

GGGGGGG	RRRRRRRRR	OOOOOOO	UUUU	UUUU	PPPPPPPPP	11				
GGGGGGGGG	RRRRRRRRR	OOOOOOOOO	UU	UU	PPPPPPPPP	111				
GG	GG	RR	RR	OO	OO	UU	UU	PP	PP	11
GG		RR	RR	OO	OO	UU	UU	PP	PP	11
GG	GGGG	RRRRRRRRR	OO	OO	UU	UU	PPPPPPPPP	11		
GG	GG	RRRRRRR	OO	OO	UU	UU	PPPPPPP	11		
GG	GG	RR	RR	OO	OO	UU	UU	PP	11	
GGGGGGGGG	RR	RR	OOOOOOOOO	UUUUUUUUU	PP	111111				
GGGGGGG	RRRR	RRRR	OOOOOOO	UUUUUUU	PPPP	111111				

Consumer
Merge Purge
System

Reference Guide

Release 2.7
November 2000
Group 1 Software, Inc.

-
+
0

Table of Contents

0	Table of Contents.	1
0	Consumer Merge/Purge--Introduction	1.1
	Organization of this manual	1.1
	Merge/Purge--Theory and Practice.	1.3
	Program Descriptions.	1.7
	System Flowchart.	1.11
0	Consumer Merge/Purge--System Overview.	2.1
	General Information	2.1
	TAPExx Parameter Card	2.3
	Parameters for TAPE/DISK Exit Routines.	2.8
	TAPE/DISK Exit Routines - Additional Notes.	2.9
	SYNCHK Parameter Card	2.11
	PAGESZ Parameter Card	2.15
	UHDnx Parameter Card.	2.17
	UFTnx Parameter Card.	2.18
	EXITOP--General-Purpose Output Exit	2.19
	Alternate names for programs.	2.23
	IOMODS Parameter Card	2.23
	CMMODS Parameter Card	2.24
	Auxiliary Parameter File.	2.26
	Region/Partition Sizes.	2.27
	Tailoring Memory Requirements	2.29
	Relocation for Subroutines (old DOS operating systems).	2.31
	Static Linkedit (OS/3 only)	2.32
0	Program CM30 -- Consumer Merge/Purge	3.1
	Narrative and Flow.	3.1
	Household Duplicate-Elimination	3.2
	Individual Duplicate-Elimination.	3.3
	All-Resident Duplicate-Elimination.	3.3
	Tightening the Comparison	3.3
	Loosening the Comparison.	3.5
	HEADER Parameter.	3.7
	TAPExx Parameter Card	3.9
	MERGES Parameter Card	3.11
	SEQERR Parameter Card	3.13
	BRKFLD Parameter Card	3.15
	RECLIM Parameter Card	3.17
	CONTRL Parameter Card	3.19
	FLCODE Parameter Card	3.27
	ZPCODE Parameter Card	3.29
	ADDRDF Parameter Card	3.31
	NAMEDF Parameter Card	3.33
	CMPFLD Parameter Card	3.35

ALGSEL Parameter Card	3.37
ZIPC Algorithm.	3.40
SURN Algorithm.	3.41
INDV Algorithm.	3.42
HSNR Algorithm.	3.43
STNM Algorithm.	3.44
POBX Algorithm.	3.45
- RRBX Algorithm.	3.46
APNR Algorithm.	3.47
ADDR Algorithm.	3.48
HHLD Algorithm.	3.49
TOTL Algorithm.	3.50
RESI Algorithm.	3.51
DUPCTL Parameter Card	3.53
DUPREC Parameter Card	3.55
CLNREC Parameter Card	3.57
PRIORT Parameter Card	3.59
FLC Parameter Card.	3.65
CHANGE Parameter Card	3.71
MULTIB Parameter Card	3.73
MLTLIM Parameter Card	3.75
CMPLIM Parameter Card	3.77
POST Parameter Card	3.79
SEQUNx Parameter Card	3.85
PEXITS Parameter Card	3.87
User Exit Routine	3.89
General Discussion.	3.89
Skeletal Example.	3.90
Functions A, B, and C	3.91
Function D.	3.92
Function L.	3.93
Function X.	3.94
INTRPT Parameter Card	3.97
CHCKPT Parameter Card	3.99
Operating Instructions.	3.101
Sample OS JCL	3.103
Sample DOS JCL.	3.104

0Program CM31 -- Extracted File Print	4.1
Narrative	4.1
HEADER Parameter.	4.3
TAPExx Parameter Card	4.5
CONTRL Parameter Card	4.7
HEADxx Parameter Card	4.9
CONSTn Parameter Card	4.11
MOVExx Parameter Card	4.13
FORMAT Parameter Card	4.15
UNPKxx Parameter Card	4.17
RECLIM Parameter Card	4.19
FLCSEL/FLCREJ Parameter Card.	4.21
NTHSEL Parameter Card	4.23
Operating Instructions.	4.25
Sample OS JCL	4.27
Sample DOS JCL.	4.28
0Program DE40 -- Duplicate Elimination Summary.	5.1
Narrative	5.1
HEADER Parameter Card	5.3
TAPExx Parameter Card	5.5
CONTRL Parameter Card	5.7
BROKER Parameter Card	5.9
- BRNAME Parameter Card	5.10
FLCSEL/FLCREJ Parameter Card.	5.11
Operating Instructions.	5.13
Sample OS JCL	5.15
Sample DOS JCL.	5.16
0Program DE41 -- State Counts by File Code.	6.1
Narrative	6.1
HEADER Parameter Card	6.3
TAPExx Parameter Card	6.5
CONTRL Parameter Card	6.7
FLCSEL/FLCREJ Parameter Card.	6.9
Operating Instructions.	6.11
Sample OS JCL	6.13
Sample DOS JCL.	6.14

0Program DE50 -- Multi-buyer Report and Match Analysis.	7.1
Narrative	7.1
HEADER Parameter Card	7.3
TAPExx Parameter Card	7.5
CONTRL Parameter Card	7.7
MULTIB Parameter Card	7.13
MLTLIM Parameter Card	7.15
PRIORT Parameter Card	7.17
SEQCHK Parameter Card	7.19
SEQERR Parameter Card	7.21
ANALIM Parameter Card	7.23
FLCSEL/FLCREJ Parameter Card.	7.25
CHCKPT Parameter Card	7.27
Operating Instructions.	7.29
Sample OS JCL	7.31
Sample DOS JCL.	7.32
0Program DE60 -- Duplicate File Print	8.1
Narrative	8.1
HEADER Parameter Card	8.3
TAPExx Parameter Card	8.5
CONTRL Parameter Card	8.7
HEADxx Parameter Card	8.9
CONSTn Parameter Card	8.11
MOVExx Parameter Card	8.13
FORMAT Parameter Card	8.15
UNPKxx Parameter Card	8.17
RECLIM Parameter Card	8.19
FLCSEL/FLCREJ Parameter Card.	8.21
NTHSEL Parameter Card	8.23
Operating Instructions.	8.25
Sample OS JCL	8.27
Sample DOS JCL.	8.28
0Program DE61 -- Duplicate File Match	9.1
Narrative	9.1
HEADER Parameter Card	9.3
TAPExx Parameter Card	9.5
- CONTRL Parameter Card	9.7
HEADxx Parameter Card	9.11
RECLIM Parameter Card	9.13
NTHSEL Parameter Card	9.15
Operating Instructions.	9.17
Sample OS JCL	9.19
Sample DOS JCL.	9.20

0DCOMPRES Program -- Extract program(s) from Delivery Tape.	10.1
Narrative and Flow.	10.1
HEADER Parameter Card	10.3
TAPExx Parameter Card	10.5
CONTRL Parameter Card	10.7
INSERT Control Card	10.9
COPY Control Card	10.11
Operating Instructions.	10.13
Sample DOS JCL.	10.15
Sample OS JCL	10.16
0File Layouts and Sample Runs	11.1
Extract Area Format	11.1
Duplicate Elimination Summary Statistics	11.3
Duplicate Elimination State-Count by File Code	11.5
List Conversion Summary Statistics (FILEOS)	11.7
Sample Runs	11.9
0Index.	12.1

Introduction

0Organization of this manual

- This manual is organized into an introductory section, a System Overview section, a section for each EXEC-level program within the system, a section on system installation, a section describing the DCOMPRES program (used to extract programs from the delivery tape), and a section containing File Layouts and a sample execution of the system.
- 0 This, the introductory section, contains a brief introduction to Merge/Purge concepts in general, a brief overview of the programs included in the system and their functions, and a system flowchart.
- 0 The System Overview section provides information about various facilities and features which are common to more than one of the programs. In general, it should be read once, carefully, and thereafter referred to as necessary.
- 0 The individual program sections cover, in detail, the functions and control facilities of the particular programs. The file assignments, parameter card formats, and sample JCL are included.
- 0 The sections on Installation and the DCOMPRES program must be read before the system is first installed, and may be referred to as required when new releases are received.
- 0 File layouts and a sample run are provided in the final section, for reference as required. Some study of the example run, in particular the control cards used for program CM30 and their results, may prove rewarding.
- 0 It is recommended that you read the entire manual once, before devoting a great deal of time to the in-depth study of any particular section. As you go through the individual program sections, refer to the appropriate section of the sample run. This should give you a general understanding of the system components, how each may be used, and the type of information which may be derived from the available reports. Once this background has been assimilated, you will find that careful study of each section is much more productive.

Introduction

0Merge/Purge--Theory and Practice

- In theory, merge/purge operations are very simple: you just look at each pair of records, and if they appear to be duplicates, you mark them as such. Of course, you should also keep track of how many duplicates you find, which one(s) you decide to keep, and so forth--and preferably you must keep track of such statistics subdivided according to the source of the records.

0 A person can do the job outlined above, much better than any computer software--as long as the number of records is kept small. A file of 10 records involves just 45 record-to-record comparisons. As the number of records grows, however, people can no longer keep up with the task. A file of 100 records requires 4950 comparisons, and a file of 1000 records requires 499,500 comparisons.

0 Even for today's fastest computers, the job (as defined above) is much too great for files of significant size. A file of one million records would require just under 500 billion comparisons. Accordingly all computer software which attempts to find duplicates must rely on sortation of large input files into some sequence which facilitate the duplicate-identification process, by eliminating most pairs of records from consideration. After all, most of the time a person or company with a Pennsylvania address is not a duplicate of the equivalent entity with a Hawaii address!

0 But some of the time, a record in Hawaii is properly a duplicate of one in Pennsylvania. Consider:

0 Pennsylvania Central Asphalt
 1050 Mirkwood Dr.
 Morrisdale, PA 16858

0 and

0 Pennsylvania Central Asphalt
 1050 Mirkwood Dr.
 Honolulu, HI 96858

0 Here, it is pretty clear that the records are duplicates, and that what has happened is that somewhere along the line the ZIP Code in the second record was mis-entered, and later some computer program "fixed" the City name based on the ZIP Code. However,

0 John Smith
 PO Box 128
 Morrisdale, PA 16858

0 and

0 John Smith
 PO Box 128
 Honolulu, HI 96858

0 might be duplicates, but they might not. Here we would probably accept the juxtaposition of these records as an interesting coinciden

Introduction

0Merge/Purge--Theory and Practice

- without giving any credence at all to the possibility that they were duplicates. Change the name again, though, to say

0 Elwood P. Krzysolski

0 and once again we might suspect that the records are for the same person.

0 Fortunately (for software vendors' customer support personnel) most people agree that computer software should not attempt to find these clever matches, at least not without being "told" to do so. Typically the sortation and "break logic", which is necessary to reduce the number of comparisons to a manageable size, also keeps such pairs from even being considered as potential matches.

0 So, now we can see three different elements of a computerized duplicate elimination process: 1) the sortation of the input files, which is designed to bring together those records which are likely to be duplicates; 2) the "break logic" which keeps some records from being compared with one another, preferably those records which are unlikely to be duplicates; and 3) the comparison logic and rules for determining which pairs of records are duplicates and which are not. Each of the three areas must be considered with care, and in relation to one another, if the duplicate identification process is to be successful. For the sortation, several types of fields come immediately to mind; in fact, the vast majority of general-purpose duplicate identification passes will use one or more of the following:

0 The City name--this field will be of great use if it has been mechanically standardized, and of somewhat less use if it has not been standardized. If standardized, then it makes a nice 11-byte (Canadian standard) or longer field which will be equal for equal records, and is unlikely to be for entities which are actually in different geographical areas. In other words, an equal standardized city name is a pre-requisite for consideration as a duplicate.

The State or Province abbreviation--this field is also of great use if standardized. If not standardized mechanically, it will still be of great use but may be weak in certain areas (is the abbreviation for Nebraska NE or NB? for Manitoba MT or MB? etc.).

The ZIP Code or Postal Code--for a consumer dupe-identification the five-digit ZIP Code or six-character Canadian Postal Code is a near-ideal sort field element--provided it is known to be correct. For a business duplicate identification, the first three digits of the ZIP Code are frequently used with great success, but the same will not be true for Canadian files. The city of Montreal alone has upwards of forty FSAs (three-character Canadian Postal Codes). The problem with business duplicate identifications is that frequently the same organization will

Introduction

0Merge/Purge--Theory and Practice

- have more than one address--most likely a Post Office box as well as a street address--while this is relatively rare with private persons.

The entity name--for a consumer duplicate identification, the surname of the person; and for a business duplicate elimination the business name. This field is clearly deserving of a place in the sortation, but particularly in the case of businesses must be included in the "break logic" very carefully if at all. For example, 'A B C COMPANY' and 'ABC CO' are pretty clearly the same, but they will not sort together. Frequently, it is possible only to remove the duplicates of each spelling of a given company name when they vary considerably. The same is almost always true when an organization has more than one proper name--if 'ALTOONA MOTOR CO' and 'BOB PERKINS CHEVROLET' are actually the same firm, it is pretty tough for a computer program to be aware of that fact.

The address--in general this field is pretty weak for sortation (remember the Post Office boxes and street addresses above), but it will be of use when the file contains no names, or when the names are to be ignored.

0 The "break logic" facility provided in this system consists of the specification of the number of characters of the sort string which must be equal in order for two records to be compared (note that if two records cannot be compared they cannot be identified as duplicate of one another). This definition of a "break" will influence your choice of the order of the elements within the sort string as well as your choice of the elements themselves. For instance, if the firm name is the first element within the sort string, then the "break length will apply (first) to the company name, and either your break definition will divide the entire file up into only twenty-six segments or you will never see 'ABC CO' as a duplicate of 'A B C COMPANY'. In general, the most general fields should be presented first in the sortation string (State, City, ZIP Code), and the most specific elements should be presented last (Name, Address).

0 The actual comparisons used to determine duplication or uniqueness, their sequence (which has a great deal to do with execution speed), and the interpretation of the results of the comparisons, are reserved for the program itself. By means of the ALGSEL cards, you may tighten or loosen various elements of the comparisons, and/or introduce special restrictions on the comparisons.

1

Introduction

0Program Descriptions

- This section gives a brief outline of the functions and capabilities of each program in the system. For a detailed description of the parameters needed to activate and control the various functions, see the individual program sections.

0 CM30:

0 Program CM30 is the heart of the Consumer Merge/Purge system. It identifies and extracts the fields to be used in record comparisons, performs the comparisons, selects the surviving records when duplicates are detected, and writes statistical summary records to facilitate reporting by the other programs.

0 The first function, identification and extraction of the fields to be used in record comparison, can be done in a separate pass (of the input file) by itself, or combined with the record comparison/duplicate-identification function. In general, the extraction must be separated from the comparison when the extracted fields are to be used in the sortation (for duplicate identification purposes) of the records. If a useful sortation is achievable without using extracted fields, then the extraction may be either done separately or combined with the comparison.

0 The second primary function of CM30 is the comparison of records and the identification of duplicates based upon the result of the comparisons. By default, the program comes prepared to do a "medium" (as opposed to "tight" or "loose") merge/purge at the Household level. An Individual merge/purge, or tightening or loosening of the comparisons of the various elements, may be specified via the ALGSEL parameter cards. In addition, you may provide CMPFLD cards which cause specific fields to be compared and used as a quick way to determine duplication or uniqueness.

0 Once a pair of records have been identified as duplicates, the question remains as to what to do with them. Here, CM30 gives you a great deal of flexibility. To begin with, each File Code may be assigned its own priority; this priority may be additionally modified based on the presence or absence of information in the individual records (see the PRIORT card); and finally the record with the highest priority is selected as the survivor. Alternatively the surviving record may be selected at random; or random selection and priority selection may be mixed.

0 The primary output files are a file of Duplicate groups (File OD), and a "clean" file with the duplicates removed. This latter file can be obtained in two copies (Files OM and OE), if desired; furthermore, the clean records may be split between those two files in two different ways under parameter card control (see the CONTRL card). File OD is necessary if the Match Analysis program, DE50, is to be executed later.

Introduction

0Program Descriptions

CM30

- Statistical files can be written which permit the production of various Summary reports, by subsequent programs. Unless you are certain that the report will not ever be required, you will be well advised to produce these files (File OT and OX), because if you do not, and if one turns out to be required later, you will be faced with re-running the entire duplicate elimination run.

0 Additional features of this program allow you to identify certain File Codes as "Purge" files to suppress all matching records; to post information from one record to another when duplicates are detected; and to treat the records from specified File Codes as 'RESIDENT' names.

0 CM31:

0 Program CM31 can be used to produce a listing of the records produced by the Extract function of program CM30. One line is printed (under parameter control) containing information from the original record, followed by the extracted fields identified as to content. Normally, you will want to print several files (or subsets of them) in this fashion early on, to get a feel for the way the data is stored; thereafter, you will need it only rarely.

0 DE40:

0 Program DE40 can be used to produce a summary report by File Code, based upon statistics from program CM30. If available, it will also incorporate statistics from any List Conversion process which may have been performed on the files.

0 If some or all of the input files have been obtained from List Brokers, those File Codes may be identified with a Broker Code and a Broker Name, and a report by List Broker may be produced also.

0 DE41:

0 Program DE41 can be used to produce State Counts by File Code in a variety of ways, based upon statistics records written by program CM30. Reports can be produced with one page per File Code, with the States listed vertically; with one page per State, with the File Codes listed vertically; or as one summary page with the States listed vertically. The first two of these reports require a sortation of the statistics file prior to report production. You have the choice of defining "state" to mean either a three-digit ZIP Code range (which will cause Puerto Rico to be split into two segments, and will separate the APO/FPO ZIPs from the state proper in New York, Florida, California, and Washington), or a common state abbreviation (which will cause only one entry/page each for the above states). If the original file contained Canadian records, then counts by Canadian province will be produced as appropriate.

Introduction

0Program Descriptions

+

-

DE50:

+

0

Program DE50 reads the duplicate records file from CM30 to analyze the File-Code-to-File-Code matches. The file of File Code Statistics is also required, and a file of List Conversion Statistics can be introduced if desired. Several different reports can be produced, as follows:

0

The Drop Summary displays, by File Code, the number of records dropped by type of drop. Drops are categorized as Purge Drops, Multi-Buyer Drops, and Single-Buyer drops, depending upon the attributes of the File Codes present in each duplicate group. The number of Intra-File (same File Code) drops is also displayed for each type of drop.

0

The Multi-Buyer report displays, for each File Code, the number of Multi-Buyer records in total, and subdivided by the "number of times Multi-Buyer". A record is considered a two-time Multi-Buyer if there are two File Codes involved in the duplicate group, a three-time Multi-Buyer if there are three File Codes, and so forth.

