAGE_CODE, 1),
         CFEU.MIGRANT_FLAG,
         IFI.OU_SEQ_ID,
         CC.PROGRAM,
         CFEU.REFUGEE_FLAG,
         FD.CAL_DATE)
LOOP
BEGIN
    INSERT INTO f_caseload_type_Details_new fcd (
FCD.ADJ_ELIGIBLE_FLAG,
FCD.CAT_CATEGORY_CODE,
FCD.CC_CLIENT_ID,
FCD.CEG_ETHNIC_GROUP_CODE,
FCD.CERTIFIED_FLAG,
FCD.CP2_ID,
FCD.CREATED_BY,
FCD.DATE_CREATED,
FCD.DATE_MODIFIED,
FCD.FAMILY_SIZE,
FCD.FBU_FFF_FFY,

```

```

FCD.FBU_FFY_MONTH,
FCD.LANG_LANGUAGE_CODE,
FCD.MIGRANT_FLAG,
FCD.MODIFIED_BY,
FCD.PARTICIPANT_FLAG,
FCD.REFUGEE_FLAG,
FCD.STATE_FUND_FLAG,
FCD.FPL_ID,
FCD.INCOME,
FCD.OU_SEQ_ID,
FCD.PLUS_FLAG,
FCD.PROGRAM,
ISSUE_MONTH

```

```

VALUES )
(
  REC3.ADJ_ELIG,
  REC3.CAT_CATEGORY_CODE,
  REC3.CC_CLIENT_ID,
  REC3.ETHNIC_GRP,
  REC3.CERT_FLAG,
  REC3.CP2_ID,
  REC3.CREATED_BY,
  REC3.Date_created,
  REC3.date_modified,
  REC3.FAMILY_SIZE,
  REC3.FFY_YEAR,
  REC3.FFY_MONTH,
  REC3.FPL_ID,
  REC3.INCOME,
  REC3.LANG,
  REC3.MIGRANT_FLAG,
  REC3.MODIFIED_BY,
  REC3.OU_SEQ_ID,
  REC3.PART_FLAG,
  REC3.PLUS_FLAG,
  REC3.PROGRAM,
  REC3.REFUGEE_FLAG,
  REC3.STATE_FUND_FLAG,
  REC3.ISSUE_MONTH);

```

```

      COMMIT;
    EXCEPTION
      WHEN OTHERS
      THEN
        ROLLBACK;
        NULL;
      END;
    END LOOP;

```

--- INSERT CLIENTS WITH FOOD INSTRUMENTS DURING THE MONTH BUT NO ACTIVE CERTIFICATION

```

      COMMIT;
    EXCEPTION

```

```

WHEN OTHERS
THEN
  P_MESSAGE := SUBSTR (SQLERRM, 1, 225);
  ROLLBACK;

  INSERT INTO HANDS_EOD_LOG (PROCEDURE_NAME,
                           GOOD_PROC_DATE,
                           ERROR_MSG,
                           CREATED_BY)
  VALUES ('REBUILD_CASELAOD_DETAILS2',
          p_in_date,
          p_message,
          'HANDSEOD');

  COMMIT;
END;

-- set Part_Flag = 0 where part_flag is null
BEGIN
  UPDATE f_caseload_type_details_new fcd
  SET participant_flag = '0'
  WHERE participant_flag IS NULL
  AND FCD.FBU_FFF_FFY = v_ffy_year
  AND FCD.FBU_FFY_MONTH = v_ffy_month;

  COMMIT;

```

-- update Participant flag of PN+ clients

Update Caselaod Details record for clients who are PN+ and the infant associated to the mother for the month received food benefits

