# COBOL Program Analysis:

EDBNAECE is a batch processing program that aggregates statistics from address standardization audits, generates detailed reports, updates a central statistics file, and, in update mode, creates a reapply file for records requiring reprocessing. It handles high-volume address-related data (e.g., Delivery Installation Numbers (DINs), Postal Installation Numbers (PINs), and Local Installation Numbers (LINs)) with robust error handling and logging.

**System Context:** NAS EVERGREENING is likely a mainframe-based system for maintaining and updating address data, ensuring accuracy for postal or delivery purposes. EDBNAECE acts as a post-processing component, summarizing results and preparing data for further action.

## Program Overview: What Does EDBNAECE Do?

EDBNAECE processes audit data generated by upstream address standardization or validation processes. Its primary functions are:

* **Aggregate Statistics:** Reads an audit file to sum counts of processed elements (e.g., DINs, PINs, LINs) and categorize outcomes (e.g., errors, updates).
* **Generate Detailed Reports:** Creates a human-readable detailed statistics file summarizing processing results, such as total records processed, errors, and quality flags (High/Low/Text).
* **Update Central Statistics:** Maintains a cumulative statistics file by adding new counts to existing totals.
* **Create Reapply File (Update Mode):** In update mode (FLAG='U'), generates a file listing records (e.g., DINs) that need reprocessing due to errors or changes.
* **Support Logging and Error Handling:** Logs processing outcomes (success/error codes) and handles file I/O errors, ensuring auditability.

The program does not perform address validation itself but processes the results of such operations. It operates in a batch environment, controlled by a JCL job (NNN7OP0A) via procedure EDGNAS50, and relies on file-based I/O (no DB2 database interactions).

### Key Operational Modes:

* **Update Mode (FLAG='U'):** Enables creation of the reapply file.
* **Logging Level (e.g., 0001):** Controls log verbosity.
* **Environment:** z/OS mainframe, COBOL with extended arithmetic (ARITH(EXTEND)).

### High-Level Workflow:

1. Read audit records.
2. Sum counts (DINs, PINs, LINs, errors).
3. Write detailed statistics.
4. Update central statistics file.
5. (If update mode) Write reapply file.
6. Log results and close files.

## Input Files Expected

The program processes three input files, defined in the JCL procedure (EDGNAS50) and the COBOL FILE SECTION. These files provide the data needed for aggregation and reprocessing.

| **File Name (DDNAME)** | **Description** | **Format/Details** | **Source/Role** |
| --- | --- | --- | --- |
| **AUDITI** | Audit trail from upstream address processing, containing counts, timestamps, error codes, addresses, and flags. Structured via copybook audit.cpy. | Variable Block (VB), LRECL=30731, BLKSIZE=32760. Includes: - Timestamps: N-PROC-START-DT (8 chars), N-PROC-START-TM (4 chars) - Counts: N-NASEVG-DIN, N-SRC-DIN (S9(18) COMP), N-PIN-COUNT, N-NON-STD-LIN-COUNT, N-DIN-COUNT (S9(04) COMP) - Arrays: N-PIN-ARRAY (up to 500), N-LIN-ARRAY (up to 200), N-DIN-REAPPLY (up to 2200) - Flags: N-ADDR-QTY (H/L/T for High/Low/Text), N-AIN-CHANGED (Y/N), N-DIN-FOUND (Y/N), N-ERROR-CODE (0001-0006), N-PROCESS-STG (E/P/U for Error/Processed/AIN-Updated) | Generated by prior NAS EVERGREENING step. DSN: E.EXGB.NASEVG.AUDIT.PROC.FILE(0). Drives main processing loop. |
| **DININP** | Sorted list of DINs for reprocessing, deduplicated by upstream SORT (STEP010). | Fixed Block (FB), LRECL=12, BLKSIZE=27996. Binary integer (positions 1-10). | Sorted from E.EXGB.NASEVG.DIN.FILE(0) to E.EXGB.NASEVG.DIN.FILE.SORT(+1). Validates/applies DINs for reapply file. |
| **STATSO** | Cumulative statistics file with running totals from prior runs. Read and updated. | Fixed Block (FB), LRECL=287, BLKSIZE=32718. Stores aggregates (e.g., total DINs processed). | Persistent file. DSN: E.EXGB.NASEVG.STATS.OUTPUT(0). Read/rewritten for cumulative stats. |

## Output Files Created and How They Are Generated

The program produces three main output files, plus additional files created by the JCL procedure. Outputs are generated through sequential writes or rewrites, controlled by PERFORM sections.