0

The Match Analysis prints one page per File Code, with all the File Codes listed vertically. The number of matches detected between the "page" File Code and each of the "column" File Codes is displayed, and expressed as a percentage of the "page" File Code input count and as a percentage of the total matches for the "page" File Code. At the bottom of each page the total number of records dropped, by drop type, is displayed, with the number of those drops which are also Intra-File drops. For those who rent lists, study of these reports can give insight as to which lists can be avoided in the future, due to apparent overlap with others.

0

Finally, the Match Matrix report lists each File Code both vertically and horizontally, with the number of matches between the File Codes displayed at the intersections of the rows and columns. The report is organized and marked in such a way that it can be used either as is (a stack of fan-fold pages), or cut and pasted into a large rectangular array and mounted on the wall (or on a boxcar, if the number of File Codes is large!).

0

DE60:

0

Program DE60 can be used to print the duplicate records file from program CM30. One line is printed per record, in a format completely controlled by parameter cards. The portion of the duplicate file to be printed can be controlled, and you may print only every Nth duplicate group. Headers can be fully specified to create an attractive report.

Introduction

0Program Descriptions

DE60

- Alternatively, the clean file or any other file not organized into duplicate groups can be printed by this program. Another option is to produce the formatted listing on tape for use by program DE61.

0 DE61:

0 Program DE61 can be used to print those duplicate groups from one duplicate-print file (produced by DE60) which are not present on another such file. By this means two duplicate files (or clean files) can be compared without the necessity for examining those duplicate groups which were not changed.

0 ISFN:

0 Dynamically loaded subroutine used to separate a natural-order name into its component parts.

0 ISAP:

0 Dynamically load subroutine used to extract city and state information from an array of address lines, and to shuffle the remaining lines to identify the most significant.

0 AMSTP:

0 Dynamically loaded subroutine used to separate a natural-order address into its component parts.

0 CANSTRS:

0 Dynamically loaded subroutine used to separate a natural-order address into its component parts.

0 When an input file consists entirely or substantially of Canadian records, you should use the CANSTRS routine instead of AMSTP (which is intended primarily for U.S. addresses). This may be accomplished by providing a CMMODS parameter card to CM30.

0 USACODR:

0 Dynamically loaded subroutine (used with AMSTP) to provide specialized street suffix tables for the routine.

0 CANCODR:

0 Dynamically loaded subroutine used (with CANSTRS) to provide specialized street suffix tables for the routine.

0 When an input file consists entirely or substantially of Canadian records, you should use the CANSTRS routine together with CANCODR. This may be accomplished by providing both routines on a CM30 CMMODS card.

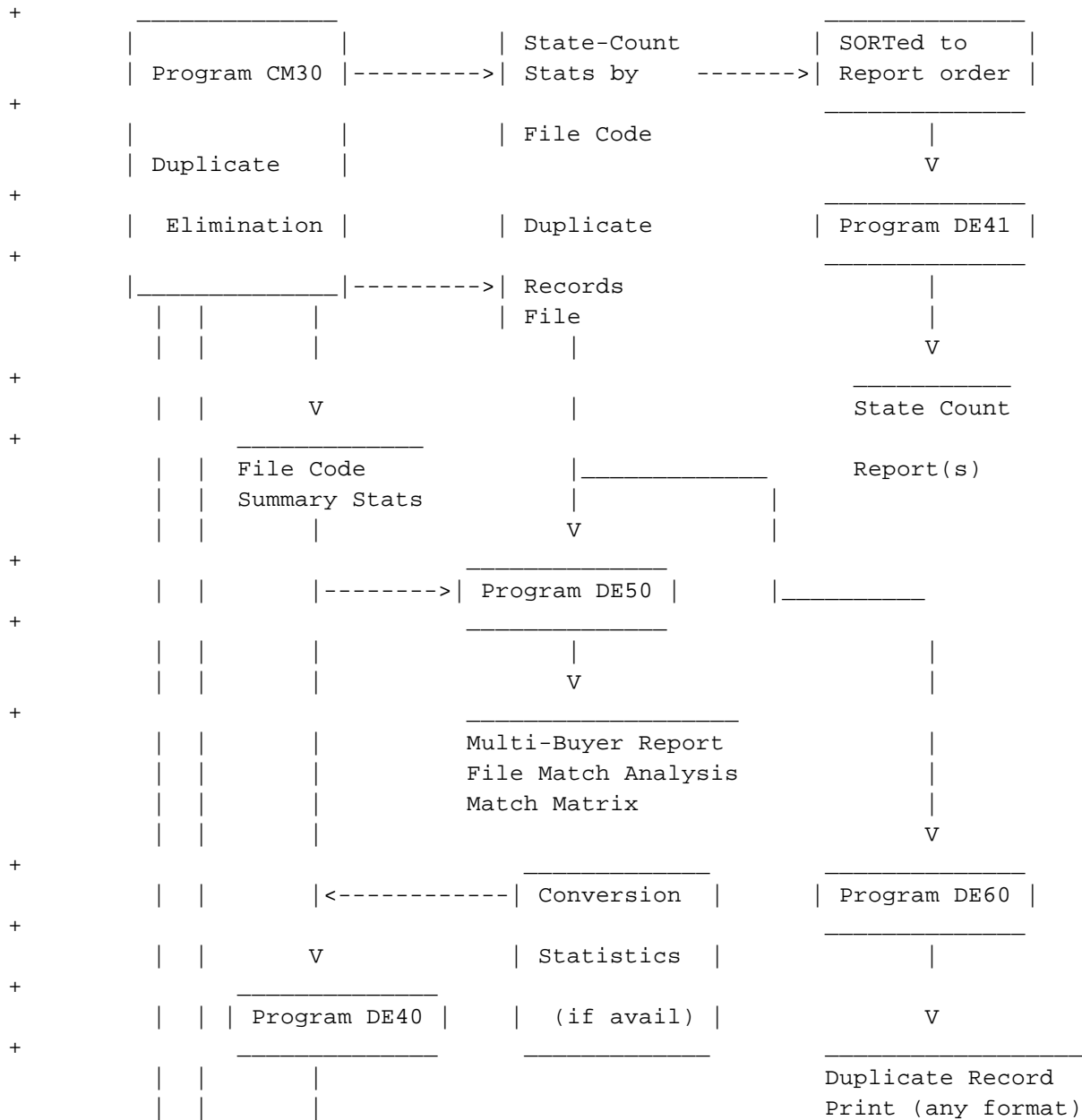
Introduction

0Consumer Merge/Purge -- System Flowchart

- Input name-and-address records, organized in up to 200 different File Codes ("Key Codes", "List Codes", "Source Codes", etc.), are first converted to a common format.



These records are then sorted into a sequence suitable for the Merge/Purge (Duplicate Elimination) process. The sorted records can be left on up to three separate files.



```
+      | |      v
      | |      -----
      | |      Summary Report
      | |      Broker Report(s)
      | |
+      | |      -----
      | |      Control Totals
      | |      File Code I/O Report
      | |      State Count Summary
      | |
      | |      v
+      | |      -----
      | |      Mail (Clean) File
1
```


1Group 1 Software -- Consumer Merge/Purge 2.1
System Overview

0General Information

- There are several features and/or options common to all programs in the Consumer Merge/Purge System. Following are descriptions of these features and/or options.

1

1Group 1 Software -- Consumer Merge/Purge 2.3
System Overview

0General Information

0TAPExx Parameter Card

- All EXEC-level GROUP 1 Mail Management programs require a TAPExx or DISKxx card to trigger processing of each input or output file (except the card reader and printer). The format of the card required is the same for all programs, as follows:

CARD COLUMNS	FIELD DESCRIPTION
0 01 - 04	PARAMETER KEYWORD - Must contain 'TAPE' or 'DISK' to indicate the DTF name to be used.
0 05 - 06	FILE ACTIVATOR - Must contain a two-position code that identifies the file to be activated. (See the list of File Activators for each EXEC-level program.)
0 08 - 11	BLOCK SIZE - Must contain the block size of the file indicated.
0	For a VSAM keyed file, this field should contain the same value as the Record Size (columns 13-15). (VSAM keyed files are used only for specific purposes, such as reference files for keyed lookup usage recording, etc. Each usage will be accompanied by specific instructions.)
0	For a VSAM numbered file, this field should contain the size of the "nominal" block, as defined in the file creation process. See the instructions for creating the file. (VSAM numbered files are used only for specific purposes, such as "indexed" Mast Files, Audit File records, etc. Each usage will be accompanied by specific instructions.)
0	Note: If necessary, this field may be extended one position to the left, allowing the entry of a five-digit block size in columns 7-11.

0General Information

0TAPExx Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	13 - 15	RECORD SIZE - Must contain the record size of the file indicated.
0		For an indexed Master file or the associated Index column 12 must contain 'X' and columns 13-15 must contain the number of blocks per track. If using VSAM (numbered), columns 13-15 should be set to '001'. Note that the indexed Master File specification is not valid for FBA devices in a DOS environment (unless using VSAM), nor available at all under OS/3.
0		Note: If necessary, this field may be extended one position to the left, allowing the entry of a four-digit record size in columns 12-15.
0		Record sizes larger than 9999 can be specified by using the letters 'A' through 'W' in column 12 to represent high-order 'digits' with the values 10 through 32, respectively.
0	16 - 16	RECORD FORM - Must be blank for fixed-length records or 'V' for variable-length blocked records.
0	17 - 17	LABELS - Must contain 'S' for standard-label files, or 'N' for non-labeled files.
0	18 - 18	ASCII TRANSLATE - For input files, if this column contains 'A', then each input record will be translated from ASCII code to EBCDIC. For variable length records, the RDW will not be translated. This option should be used only if the records contain no binary or packed fields that will be needed, as fields of this sort can not be processed correctly via the translate.
0	19 - 19	BUFFERS - Must contain '1' or '2', indicating the number of buffers to use.

0General Information

0TAPExx Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	21 - 24	DEVICE TYPE - May contain one of the following to indicate the device to which the file is assigned.
0		For OS or DOS: VSAM if the file is either an Indexed Master File stored under VSAM, or a VSAM keyed or numbered file constructed in accordance with specific instructions for that file. For OS, no value of this field except 'VSAM' has any significance.
		For DOS: TAPE, 2311, 2314, 3310, 3330, 3340, 3350, 3370, 3375, 3380, or FBA
		For OS/3: TAPE or DISK
		For either: IGNR, which causes the following: OPEN and CLOSE requests will be ignored. READ requests will be treated as if end-of-file were reached.
		WRITE requests will be ignored.
0		Note that for DOS, the IGNR device type is more powerful than the IGNORE JCL option, because it works properly for input files.
0		Blank defaults to TAPE for TAPExx cards, and to 2314 or DISK for DISKxx cards.

0General Information

0TAPExx Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	26 - 26	TAPE REEL HANDLING - Must contain one of the following codes to indicate the desired handling of each reel of magnetic tape. This column is ignored for files assigned to disk devices. The codes are:
0		blank - (normal) Magnetic tape reels will be rewound at each Open or Close operation. For no-label Tape files or multi-volume standard-label files, this treatment will be changed to 'U' (see below) if a multi reel or multi file situation is detected
0		'U' - Magnetic tape reels will be rewound at Open operations, and unloaded at Close operations.
0		'X' - Magnetic tape reels will not be repositioned at either Open or Close operations.
0		In general, blank should be used for intermediate occurrences of a single-reel file; 'U' should be used for multi-reel files and for the final occurrence of single-reel files; and 'X' should be used for files which require non-standard positioning of the tape reel (which can be accomplished via MTC commands or the equivalent).
0	28 - 28	MULTIPLE STANDARD-LABEL FILES - May contain 'M' to indicate that the input file consists of more than one standard-label file. The operator will be queried for EOF/EOV at the end of each of the input files.
0	54 - 63	ALTERNATIVE FILE TREATMENT - May contain values which cause alternative treatment of the file, as follows:
0	54 - 56	ALTERNATE SYS NUMBER - If these columns are numeric, then the value entered will be used as the SYS number for the file, instead of using the SYS number noted in the Operations section. In this case, columns 57-63 are ignored.

0General Information

0TAPExx Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	54 - 61	EXIT ROUTINE NAME - If non-blank, these columns are treated as the name of an exit routine which is to handle the I/O operations. In this case, the TAPE/DISK card is ignored, the exit routine will be given control for each operation on the file (see description following), and column 63 is used as the COBOL indicator as follows.
0	63 - 63	COBOL INDICATOR (DOS ONLY) - If the exit routine is written in ANS COBOL under DOS, place a 'C' in this column.
0	65 - 71	PRECISE REEL CUT-OFF - May contain a number to specify a precise number of records for each reel of an output tape file. If specified, then each reel of the file will contain exactly this number of records. This field will be ignored if the file is not assigned to tape, if it is not numeric if it is zero, or if the file for which specified is an input file. This option is ignored if an exit routine is specified in columns 54-63.

0 General:

0 The TAPExx/DISKxx card is required in all systems to activate the file indicated. Columns 54 and up are used in all systems for the purposes indicated. If an exit routine is not specified, then columns 7-28 are ignored in OS systems (this information is taken from the JCL provided), and are required in DOS and OS/3 systems.

0 Under DOS and OS operating systems, an Indexed Master File, or the non-sequential Index for an Indexed Master File, may optionally be stored as a VSAM relative-record file. For either Operating System, the word 'VSAM' in the device type field (columns 21-24) will cause the file defined to be treated as a VSAM file. Under OS, the file attributes are obtained when the file is opened. For DOS, the block size must be entered in columns 7-11, an 'X' must be placed in column 12, and the value '001' must be placed in columns 13-15; the remaining fields are not used when the file is VSAM.

0General Information

0TAPExx Parameter Card

Parameters for TAPE/DISK Exit Routines

- The following parameters are passed to the TAPE/DISK Exit Routines via standard linkage conventions at each I/O operation request:
- 1) A word-aligned 16-byte area as follows:
 - 0 File Activator, 2 bytes
 - Function byte, as follows:
 - 'O' for open
 - 'C' for close
 - 'R' for read
 - 'W' for write
 - Binary SYS number, one byte
 - The remainder of the area is reserved
- 0 2) A word-aligned 16-byte area which must be filled by the Exit Routine as follows:
 - 0 Record length (halfword) - Must be set to fixed or maximum record length by the Open call
 - Must be set to the length of the current record by the Read call, or to zero to indicate end-of-file
 - Not used by the Close call.
 - Record type (byte) - Must be set to 'F' (fixed-length records), 'V' (variable-length records), or 'U' (undefined records) by the Open call
 - Not used by the Read, Write, or Close calls
 - The remainder of the area is reserved.
- 0 3) The area containing (for Write) or to contain (for Read) the next record. This area should not be used by the Open or Close routines.
- 0 4) A word-aligned 80-byte area containing the TAPE or DISK card image.

0General Information

0TAPExx Parameter Card

TAPE/DISK Exit Routines - Additional Notes

- 1. For OS and DOS, the Tape/Disk exit routines are always dynamically loaded at execution time. For Univac OS/3, the exit routine must be linked with the primary program (see below).
- 0 2. Under DOS, if the Exit Routine is written in ANS COBOL, the following must be observed in order for the exit routine to "know" that it is a called routine:
 - 0 A. Create an assembler program named MNS0ADDR as follows:
 - 0 MNS0ADDR START
 - 0 DC V(ILBDMNS0)
 - 0 END
 - 0 B. Link the exit routine with this 'program', as follows:
 - 0 INCLUDE MNS0ADDR
 - 0 INCLUDE exit
 - 0 ENTRY exit
 - 0 C. Now the eight-byte 'program' MNS0ADDR will immediately precede the entry-point of the exit, providing its caller with the address of the flag byte to be set when an ANS COBOL exit is to be called.
 - 0 D. Place a 'C' in column 63 of the TAPE/DISK card to cause the COBOL logic to be invoked.
- 0 3. For Univac OS/3 systems, the exit routine must be linked statical with the main program. This is accomplished as follows:
 - 0 A. Find the label TAPDSK in the main program.
 - 0 B. Following this label is an entry for each TAPE/DISK card (file) possible. Locate the entry for the file that is to be handled by the exit routine.
 - 0 C. Change the definition of the entry to place a V-type address-constant referring to the exit's entry-point at displacement 36 in the entry located.
 - 0 D. Regardless of the contents of the TAPE/DISK card, the exit routine will now be invoked as noted above.

0General Information

0SYNCHK Parameter Card

- The SYNCHK parameter allows the content of any sequential file activated via a TAPExx or DISKxx parameter to be checked at defined intervals, ensuring that the correct file is read and that no data has been lost. For an output file, a record is written every nth record, containing the record number and the contents of a user defined control field, up to 30 characters. When the same file is read (it must still be in the same sequence), the synchronization file is also read, and the contents of the input file are checked against the data stored in the synchronization file. When a discrepancy is detected, it is noted on the console, and one of three dispositions takes effect: 1) continue; 2) cancel the job; or 3) ask the operator for one of the first two choices.

0 Normally, this procedure is done when there is a large file which must be transmitted exactly as written. If checkpoints are also taken, then if a discrepancy is detected, the job can be cancelled. Then the cause of the error can be found, corrective steps taken if necessary, and the job restarted again from the last checkpoint at which the data was in sync. Note that the field selected should most likely contain some field which indicate the point in the file at which any record is found, so that the discrepancy messages can be readily interpreted. For many files, the ZIP or Postal code serve nicely.

0 The format of the SYNCHK parameter is the same for all program as For simplicity, the (large) file whose content is critical, will be referred to as the "checked" file, and the file of synchronization records will be referred to as the "sync" file.

-	CARD COLUMNS	FIELD DESCRIPTION
0	01 - 06	PARAMETER KEYWORD - Must contain 'SYNCHK'.
0	08 - 09	ACTIVATOR OF CHECKED FILE - Must contain the two-character File Activator of the file to be checked. If that file is not opened, then the sync file will not be opened either. Up to 3 SYNCHK cards (for different files) may be supplied for any single execution.
0	11 - 13	CONTROL FIELD LOCATION - Must contain the location, in the checked file records, of the field which is to be checked. This location is applied to the records of the checked file at read or write time, in the format of the records on the tape or disk.
0		If necessary, this field may be extended one position to the left, permitting the entry of a four-digit location in columns 10-13.