```

UPDATE f_caseload_type_details_new fcd
SET participant_flag =
  NVL (
    (SELECT DECODE (IFI.FIS_RECEIVED, 'Y', 1, 0)
     FROM AIM.c_clients cc,
          AIM.c_certifications cert,
          c_client_participation_base ifi
     WHERE cc.cc_client_id = fcd.cc_client_id
           AND cert.cc_client_id = cc.client_id
           AND cert.cat_category_code = 'IPN+'
           AND ifi.cc_client_id = cert.cc_client_id
           AND IFI.ISSUE_MONTH = v_in_date
           AND ROWNUM = 1),
    0
  ),
date_created =
  NVL (
    (SELECT ifi.participation_date
     FROM AIM.c_clients cc,
          AIM.c_certifications cert,
          c_client_participation_base ifi
     WHERE cc.cc_client_id = fcd.cc_client_id
           AND cert.cc_client_id = cc.client_id
           AND cert.cat_category_code = 'IPN+'
           AND ifi.cc_client_id = cert.cc_client_id
           AND IFI.ISSUE_MONTH = v_in_date
           AND ROWNUM = 1),
    V_IN_DATE
  )

```

```

WHERE FCD.CAT_CATEGORY_CODE = 'PN+'
AND (FCD.PARTICIPANT_FLAG = '0'
     OR FCD.PARTICIPANT_FLAG IS NULL)
AND MONTHS BETWEEN (v in date, actual_delivery_date) >= 6
AND FCD.FBU_FFF_FFY = v_ffy_year
AND FCD.FBU_FFY_MONTH = v_ffy_month;

COMMIT;
EXCEPTION
WHEN OTHERS
THEN
  P_MESSAGE := SUBSTR (SQLERRM, 1, 225);
  ROLLBACK;

  INSERT INTO HANDS_EOD_LOG (PROCEDURE_NAME,
                           GOOD_PROC_DATE,
                           ERROR_MSG,
                           CREATED_BY)
  VALUES ('REBUILD_CASELAOD_DETAILS4',
          p_in_date,
          p_message,
          'HANDSEOD');

COMMIT;
END;

-- update Participant flag of IEN and IPN clients
v_part_flag := 0;

```

```
BEGIN
```

Update Caselaod Details record for clients who are IEN,IPN,IPN+ and the mother associated to the Infant for the month received food benefits

```

UPDATE f_caseload_type_details_new fcdn
SET fcdn.PARTICIPANT_FLAG =
  NVL (
    (SELECT DECODE (IFI.FIS_RECEIVED, 'Y', 1, 0)
     FROM AIM.c_clients cc,
           c_client_participation_base ifi
     WHERE cc.client_id = fcdn.cc_client_id
           AND ifi.cc_client_id = cc.cc_client_id
           AND ifi.issue_month = v_in_date),
    0
  ),
date_created =
  NVL (
    (SELECT ifi.participation_date
     FROM AIM.c_clients cc,
           c_client_participation_base ifi
     WHERE cc.client_id = fcdn.cc_client_id
           AND ifi.cc_client_id = cc.cc_client_id
           AND ifi.issue_month = v_in_date),
    V_IN_DATE
  )
WHERE FCDN.CAT_CATEGORY_CODE IN ('IPN', 'IEN', 'IPN+')
AND (FCDN.PARTICIPANT_FLAG = '0'
     OR FCDN.PARTICIPANT_FLAG IS NULL)
AND fcdn.fbu_ffy_month = V_FFY_MONTH
AND fcdn.fbu_fff_ffy = V_FFY_year;

COMMIT;

```

Update Caseload Details record for clients who are IEN and were assigned to no food food package (not applied after 4/1/2015)

```

IF v_in_date < '01-apr-2015'
-- Per Karen Sell this logic is not longer needed after march 1st,
2015
THEN
  UPDATE f_caseload_type_details_new fcd
  SET participant_flag =
    (SELECT DECODE (COUNT (1),
      1, '1',
      2, '1',
      3, '1',
      4, '1',
      5, '1',
      6, '1',
      7, '1',
      8, '1',
      9, '1',
      10, '1',
      0)
    FROM AIM.c_food_package_prescriptions cfpp,
    f_caseload_type_details_new fcdn
  WHERE cfpp.cc_client_id = fcdn.cc_client_id
    AND fcdn.cc_client_id = fcd.cc_client_id
    AND FCDN.PARTICIPANT_FLAG = 0
    AND FCDN.FBU_FFF_FFY = v_ffy_year
    AND FCDN.FBU_FFY_MONTH = v_ffy_month
    AND v_in_date BETWEEN TRUNC (
      effective_date,
      'mm'
    )
    AND LAST DAY (end date)
    AND CFPP.IFP_FOOD_PACKAGE_ID = 'AA900888')
  WHERE fcd.CAT_CATEGORY_CODE IN ('IEN')
    AND (FCD.PARTICIPANT_FLAG = '0'
    OR FCD.PARTICIPANT_FLAG IS NULL)
    AND FCD.FBU_FFF_FFY = v_ffy_year
    AND FCD.FBU_FFY_MONTH = v_ffy_month;
END IF;