| **File Name (DDNAME)** | **Description** | **Format/Details** | **How Created** |
| --- | --- | --- | --- |
| **STATSD** | Detailed statistics file for human-readable reporting of totals (e.g., DINs, PINs, LINs, errors, quality changes). | Fixed Block (FB), LRECL=57, BLKSIZE=27987. Formatted text for display. | - Sums counts from audit records (e.g., total DINs = sum of N-DIN-COUNT) - Formats into STATS-DETAIL record in C0002-POP-STATS-DETAIL-DATA - Writes via U0008-WRITE-STATS-DETAIL-FILE. DSN: E.EXGB.NASEVG.DISP.STATS(+1) |
| **REAPPLY** | List of records (e.g., DINs, subject IDs, sequences) for reprocessing, created only in update mode (FLAG='U'). | Variable Block (VB), LRECL=207, BLKSIZE=27998. Derived from N-DIN-REAPPLY array. | - Checks FLAG='U' and flags like N-DIN-FOUND='Y' or N-AIN-CHANGED='Y' - Formats array data into REAPPLY-REC in B0003-FORMAT-REAPPLY-DATA - Writes via U0005-WRITE-REAPPLY-FILE. DSN: E.EXGB.NASEVG.REAPPLY.FILE(+1) |
| **STATSO** | Updated cumulative statistics file with new totals added to existing counts. | Fixed Block (FB), LRECL=287, BLKSIZE=32718. Same as input. | - Reads existing STATS-REC - Adds audit sums (e.g., total\_DIN += N-DIN-COUNT) - Rewrites via U0007-REWRITE-STATS-FILE in C0000-TERMINATION. DSN: E.EXGB.NASEVG.STATS.OUTPUT(0) |

### Additional Outputs (via Procedure):

* **Audit Error File:** Filters audit records with N-PROCESS-STG='E' (SETP040, SORT with EDCNAEI2.txt). DSN: THC.E.EXGB.NASEVG.AUDIT.ERROR(+1).
* **Audit Backlog:** Archives audit file (STEP050, IEBGENER). DSN: E.EXGB.NASEVG.AUDIT.PROC.BACKLOG(+1).
* **Stats Tape:** Archives stats file to tape (STEP060, IEBGENER). DSN: THC.E.EXGB.NASEVG.STATS.OUTPUT(+1).
* **SAR Output:** Copies detailed stats to System Archive/Retrieval for printing (STEP030, IEBGENER).

### Generation Process:

* **STATSD:** Aggregated in memory, written once at end of processing.
* **REAPPLY:** Written per qualifying audit record in update mode, after validating against DININP.
* **STATSO:** Updated in memory during processing, rewritten during termination.

## Step-by-Step: How the Program Works

The program follows a structured COBOL flow with divisions (IDENTIFICATION, ENVIRONMENT, DATA, PROCEDURE). Key logic resides in PROCEDURE DIVISION sections, executed as a batch job.

### **1.** Initialization (A0000-INITIALIZATION):

* Opens files: AUDIT, DIN-INP (inputs), STATS (I/O), REAPPLY, STATSD (outputs).
* Sets flags: EOF-AUDIT=FALSE, UPDATE-MODE if PARM FLAG='U'.
* Initializes counters (e.g., WC-AUDIT-READ=0) and working storage for sums.
* Reads initial STATS-REC into memory.
* Calls E1902 module for CPU/elapsed time tracking.

### **2.** Main Processing Loop (B0000-MAIN-PROCESSING):

* Reads AUDIT records until EOF (U0003-READ-AUDIT-FILE).
* For each record:
  + Checks N-PROCESS-STG: If 'E' (error), increments error count and skips.
  + Sums arrays: Adds N-PIN-ARRAY, N-LIN-ARRAY, N-DIN-REAPPLY to running totals.
  + If FLAG='U' and N-DIN-FOUND='Y' or N-AIN-CHANGED='Y', queues DINs for reapply.
  + Increments WC-AUDIT-READ.
* Reads DIN-INP (U0004-READ-DIN-FILE) to validate/append reapply DINs.
* Formats detailed stats (C0002-POP-STATS-DETAIL-DATA) for STATSD.

### **3.** Reapply File Creation (B0003-FORMAT-REAPPLY-DATA, Update Mode Only):

* Iterates N-DIN-REAPPLY array, formats records with DIN, N-SUBJ-ID-NB, N-SUBJ-ID-SEQ-NB.
* Writes to REAPPLY via U0005-WRITE-REAPPLY-FILE.
* Clears REAPPLY-REC after each write.

### **4.** Termination (C0000-TERMINATION):

* Updates STATS-REC with final sums (e.g., total DINs, errors).
* Rewrites STATSO (U0007-REWRITE-STATS-FILE).
* Writes STATSD (U0008-WRITE-STATS-DETAIL-FILE).
* Closes files (U0006-CLOSE-FILES, U0009-CLOSE-STAT-FILES).
* Calls ET530 logging module with return code (e.g., '0000' for success).
* Finalizes CPU/elapsed time (E1902).

### **5.** Error Handling (Z999-ABEND):

* On file I/O errors or CALL exceptions (e.g., ET530 failure), displays error message (e.g., "FAIL: CALL TO ET530").
* Sets abend code (e.g., 020) and calls ABEND routine.
* Logs errors via ET530 (Z900-CALL-LOGGING).

### Key Data Structures (audit.cpy):

* **Fixed Fields:** Timestamps, counts (e.g., N-DIN-COUNT), flags (e.g., N-ADDR-QTY 'H'/'L'/'T', N-ERROR-CODE '0001'-'0006').
* **Arrays:** N-PIN-ARRAY (0-500), N-LIN-ARRAY (0-200), N-DIN-REAPPLY (0-2200).
* **Formatted Address:** N-FORMATTED-ADDR with 7 lines (e.g., N-ADDR-LINE1 to N-ADDR-LINE7).