0General Information

0SYNCHK Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	15 - 16	CONTROL FIELD LENGTH - Must contain the length of the control field, up to 30.
0	18 - 24	FREQUENCY OF SYNCHRONIZATION - Must contain the number of records between successive synchronizations. For an input file, of course, the actual number of records between checks is determined by the input sync file, but this field is still validated.
0	26 - 26	ERROR DISPOSITION - Must contain one of the following codes to indicate the treatment desired when a discrepancy is detected between the checked file and the sync file: C - continue processing; in this case, the only evidence of the discrepancy will appear on the console log E - terminate processing O - ask the operator for one of the above treatments
0	28 - 30	SYS NUMBER FOR SYNC FILE - Must contain the SYS number to use for the sync file. No TAPE/DISK card may be provided for the sync file, but it must be assigned (that is, a DD or ASSGN card must be present, depending on the operating system). For DOS and OS/3 systems, the DTF name used for the sync file will be SYNCxx, where xx is the File Activator for the checked file.
0	32 - 35	DEVICE TYPE FOR SYNC FILE - For DOS and OS/3, must contain one of the device types noted under the TAPExx/DISKxx parameter card. If omitted, TAPE will be assumed. This field is neither used nor verified under OS.
0	37 - 37	LABELS FOR SYNC FILE - For DOS and OS/3, must contain one of the following codes to indicate whether or not the sync file has or is to have standard labels. For DOS and OS/3, the value will always be verified, but will be ignored if the sync file device is not tape. For OS, this field is neither used nor verified. The codes are: S - Standard labels N - no labels blank - defaults to 'S'

1Group 1 Software -- Consumer Merge/Purge 2.13
 System Overview

0General Information

0SYNCHK Parameter Card

+

- The records written to the sync file will be 80 characters long, and will be written unblocked (for OS, the blocking will be determined by the DD statement). Each record will have the following form:

0	Field Description/Contents	Format	Length	Positions
0	Record Identification contains 'SYNC POINT'	C	10	001 - 010
0	(blank)	-	1	011 - 011
0	File Activator of checked file	C	2	012 - 013
0	(blank)	-	1	014 - 014
0	Number of record in checked file	9	9	015 - 023
0	(blank)	-	1	024 - 024
0	End-of-file indicator X if the checked file should be at end-of-file blank otherwise	C	1	025 - 025
0	(blank)	-	1	026 - 026
0	constant 'DATA=', or blank if 25 is 'X'	C	5	027 - 031
0	control field data, or blank if position 25 is 'X'	C	30	032 - 061
0	(blank)	-	19	062 - 080

1

1Group 1 Software -- Consumer Merge/Purge 2.15
 System Overview

0General Information

0PAGESZ Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'PAGESZ'.
0		This parameter card is provided to permit you to alter the standard pagination of reports, which is set up for 11-inch paper printed at six lines per inch.
0	08 - 10	PAGE SIZE - May contain the maximum number of lines to be printed on any page, including headers. The minimum permissible value is 25 and the default is 60.
0	12 - 12	PRINT TRANSLATION - Must contain one of the following codes to indicate whether a lower-to-upper case character translation is to occur before printing a print line:
0		Y - Yes, a translation is to occur
		N - No translation is to occur
		blank - defaults to N

1

1Group 1 Software -- Consumer Merge/Purge 2.17
System Overview

0General Information

0UHDnx Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 03	PARAMETER KEYWORD - Must contain 'UHD'.
0		UHDnx parameter cards may optionally be used in any Group 1 EXEC-level program to define one to four lines of user-controlled report heading, which (if present) will be printed above the normal (program-generated) report headings. When UHDnx cards are used, the normal report headings will be double-spaced beneath those generated by the UHDnx cards.
0	04 - 04	USER HEADING LINE NUMBER - Must contain '1' to '4' to specify the first through the fourth line of heading.
0	05 - 05	LEFT OR RIGHT SIDE OF LINE - Must contain an 'A' or a 'B' to indicate that the left ('A') or right ('B') side of the line is being defined by this card. The left side of a line is print positions 1 through 66, and the right side is print position 67 through 132.
0	07 - 72	DATA TO PRINT - Must contain the 66 characters of text which is to be used as the left or right side of the specified heading line.

1Group 1 Software -- Consumer Merge/Purge 2.18
System Overview

0General Information

0UFTnx Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 03	PARAMETER KEYWORD - Must contain 'UFT'.
0		UFTnx parameter cards may optionally be used in any Group 1 EXEC-level program to define one to four lines of user-controlled report footer, which (if present) will be printed at the bottom of each page of the report(s) produced. When UFTnx cards are used, the footers defined will be printed double-spaced below the last line of a full page of report.
0	04 - 04	USER FOOTER LINE NUMBER - Must contain '1' to '4' to specify the first through the fourth line of footer.
0	05 - 05	LEFT OR RIGHT SIDE OF LINE - Must contain an 'A' or a 'B' to indicate that the left ('A') or right ('B') side of the line is being defined by this card. The left side of a line is print positions 1 through 66, and the right side is print position 67 through 132.

0

07 - 72

DATA TO PRINT - Must contain the 66 characters of text which is to be used as the left or right side of the specified footer line.

0General Information

0EXITOP--General-Purpose Output Exit

- All GROUP 1 EXEC-level programs will accept an EXITOP Parameter card which activates an exit routine which receives control just before each output sequential tape or disk record is written (and before an output TAPExx Exit if one is activated). If the program does not write any output sequential files, or if none are activated, the EXITOP card will have no effect. The format of the EXITOP Parameter card is as follows:

-	CARD COLUMNS	FIELD DESCRIPTION
0	01 - 06	PARAMETER KEYWORD - Must contain 'EXITOP'.
0	08 - 15	EXIT ROUTINE NAME - Must contain the name of the Exit Routine which is to be activated. For OS and DOS systems, the Exit routine will be loaded dynamically when the card is encountered. For OS/ the Exit routine must be linked with the EXEC-level program, but will not be activated unless the EXITOP card is present.
0	17 - 17	COBOL INDICATOR (DOS ONLY) - If the exit routine is written in ANS COBOL under DOS, place a 'C' in this column.
0	18 - 80	These columns are reserved for use by the Exit routine, and permit information to be conveyed to it by the user.

- General:

0 The EXITOP routine receives control before the OPEN, WRITE, and CLOSE operations for each output sequential Tape or Disk file. The Exit routine receives the following parameter list using standard linkage:

- 0 1) A single byte indicating the function which is about to be performed as follows:
 - 0 O - Open
 - W - Write
 - C - Close
- 0 2) A two-byte field which contains the File Activator characters for the file involved (by examination of this field, an Exit routine can be activated which operates on only some of the output records which are written by a program).
- 0 3) The record which is about to be written. For the Open and Close calls, this parameter is undefined and should not be referenced
- 0 4) The EXITOP card image.

0General Information

0EXITOP--General-Purpose Output Exit

Skeletal sample EXITOP routine

- Following is a skeletal example of an EXITOP routine, in assembler language.

```
0      EXOP      TITLE 'SAMPLE EXITOP ROUTINE'
      EXITOP    START
                STM   R14,R12,12(R13)
                BALR  R12,0
                USING *,R12
                L     R15,=A(SAVEAREA)      * SAVE
                ST   R13,4(R15)            * AREA
                ST   R15,8(R13)            * LINKAGE
                LR   R13,R15                * ESTABLISH NEW SAVE AREA
                LM   R8,R11,0(R1)          * PARAMETER ADDRESSABILITY:
                *                                     R8 -> O/W/C INDICATOR BYTE
                *                                     R9 -> FILE ACTIVATOR
                *                                     R10 -> RECORD TO BE WRITTEN
                *                                     R11 -> EXITOP CARD IMAGE
                CLI  0(R8),C'C'
                BE   CLOSE
                CLI  0(R8),C'W'
                BE   WRITE
      OPEN    .
                .           OPEN LOGIC
                .
                B    EXIT
      WRITE   .
                .           WRITE LOGIC
                .
                B    EXIT
      CLOSE  .
                .           CLOSE LOGIC
                .
      EXIT   L     R13,4(R13)                * UNLINK SAVE AREAS
                LM   R14,R12,12(R13)        * RESTORE REGISTERS
                SR   R15,R15                * SET RETURN CODE = ZERO
                BR   R14                    * RETURN TO CALLER
                TITLE 'ADDRESSABLE DATA AREAS'
                .
                .           ADDRESSABLE DATA AREAS
                .
                TITLE 'REMOTE (UNADDRESSABLE) AREAS'
      EXITOP1 CSECT
      SAVEAREA DS  90D                      10 LEVELS OF SAVE AREA
                .
                .           ADDITIONAL REMOTE AREAS
                .
```

```
TITLE 'LITERAL POOL'  
END   EXITOP
```

0General Information

0EXITOP--General-Purpose Output Exit

EXITOP Exit Routines - Additional Notes

- 1. For OS and DOS, the EXITOP exit routine is always dynamically loaded at execution time. For Univac OS/3, the exit routine must be linked with the primary program (see below).
- 0 2. Under DOS, if the Exit Routine is written in ANS COBOL, the following must be observed in order for the exit routine to "know" that it is a called routine:
 - 0 A. Create an assembler program named MNS0ADDR as follows:
0 MNS0ADDR START
DC V(ILBDMNS0)
END
 - 0 B. Link the exit routine with this 'program', as follows:
0 INCLUDE MNS0ADDR
INCLUDE exit
ENTRY exit
 - 0 C. Now the eight-byte 'program' MNS0ADDR will immediately precede the entry-point of the exit, providing its caller with the address of the flag byte to be set when an ANS COBOL exit is to be called.
 - 0 D. Place a 'C' in column 17 of the EXITOP card to cause the COBOL logic to be invoked.
- 0 3. For Univac OS/3 systems, the exit routine must be linked statical with the main program. Thereafter, it will be ignored unless an EXITOP card is present.

1

0General Information

0Alternate names for programs

- Some installations may wish to assign different names to the programs of the system, to conform to local standards, or for testing a new release, etc. While this presents no problem in the case of the EXEC-level programs, some systems also include subroutines which perform I/O functions in specialized areas. The names of these routines may also be changed, but the EXEC-level programs which invoke them must be informed of the name under which each routine is to be found. The IOMODS parameter card is provided for this purpose. All GROUP 1 EXEC-level programs will accept an IOMODS card, but it will have no effect unless the I/O subroutine is to be invoked.

0 The format of the IOMODS card is as follows:

-IOMODS Parameter Card

0	CARD COLUMNS	FIELD DESCRIPTION
0	01 - 06	PARAMETER KEYWORD - Must contain 'IOMODS'.
0	08 - 15	VSAM I/O ROUTINE - May contain the name

assigned to the subroutine which is delivered
as 'VSIO', and which performs direct-access
on VSAM Master and Work files.

0General Information

0Alternate names for programs

-CMMODS Parameter Card

0	CARD COLUMNS	FIELD DESCRIPTION
0	01 - 06	PARAMETER KEYWORD - Must contain 'CMMODS'
0		The CMMODS parameter card permits you to store the subroutines provided with the Consumer Merge/Purge system under different names, and still have the EXEC-level programs "know" the correct name for each function. The CMMODS card will be accepted by any EXEC-level Consumer Merge/Purge program, but is required only when overriding subroutine names.
0	08 - 15	STREET ADDRESS ANALYSIS ROUTINE NAME - May contain an alternate name to be used for the Street Address Analysis routine which is delivered as 'AMSTP'. If this field is omitted, or if the CMMODS card is omitted the default name of AMSTP will be used.
0		Note: If the addresses on your input file are primarily Canadian addresses the CANSTRS address parsing routine should be used.
0	17 - 24	ADDRESS LINE LOCATION ROUTINE - May contain an alternate name to be used for the Address Line Location routine which is delivered as 'ISAP'. If this field is omitted, or if the CMMODS card is omitted, then the default name of ISAP will be used.
0	26 - 33	NAME PARSING ROUTINE - May contain an alternate name to be used for the Name Parsing routine which is delivered as 'ISFN'. If this field is omitted, or if the CMMODS card is omitted, then the default name of ISFN will be used.

1Group 1 Software -- Consumer Merge/Purge 2.25
System Overview

0General Information

0Alternate names for programs

-CMMODS Parameter Card

- 35 - 42 STREET SUFFIX ROUTINE - May contain an alternate name to be used for the Street Suffix Decoder routine which is delivered as 'USACODR'. If this field is omitted, or if the CMMODS card is omitted then the default name of USACODR will be used.

0 Note: If the addresses on your input file are primarily Canadian, the CANCODR decoder routine should be used. If CANSTRS is specified as the the Street Analysis Routine, CANCODR must be used.

0 44 CANADIAN INDICATOR - Must contain an 'X' if both the Street Analysis Routine and Street Suffix Decoder must process Canadian addresses.

1Group 1 Software -- Consumer Merge/Purge 2.26
System Overview

0General Information

0Alternate names for programs

+

-

0 If either a TAPEIP or DISKIP parameter card is encountered in the parameter stream (SYS004), then an additional file, SYS002, will be opened and processed as an extension of the parameter stream.

0 The record size of this file must be 80 bytes.

0 This SYS002 auxiliary file will accommodate parameters shared by multiple programs, if needed. This file can also serve other purposes as required by your installation.

0General Information

0Region/Partition Sizes

- Following is a rough guide to the amount of memory required by each program in the Consumer Merge/Purge System. For OS systems, buffer requirements should be added depending upon the files used, their block sizes, and the (specified or default) number of buffers for each. For DOS and OS/3 systems, buffer size is included, except that under DOS if FBA devices are used, memory must be added for the Control Interval buffers.

0 Program CM30: 900K
CM31: 250K
DE40: 150K
DE41: 150K
DE50: 350K
DE60: 250K
DE61: 850K

0 For some DOS users, depending upon the operating system version and system generation options, a SIZE parameter may need to be entered on the EXEC card to restrict the amount of memory allocated to the primary program. In these cases, the following table may be used to determine the SIZE value:

0 Program CM30: 600K
0 Remaining programs: none required

0General Information

0Tailoring Memory Requirements

- Each of the Duplicate Elimination programs has a statically defined memory requirement. As delivered, the programs can accommodate most 'normal' dupe-elim jobs. However, some users will find that the default capacities of one or more elements are not sufficient for some jobs. Following is a list of the capacity elements which may need to be changed:

0	Program(s)	Statement Label	Description
+			
0	CM30	MAXSAVIN	Controls the amount of memory dedicated to tabling the records for comparison
0	CM30	MAXCHANG	Controls the maximum number of CHANGE parameter cards which can be accommodated
0	CM30	MAXPOST	Controls the maximum number of POSTxx parameter cards which can be accommodated
0	CM30 CM31 DE40 DE41 DE50 DE60 DE61	BUFFERSZ	(DOS and OS/3 only) Controls the amount of memory reserved for the buffers for the input and output files.
0	CM30 DE40 DE41 DE50	NRKEYS	Controls the number of different Key Codes (File Codes, List Codes, etc.) which can be accommodated
0	CM30 DE50	MAXRECSZ	Controls the maximum-sized record which can be processed
0	CM30 DE50 DE61	MAXDUPAT	Controls the maximum number of records which may be present in a single duplicate group
0	CM31 DE40 DE41 DE50 DE60	MAXFLCSR	Controls the maximum number of value operands which can be entered in all FLCSEL or FLCREJ cards combined
0	DE40	NRBRKR	Controls the maximum number of BRNAME parameter cards which can be accommodated

1Group 1 Software -- Consumer Merge/Purge
System Overview

2.30

0General Information

0Tailoring Memory Requirements

0	Program(s)	Statement Label	Description
+	DE50	MAXDUPGR	Controls the maximum number of key codes which can be present in a single group of duplicates
0	DE61	MAXTABLE	Controls the number of records which can be tabled for the purpose of duplicate-group comparison

0General Information

0Relocation for Subroutines (old DOS operating systems)

- Old DOS operating systems do not provide relocation information in the core-image library copy of programs, so that when these programs are dynamically loaded (at a random address), their internal address constants do not contain the correct values, which will prevent the system from operating correctly.

The following procedure may be used to overcome this problem.

- 0 1) When the subroutines (STRS, ISAP, ISFN, and any exit routines which are to be dynamically loaded) are assembled, punch out an object deck.
- 0 2) Linkedit the object decks producing core-image modules and retain the printed listing. A following step will need to know the address at which the program is linked.
- 0 3) Create a tape or disk file which consists of the object decks for the subroutines, each preceded by a PROGRAM card (see below). The order of the programs in this file is not significant. (Actually, only the RLD cards from the object decks will be used, and if you wish to reduce the size of the file, you may eliminate the other cards. RLD cards have the characters 'RLD' in columns 2-4.) The PROGRAM cards are used to identify the object deck (or RLD cards) which follow, and must be punched as follows:
 - 0 Cols. 1-8: Must contain 'PROGRAM '
 - 9-16: Must contain the name of the subroutine, left-justified and padded on the right with blanks. If an alternate name is used for the subroutine, the alternate name must be entered.
 - 17-27: Must contain ' LINKED AT '
 - 28-33: Must contain the address at which the subroutine is linked, from the linkedit listing from step 2. This address must be entered as a six-digit hexadecimal value, with leading zeroes as required.
 - 34-80: These columns are ignored.The file created must have a record size of 80, and may be blocked. A relatively small block size is recommended, since this file will take up buffer space.
- 0 4) When executing any program, assign the file created in step 3 to SYS009, and provide a TAPEI9 card in the parameter list. The programs will scan the TAPEI9 file for the relocation information necessary for proper execution of the subroutines which they require.

0General Information

0Static Linkedit (OS/3 only)

- Under the Univac OS/3 Operating System, all main programs (in this system) must be statically linkedited with any required exit routines and subroutines. The remarks throughout the remainder of this manual concerning "dynamic loading" do not apply under OS/3. Following is a table of the required Link editing.

0 Program CM30: Link together with

STRS

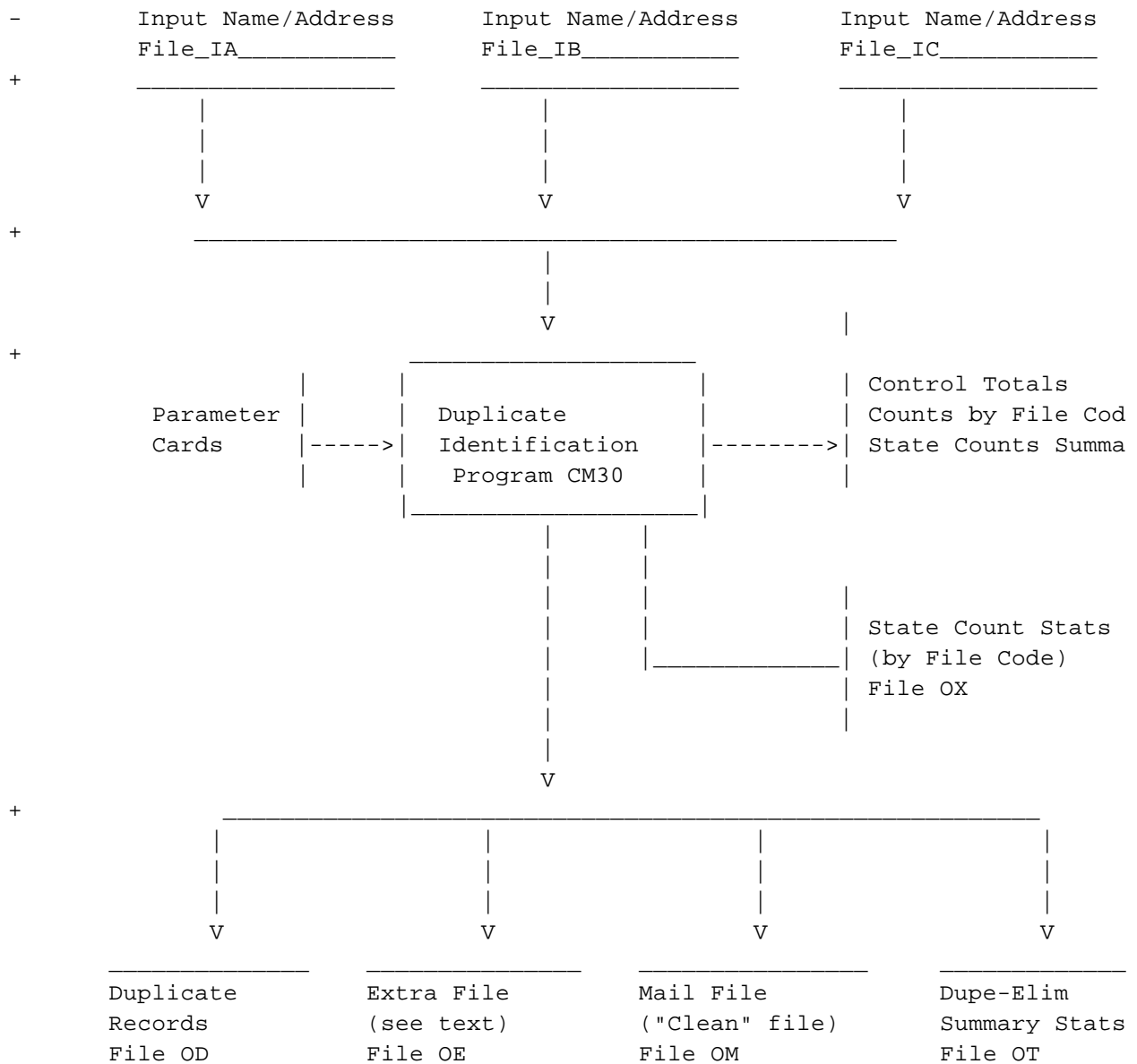
ISAP

ISFN

Any user exit routines required

0 The remaining programs require no composite link-edit.

- Narrative_and_Flow:
 +



- Name-and-Address records are read from one, two, or three input files. If more than one input file (that is, device) is specified, then the records are merged according to the sequence defined by the MERGES parameter card. The ZIP Code, Address, and Name are extracted and assembled in a fixed-field array at the front of each record as it is placed in memory. Records which share the same first n characters of the MERGES string are loaded in as groups ("memory loads") for the purposes of comparison, to the extent permitted by the size of the memory load area and the record size. For each memory load, the records are then compared to one another. The tightness or looseness of the match required, as well as

specific additions or deletions to the comparison process ("options") may be specified by the use of ALGSEL cards, if desired.