COMMIT;
EXCEPTION
WHEN OTHERS
THEN
  P_MESSAGE := SUBSTR (SQLERRM, 1, 225);
  ROLLBACK;

  INSERT INTO HANDS_EOD_LOG (PROCEDURE_NAME,
    GOOD_PROC_DATE,
    ERROR_MSG,
    CREATED_BY)
  VALUES ('REBUILD_CASELAOD_DETAILS5',
    p_in_date,
    p_message,
    'HANDSEOD');

  COMMIT;
END;
```

BEGIN

Update Caseload Details record for clients who PN+ and are assigned the food package AZ200888 (for AZ) in a valid food package prescription

```

UPDATE f_caseload_type_details_new fcd
SET participant_flag =
    (SELECT DECODE (COUNT (1),
        1, '1',
        2, '1',
        3, '1',
        4, '1',
        5, '1',
        0)
        c1
    FROM c_food_package_prescriptions cfpp
    WHERE cfpp.cc_client_id = fcd.cc_client_id
    AND cfpp.effective_date <=
        LAST_DAY (v_in_date)
    AND cfpp.ccl_cert_start_date =
        fcd.cert_start_date
    AND cfpp.end_date >= v_in_date
    AND cfpp.ifp_food_package_id = 'AZ200888'
    AND cfpp.disable_flag = 'N'
    AND TRUNC (cfpp.date_created) <=
        LAST_DAY (v_in_date))
WHERE (FCD.PARTICIPANT_FLAG = '0' OR FCD.PARTICIPANT_FLAG IS NULL)
AND FCD.CAT_CATEGORY_CODE = 'PN+'
AND MONTHS_BETWEEN (v_in_date, fcd.actual_delivery_date) >=
    6
AND FCD.FBU_FFF_FFY = v_ffy_year
AND FCD.FBU_FFY_MONTH = v_ffy_month;

COMMIT;
EXCEPTION
WHEN OTHERS
THEN
    P_MESSAGE := SUBSTR (SQLERRM, 1, 225);
    ROLLBACK;

    INSERT INTO HANDS_EOD_LOG (PROCEDURE_NAME,
        GOOD_PROC_DATE,
        ERROR_MSG,
        CREATED_BY)
    VALUES ('REBUILD_CASELAOD_DETAILS6',
        p_in_date,
        p_message,
        'HANDSEOD');

COMMIT;
END;
```

Calculate CSFP caseload details records

```

FOR rec4
IN (SELECT cc_client_id,
    ou_seq_id,
    FCD.FBU_FFF_FFY,
    FCD.FBU_FFY_MONTH
    FROM f_caseload_type_details_new fcd
    WHERE program = 'C'
    AND fcd.fbu_ffy_month = V_FFY_MONTH
```

```

        AND fcd.fbu_fff_ffy = V_FFY_year)
LOOP
  UPDATE f_caseload_type_details_new fcd
  SET participant_flag =
      NVL (
        (SELECT COUNT (1)
         FROM c_food_box_dists fb
         WHERE FB.CC1_CC_CLIENT_ID = rec4.cc_client_id
               AND TRUNC (FB.PICK_UP_DATE, 'mm') =
                   V_IN_DATE
               AND FB.OU_SEQ_ID = rec4.ou_seq_id),
          0
      )
  WHERE cc_client_id = rec4.cc_client_id
        AND program = 'C'
        AND fcd.fbu_ffy_month = rec4.FBU_FFY_MONTH
        AND fcd.fbu_fff_ffy = rec4.FBU_FFF_FFY;