0Narrative and Flow

+ _____

- The comparisons may result in various records being identified as duplicates of one another. All such records are written to the output Duplicate Names file, File OD. Each group of duplicates will have one record selected (according to priority) as the survivor, unless one of the records involved is from a Purge file (in which case, there will be no survivor from that group). The surviving records from the duplicate groups, as well as any input (and non-Purge File Code) records which were unmatched, will be written to the Mail and Extra files (Files OM and OE) in their original sequence. Several CONTRL-card options permit you to split these "clean" records in various ways between the two files, but by default the files will contain exactly the same records.

0 You may also specify (again on the CONTRL card) that no records which are involved in duplication are to be written to either clean file, or that no "Multi-Buyers" are to be written to either clean file (a "Multi-Buyer" is the survivor of a duplicate group which contained records from more than one File Code).

0 At the end of all input files, various reports are produced: a page of control totals, an input/output summary by File Code, an input/output summary by state, and a Missing Element Analysis. Additional reports are printed which reflect the actions of CHANGE and/or POST cards, if any were provided.

0 Optionally, a Summary Statistics file (File OT) may be produced, to drive the subsequent report programs.

0 Another optional output file, File OX, may be produced if it is necessary to have counts printed by State for each File Code, or vice versa. (Note that an overall State Counts report is automatically produced by this program--the optional file is for a State-count breakdown for each File Code separately). After execution of program CM30, this file may then be sorted to either State/File Code sequence or File Code/State sequence, and used as input to program DE41 to produce the required report.

- Household_Duplicate-Elimination:

0 In the absence of any ALGSEL cards, the program will do a basic "Household" dupe-elimination of "medium" tightness. A "Household" duplicate-elimination is one for which members of the same family, at the same address, are considered as duplicates of one another. No ALGSEL cards are needed to tell the program to perform a household dupe-elimination (the ALGSEL HHLDD card is to tighten, loosen, or eliminate logic which is used in some circumstances to restrict matches to members of the same household, as best as

0Household Duplicate-Elimination

+

- that can be determined by the names; see the discussion under HHLDD
Algorithm of the ALGSEL card).

0 If any records which have blank names, or names which are
'RESIDENT' or 'OCCUPANT', are encountered, these records will by
default be compared as "RESIDENT" records--that is, the comparison
will be based solely on the address, and will be pretty tight. If
these names (blank, RESIDENT, or OCCUPANT) are to be treated as
"names", the N option of the ALGSEL SURN card must be specified.

- Individual_Duplicate-Elimination:

0 If an "Individual" duplicate-elimination is required (an "Individual"
duplicate elimination is one in which different people, even within
the same family, are not duplicates), then an ALGSEL INDV parameter
which specifies T, M, or L (tight, medium, or loose) must be
provided; its presence tells the program to perform an individual
dupe-elim. The M setting (medium) is recommended unless you have
a specific reason to use T or L.

- All-Resident_Duplicate_Elimination:

0 Sometimes circumstances arise under which it is necessary to do a
duplicate-elimination based solely on address. These are usually
called "Resident" or "All-Resident" duplicate eliminations. This
function may be achieved simply by supplying an ALGSEL SURN card
which specifies option X. The tightness or looseness of the
resident (address) comparison can be adjusted by means of the
ALGSEL RESI card, which is set by default at "medium".

- Tightening_the_Comparison:

0 As noted above, the program by default performs a "medium"
dupe-elimination on a Household basis; for RESIDENT names, the
(resident) comparison is also "medium". In various circumstances,
it may be necessary to "tighten" the record comparison, to
restrict the dupe-identification to only the "best" duplicates.
This "tightening" of the comparisons can be achieved by providing
ALGSEL parameter cards which tighten various comparison
algorithms, or which specify various options, or both. A careful
study of the ALGSEL card details later in this section will be
necessary to achieve the optimum control, probably in conjunction

0Tightening the Comparison

+_____

- with various experimental runs, but the following is provided as
a quick overview of the tightening mechanism.

0 The maximum tightness can be achieved with the following ALGSEL
cards:

- 0 ALGSEL ZIPC E (if necessary--this is most frequently
achieved with the break logic)
- ALGSEL SURN E
- ALGSEL INDV T M (only if doing an Individual dupe-elim)
- ALGSEL HSNR E
- ALGSEL STNM E D
- ALGSEL POBX E P
- ALGSEL RRBX E
- ALGSEL APNR E
- ALGSEL ADDR T
- ALGSEL HHL D T
- ALGSEL TOTL T
- ALGSEL RESI E (for RESIDENT comparisons)

0 Note that the above cards specify that all individual elements must
be equal. This permits the program very little latitude for
judgment; in fact, the only areas in which the program could use
its "judgment" are in the identification of individuals, if the
ALGSEL INDV card is provided, and the identification of households
(the program attempts to confirm, from the names, that two records
are in the same household when there is no "specific" address
element like House number which is equal; in the example above, for
instance, this judgment would be made if the two addresses were
'RR 1' without a box number).

0 A more "reasonable" approach to tightness (that is, one which allows
mild variations in the individual elements) could be achieved with
the following:

- 0 ALGSEL ZIPC E (if necessary)
- ALGSEL SURN T I (if Option I is necessary after the
sort/break specification)
- ALGSEL INDV T M (if doing an Individual dupe-elim)
- ALGSEL HSNR T L
- ALGSEL STNM T 4D
- ALGSEL POBX T P
- ALGSEL RRBX T
- ALGSEL APNR T A
- ALGSEL ADDR T
- ALGSEL HHL D T
- ALGSEL TOTL T
- ALGSEL RESI T 4 (for RESIDENT comparisons)

0Tightening the Comparison

- A somewhat looser approach to tightness would be provided by the following cards:

0 ALGSEL ADDR T
ALGSEL INDV T M (for an Individual dupe-elim)
ALGSEL HHL D T
ALGSEL TOTL T
ALGSEL RESI T (for RESIDENT comparisons)

0 This would allow individual elements to vary up to "medium", but impose "tight" limitations on the overall address, the household determination where appropriate, and the overall name and address combination.

- Loosening_the_Comparison:

0 Sometimes it is necessary to "loosen" the comparison to achieve what direct mailers refer to as "overkill" (the identification of records as duplicates which are not clearly dupes). The maximum looseness can be achieved as follows:

0 ALGSEL ZIPC X N
ALGSEL SURN L
ALGSEL INDV L G (for an Individual dupe-elim)
ALGSEL HSNR L
ALGSEL STNM L
ALGSEL POBX L
ALGSEL RRBX L
ALGSEL APNR X
ALGSEL ADDR L U
ALGSEL HHL D X
ALGSEL TOTL L
ALGSEL RESI L A (for RESIDENT comparisons)

0 A job run with the above cards would almost certainly be too loose. More reasonable approaches would be either to loosen the individual comparisons more, while retaining "medium" for the combination checks, or to loosen the combination checks (permitting more simultaneous "medium" mismatches in the individual elements). The following cards illustrate the loosening of the individual elements:

0 ALGSEL ZIPC X (with or without option N)
ALGSEL SURN L
ALGSEL INDV L G (for an Individual dupe-elim)
ALGSEL HSNR L
ALGSEL STNM L
ALGSEL POBX L
ALGSEL RRBX L
ALGSEL APNR X (or L, for a little more discipline)
ALGSEL HHL D X (or L, for a little more discipline)
ALGSEL RESI L (for RESIDENT comparisons)

1Group 1 Software -- Consumer Merge/Purge
Program CM30

3.6

0Loosening the Comparison

+ _____

- The following cards illustrate loosening the composite checks while retaining the "medium" default for the individual elements:

0 ALGSEL ADDR L U
ALGSEL INDV L (for an Individual dupe-elim)
ALGSEL TOTL L

0 Of course, these approaches may be combined to the extent desired, in combination with the options. It is perfectly rational to loosen address-element comparisons at the same time as you tighten the Name or Household comparison, and so forth.

1Group 1 Software -- Consumer Merge/Purge
Program CM30

3.7

0HEADER Parameter

-	COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'HEADER'.
0	08 - 17	DATE - May contain a date in the form MM/DD/YYYY. This date will appear on the output reports. If this field is left blank, the current date will be inserted.
0	19 - 58	RUN DESCRIPTION - May contain any run description the user wishes. This run description will appear on the output reports.
0	60 - 65	START POSTAL CODE - May contain a Postal Code, left-justified, at which the user wishes to start processing the input data. All records with a Postal Code lower than the Postal Code in this field will not be processed.
0	67 - 72	STOP POSTAL CODE - May contain a Postal Code, left-justified, after which the user wishes to stop processing. When a Postal Code of higher value than that which is entered here is encountered, the program will act as if it has reached end of file.
0		Note: The START POSTAL CODE and the STOP POSTAL CODE fields may be both left blank. If they are both blank, all input records will be processed.
0		Column 66 must be either blank or a dash ('-'), to ensure that processing will not continue when the STOP POSTAL CODE is inadvertently entered one column to the left, in columns 66-71 instead of 67-72.

0TAPExx Parameter Card

- TAPExx or DISKxx cards must be used to activate each input or output
file. Following is a brief description of each of the possible files
See the System Overview section for a detailed description of the
TAPExx/DISKxx card fields.

- File_Activators:

+

- 0 IA - Input Name and Address file
0 IB - Input Name and Address file
0 IC - Input Name and Address file
0 OD - Output Duplicate Names file
0 OE - Output Extra File
0 OM - Output Mail File (Clean file)
0 OT - Output Summary file (for subsequent reporting)
0 OX - Output State Count Statistics by File Code
(for subsequent reporting)

0 Notes: The presence of the TAPExx/DISKxx card triggers processing
or creation of the corresponding file.

0 At least one of the files IA, IB, and IC is required.

0 At least one of the files OD, OE, and OM is required.

0 File OT may be omitted if the reports produced by
program DE40 and DE50 are not desired.

0 File OX may be omitted if the reports produced by
program DE41 are not desired.

1

OMERGES Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'MERGES'.
0		The MERGES card is required to define the sequence of the input file(s), by specifying the fields which make up the sequencing key.
0	08 - 10 15 - 17 22 - 24 29 - 31 36 - 38 43 - 45 50 - 52 57 - 59 64 - 66 71 - 73	SEQUENCE FIELD SEGMENT LOCATIONS - Must contain the starting positions of the fields that define the sequence of the input name and address files (TAPEIA, TAPEIB, and TAPEIC). If necessary, these fields may be extended to the left one position, allowing the entry of a 4-digit number in columns 7-10, 14-17, etc.
0	11 - 11 18 - 18 25 - 25 32 - 32 39 - 39 46 - 46 53 - 53 60 - 60 67 - 67 74 - 74	PACKED SEQUENCE FIELD - Must be blank, or contain 'P' to indicate that this sequence field segment is in packed form, and should be unpacked prior to use. If entered as 'P', then the field will occupy (2n-1) characters in the total merge control field, where n is the length specified.
0	12 - 13 19 - 20 26 - 27 33 - 34 40 - 41 47 - 48 54 - 55 61 - 62 68 - 69 75 - 76	SEQUENCE FIELD SEGMENT LENGTHS - Must contain the length of each field defined in the previous parameter card field. The total of all field lengths must not exceed 99.

0SEQERR Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'SEQERR'. The SEQERR card may optionally be supplied to specify the treatment of sequence errors in the input file. If omitted, the default treatment will be that described under 'E' below.
0	08 - 08	SEQUENCE ERROR TREATMENT - Must contain one of the following values to specify the desired treatment of an out-of-sequence condition on any of the defined input name-and-address files: B - Bypass the offending record, and continue sequence checking C - Continue processing of the offending record, and continue sequence checking E - Bypass the offending record, and do not read any more records from any input file. That is, bring the execution to a graceful close with what has been read so far. I - Continue processing of the offending record, and abandon any further sequence checking of the file which had the sequence error (that is, if the first sequence error occurs on file IA, sequence checking will be abandoned on file IA but will continue on files IB and IC) O - Ask the operator for one of the above treatment options, and honor it.
0		Note: If the SEQERR card is omitted, the default treatment option is E. This is recommended because a sequence error is almost certain to indicate either an error in the sortation, or an error in the definition of the sequence control field.
0		In all cases, the sequence error will be displayed on the console, and the occurrence of a sequence error will be noted on the print output (SYS005). If option E is selected, via SEQERR card or by the operator, a note will be displayed on the console and the print output that processing was terminated by a sequence error condition.

0BRKFLD Parameter Card

```
-      CARD COLUMNS      FIELD DESCRIPTION
-      01 - 06            PARAMETER KEYWORD - Must contain 'BRKFLD'.
0                                     The BRKFLD parameter card may optionally be
                                     provided to specify that some segments of the
                                     MERGES string contribute to the break length
                                     only if they are not blank.
0                                     See the MERGES card and CONTRL card column 8
                                     and the associated discussions.
0      08 - 10            SEGMENT DEFINITIONS - These fields define up to
      11 - 13            twenty segments (of the MERGES string). Each
      14 - 16            3-character segment definition contains a two
      .                  position length followed by one of the following
      .                  codes:
      .
0      65 - 67            blank - to indicate that this segment is to
                                     contribute to the break length
                                     whether it is blanks or not
                                     X - to indicate that this segment is not
                                     to contribute to the break length if
                                     it is blank
0                                     Columns 8-10 may not be blank. Thereafter, the
                                     first blank segment definition encountered will
                                     terminate processing of the card. The sum of the
                                     segment lengths may not exceed 99.
0      Example: Suppose that the MERGES string consists of a three-digit
      ZIP Code, a 30-byte company name, and a 13-byte surname; and
      that the break length is four. Now in addition, suppose that
      when the file is examined, it turns out that three-fourths of t
      records contain no company name. Without a BRKFLD parameter
      card, those records with a blank company name would actually be
      compared in memory-loads effectively "broken" only on three-dig
      ZIP Code. The net result would be that some duplicates would b
      missed (due to memory load overflows) and the number of
      comparisons performed would be larger than necessary. The
      following BRKFLD parameter card would alleviate these problems:
0      BRKFLD 03 30X13
0      With this card, the break comparison is effectively: the first
      three bytes of the ZIP Code and the first byte of the company
      name when there is a company name; and the first three bytes of
      the ZIP Code and the first byte of the surname when the company
      name is blank.
```

0RECLIM Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'RECLIM'. The RECLIM card may optionally be used to limit the records processed for a particular run. A number of records to be skipped may be specified, as well as a number of records to process.
0		The most frequent use of the RECLIM card is to restrict the file to a small number of records, for parameter checkout or calibration. It can also be used, of course, to break what would otherwise be an excessively large job into several smaller independent executions without dividing the input files.
0	08 - 16	RECORDS TO SKIP - May contain the number of records to skip before processing begins. If omitted, processing will begin with the first record.
0		Note: This option will only function as indicated when a single input file is used.
0	18 - 26	RECORDS TO PROCESS - May contain the number of records to be processed. If omitted, all records will be processed (once the specified number are skipped).

1

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'CONTRL'.
0		The CONTRL parameter card is required to provide control information for the duplicate elimination.
0	08 - 08	MERGES STRING BREAK LENGTH - In preparation for the comparison of records for the purpose of duplicate identification, name-and-address records are loaded into memory until this loading process is stopped by one of two conditions:
0		1) there is no room for another record, or 2) the program detects a change in the first n characters of the string defined by the MERGES card
0		This field must contain the number n needed in check 2) above. When choosing a number, the following should be kept in mind:
0		Two records which differ within the first n characters of the MERGES string will never be found to be duplicates. If n is too large (for the file being processed), then many duplicate pairs which could be found (if a smaller n were chosen) will not be found. then all (or most) memory loads will be stopped by running out of room, and two records which are identical will sometimes fall on opposite sides of the division enforced by the memory capacity. When this happens, the number of "overflows", which is displayed on the Control Total page, will be large.
0		If n is too small (again, for the file being processed), then there will be "too many" record in the memory at one time, and the comparison time required will be too large (or at least, larger than it need be). The total comparison time for a memory load will vary with the square of the number of records in the memory load, because each record is compared with every other record (in the same load).
0		In general, a "controlled" termination of a memory load, such as is provided by the n character comparison, is more desirable than the "uncontrolled" termination which results

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-		from a memory load overflow. Ideally, n should be chosen small enough that all potential duplicates are always in the same memory load, and at the same time large enough that there are no memory load overflows.
0		Note: If necessary, this field may be extended one position to the left, allowing the entry of a two-digit number (up to 99) in columns 7-8.
0	10 - 12	USEFUL RECORD LENGTH - Must contain the length of that portion of the records which is needed to properly form the required output files. Normally this will be the same as the input record length; however, if the input records are longer than the output records need to be, then entering a smaller number here permits the program to shorten the records as they are loaded into the memory, thus permitting more to be loaded in as a single memory load.
0		If necessary, this field may be extended to the left one position, allowing the entry of a 4-digit number in columns 9-12.
0	14 - 16	SAVE AREA RECORD LOAD LIMIT - Normally, records are loaded into the program's "Memory Load" area until it is full, or until a change in the first n characters of the MERGES string is detected, as noted above. Also as noted above, the more record in each memory load, the longer the comparison will take; however, finding all the possible duplicates requires the largest possible memory loads.
0		This field permits you to enter a maximum number of records for each memory load. If you have not changed the size of the area (at assembly time), then you will not be likely to ever need to enter a value here. However, to accommodate a large (or very dense) job where every duplicate must be identified, you may find it necessary to drastically increase the default setting. When you do so, it may adversely affect the execution time of other jobs for which the identification of every duplicate is not necessary. Should this prove to

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-		be the case, you can have the effect of the origin area, or a smaller one still, by entering the maximum number of records to be loaded at once here.
0		In any case, any number entered here may not exceed the actual memory capacity (in records) of the program, considering the record size entered in columns 10-12. This capacity can be computed as follows (round the division down):
0		capacity = MAXSAVIN/(((rrr + 208 + 3) / 4) * 4)
0		where MAXSAVIN represents the size of the SAVIN area, controlled by the (program-internal) EQU statement with the label MAXSAVIN; rrr is the record length defined in columns 10-12; and the other numbers are constants determined by the internal design of the program.
0		As the program is delivered, the value of MAXSAVIN is set to 100,000.
0		If necessary, this field may be extended to the left one position, allowing the entry of a 4-digit number in columns 13-16.
0	18 - 18	RUN TYPE - Must contain one of the following codes to define the type of run:
0		blank - The input name-and-address file is in its original form, and a normal duplicate elimination process is to be performed.
		X - The input name-and-address file is in its original form; no duplicate elimination is to be performed; instead, the records are to be extended by the extracted information, stored in memory load format, and written to File OM (and also to OE if it is present). In this case, 208 bytes are added to the front of each record (plus 4 if the

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-		output file is variable-length); this area contains the extracted information in the format noted in the File Layouts section.
		E - The input name-and-address file is in extracted format; and a duplicate elimination process is to be performed. In this case, all parameter cards refer to positions in the original record, except for the MERGES card (because you may wish to sort on one or more of the extracted fields). The output records will have the extracted-information area stripped off.
0		Note: If the 'X' option is selected, then the cards used to control the duplicate elimination process and its output (DUPCTL, ALGSEL, etc.) will be ignored if present.
0		If the 'E' option is selected, then the cards used to control the extraction of data from the records (FLCODE, NAMEDF, etc.) will be ignored if present.
0		In the normal case (blank), all cards have their normal meaning.
0		If the 'X' option is selected, then FLCnnn cards are not required. If none are present, then statistics will be accumulate for each File Code actually encountered. If FLCnnn cards are present, then statistics will be accumulated for those File Codes which are defined, and any unrecognized File Codes will be treated as specified by CONTRL card column 28.