  COMMIT;
END LOOP;

INSERT INTO HANDS_EOD_LOG (PROCEDURE_NAME,
                          GOOD_PROC_DATE,
                          ERROR_MSG,
                          CREATED_BY)
VALUES ('FINISH-REBUILD_CASELOAD_DETAILS',
        p_in_date,
        NULL,
        'HANDSEOD');

COMMIT;

```

7.3.1 Certification Flag

Any client with an active certification is enrolled (certified) in the system for the month being processed. To have an active certification the client must have a certification end date greater than the first day of the month being processed and a certification start date less than or equal to the last day of the month being processed. Certification end date is the lesser of the certification end date and the termination date of the certification.

7.3.2 Participation Flag

An enrolled client is a participant if they have been issued food benefits for the month being processed. Food benefits voided for the following reasons are also included when determining if an enrollee is a participant (Lost or Stolen, Returned by Client/Unused, Signed Before Going to Store, Stale Dated by EOD, Replacement, Termination, Late Pickup/reduced Food Package).

Special Participation Processing Rules:

- **Enrollee of Category PN+** – The client’s child has been issued food benefits and the difference between the process month and the actual delivery date of the mother is greater than or equal to 6 months.
- **Enrollee of Category Type IPN, IEN, IPN+** – The client’s mother has been issued a food benefit for the month being processed.
- **Enrollee of Category Type IEN** – An Enrollee who was issued the food benefit package “AA900888” for the month being processed.

Note – This processing rule is no longer valid after March 31, 2015.

- **Enrollee of Category Type PN+** – Client that has been prescribed the default food package for the PN+ category (“AZ200888”) for the month being processed. The food package prescription must be valid for the month being processed.
- Participating but NOT Certified Rule – A client who does not have an active certification but has issued food benefits for the month being processed. These clients will be listed as participants, but not enrollees.

Refugee Flag:

An enrollee who’s family record indicates that they are a refugee.

Migrant Flag:

An enrollee who’s family record indicates that they are a migrant.

7.4 Rebuild Caseloads

The caseload summary table rolls up the caseload type details table by the various flags on the detail record. There is record in the Caseload table for the following column groupings (Clinic Id, Category Code, Priority, Ethnic Group, Language, Program and Funding Flag) for the month being processed. A flag set to “Y” translates to a count of one. Those counts are then summed up and grouped as listed above.

The Caseload Summary table is rebuilt every night during the End of Day processing for the three month period starting one month prior to the month being processed.

Caseload Type Code: (Caseload types are not exclusive)

- 1 = Participant
- 2 = Enrollee
- 3 = Migrant
- 4 = Refugee
- 5 = Migrant Participant
- 6 = Refugee Participant
- 7 = Farmers Market Enrollee
- 8 = Farmers Market Participant

8 FINANCIAL SUMMARY PROCESSING

The financial summary process is designed to roll up the food benefit data for federal, state and local agency reporting. The summary tables are based on the federal fiscal year, meaning they run from October through September.

8.1 Building Food Benefit Obligation Summary

The obligation summary is based on the obligated amount for each food benefit that has not been redeemed or voided. Once a check is redeemed or voided it is not used in the summary of outstanding food benefit obligations. The summary table F_OBLIGATIONS groups the food benefit obligations by the following fields:

- Benefit Type – FI_TYPE
- Clinic – OU_SEQ_ID
- Obligation Month –Month of Issuance
- Obligation Year – Fiscal Year of Obligation
- Category – Client Category at time of Issuance (CATEGORY_CODE)
- Priority – Client Priority at time of Issuance (CP2_ID)

8.2 Building Food Benefit Outlays

The food benefit outlays summary process is designed to calculate the total food benefits redeemed by the clients in a given month. The F_OUTLAYS table contains the obligation month of the food benefit and the activity month (redemption month) of the food benefit. This is to facilitate reporting on the FNS-798 report which asks for redemptions by month of issuance and month of cashing. The F_OUTLAYS table groups the food benefit outlays by the following fields:

- Benefit Type- FI_TYPE
- Clinic – OU_SEQ_ID
- Obligation Month – Month of Issuance
- Obligation Year – Year of Issuance
- Activity Month – Month of Redemption
- Activity Year – Year of Redemption
- Category – Client Category at time of Issuance (CATEGORY_CODE)
- Priority – Client Priority at time of Issuance (CP2_ID)

8.3 Building Divisibility Records

The divisibility process determines the actual number of formula cans purchased by the participant on a check by check basis. The process looks at the purchase price of the food benefit instrument (check) and divides by the formula can quantity on the check. This is done

by multiplying the purchase price by 100 and dividing by the can quantity, if there is no remainder (divides evenly) than the participant purchased the number of cans listed on the check. If there is a remainder then a loop is created to perform the same calculation reducing the can quantity by one each time the calculation is repeated, until the result of the division has no remainder.

Once the quantity purchased is determined based on the logic above a record for that food benefit record (check) is created in the F_DIVISIBILITY table.

- **IFI_SERIAL_NUMBER** – Unique identifier for the food benefit issuance
- **QUANTITY**–Number of cans on the food benefit issuance
- **REBATE_PER- UNIT**- Dollars per can from the active rebate contract
- **VENDOR_ID**–Unique identifier of the vendor where the food benefit was redeemed
- **REDEMPTION_AMT**–Amount for which the food benefit was redeemed
- **CLEARED_DATE**–Date when the financial institution paid the vendor
- **IFIT_FI_TYPE_CODE**–Code to identify the food benefit issuance
- **FORMULA_TYPE** – The type of formula to be rebated
- **PURCHASED** – The calculated number of cans actually purchased by the participant
- **FIRST_DATE_TO_USE** – The day on which the participant can use the check
- **MAN_MFR_ID** – Unique identifier for the formula manufacturer
- **ISSUE_MONTH** – The month for which the food benefit was issued
- **VPG_PEER_GROUP_ID** – Unique identifier of the peer group that the vendor belongs in.

8.4 Building Food Benefit Rebate Details

During the EOD process the rebate detail information is re-built for food benefits redeemed during the current month. The rebate amount is based on the quantity of food items on the food benefit issuance and the negotiated rebate contract price with the manufacturer.

Note – Each food benefit has an estimated rebate value calculated at the time of issuance. This is calculated by taking the quantity of food items with rebate contracts² on the food benefit and multiplying that by the currently effective rebate amount for that food item.

8.4.1 Rebate Detail Calculation:

The first step is to determine if a food item with a rebate contract is included on the food benefit issuance redeemed during the current month. The cleared date of the food benefit must be between the start and end date of the rebate contract for the food item included on the redemption record. The rebate contract amount is found and multiplied by the quantity of food items on the food benefit redemption record. This is the rebate amount of the food benefit.

² Rebate contracts are entered into between the state and the food item manufacturer. A food currently under rebate contract is defined as where the current date is between the start and end date of a rebate contract.

8.4.2 Determine Divisibility

A WIC program has the option of using divisibility or not on a system wide level. Divisibility determines the actual number of items purchased of a given food item with a rebate contract. This is done by multiplying the redemption price by 100 and then dividing it by quantity of the food item on the food benefit. If the result of this operation is a whole number with no remainder than the participant purchased the “full” quantity on the food benefit. If there is a remainder then the same operation is preformed again using quantity-1 as the divisor. If the result is a whole number than the participant purchased one less than the full quantity. This loop continues until a whole number results from the operation or a quantity of one (1) is reached by the subtraction of quantity minus the loop number. Only food benefit redemption records where less than the full quantity are purchased are stored in the divisibility table in the system.