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	20 - 20	DUPLICATE RECORD TREATMENT- Must contain one of the following codes to define how duplicate records are to be treated:
0		N - Will leave one record from each duplicate group on the Mail File.
0		E - Will eliminate all records involved in duplication from the Mail File.
0		M - Will eliminate all multi-buyer records from the Mail File.
0		Note: Any culling of the normal output file which is accomplished through this column affects both the OM and OE files, and will be reflected in the output reports.
0	22 - 22	DROP OPTIONS FOR THE MAIL FILE - Must contain one of the following values to specify which (if any) records are to be excluded from the Mail File (File OM). By means of this column and the following one, the total output (reflected in the reports) may be split between the OM and OE files. The possible values are:
0		blank - Do not exclude any records from File OM
		M - Exclude Multi-Buyer records from File OM
		N - Exclude from file OM any records that are not Multi-Buyer records
		S - Exclude Single-Buyer records from File O
		I - Exclude from file OM any records that are not Single-Buyer records
		D - Exclude from File OM all records involve in any duplication
		U - Exclude from File OM any records not involved in any duplication (Uniques)

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	24 - 24	DROP OPTIONS FOR THE EXTRA FILE - Must contain one of the following values to specify which (if any) records are to be excluded from the Extra File (File OE). By means of this column and the preceding one, the total output (reflected in the reports) may be split between the OE and OM files. The possible values are: blank - Do not exclude any records from File OE M - Exclude Multi-Buyer records from File OE N - Exclude from file OE any records that are not Multi-Buyer records S - Exclude Single-Buyer records from File O I - Exclude from file OE any records that are not Single-Buyer records D - Exclude from File OE all records involve in any duplication U - Exclude from File OE any records not involved in any duplication (Uniques) X - Do not exclude any records from File OE. Instead, write all surviving records, in the Extracted File format (so they ma be re-used in a subsequent merge/purge without re-running the Extract). This option is effective regardless of the value specified in column 22, and regardless of whether File OM is also present.
0		
0	26 - 26	FILE CODE SEQUENCE ON REPORT - Must contain one of the following codes to specify the sequence in which the totals by file code will be presented blank - List the File Codes in the order in whic the FLCnnn parameter cards were presente A - List the File Codes in alphabetical orde P - List by the relative priority (as determined by the FLCnnn cards and the PRIORT card if one was provided), major; and File Code, minor. D - List by File Code Description, major; relative priority, intermediate; and File Code, minor.
0		
0	28 - 28	NON-MATCH FILE CODE TREATMENT - Must contain one of the following values to specify the desired treatment of an input record with a File Code

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-		which does not match any of the FLCnnn parameter cards:
0		blank - Continue processing, treating all unknown File Codes as a single class, having a priority lower than that of any defined File Code
		I - Bypass the records whose File Codes are not recognized
		X - Terminate processing as soon as an unknown File Code is detected, producing an appropriate error message.
0		For an Extract run (column 18 is 'X'), and provide that no FLCnnn cards are present, then this action will be taken if the number of File Codes actually encountered exceeds 200 (or the value specified by the NRKEYS variable).
0	30 - 32	LOCATION OF FILE CODE IN OUTPUT - May contain the starting position of the File Code desired in the output record.
0		If this field is specified, then the File Code will be stored in the specified location of all records written to Files OM, OE, and OD. If it is omitted, then any Multi-Buyer File Code specified in the MULTIB card will not be placed in any output records.
0		The File Code placed on the File OD records will be that associated with the input record (after any CHANGE cards have been applied); the one placed in records written to File OM or file OE will be that associated with the output record; that is, it will reflect the Multi-Buyer code for multi-buyer records, if one is specified via the MULTIB parameter card.
0		If necessary, this field may be extended to the left one position, allowing the entry of a 4-digit location in columns 29-32.
0		Note: An output File Code position should be specified whenever any of the following is true:
0		MULTIB card is present
		CHANGE card(s) are present
		The FLCODE card does not specify option F or C

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	34 - 34	OUTPUT FILE CODE LENGTH - If columns 30-32 are not blank, then this column must contain the length of the output File Code.
0	36 - 36	PRINT INACTIVE FILE CODES - Must contain one of the following codes to control the printing of inactive File Codes (those with no input or output) on the File Code Input/Output Control Total report: blank - Do not print inactive File Codes. X - Print inactive File Codes.

0FLCODE Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'FLCODE'. The FLCODE card is required to specify the location length and format of the File Code on the name and address records.
0	08 - 08	FILE CODE FORMAT - Must contain one of the following codes to indicate the form of the File Code in the input records: C - The File Code is a character-form string of from one to nine characters P - The File Code is a packed number of from one to five bytes. Regardless of the length specified, the number will be treated as a positive number, and will be left-justified in a nine-character field with trailing blanks. Therefore, the FLC Parameter cards (see below) must specify the File Codes in this fashion also. B - The File Code is a binary number of length one or two. This number will be treated as unsigned, and the resultant value will be left-justified in a nine-character field with trailing spaces. Therefore, the File Code Parameter cards (see below) must specify the File Codes in this fashion also. F - The File Code does not exist in the record but is to be derived from the input file on which each record occurs, and stored in the record position specified in columns 10-14. The value stored will be 'A', 'B', or 'C' depending upon whether the record was read from File IA, IB, or IC. Note: IF an option other than 'F' or 'C' is specified here, then an output File Code location should be specified in the CONTRL card. Subsequent programs in the system locate the File Code only as a simple character string.
0	10 - 12	FILE CODE LOCATION - Must contain the location of the File Code in the input name-and-address records. Note: If necessary, this field may be extended one position to the left, allowing the entry of a four-digit position in columns 9-12.

1Group 1 Software -- Consumer Merge/Purge 3.28
Program CM30

0FLCODE Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	14 - 14	FILE CODE LENGTH - Must contain the length of the File Code in the input name-and-address records. The maximum value permitted depends on column 8, as follows:
0		C - maximum length is 9
		P - maximum length is 5
		B - maximum length is 2
		F - maximum length is 1

1Group 1 Software -- Consumer Merge/Purge 3.29
Program CM30

0ZPCODE Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'ZPCODE'. The ZPCODE card is required to define the location and length of the field containing the ZIP Code or Postal Code.
0	08 - 08	ZIP CODE LOCATION ALGORITHM - Must contain one of the following codes to direct the program in extraction of the ZIP Code (or Canadian Postal Code):
0		M - The ZIP Code may be moved from a fixed location
		P - The ZIP Code is in a fixed location in packed decimal format. The assumed length of the packed field is three, and a sign digit of hex 'F' will be used regardless of the actual sign digit.
		F - The ZIP Code is floating within the record. The location specified in columns 10-12 is the rightmost location in which the ZIP Code or Postal Code may end. The program will search leftward from that location and take the first non-blank field encountered as the ZIP Code or Postal Code.
0	10 - 12	ZIP CODE LOCATION - Must contain the location of the ZIP Code or Postal Code in the input name-and-address records.
0		Note: If necessary, this field may be extended one position to the left, allowing the entry of of a four-digit location in columns 9-12.
0	14 - 15	ZIP CODE LENGTH - If column 8 is 'M', then this field must contain the length of the fixed ZIP Code. The value entered must be one of the following:
0		Possible lengths: 3 through 7, 9, and 10.

0

Note: The ZIP Code extracted by means of this card will be used to apply the Start/Stop ZIP Code option of the HEADER card as the basis for the State Counts, and as the basis for any ZIP Code comparisons.

0ADDRDF Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'ADDRDF'. The ADDRDF card is required to specify the location and lengths of the fields which comprise the address in the input records.
0	08 - 08	ADDRESS LINE LOCATION METHOD - Must contain one of the following codes to indicate the method to be used to locate the address line: 'F' - The address line is to be formatted from two or more separated segments; the location/length fields starting in column 10 define up to six fields which are to be strung together in the order defined to produce the single line of address data. 'I' - The address line is to be found in two or more lines, defined in the order in which they would be printed on a mailing label, and the last of which is to be ignored (typically because it is a City line). 'L' - The address line is to be found in two or more lines, defined in the order in which they would be printed on a mailing label, all of which are address lines. This form should be used when there are floating address lines as well as a separate fixed City location in the records. 'M' - There is a single address line in a fixed location which should be used as is. 'Z' - The address line is to be found in two or more lines, defined in the order in which they would be printed on a mailing label, and which also contain City/State/ZIP information on one or more lines. When this option is selected, the program examines the lines in the reverse order, looking for and eliminating a ZIP Code or Postal Code, a State or Province, and a City. Whatever is left is then treated as in option L above.

0 Note: For option M, one location/length field is required. For the remaining options, two to six fields must be provided and the first blank field terminates the card.

1Group 1 Software -- Consumer Merge/Purge
Program CM30

3.32

0ADDRDF Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	10 - 12	FIELD LOCATION - Must contain the location of the field(s) required, in the name-and-address records
	17 - 19	
	24 - 26	
	31 - 33	
	38 - 40	
	45 - 47	Note: If necessary, each of these fields may be extended one position to the left, allowing the entry of a four-digit position in columns 9-12, 16-19, etc.
0	14 - 15	FIELD LENGTH - Must contain the length of the corresponding field, from one to 35.
	21 - 22	
	28 - 29	
	35 - 36	
	42 - 43	
	49 - 50	

1Group 1 Software -- Consumer Merge/Purge
Program CM30

3.33

0NAMEDF Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'NAMEDF'.
0		The NAMEDF card is required to specify the location and length of the name field(s) on the input records.
0	08 - 10	LOCATION OF PRIMARY NAME FIELD(S) - Must contain the location of the primary name field(s). If more than one field is specified, then the fields will be formatted together in the order presented to form the name. If four or more fields are specified, the first field will be checked for an asterisk ('*') the first position, and if one is found a name in 'Company form' will be assumed; that is, the name will be assembled from the fourth, second, and third fields, in that order and with no additional space
	15 - 17	
	22 - 24	
	29 - 31	
	36 - 38	
	43 - 45	
0		Note: Each of these fields may be extended one position to the left, allowing the entry of a four-digit location in columns 7-10, etc.
0	12 - 13	LENGTH(S) OF PRIMARY NAME FIELD(S) - Must contain the length of the corresponding primary name field(s).
	19 - 20	
	26 - 27	
	33 - 34	
	40 - 41	
	47 - 48	
0	50 - 52	LOCATION OF ALTERNATE NAME FIELD - May contain the location of an alternate formatted name field. This field, if provided, will be used only if the name specified by the above field(s) is blank.
0		Note: If necessary, this field may be extended one position to the left, allowing the entry of

0

54 - 55

a four-digit location in columns 49-52.

LENGTH OF ALTERNATE NAME FIELD - If columns 50-52 are not blank, then this field must contain the length of the alternate name field.

ONAMEDF Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	57 - 57	TREAT ALL NAMES AS FORMATTED - Must contain one of the following codes to specify whether or not all names are to be treated as 'formatted' names (that is, compared as strings without complex parsing, as is appropriate for 'company' names).
0		blank - analyze each name, and treat it as a 'formatted' name if it appears to be a 'company' name. F - treat all names as 'formatted', regardless of content

0CMPFLD Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'CMPFLD'. The CMPFLD card may optionally be provided to add comparisons for "special" fields in the record which can aid in the dupe-identification process. If present, the specified comparisons are performed following Exit L, if one is present, and before the normal program-controlled record comparison.
0		Up to 20 CMPFLD cards may be provided.
0	08 - 10	POSITION OF FIELD - Must contain the location in the input records of the field to be compared. In general, this will be a position in the original record, but column 22 permits the specification of a field in the extracted-form record, so that a comparison (for instance, for PO Box) which relies on the program's ability to parse a name and/or address may also be specified. Note: if necessary, this field may be extended one position to the left, allowing the entry of a four-digit location in columns 7-10.
0	12 - 13	LENGTH OF FIELD - Must contain the length of the field to be compared.
0	15 - 16	RESULT OF COMPARISON - Must contain one of the following codes to indicate what comparison result is to make this card effective: EQ - the card is to be effective if the two fields are equal NE - the card is to be effective if the two fields are not equal
0	18 - 18	EFFECT OF CARD - Must contain one of the following codes to indicate the desired effect on the pair of records, provided that the comparison is effective according to columns 15-16: D - the two records are Duplicates U - the two records are Unique (not dupes)
0		

0CMPFLD Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	20 - 20	NULL CHECK - Must contain one of the following codes to indicate the treatment of a null (blank) field in either record:
0		blank - The comparison specified will be performed on each pair of records, whether or not the field in either record is blank.
		X - The comparison will not be performed if the field is blank in either record.
		0 - The comparison will not be performed if the field is blank or 0 filled in either record.
0	22 - 22	EXTRACT-AREA FIELD - Must be blank, or contain 'X' to specify that the location provided in columns 8-10 is relative to the 208-byte Extract Area at the front of each record in memory. See the File Layouts section for the content of the Extract Area.
-	Example:	
0	Suppose that in your records there is a telephone number in positions 101-107, and that you have determined that if two records contain equal telephone numbers (except blank ones) then they are to be duplicates, regardless of name and address. In this case, the following card would accomplish the function:	
0	CMPFLD 101 07 EQ D X	

0ALGSEL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'ALGSEL'.
0		ALGSEL parameter cards may optionally be supplied to modify the comparison algorithms used. Twelve different algorithms may be individually tightened or loosened, or have special options invoked.
0		In addition, the comparison may be changed only for some records (that is, for some memory loads), if there is some geographic or demographic code available in the records for algorithm control.
0		The sequence of the ALGSEL cards is not significant (except that all "default" cards must precede any "override" cards--see column 24).
0		Below is a general description of the basic ALGSEL card format. This is followed by a separate page for each algorithm, with details as to its use and the special options available for it.
0	08 - 11	ALGORITHM AFFECTED - Must contain one of the algorithm selection codes. See the following pages for details as to each of the algorithms. The available algorithm selection codes are:
0		ZIPC - ZIP Code (or Postal Code) comparison SURN - Surname (or Company Name) comparison INDV - Individual name comparison HSNR - House Number comparison STNM - Street Name comparison POBX - Post Office Box comparison RRBX - Rural Route and Box comparison APNR - Apartment number comparison ADDR - General Address comparison HHLN - Household determination TOTL - Overall comparison RESI - comparison when either name is treated as 'RESIDENT'

0ALGSEL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	13 - 13	TIGHTNESS OR LOOSENESS - Must contain one of the following codes to specify the tightness or looseness of the comparison specified. A "tight" comparison will require that the fields be nearly equal in order that two records be duplicates, and a "loose" comparison will permit the values to be "farther apart". The codes permitted are: 0 E - the fields must be equal T - the comparison is to be "tight" M - the comparison is to be "medium" L - the comparison is to be "loose" X - the comparison is not to be performed 0 Note that some choices are not available for some comparisons. See the following pages on the specific algorithms.
0	15 - 15	OPTIONS - Most Algorithm choices permit one or more options to be specified, which serve to enable or disable a special segment of the comparison logic. See the following pages for details of the options available for each Algorithm. 0 An option is specified as on by entering the next available of these option columns. Each option may be specified only once. 0 The options for any Algorithm may be specified in any order. 0 The first option column which is blank terminates examination of the card for options. All remaining option columns must be blank. Columns 14 and 23 are verified to be blank so that an option inadvertently punched in these columns will be flagged as an error. 0 By default all options are off.
	16 - 16	
	.	
	.	
	.	
	22 - 22	

0ALGSEL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	24 - 24	DEFAULT OR OVERRIDE - Must be blank to specify a default ALGSEL card, or contain 'X' to specify that this card is an override ALGSEL card.
0		Default ALGSEL cards are used to establish a "background" comparison environment. For most applications (where the comparison is the same for the entire file), only this "background" need be established.
0		All default ALGSEL cards must precede any override ALGSEL cards.
0		Override ALGSEL cards are used to modify the background comparison for subsets of the file, based on a value in the Name/Address records. When override cards are effective, they are modifications to the background situation, not to the original (default) settings.
0	26 - 28	LOCATION OF CODE - Must be blank for default ALGSEL cards. For the first override card, must contain the location in the Name/Address records at which the override value may be found. For all additional override cards, this field must be blank.
0		Note: If necessary, this field may be extended one position to the left, allowing the entry of a four-digit position in columns 25-28.
0	30 - 30	LENGTH OF CODE - For the first override card, must contain the length of the override value in the Name/Address records. For all other cards, this field must be blank.
0	32 - 40	VALUE OF OVERRIDE CODE - For all override cards, must contain the value (of the code in the Name/Address records) which causes this card to be effective.

+
- Synopsis_of_Override_Processing:
+

0 For each memory load, the first record of the memory load is examined (compared against the stored override codes) to see if any override ALGSEL cards are to be used. If not, the default ALGSEL cards are used unmodified. If an override code is found which was specified, then the default ALGSEL cards are used, as modified by the override cards with the code in the record.

0ALGSEL Parameter Card

ZIPC Algorithm

- The ZIPC Algorithm is used to compare the ZIP Code or Postal Code of two records.

0 TIGHTNESS_OR_LOOSENESS:

0 Column 13 must contain one of the codes E, T, M, L, or X. If no ALGSEL ZIPC card is provided, the default is X (no comparison-- because the vast majority of the sortations used will include three or five characters of the ZIP Code, depending on the degree of match required).

0 OPTIONS:

0 The options available are as follows:

0 Option 3 specifies that the first three characters of the ZIP Code, or Postal Code, must be equal in order for two records to be duplicates.