8.4.3 Update I_FOODS for rebate information:

The EOD process will update I_FOODS.CONTRACT_FLAG and I_FOOD_MAN_MFR_ID for foods that are currently under a rebate contract, all other food will have the CONTRACT_FLAG set to 'N' and MAN_MFR_ID to null.

8.5 Building Food Benefit Peer Group Averages

The Peer Group Average for each Food Instrument (FI) Type is calculated on a weekly basis, (currently run on Thursday Evenings, the rundate) for limits that go into effect for the following week from Monday through Sunday. There are 2 steps to the calculation:

- 1) A new peer group average is calculated for FI Types redeemed 5 or more times in the 30 days prior to the calculation date, including all redemptions with cleared dates falling into the 30 day period ending on the Wednesday night prior to the run date. This is based on good proc dates which may result in a time period of 29 days instead of 30.
- 2) The default Maximum FI Amount of \$200 is set as the new peer group average for FI Types that were redeemed less than 5 times in 90 days prior to the calculation date.

Note – The Peer Group Average will remain the same for FI Types that were not redeemed 5 or more times in the 30 days prior to the calculation date and that were redeemed more than 5 times in the 90 days prior to the calculation date.

Note – Historical records of past Peer Group Averages are not maintained by HANDS, but can be found in bank files.

Example 1, New Peer Group Average is calculated:

Date of Calculation: 6/4/2015

Test 1 30 day calculation period: 5/4/2015 – 6/3/2015

Test 2 90 day calculation period: 3/5/2015 – 6/3/2015

Redemption Details:

- a) FI 1234 redeemed at Peer Group 1 location for \$20.00 on 5/5/2015
- b) FI 1234 redeemed at Peer Group 1 location for \$22.50 on 5/15/2015
- c) FI 1234 redeemed at Peer Group 1 location for \$21.55 on 5/16/2015

- d) FI 1234 redeemed at Peer Group 1 location for \$18.98 on 5/20/2015
 - e) FI 1234 redeemed at Peer Group 1 location for \$19.88 on 6/3/2015
- Result Test 1: New Peer Group Average for Peer Group 1, FI 1234: \$20.58
($\$20.00 + \$22.50 + \$21.55 + \$18.98 + \$19.88$) / 5 = \$20.58
- Result Test 2: Step 2 not applicable.

Example 2, Peer Group Average set to Default Maximum (\$200.00):

Date of Calculation: 6/4/2015
90 day calculation period: 3/5/2015 – 6/3/2015
Redemption Details:
FI 4321 redeemed fewer than 5 times at Peer Group Location 1 during the 90 day calculation period.
Result Test 1: 5 or more FIs were not redeemed, nothing to do.
Result Test 2: At least 5 FIs were not redeemed: New Peer Group Average for Peer Group 1, FI 4321: \$200.00 (Default Maximum)

Example 3, Peer Group Average is NOT changed:

Date of Calculation: 7/16/2015
30 day calculation period: 6/15/2015 – 7/15/2015
Redemption Details:
FI 1234 redeemed fewer than 5 times at Peer Group Location 1 during the step 1 calculation period (30 days prior to calculation date). As noted in example 1, there were at least 5 redemptions discovered in the step 2 calculation period (90 days prior to calculation date) for FI 1234.
Result Test 1: 5 or more FIs were not redeemed, nothing to do.
Result Test 2: At least 5 FIs were redeemed: Peer Group Average remains: \$20.58

Peer Group Table Fields:

- **Food Benefit Type** – Unique Identifier of Food Items on a Food Benefit
- **Peer Group Id** – Unique identifier of a group of vendors
- **Total Redemption Amount** – Sum of redemptions for Peer Group and Food Benefit Type
- **Total Number of Food Benefits Redeemed** – The count of Food Benefits redeemed for the Peer Group and the Food Benefit Type
- **Average Participant Count** – Three month average of participants redeeming this food benefit type
- **Average Redemption Amount** – Total Redemption Amount divided by the Total Number of Food Benefits Redeemed
- **Maximum Redemption Amount** – The largest redemption amount for the Peer Group and Food Benefit Type
- **Standard Deviation** – Calculated using the Oracle Standard Deviation function for the Peer Group and Food Benefit Type
- **Number of Standard Deviations** – Currently defaulted to 3 – changed to 2.5 on April 1, 2015.

Note: Peer Group 8 is defined by the Vendor Management Team as the State wide average across all vendors with the EXCEPTION of peer group 8 that cleared during the same time period.

9 STATE CLIENTS PROCESSING

The State Clients process maintains a list of all clients in the HANDS system. The most recent information for the following fields is maintained in the table.