0 Option N specifies that if in two records the Name and ZIP Code are exactly equal, then the two records are duplicates regardless of any other address information.

- Note: Options 3 and N will be effective if specified, whether or not column 13 is 'X'.

0ALGSEL Parameter Card

SURN Algorithm

- The SURN Algorithm is used to compare the Surnames of two records, or the Formatted Names if the name in either record is interpreted as a 'company name'. The options available (see below) also permit the treatment of some or all names as 'RESIDENT' records; see also the RESI algorithm.
- 0 TIGHTNESS_OR_LOOSENESS:
- 0 Column 13 must contain one of the codes E, T, M, or L. If no ALGSEL SURN card is provided, the default is M (medium).
- 0 OPTIONS:
- 0 The options available are as follows:
- 0 Option I specifies that the initial letter of the names must be equal in order for two records to be duplicates.
- 0 Option N specifies that all names are to be treated as "natural" names (that is, they are to be assumed to be names). If Option N is not specified, then a record in which the name is blank, one which is 'RESIDENT' or 'OCCUPANT', will be treated as a "RESIDENT" record; that is, the name will not contribute to the comparison, and the address comparison will be controlled by the ALGSEL RESI card. If Option N is specified, then such names will be treated as "names", and, for instance, RESIDENT will not match OCCUPANT at the same address.
- 0 Option X specifies that all names are to be treated as 'RESIDENT' names (see above).

0ALGSEL Parameter Card

INDV Algorithm

- The INDV Algorithm is used to determine the uniqueness of two names, as individuals. If specified (with column 13 not 'X'), then what is generally known as an Individual duplicate elimination will be performed; if column 13 is 'X' by specification or by default, a Household duplicate elimination will result. In an Individual duplicate elimination, two records which represent two different individuals will not be duplicates, even if they are in the same household. For example, JOHN SMITH and TOM SMITH at the same address will not be duplicates.
- 0 TIGHTNESS_OR_LOOSENESS:
- 0 Column 13 must contain one of the codes T, M, L, or X. If no ALGSEL INDV card is provided, the default is X (for a household duplicate elimination).
- 0 OPTIONS:
- 0 The options available are as follows:
- 0 Option G specifies that if two records have the same name, except that one has the title MRS, and the sex of the name (alone) is indeterminate because the name is ambiguous or just initials, then the names are permitted to match. If Option G is not specified, then A B SMITH and MRS A B SMITH at the same address will not be (individual) duplicates. If Option G is specified, then those records will be duplicates.
- 0 Option M specifies that two records are not duplicates if the title of one is MISS and the title of the other is MRS. If Option M is not specified, two such records with the same name, at the same address, will be considered as duplicates.
- 0 Option C specifies that two records are not duplicates if one of them has been determined to be a Company name. If Option C is not specified, two records, the first containing a surname, the second containing a matching company name at the same address, will be considered duplicates.
- Note: Options C, G, and M are effective only if column 13 is not 'X'.

1Group 1 Software -- Consumer Merge/Purge 3.43
Program CM30

0ALGSEL Parameter Card

HSNR Algorithm

- The HSNR Algorithm is used to compare the House Numbers in
two records which each have a House Number.
0 TIGHTNESS_OR_LOOSENESS:
0 Column 13 must contain one of the codes E, T, M, or L. If no
ALGSEL HSNR card is provided, the default is M (medium).
0 OPTIONS:
0 The options available are as follows:
0 Option L specifies that two records are not duplicates if
the lengths of their House Numbers are not equal.

1Group 1 Software -- Consumer Merge/Purge 3.44
Program CM30

0ALGSEL Parameter Card

STNM Algorithm

- The STNM Algorithm is used to compare the Street Names in
two records which each have a Street Name.
0 TIGHTNESS_OR_LOOSENESS:
0 Column 13 must contain one of the codes E, T, M, or L. If no
ALGSEL STNM card is provided, the default is M (medium).
0 OPTIONS:
0 The options available are as follows:
0 Option H specifies that two records are not duplicates if:
0 The Highway number is unequal, for streets of the form
ROUTE 28 or HIGHWAY 1, and similar;
The first four characters of numeric Street Names are
unequal (such as 123RD AV)
The first four characters of Street Names which begin
with 'AV' (such as AVENUE A) are unequal (after
transformation by the address parsing routine).
0 Option I specifies that two records are not duplicates if
the first character of the Street Name is not equal (or
the first three characters for numeric streets, or the
first three characters of the Highway number).
0 Option 4 specifies that two records are not duplicates if
the first four characters of the Street Name (or Highway
Number, etc.) are not equal.
0 Option D specifies that two records are not duplicates if
their Street Names have unequal Directionals.

1Group 1 Software -- Consumer Merge/Purge
Program CM30

3.45

0ALGSEL Parameter Card

POBX Algorithm

- The POBX Algorithm is used to compare the PO Boxes in two records which each have a PO Box. Unless Option P is specified (see below), this algorithm also controls the comparison of the Box Numbers in two records when one appears to have a PO Box and the other appears to have a Rural Route Box.
- 0 TIGHTNESS_OR_LOOSENESS:
- 0 Column 13 must contain one of the codes E, T, M, or L. If no ALGSEL POBX card is provided, the default is M (medium).
- 0 OPTIONS:
- 0 The options available are as follows:
- 0 Option P specifies that if one record contains (what appears to be) a Post Office Box, and the other contains (what appears to be) a Rural Route Box, then the two box numbers are not to be compared.

1Group 1 Software -- Consumer Merge/Purge
Program CM30

3.46

0ALGSEL Parameter Card

RRBX Algorithm

- The RRBX Algorithm is used to compare the Rural Route and Box numbers in two records which each have a Rural Route or Highway Contract number.
- 0 TIGHTNESS_OR_LOOSENESS:
- 0 Column 13 must contain one of the codes E, T, M, or L. If no ALGSEL RRBX card is provided, the default is M (medium).
- 0 OPTIONS:
- 0 No options are available to the RRBX Algorithm.

0ALGSEL Parameter Card

APNR Algorithm

- The APNR Algorithm is used to compare the Apartment Numbers in two records when either has an Apartment Number.
- 0 TIGHTNESS_OR_LOOSENESS:
- 0 Column 13 must contain one of the codes E, T, M, L, or X. If no ALGSEL APNR card is provided, the default is M (medium).
- 0 OPTIONS:
- 0 The options available are as follows:
- 0 Option A specifies that if one record contains an Apartment Number and the other does not, then the two records are not duplicates.

0ALGSEL Parameter Card

ADDR Algorithm

- The ADDR Algorithm is used to "compare" two complete addresses.

+

It is invoked after each of the address elements has been compared (and passed, or the comparison of the records would already have stopped).

0 TIGHTNESS_OR_LOOSENESS:

+

0 Column 13 must contain one of the codes T, M, or L. If no ALGSEL ADDR card is provided, the default is M (medium).

0 OPTIONS:

+

0 The options available are as follows:

0

Option U specifies that two records which have similar but non-identical names, and no address elements in common, are permitted to match. In the absence of Option U, JOHN P SMITH at RR 4 BOX 567 will match JOHN P SMITH at 123 MAIN ST, but will not be permitted to match J P SMITH at 123 MAIN ST. With Option U specified, both of the MAIN ST records will match the rural address.

0 Note that the absence of Option U will not keep two records,

+

which each have no address information, from matching one another.

0

Option X specifies that two records which have no address elements in common are not duplicates, regardless of the names. If Option X is specified, then JOHN P SMITH at RR 4 BOX 567 will not match JOHN P SMITH at 123 MAIN ST.

+

0 Note that Option X will not keep two records, of which at

+

least one has no address information, from matching one

+

another.

0

Option Y specifies that two records with "equal" names will not be considered duplicates before the street name element of the addresses are compared. JOHN SMITH at 123 MAIN ST will not match JOHN SMITH at BOSTON COLLEGE if

+

Option Y is specified. Address elements such as BOSTON COLLEGE or CAPITOL PLAZA are considered to be street names by the program.

0 Notes:
0 If Options U and X are both specified, then Option X will take
precedence and Option U will be ineffective.
0 If ADDR Option X and ZIPC Option N are both specified, then
ZIPC Option N will take precedence; that is, two records with
equal ZIP Codes and equal Names will be duplicates.

+ _____
1Group 1 Software -- Consumer Merge/Purge 3.49
Program CM30

0ALGSEL Parameter Card

HHL D Algorithm

- The HHL D Algorithm is used when the address comparison gives no
+ _____
positive indication that two records belong to the same
household (that is, when no equality has been established

+ _____
involving House Number, Apartment Number, PO Box, or Rural
Route Box). Its function is to determine if the two records
are members of the same household, by examination of the two
name fields.

0 TIGHTNESS_OR_LOOSENESS:

+ _____
0 Column 13 must contain one of the codes T, M, L, or X. If no
ALGSEL HHL D card is provided, the default is M (medium).

0 OPTIONS:

+ _____
0 No options are available for the HHL D Algorithm.

1Group 1 Software -- Consumer Merge/Purge 3.50
Program CM30

0ALGSEL Parameter Card

TOTL Algorithm

- The TOTL Algorithm is used to "compare" two complete records.
+ _____
It is invoked after the ZIP Code, Name, and Address have
been compared (and passed, or the comparison of the records
would already have stopped).

0 TIGHTNESS_OR_LOOSENESS:

+ _____
0 Column 13 must contain one of the codes T, M, or L. If no
ALGSEL TOTL card is provided, the default is M (medium).

0 OPTIONS:

+ _____
0 No options are available for the TOTL Algorithm.

0ALGSEL Parameter Card

RESI Algorithm

- The RESI Algorithm is used to compare the address information
+

of two records when either or both is to be treated as
'RESIDENT', or when both names are blank.
0 When two records are compared using the Resident Algorithm (which
this card controls), then no other ALGSEL cards have any effect
on the comparison. Accordingly, a file which contains some
RESIDENT names and some normal names can be processed in a single
pass, using the ALGSEL RESI card to control the Resident
comparisons and the other ALGSEL cards to control the normal
comparisons.

0 TIGHTNESS_OR_LOOSENESS:

+
0 Column 13 must contain one of the codes E, T, M, or L. If no
ALGSEL RESI card is provided, the default is M (medium).

0 OPTIONS:

+
0 The options available are as follows:

0 Option A specifies that the Apartment Numbers in the two
records are not to be compared.

+
0 Option R specifies that the two records are not duplicates
if the only address information available is a Rural
Route number or Highway Contract number.

0 Option 4 acts just the same as STNM Option 4 (which see).

- Note: The presence of the ALGSEL RESI card does not cause any

+
more or fewer names to be treated as 'RESIDENT'. This is done on
a per-File-Code basis by the FLCnnn card, column 20, or on a
universal basis by options in the ALGSEL SURN card.

0 The ALGSEL RESI card controls the logic used to compare two records
when either or both is to be treated as 'RESIDENT'.

0DUPCTL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'DUPCTL'. The DUPCTL parameter card is required to provide the locations for the information field(s) which are to be placed in the Duplicate File (File OD) for printing and/or analysis of the duplicates.
0		
0	08 - 10	DUPLICATE-CONTROL CHARACTER - Must contain the location at which will be placed the duplicate control character. The specified location in the Duplicate Names file (File OD) will contain a zero ('0') for the first record of each group of duplicates, and a blank for each additional record in the group.
0		
0		If necessary, this field may be extended one position to the left, allowing the entry of a four-digit location in columns 7-10.
0	12 - 14	DUPLICATE-GROUP DRIVER - May contain the location at which a one-character indicator will be placed. For the duplicate-group "driver" (the record which is compared against each other record), this location will contain 'X'. For all other records in each duplicate group, it will be blank.
0		
0		If necessary, this field may be extended one position to the left, allowing the entry of a four-digit location in columns 11-14.

0DUPCTL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	16 - 18	DEGREE OF MATCH - May contain the location at which will be placed a two-character value which will reflect the degree of match between each record in a duplicate group and the duplicate-group driver. The possible values are:
0		XX - this record was the duplicate-group driver XL - duplication was forced by User Exit L CF - duplication was forced by a CMPFLD card EQ - duplication was forced by Name/Address equality NM - duplication was forced by name equality; either the only address element common to both records was the Street Name, or else all individual address elements passed the individual element comparisons RS - duplication was forced by the logic used to compare two records when either is treated as 'RESIDENT', or when both names are blank ZN - duplication was forced by ZIPC option N numeric - indicates the degree of match for record compared "normally"; the higher the number, the better the match. A value of '00' represents 100.
0		If necessary, this field may be extended one position to the left, allowing the entry of a four-digit location in columns 15-18.
0	20 - 22	DEGREE OF MATCH FOR INDV COMPARISON - May contain the location at which will be placed a score which will reflect the degree of match of the individual name between each record in a duplicate group and the duplicate-group driver. The duplicate-group driver will have the value of 'X'. If there was n individual comparison, then all non-driver records will have a score of '0'; otherwise a value from 1 will be stored with 9 being the best match and 1 being the worst.
0		If necessary, this field may be extended one position to the left, allowing the entry of a four-digit location in columns 19-22.

0DUPREC Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'DUPREC'.
0		The DUPREC card may optionally be included to change the content of the Duplicate Names (file OD) records from what it would otherwise be.
0	08 - 10	PRIORITY NUMBER - May contain the starting position of a 4-byte field in the Duplicate Names file (file OD) which will be filled with the four-digit Priority Number associated with each record by the combined effect of the FLCnnn cards and the PRIORT card. Records for File Codes which are not defined by any FLCnnn card will be assigned a priority number of 9999.
0		If necessary, this field may be extended to the left one position, allowing the entry of a 4-digit location in columns 7-10.
0	12 - 14	UNIQUENESS CODE - May contain the starting position of a 4-byte (binary) field. This field will contain equal values for records determined to be duplicates of one another, and unequal values for records which are determined to be non-duplicates. The surviving record (on File OM and OE) will be assigned the same uniqueness code as that assigned to the duplicate group.
0		If necessary, this field may be extended to the left one position, allowing the entry of a 4-digit location in columns 11-14.
0	16 - 18	SELECTION INDICATOR - May contain the position in the output Duplicate records at which a single character will be stored as follows:
0		blank - this record not involved in duplication S - Single-buyer survivor M - Multi-buyer survivor 0 - Purge-file drop 1 - Single-buyer drop 9 - Multi-buyer drop
0		If necessary, this field may be extended to the left one position, allowing the entry of a 4-digit location in columns 15-18.

0DUPREC Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	20 - 22	<p>MULTI-BUYER LEVEL - May contain the position in the output Duplicate records at which a two-byte Multi-Buyer Level will be stored. This field will contain a two-digit number, as follows:</p> <ul style="list-style-type: none">00 for matched records whose duplicate group contains no Multi-Buyer contributing File Codes; and for those records in a duplicate group which were purged by a Purge File Code record.01 for unmatched records whose File Code is a Multi-Buyer contributor; and for matched records, which were not purged by a Purge File Code record, from a duplicate group with only one Multi-Buyer contributing File Code in the group.02-99 for records involved in duplication, which were not purged by a Purge File record, and whose duplicate group contained two or more File Codes which were unique Multi-Buyer contributors.
0		<p>Note: The number stored will be the number of unique Multi-Buyer contributing File Codes in the duplicate group, except that if there are more than 99, 99 will be stored. Note that the word "unique" is defined to mean unique as per the MLTLIM parameter card if one is present.</p>
0		<p>If necessary, this field may be extended to the left one position, allowing the entry of a 4-digit location in columns 19-22.</p>
0	24 - 26	<p>NR OF RECORDS IN DUPLICATION - May contain the position in the output Duplicate records at which a four-byte number will be stored. For unmatched records, the value of the number will be 0001. For records involved in duplication, this number will be the total number of records in the duplicate group.</p>
0		<p>If necessary, this field may be extended to the left one position, allowing the entry of a 4-digit location in columns 23-26.</p>

0CLNREC Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'CLNREC'. The CLNREC card may optionally be included to change the content of the Mail File records (files OM and OE) from what it would otherwise be.
0	08 - 10	PRIORITY NUMBER - May contain the starting position of a 4-byte field in the Mail File records (files OM and OE) which will be filled with the four-digit Priority Number associated with each record by the combined effect of the FLCnnn cards and the PRIORT card. Records for File Codes which are not defined by any FLCnnn card will be assigned a priority number of 9999. If necessary, this field may be extended to the left one position, allowing the entry of a 4-digit location in columns 7-10.
0	12 - 14	UNIQUENESS CODE - May contain the starting position of a 4-byte (binary) field. This field will contain equal values for records determined to be duplicates of one another, and unequal values for records which are determined to be non-duplicates. If a record is involved in duplication, then the value stored here will be the same as that stored in each of the Duplicate (file OD) records. If necessary, this field may be extended to the left one position, allowing the entry of a 4-digit location in columns 11-14.
0	16 - 18	SELECTION INDICATOR - May contain the position in the output Mail file records at which a single character will be stored as follows: blank - this record not involved in duplication S - Single-buyer survivor M - Multi-buyer survivor 0 - Purge-file drop 1 - Single-buyer drop 9 - Multi-buyer drop If necessary, this field may be extended to the left one position, allowing the entry of a 4-digit location in columns 15-18.

0CLNREC Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	20 - 22	MULTI-BUYER LEVEL - May contain the position in the output Mail File records at which a two-byte Multi-Buyer Level will be stored. This field will contain a two-digit number, as follows:
0		00 for unmatched records whose File Code is not a Multi-Buyer contributor, and for matched records whose duplicate group contains no Multi-Buyer contributing File Codes. 01 for unmatched records whose File Code is a Multi-Buyer contributor, and for matched records with only one Multi-Buyer contributing File Code in the duplicate group. 02-99 for records involved in duplication whose duplicate group contained two or more File Codes which were Multi-Buyer contributors. The number stored will be the number of unique Multi-Buyer contributing File Codes in the duplicate group, except that if there are more than 99, 99 will be stored. Note that the word "unique" is defined to mean unique as per the MLTLIM parameter card if one is present.
0		If necessary, this field may be extended to the left one position, allowing the entry of a 4-digit location in columns 19-22.
0	24 - 26	NR OF RECORDS IN DUPLICATION - May contain the position in the output Mail File records at which a four-byte number will be stored. For unmatched records, the value of the number will be 0001. For records involved in duplication, this number will be the total number of records in the duplicate group.
0		If necessary, this field may be extended to the left one position, allowing the entry of a 4-digit location in columns 23-26.

0PRIORT Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'PRIORT'. The PRIORT parameter card may be used to specify the algorithm to be used to select the surviving record from a duplicate set. If the PRIORT card is omitted, then the relative priority entered in the FLC cards will be used (except for Purge files, which will be treated as under the blank value for column 7, below).
0	07 - 07	PRIORITY TREATMENT FOR PURGE FILES - Must contain one of the following codes to specify the treatment of Purge File records with respect to priority: blank - The priority assigned to each Purge File Code record will be 0000, regardless of the value entered in the corresponding FLCnnn parameter card. With this zero priority, a Purge ('P') file record will purge all records which match it. P - The priority assigned to each Purge File Code ('P' or 'I' in the FLCnnn card) is to be left as is. With this treatment, a Purge File record will only cause those records to be purged which match it and + which have a lower priority (that is, a higher priority number). 0 Note: If Random or Semi-random is specified in + _____ column 8, then column 7 is verified, but treated as described above under blank. If Mixed is specified in column 8, then those Purge File Codes included within the final n File Codes will be treated as + _____ having a specific priority (dependent upon their position), and the remaining Purge File Codes will be treated as having a higher priority than the remaining non-Purge File Codes, but a lower priority than non-Purge File Codes included within the last n File Codes.