- **Client Id** – Unique Client identifier
- **Clinic Id** – Unique Clinic Id that the client is associated with
- **Client Last Name**
- **Client First Name**
- **Client Middle Initial 1**
- **Client Middle Initial 2**
- **Client Date of Birth**
- **Clients Gender**
- **Clients Mailing Address**
- **Family Id** – Unique Identifier for an economic family unit
- **Authorized Reps Last Name**
- **Authorized Reps First Name**
- **Authorized Reps Middle Initial 1**
- **Authorized Reps Middle Initial 2**
- **Client Category Code**
- **Clients Mailing Address**
- **Clients Mailing Zip Code**
- **Clients Certification Start Date** – The First date that the most recent (current) certification is active
- **Clients Certification End Date** – The lesser of the Certification end date or the termination date for the most recent (current) certification.
- **Last Food Benefit Issuance Date**
- **Last Food Benefit Cashed Date**
- **Clients Program**
 - **W** – WIC
 - **C** – CSFP
- **Clients Birth Month** – Month in which the client was born
- **Geo Location ID** – Unique ID for identifying the client's address location.

10 UPDATE BUDGET FACTORS

The Final part of the EOD process is to update the budget factors table with revised actual dollar amounts and counts from the most recent redemption information.

The following fields are calculated and revised by the Update Budget process:

- Monthly Redemption Percentages
- Actual Redemption Participation
- Actual Food Package Inflation Factor
- Actual Pre-Rebate food Package Cost
- Actual Post-Rebate Food Package Cost
- Actual Rebate per Clients
- Actual Redemption Factor

10.1 Monthly Redemption Percentages:

The redemption percentage (expressed as a decimal) is calculated for each issue month based on the redemption month. The system calculates a percentage for food benefits issued and cashed in the same month (month 0), issued and cashed in the following month (month 1) and finally issued and cashed two or more months in the future (month 2).

10.2 Actual redemption Participation:

The system also calculates the actual redemption participation for each month. The redemption participation is based on issued checks that were redeemed for a particular issue month.

10.3 Actual Food Package Inflation Factor:

The food package inflation factor is calculated by taking the average food package cost less the prior months average food package cost divided by the average food package cost.

10.4 Actual Pre-Rebate Food Package Cost:

The actual pre-rebate food package cost is determined by taking the total dollars spent on food benefit type for a month and dividing it by the distinct count of clients who cashed those food benefits.

10.5 Actual Post-Rebate Food Package Cost:

The actual pre-rebate food package cost is determined by taking the total dollars spent on food benefit type for a month less the total rebate dollars for those food benefits and dividing it by the distinct count of clients who cashed those food benefits.

10.6 Actual Rebate per Client:

The actual rebate per client is calculated by taking the total rebate dollars for a month and dividing by the distinct count of clients redeeming food benefits that month.

10.7 Actual Redemption Factor:

The actual redemption factor is calculated by taking all issued food benefits redeemed for a month and dividing by the count of food benefits redeemed in that same month.

11 REPLENISH FOOD INSTRUMENT INVENTORIES

The replenish food instrument (FI) inventories process looks at the I_FI_INVENTORIES table to determine if any clinic has less than 10% of the current allocation of serial numbers still remaining. If the clinic falls below 10% then the process creates a new range of check number for that clinic based on the clinic's demographic information (see below).

- Large Clinic – 24,000 Serial Numbers
- Medium Clinic – 12,000 Serial Numbers
- Small Clinic – 4,000 Serial Numbers

12 LOAD LOCAL AGENCY & CLINIC FOOD BENEFIT AVERAGES

The average food benefit redemption is calculated for each food benefit type at the local agency and clinic. The table contains the total food benefit redemptions, count of food benefits redeemed and the average food benefit. The numbers are calculated on a nightly basis for the month being processed. The date created field indicates which month the averages apply to.

13 CLEANUP LOGS

This cleanup logs process deletes error logs and optimizations logs which are older than 365 days from the current date.