OPRIORT Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	08 - 08	RECORD SELECTION ALGORITHM - Must contain one of the following codes to specify the algorithm which is to be used to determine which record of each duplicate-group is to survive (that is, which will be written to File OM and/or OE).
0		blank - The priority values entered in columns 4-6 of the FLC parameter cards will be the relative priority of the File Codes. The lowest number will be the highest priority, and so forth. If more than one File Code have the same relative priority then the surviving record will be chosen at random.
		P - Priority by Position; the last File Code defined by an FLC card will have the highest priority, the next-to-last File Code will have the next-highest priority and so forth.
		R - Random; the survivor will be selected on a statistically random basis.
		S - Semi-random priority; the surviving record will be the first record found in a duplicate set. If there is a bias introduced in the sortation of the input files, then this bias will be reflected in the record selection.
+		M - Mixed priority; the last n (where n is the value entered in columns 10-12) FLC parameter cards have priority (the last having the highest priority, the next to last having the next to highest priority and so forth), and the rest of the FLC parameters have random priority
0	10 - 12	EXTENT OF MIXED PRIORITY - Must be blank if column 8 is not 'M'. If column 8 does contain 'M', then this field must contain a number between 001 and the value one less than the number of File Codes defined by the FLC parameter cards. The number entered specifies how many of the defined File Codes (beginning with the last-defined) will be treated as 'P' above, with the remaining File Codes being treated as 'R' above.

OPRIORT Parameter Card

```
- CARD COLUMNS          FIELD DESCRIPTION
- Note: The following fields permit the modification of the priority
+
  for individual records, either because the name is treated as
  'RESIDENT' or because one or more elements is missing. When
  specifying the priorities for such situations, it will be important
  for you to know that when a File Code is to be treated as "random"
  or "semi-random" (see above) it will actually have a priority
  of 9000 assigned. Therefore, even in an all-random situation it
  is possible to specify that "normal" records are to be preferred to
  "resident" records, by ensuring that the priority for "resident"
  records ends up being greater than 9000.
0 No priority modification is applied to records whose File Code
  is identified as a Purge File.
0 14 - 18          PRIORITY MODIFICATION FOR RESIDENT - May contain
  a value which will be used to affect the actual
  priority of all records which are to be treated
  as 'RESIDENT' (see the FLC card, column 20, and
  the ALGSEL SURN card). This field consists of
  two subfields, as follows:
0 14 - 14          MODIFICATION TYPE - Must contain one of the
  following values:
0 '+' - the value in columns 15-18 is to be
  added to the original (via File Code)
+
  _____
  priority
0 '-' - the value in columns 15-18 is to be
  subtracted from the original priority
+
  _____
0 'R' - the value in columns 15-18 is to
  replace the original priority
+
  _____
0 15 - 18          MODIFICATION VALUE - Must contain the non-zero
  value which is to be used to modify the priority
  as specified by column 14.
0 Note: For records treated as 'RESIDENT',
+
  _____
  regardless of whether columns 14-18 are provided,
  the following actions are taken:
0 The original priority is modified by 14-18,
  if those columns are non-blank
  If there is no address information, then the
  value in columns 26-28, if any, is added
  to the priority (whether original or from
  columns 14-18)
  No further modifications to the priority are
  made by this card.
```

OPRIORT Parameter Card

```
- CARD COLUMNS          FIELD DESCRIPTION
- Note: The following eight fields may be used to alter the
+
+      priority of records which have one or more fields omitted
+      (or unrecognizable). Columns 20-24 are required if any of
+      the remaining fields are present. If the field corresponding
+      to a missing element is omitted, then that element being
+      missing will have no effect on the result priority.
0      20 - 24          PRIORITY MODIFICATION FOR MISSING ELEMENTS - If
+      columns 26-44 are all blank, then this field
+      must be blank. Otherwise, it must be present to
+      specify the basic modification to a record's
+      priority which is to occur when one or more
+      elements are missing from the name and address.
+      The modification process proceeds as follows:
0      If one or more elements (for which addends
+      are entered below) are missing from a record,
+      then the priority of the record is first
+      modified as specified by columns 20-24, and
+      the addend(s) for the missing element(s)
+      are added to the result. If the final sum
+      is less than one, one is retained as the
+      priority, and if greater than 9999, then
+      9998 is retained as the final priority.
0      The modification specification consists of two
+      subfields, as follows:
0      20 - 20          MODIFICATION TYPE - Must contain one of the
+      following values:
0      '+' - the value in columns 21-24 is to be
+      added to the original (via File Code)
+      priority
+      '-' - the value in columns 21-24 is to be
+      subtracted from the original priority
+      'R' - the value in columns 21-24 is to
+      replace the original priority
+
0      21 - 24          MODIFICATION VALUE - Must contain the number
+      which is to be used to modify the priority as
+      specified by column 20.
0      26 - 28          ADDEND FOR MISSING ADDRESS - May contain a
+      non-zero value which is to be added to the
+      (modified) priority if there is no address
+      information in the record.
```

0PRIORT Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	30 - 32	ADDEND FOR MISSING NAME - May contain a non-zero value which is to be added to the (modified) priority if there is no name in the record (column 34 through 44 will also be added in this case). Note that if a blank name is to be treated as 'RESIDENT' (see ALGSEL SURN card), then the priority will be adjusted as specified by columns 14-18 rather than by this field.
0		Note: The fields in columns 34-48 will have no effect on records
+	_____	whose names are stored in 'Company' form (that is, whose names the program decides are 'Company' names).
0	34 - 36	ADDEND FOR MISSING FIRST NAME - May contain a non-zero value which is to be added to the (modified) priority if there is no first name in the record.
0	38 - 40	ADDEND FOR MISSING TITLE OF RESPECT - May contain a non-zero value which is to be added to the (modified) priority if there is no title of respect in the record (such as 'MR' or 'MRS', at the front
0	42 - 44	ADDEND FOR MISSING PROFESSIONAL TITLE - May contain a non-zero value which is to be added to the (modified) priority if there is no business title in the record (such as 'CHAIRMAN', at the end).

0PRIORT Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	46 - 48	ADDEND FOR INITIALS-ONLY - May contain a non-zero value which is to be added to the (modified) prior for records whose first name is an initial (only) and whose middle name is either missing or a single initial.
0	50-52	ADDEND FOR MISSING APARTMENT NUMBER - May contain a non-zero value which is to be added to the (modified) priority if there is no apartment information in the record.
0	54-56	ADDEND FOR COMPANY NAME - May contain a non-zero value which is to be added to the (modified) priority for a company name record.
0		For example:
0		SMITH 20 South Main St 12345
		JOHN SMITH 20 South Main St 12345

These 2 records will be treated as duplicates, and it is possible for the SMITH record to be kept, when both records have the same priority. If this option is used, the SMITH record can be kept or dropped based on the priority.

0FLC Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 03	PARAMETER KEYWORD - Must contain 'FLC'.
0		FLC parameter cards are required to identify the File Codes which are to be present in a given execution, and to specify how each is used. The relative priority of the various File Codes must be specified, although these priorities may be over-ridden by use of the PRIORT card.
0	04 - 06	RELATIVE PRIORITY - Must contain a number from 001 to 999, expressing the relative priority of the various File Codes. 001 indicates the highest priority, and 999 the lowest. When a duplicate group contains records from File Codes which have different priorities, then the surviving record will be that from the File Code which has the highest priority (that is, the lowest number). If two File Codes have the same priority, then a record will be chosen at random.
0		If one of two matching records is from a Purge File (its FLCnnn card column 18 is 'P'), the other record will be dropped if its priority is lower than that of the Purge record (that is, if the non-Purge record has a higher priority number than the Purge record).
0		For Purge File Codes (column 18 containing 'P'), the priority entered may be '000'. This permits a Purge File Code to be entered with a priority which lets it purge records from all non-Purge File Codes. See the PRIORT card, column 7, for a full discussion of the interaction of the Priority field with Purge File Codes.
0		Note that the relative priorities entered here may be over-ridden by the action of the PRIORT parameter card, if one is provided.
0	08 - 16	FILE CODE - Must contain the file code as it will appear after the action of the FLCODE card, left-justified.

0FLC Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	18 - 18	SPECIAL TREATMENT - Must contain one of the following codes:
0		blank - normal treatment
		P - this File Code is to be treated as a "Purge file" or "suppression file". That is, none of its records are to be written to the output Mail file, and no records which match a record from this File Code are to be written to the output Mail File
		S - this File Code is not to contribute to

+

the Multi-Buyer status of a group of duplicates. That is, suppose that there are three File Codes, A, B and C. Normally, if a duplicate group is detected with records from any two or more of the File Codes, then the surviving record will be identified as a "Multi-Buyer". However, if File Code A (for example) is marked with the 'S' in this column, then neither an A-B match nor an A-C match will have the survivor marked as a Multi-Buyer. An A-B-C match, or a B-C match, will still have the survivor marked as a Multi-Buyer.

0FLC Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	19 - 19	DO NOT USE AS A DRIVER - Must be blank (normal), or contain 'X' to indicate that records from this File Code are not to be used as "Duplicate-group Drivers".
0		During the duplicate-identification process, each record is normally compared against every other. That is, one record is chosen as a potential "driver" for a duplicate group, and then it is compared against the remaining records. If a match (or more than one) is found, then a duplicate group is formed and the original record is marked as the "driver". If this column is 'X', then records from this File Code will not be selected as potential drivers. The result of this will be that no match between non-driver File Codes will be possible, although it is possible that two such records will each be found to match a third record which is permitted to be a driver.
0		Unless it is necessary to have purge-purge matches in the Duplicates file, it is generally advisable to mark all Purge File Codes with the 'X' in this column, to save on comparison time. If the number of purge records is large compared to the number of non-purge records, the bulk of the comparison time will consist of comparing one purge record to another, and in most cases this is not necessary.

OFLC Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	20 - 21	FILE-SPECIFIC OPTIONS - These columns may be used to specify that all or a portion of the records for this File Code are to be ignored (treated as if not present) or treated as 'RESIDENT', based upon the input Tape on which the records are found. If both columns are blank, then all records for the File Code will be processed normally. Column 20 must be blank, 'I', or 'R'. Column 21 must be blank or contain one of the input Tape IDs, 'A', 'B', or 'C'. If column 20 is blank, then: If column 21 is also blank, all records are processed normally. If column 21 contains 'A', 'B', or 'C', then all records from the Tape specified will be processed, and those from the remaining Tapes will be ignored. If column 20 is 'I', then: If column 21 is blank, then all records will be ignored. If column 21 contains 'A', 'B', or 'C', then all records from the Tape specified will be ignored, and those from the remaining Tapes will be processed normally. If column 20 is 'R', then: If column 21 is blank, then all records will be treated as 'RESIDENT'. If column 21 contains 'A', 'B', or 'C', then all records from the Tape specified will be treated as 'RESIDENT', and those from the remaining Tapes will be processed normally.
0		
0		
0		
0		
0		
0	23 - 37	FILE DESCRIPTION - May contain the description or name of the file that the File Code corresponds to

0FLC Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	44 - 44	OUTPUT CLEAN FILE SELECTION - Must contain one of the following codes, to specify the file or files to which surviving records with this File Code are to be written. If this column is left blank, then surviving records for this File Code will be written to both Files OM and OE, unless restricted
+		by CONTRL card columns 22 or 24. The possible values for this column are:
0		blank - write surviving records with this File Code to both File OM and File OE, unless restricted by CONTRL Card columns 22 or 24
		M - write surviving records with this File Code only to File OM. Restrictions applied by CONTRL Card column 22 will be applied
		E - write surviving records with this File Code only to File OE. Restrictions applied by CONTRL Card column 24 will be applied.
		X - do not write surviving records with this File Code to either of the output clean files. This option should only be used when you want to suppress a particular File Code altogether from the output
0		Notes: Normally, splits between Files OM and OE are accomplished either by use of this column (in all FLC cards), with CONTRL card columns 22 and 24 left blank; or by the use of CONTRL card columns 22 and 24 with this column left blank in all FLC cards. Using both this column and CONTRL card columns 22 and 24 in the same run may result in some surviving records not being written to either File OM or File OE. Use of this column in some but not all FLC cards may result in some records being written to both File OM and File OE, while other records are written to only one file or the other. (Combining these two non-recommended practices may result in an even worse mess!)
0		If this column is 'M' in any FLC card, then File OM is required, and similarly for 'E' and File OE.
0		To ensure that the character in column 44 is not a part of a description which was inadvertently made too long, both columns 43 and 45 are verified to be blank.

0CHANGE Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
0	01 - 06	PARAMETER KEYWORD - Must contain 'CHANGE'. Up to 100 CHANGE cards may be provided to specify that specific File Codes, each on a specific input file, are to be changed. Normally, this is necessary only when an incorrect File Code was specified for one or more conversion runs. 0 If any File Codes are to be changed, then it is imperative to specify a location for the Output File Code in the CONTRL card. The CHANGE card affects the internal representation of the File Code only.
0	08 - 16	ORIGINAL (ERRONEOUS) FILE CODE - Must contain the File Code which is to be changed, as it will appear after the action of the FLCODE card, left-justified.
0	17 - 17	ALL FILE CODES - Must contain one of the following codes: 0 blank - change the File code specified in columns 8-16 X - change all File codes on the specified (column 18) file to the File code specified in columns 23-28. If there are any other CHANGE cards (with column blank) they will be effective for the specific File codes entered; all other File codes will be changed to the value in this card (column 17 = X). If column 17 is 'X', then columns 8-16 must be blank.
0	18 - 18	INPUT FILE ID - Must contain one of the following values: A - the specified File Code is to be changed only in records from input file IA. B - the specified File Code is to be changed only in records from input file IB. C - the specified File Code is to be changed only in records from input file IC. 0 Of course, if the File Code is to be changed on two or all of the input files, you must enter one CHANGE card for each input file.

1Group 1 Software -- Consumer Merge/Purge 3.72
Program CM30

0CHANGE Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	20 - 28	RESULT (CORRECTED) FILE CODE - Must contain the File Code as it should have appeared.
+		
0		All records whose File Codes were changed will match the FLCnnn card which matches the new
+		File Code.

1Group 1 Software -- Consumer Merge/Purge 3.73
Program CM30

0MULTIB Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'MULTIB'. The MULTIB parameter card may be used to change the File Code of Multi-Buyer records on the output Mail file(s) to the same value for all, regardless of the original File Code, without affecting any
+		of the reports.
0	08 - 16	MULTI-BUYER FILE CODE - Must contain the File Code to be assigned to the survivor(s) of a multi-buyer duplicate set.
0		Note: For the Duplicate Elimination program, an output File
+		Code location must be specified in the CONTRL card if
+		the File Code specified here is to appear in the output Mail files (OM and/or OE).

OMLTLIM Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'MLTLIM'. The MLTLIM parameter card may optionally be provided to limit the identification of Multi-Buyer records.
0		Normally, if a duplicate-group contains records from a single File Code, then the group is identified as a Single-Buyer group, any drops are recorded as Single-Buyer drops, and any survivors are identified as Single-Buyer survivors. If more than one File Code is present in the group, it is considered a Multi-Buyer group, any drops are counted as Multi-Buyer drops, and the survivors are marked as Multi-Buyer survivors.
0		By means of this card, the File Code comparison, for the purpose of Multi-Buyer identification only, may be restricted to a substring of the File Code proper.
0	08 - 08	POSITION OF SUBSTRING - Must contain the starting position of the relevant substring, within the File Code as stored by the FLCODE parameter.
0	10 - 10	LENGTH OF SUBSTRING - Must contain the length of the substring.
0		Notes: The sum of column 8 and column 10 may not exceed 10.

0CMPLIM Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'CMPLIM'. The CMPLIM parameter card may optionally be provided to limit the comparison of records to those whose File Codes have a specified relationship.
0		Normally, each record is compared to every other (within a single memory load), and when a match is detected the two records are identified as duplicates. If two records cannot be compared, they cannot be identified as duplicates (although it is possible that they will end up in the same duplicate group because they each match a third record).
0	08 - 09	RELATIONSHIP FOR COMPARISON - Must contain one of the following codes:
0		EQ - two records will be compared only if the specified substrings (of their File Codes) are equal to one another
+		_____
+		NE - two records will be compared only if the specified substrings are not equal to
+		_____
+		one another
0	11 - 11	POSITION OF SUBSTRING - Must contain the position of the substring within the File Code.
0	13 - 13	LENGTH OF SUBSTRING - Must contain the length of the substring.
0		Notes: The sum of column 11 and column 13 may not
+		_____
+		exceed 10.
0		File Codes must be assigned systematically if this card is to be used. Note that the relationship defined here will apply to all
+		_____
+		pairs of File Codes.

OPOST Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 04	PARAMETER KEYWORD - Must contain 'POST'. The POST parameter card may optionally be provided to move or accumulate information from one record to another, when the records are determined to be duplicates of one another. See the notes below for a detailed discussion of exactly which records are posted.
0		
0		
+		
0		Up to 50 POST operations may be specified.
0	05 - 06	Any value may be entered here.
0	08 - 08	TYPE OF POSTING - Must contain one of the following codes, to indicate the nature of the POSTing which is to be performed by this card: A - additive posting; a field from the second record is added to the first, regardless of the value of the two fields. M - move posting; a field from the second record is moved to a field in the first, provided that the field in the first record is blank, and the field in the second record is not blank. O - posting by logical OR; a field from the second record is ORed with a field in the first, so that each bit is left '1' if either of the corresponding bits was originally '1'. R - replace posting; a field from the second record is moved to (replaces) a field in the first, regardless of the original value of either field.
0		
0	10 - 31	FIRST RECORD DATA - These columns contain data concerning the first record, and the field within it which is to be posted. The individual elements of this "first record" data are described below.

OPOST Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	10 - 10	LOCATION OF FILE CODE SUBSTRING - May contain the location, within the File Code, of a substring which is to be compared against the data in column 14-22. If present, then columns 12 and 14-22 are required also. If omitted, columns 12 and 14-22 will not be examined.
0		If this field is present (in the card), then the first record will be posted (by this card) only if
+		the defined substring of the File Code matches the data presented in columns 14-22. If omitted, then all first records will be posted (provided that the field to be posted is blank, when column
+		8 is 'M').
0	12 - 12	LENGTH OF SUBSTRING - If column 10 is not blank, then this field must contain the length of the substring. The sum of the length and the location (column 10) may not exceed 10.
0	14 - 22	COMPARAND FOR SUBSTRING - If column 10 is not blank, then any "first" record will only be posted if the defined substring of its File Code matches the data stored here exactly. If column 10 is blank, then this field will be ignored.
0	24 - 26	LOCATION OF FIELD TO BE POSTED - Must contain the location within the record at which is the field which is to be posted.
0		If necessary, this field may be extended one position to the left, permitting the entry of a four-digit location in columns 23-26.
0	28 - 29	LENGTH OF FIRST FIELD - Must contain the length of the field to be posted (in the first record).
0		If column 8 is 'A', then this length must meet the requirements of column 31.

OPOST Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	31 - 31	FORMAT OF FIRST FIELD - If column 8 is 'A', then this column must contain a code which indicates the format of the field to be posted (the sum field). If column 8 is not 'A', then this column will be ignored. The valid codes are:
0		B - the sum field is a binary number. In this case, the length (columns 28-29) must be 1, 2, or 4. A field of length 1 will be treated as unsigned. Fields of length 2 and 4 will be treated as signed.
		C - the sum field is a number stored as an EBCDIC character string. In this case, the length above may not exceed 16. The number stored will consist of all numeric digits unless negative, in which case the rightmost digit will have a negative sign.
		P - the sum field is a packed number. In this case, the length above may not exceed 16.
0	33 - 54	SECOND RECORD DATA - These columns contain data concerning the second record, and the field within it which is to be posted to the first record. The individual elements of the "second record" data are defined below.
0	33 - 33	LOCATION OF FILE CODE SUBSTRING - May contain the location, within the File Code, of a substring which is to be compared against the data in column 37-45. If present, then columns 35 and 37-45 are required also. If omitted, columns 35 and 37-45 will not be examined.
0		If this field is present (in the card), then the posting will take place only if the specified substring of the File Code in the second record matches the data presented in columns 37-45. If omitted, then posting will always take place (provided that the field defined by columns 47-52 is not blank, if column 8 is 'M').
0	35 - 35	LENGTH OF SUBSTRING - If column 33 is not blank, then this field must contain the length of the substring. The sum of the length and the location (column 33) may not exceed 10.

OPOST Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	37 - 45	COMPARAND FOR SUBSTRING - If column 33 is not blank, then posting will only take place if the defined substring of the "second" record File Code matches the data stored here exactly. If column 33 is blank, then this field will be ignored.
0	47 - 49	LOCATION OF FIELD TO POST - Must contain the location of the field within the "second" record which is to be used to post the "first" record. If necessary, this field may be extended one position to the left, permitting the entry of a four-digit location in columns 46-49.
0	51 - 52	LENGTH OF SECOND FIELD - Must contain the length of the field (in the second record) which is used to post the first record. If column 8 is 'A', then this length must meet the requirements of column 54. If column 8 is not 'A', then this length must be the same as that entered in columns 28-29.
0	54 - 54	FORMAT OF SECOND FIELD - If column 8 is 'A', then this column must contain a code which indicates the format of the field to be posted (the sum field). If column 8 is not 'A', then this column will be ignored. The valid codes are: <ul style="list-style-type: none">B - the sum field is a binary number. In this case, the length (columns 51-52) must be 1, 2, or 4. A field of length 1 will be treated as unsigned. Fields of length 2 and 4 will be treated as signed.C - the sum field is a number stored as an EBCDIC character string. In this case, the length above may not exceed 16. The number stored will consist of all numeric digits unless negative, in which case the rightmost digit will have a negative sign.P - the sum field is a packed number. In this case, the length above may not exceed 16.

0POST Parameter Card

- CARD COLUMNS FIELD DESCRIPTION
- 56 - 56 NULL OUT SOURCE FIELD - If this column contains 'X', then when POSTing is performed the source field (in the second record) will be set to null following the POST operation. This permits multiple POST commands to be entered for the same source field (and a number of different destination fields) while ensuring that each source field will only be posted once.

0 The value stored (by the nulling operation) will depend on the type of posting (column 8) and on the format of the second-record field (column 31), as follows:

0	Column 8	Column 31	Null value stored
0	A	B	Binary zeroes
	A	C	EBCDIC zeroes
	A	P	Packed zero
	M	-	Blanks
	O	-	Binary zeroes
	R	-	Blanks

0 Note: This option is not recommended when column 8 is 'R' (replace posting).

- Execution_Summary:

0 For each duplicate group, posting is performed as follows (if POST cards are present).

0 1. The record from the duplicate group with the highest priority (lowest number) is selected. If multiple records with the same priority exist, then one is picked at random. If the group contains a Purge-file record, then a record is picked at random (Purge files and POST cards do not mix very well!). The record selected will be the "first" record for the entire duplicate group--that is, it is the record which will be posted from the remaining records of the group.

0 2. For each other record in the duplicate group, the POST cards are examined, and operated upon as required. For each POST card, the following steps are performed for each "second" record:

0 a. If the File Code of the first record does not permit

OPOST Parameter Card

Execution Summary:

- posting (per columns 10-22 of the card), then this POST card is abandoned.
 - b. If column 8 is 'M' and the result field in the first record is not blank, then this POST card is abandoned.
 - c. If the File Code of the second record does not permit posting (per columns 33-45), then this POST card is abandoned.
 - d. If column 8 is 'M' and the source field in the second record is blank, then this POST card is abandoned.
 - e. The posting operation is performed.

0 Notes:

0 If the duplicate group is to have a survivor (that is, if there are no Purge records in the group), then the record selected as the "first" record will be the survivor (unless a user exit function alters the Selection Indicator(s) of the records in memory).

0 The POST cards will be performed in the order presented.

0 Program check interruptions may occur in any of the following circumstances:

0 Either field specified is beyond the record as defined by the length in CONTRL card columns 10-12.

0 For an Add posting, an operand field specified as character or packed is not of valid form.

0 For an Add posting with a binary result field, the sum generated is larger than the capacity of a machine register (which is approximately 2.4 billion).

0 At end-of-job, a report is produced showing the number of posting operations performed for each POST card. On this report, the entire card is listed, so it may be desirable to place appropriate comments in the columns following the last used column.

0SEQUNx Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 05	PARAMETER KEYWORD - Must contain 'SEQUN'. The SEQUNx card may optionally be included to add a sequence number to each of the output name-and-address files.
0	06 - 06	OUTPUT FILE SELECTION - Must contain one of following values to identify the output file which is to be numbered: D - File OD E - File OE M - File OM
0	08 - 10	LOCATION OF SEQUENCE NUMBER IN OUTPUT - Must contain the location in output records where the sequence number is to be placed. If necessary, this field may be extended to the left one position, allowing the entry of a 4-digit number in columns 7-10.
0	12 - 12	LENGTH OF SEQUENCE NUMBER IN OUTPUT - Must contain the length of the sequence number field being placed in the output record. If the sequence number is to be a packed number this length must not exceed 5.
0	14 - 14	PACKED SEQUENCE INDICATOR - Must contain either a 'P' or 'C'. 'P' specifies that the sequence number is to be in a packed format, while 'C' represents a character format.
0	16 - 24	SEQUENCE NUMBER LIMIT - May contain a numeric value between 000000002 and 999999999 inclusive. This number defines the limit that the sequence number will have. When this number is reached, sequencing will start again from the value specified by columns 26-34.
0	26 - 34	SEQUENCE NUMBER STARTING VALUE - May contain a numeric value between 000000002 and 99999999. This number defines the starting number at which sequencing will take place. If this field is blank, sequencing will start with the number one.

OPEXITS Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'PEXITS'.
0		The PEXITS card may optionally be used to activate a user-written Exit routine at one or more points in the duplicate-identification process. A single exit routine must be constructed, and must be able to distinguish the function invoked (this way, the various functions may communicate with one another through the common memory, if necessary). See the detailed discussions following.
0	08 - 15	EXIT ROUTINE NAME - Must contain the name of the user exit routine. For OS/3 the exit routine must be linked with the main module and this field may be blank. Otherwise this is the name of the exit that will be dynamically loaded at execution time. (see User Exits for specifications)
0	17 - 17	COBOL INDICATOR (DOS) - Must contain 'C' if the User Exit is written in ANS COBOL under DOS.
0	19 - 29	EXIT ROUTINE ACTIVATORS - Each of these columns must contain a blank or the letter associated with a particular Function, as follows. Only those functions which are activated by this card will be invoked.
0		Column 19: 'A' to activate exit function A. 21: 'B' to activate exit function B. 23: 'C' to activate exit function C. 25: 'D' to activate exit function D. 27: 'L' to activate exit function L. 29: 'X' to activate exit function X.
0		Note: For this card only, the even-numbered
+		columns between 18 and 30, inclusive, must be blank. This allows the program to stop when an Exit Activator is inadvertently punched in the wrong column.
0	31 - 80	The remainder of this card may be used to pass information to the user-written Exit routine.

OPEXITS Parameter Card

+

-Additional_Notes:

- 0 1. For OS and DOS, the PEXITS exit routine is always dynamically loaded at execution time. For Univac OS/3, the exit routine must be linked with the primary program (see below).
- 0 2. Under DOS, if the Exit Routine is written in ANS COBOL, the following must be observed in order for the exit routine to "know" that it is a called routine:
 - 0 A. Create an assembler program named MNS0ADDR as follows:

```
0 MNS0ADDR START  
DC V(ILBDMNS0)  
END
```
 - 0 B. Link the exit routine with this 'program', as follows:

```
0 INCLUDE MNS0ADDR  
INCLUDE exit  
ENTRY exit
```
 - 0 C. Now the eight-byte 'program' MNS0ADDR will immediately precede the entry-point of the exit, providing its caller with the address of the flag byte to be set when an ANS COBOL exit is to be called.
 - 0 D. Place a 'C' in column 17 of the PEXITS card to cause the COBOL logic to be invoked.
- 0 3. For Univac OS/3 systems, the exit routine must be linked statical with the main program. Thereafter, it will be ignored unless an PEXITS card is present. The name of the Exit Routine must be 'USEREX'.

0User Exit Routine

General Discussion

- In some circumstances, a user-written Exit Routine must be used to accomplish a function which is outside the scope of the system as designed. There are six different points in the logic of the duplicate-identification process at which linkage has been provided to such a routine, if required. The Exit routine receives the same parameter list at each call; however, some of the parameters are only presented at certain linkage points, so a thorough understanding of the function and parameters for each linkage is necessary. One of the parameters (always present) contains a function indicator byte which indicates, by its value, which linkage is being performed and therefore which parameters are usable.

0 The same Parameter List is provided to the Exit routine at each invocation. As noted, some parameters will not be usable for certain functions. The parameters are:

0	Number	Description_and_Function_____	For_Function(s)
+			
0	1	A word in which the Exit routine may store a return code which will affect the main program's subsequent execution. See the individual functions for details.	All
	2	A 100-byte array of information for the Exit routine, as follows:	All
		01-01 A single character whose value (A, B, C, D, L, or X) indicates the function being invoked.	All
		02-02 A single character whose value indicates the timing of the call for function D: '2' indicates that a pair of duplicates is being presented, and '9' indicates that the previous duplicate-group has no more records.	D
		03-100 (reserved)	
	3	An input record, in its original form (that is, the 208-byte appendage is never included, regardless of the type of run being performed).	A, B, C
	4	The first record of a duplicate group, or the first of two records which are to be compared. In either case, the area presented does include the 208-byte appendage.	D, L
	5	The next record of a duplicate group, or the second of two records which are	D, L

	to be compared. Same form as parameter 4.	
6	A 133-byte area in which a print line may be constructed.	X
7	The 80-character PEXITS card image.	All

0User Exit Routine
 0Skeletal Example

+ _____
 - The following skeletal example Exit Routine illustrates the use
 of the parameter list, and the logic required to execute each functio
 at the proper time.

```

0      USRX      TITLE 'USER EXIT ROUTINE FOR CM30'
      USEREX    START
              STM  R14,R12,12(R13)
              BALR R12,0
              USING *,R12
              ST   R13,SAVE+4
              LA   R13,SAVE
              MVC  PARAMS,0(R1)
              L    R1,AMODIF
              CLI  0(R1),C'B'
              BL   FUNCTA
              BE   FUNCTB
              CLI  0(R1),C'D'
              BL   FUNCTC
              BE   FUNCTD
              CLI  0(R1),C'L'
              BE   FUNCTL
              CLI  0(R1),C'X'
              BE   FUNCTX

*      INVALID FUNCTION -- RETURN 0
RETURN0  L      R1,ARTRNCD
         MVC    0(4,R1),=F'0'
         B      RETURN
RETURN4  L      R1,ARTRNCD
         MVC    0(4,R1),=F'4'
         B      RETURN
RETURN8  L      R1,ARTRNCD
         MVC    0(4,R1),=F'8'
RETURN   L      R13,4(R13)
         LM     R14,R12,12(R13)
         SR     R15,R15
         BR     R14

REGS                                REGISTER EQUATES
.
.      (FUNCTA, FUNCTB, ETC.)
.

SAVE     DS     9D                    SAVE AREA
PARAMS   DS     0XL28                 PARAMETER ADDRESS LAYOUT
ARTRNCD  DS     A                      -> RETURN CODE          (ALL)
AMODIF   DS     A                      -> 100-BYTE MODIFIER AREA (ALL)
AINPREC  DS     A                      -> INP RECORD            (A/B/C)
ADUPL    DS     A                      -> 1ST OF TWO RECORDS   (D/L/P/Q)
    
```

ADUPN	DS	A	-> SECOND OF TWO RECORDS	(D/L/P/Q)
APLINE	DS	A	-> 133-BYTE PRINT LINE	(X)
APEXITS	DS	A	-> PEXITS CARD IMAGE	(ALL)
	END	USEREX		

0User Exit Routine

0Functions A, B, and C

- Functions A, B, and C receive control immediately following the read of each record from Files IA, IB, and IC respectively. Only parameters 1, 2, 3, and 7 are usable. By means of this function, a user may modify or reject the records as they are read, before the extraction of any data by the duplicate identification program.

0 Return Code values (parameter 1) for functions A-C are:

0 0 - continue processing the record normally

4 - reject this record (do not process it further)

- Sample_Function_B:

0 The following code (which is a continuation of our skeletal sample Exit Routine) illustrates a Function B routine whose job is to reject all records from File IB which contain 'XX' in positions 38-39.

-
FUNCTB DS 0H
L R1,AINPREC
CLC 37(2,R1),=C'XX'
BE RETURN4
B RETURN0

0User Exit Routine
0Function D

+ _____
- Function D receives control when a duplicate group has been identified and completed. It will be invoked once for each record other than the first in the group, and again at the end of the group. For all these invocations, parameter 4 will be the first record of the group; for all invocations except the last (end-of-group), parameter 5 will be the (next) other record in the duplicate pair.
0 The value of the second byte of parameter 2 will indicate the timing of each invocation to Function D, as follows:
0 '2' indicates that a duplicate pair is being presented
0 '9' indicates the end-of-group call
0 Parameters 3 and 6 are not usable by Function D, and parameter 5 is not usable for the end-of-group invocation.
0 For each duplicate group, all duplicate-pair calls will precede the end-of-group call.
0 The return code has no effect for Function D.
- Sample_Function_D:
0 The following code illustrates a Function D which adds a packed number in positions 124-128 of the (original) records from a non-survivor into the survivor of a group. Note that the presence of the 208-byte appended area (in memory) must be allowed for.
- FUNCTD DS 0H
 LM R5,R8,AMODIF MODIFIERS THRU DUPN
 CLI 1(R5),C'9'
 BE RETURN0 NO ACTION AT END-OF-GROUP
 CLI 35(R7),C'0' SEE IF FIRST REC SURVIVES
 BNL RETURN0 NO ACTION IF IT DOES NOT
 AP 208+123(5,R7),208+123(5,R8) ADD THE FIELD
 B RETURN0 THAT'S IT

0User Exit Routine

0Function L

- Function L receives control just before each pair of records is compared (that is, before the action of any CMPFLD cards, and before the program-controlled comparison). It can be used to determine that the two records positively are duplicates, or that they positively are not duplicates. Parameter 4 will be the first of the two records to be compared, and parameter 5 will be the second.

0 Parameters 3 and 6 are not usable by Function L.

0 The return code (parameter 1) will be honored as follows (all values for the return code must be stored in binary):

0 0 - no determination could be made (by the Exit routine) as to whether or not the two records are duplicates; the main program must compare them.

4 - the two records are not duplicates; no further comparison need be performed.

8 - the two records are duplicates; no further comparison need be performed.

- Sample_Function_L:

0 The following code (continuing our skeletal example) applies the following logic (ahead of the normal record comparison): If either record contains the word 'SEED' in position 100, then the two records do not match. Note that the routine must allow for the 208-byte area which is present at the front of each record.

```
- FUNCTL DS 0H
      LM R7,R8,ADUP1 FIRST AND SECOND RECORD
      CLC 208+99(4,R7),=C'SEED'
      BE RETURN4
      CLC 208+99(4,R8),=C'SEED'
      BE RETURN4
      B RETURN0
```

0User Exit Routine

0Function X

+ _____
- Function X is invoked at end-of-job, after all other processing
(and printing) has been completed by the main program. It can
be used to print a report based on values accumulated in memory
by the other functions, or to close additional files which may
have been opened by the other functions.

0 Parameters 3, 4, and 5 are not usable by Function X.

0 Line(s) to be printed, if any, may be prepared in parameter 6.
By storing values in the return code (parameter 1), multiple
lines may be formatted (by Function X) and printed (by the main
program), as follows:

0 0 - do not print the line, and do not re-invoke Function X

+ _____

4 - print the line stored in parameter 6, and then invoke
Function X again

8 - print the line stored in parameter 6, but do not invoke

+ _____

Function X again

0 Note that if an attractive report is to be produced via Function X,
then it must supply headers at the appropriate time(s).

- Sample_Function_X:

+ _____

0 The following code illustrates a (rather primitive) Function X which
must print ten lines of information (which presumably were stored
by other Functions).

- FUNCTX DS 0H
L R8,APLINE
L R7,ANXTLINE
C R7,=A(LINEX)
BE RETURN0
MVC 0(133,R8),0(R7)
LA R7,133(R7)
ST R7,ANXTLINE
B RETURN4
ANXTLINE DC A(LINES) -> NEXT LINE TO PRINT
LINES DS 10CL133 DATA TO PRINT
LINEX EQU *

1Group 1 Software -- Consumer Merge/Purge 3.95
Program CM30

0User Exit Routine
0General Information

+-----
- Return Code values must be stored as a binary fullword in parameter
1 (PIC S9(9) COMP for COBOL programs). If a value other than one
of those noted is stored, the main program will interpret it as
zero.

1
1Group 1 Software -- Consumer Merge/Purge 3.97
Program CM30

0INTRPT Parameter Card

CARD COLUMNS	FIELD DESCRIPTION
01 - 06	PARAMETER KEYWORD - Must contain 'INTRPT'.
0	This parameter card permits the program to be interruptible by the operator.
0	For DOS systems when this card is used ATTENTION will cause the display of the message below. For OS systems, the presence of this card will cause the WTOR described below to be issued. Under OS/3 this card will allow an unsolicited message to be routed to the program; such a message should be one of the responses described below.
0	The message issued is:
0	TO HALT REPLY: NOW/ZIP/SCF/STE
0	The responses are treated as follows:
0	NOW - this response is treated the same as an immediate end-of-file on all input files.
	ZIP - this response is treated the same as the detection of end-of-file on all input files at the end of the current ZIP Code.
	SCF - this response is treated the same as the detection of end-of-file on all input files at the end of the current SCF (3-digit ZIP)
	STE - this response is treated the same as the detection of end-of-file on all input files at the end of the current state.
	other - in all systems, the following message will first be displayed:
0	UNRECOGNIZED OPTION IGNORED
0	Then, in DOS and OS/3 systems, processing will continue until the next ATTENTION. In OS systems, the WTOR described above will be repeated.

0CHKPT Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'CHKPT'.
0	08 - 14	CHECKPOINT COUNT - The checkpoint parameter card activates the IBM checkpoint facility. If this option is used, checkpoints will be taken for every 'n' input records processed. This field must contain a number which will be used as 'n'. The value must be numeric and greater than one, entered with leading zeroes as required.
0	68 - 71	(DOS only) DEVICE - May contain the device to which the checkpoint file is to be assigned. See the TAPExx card for a list of available device types. The default is 2314.
-	OS_CHECKPOINT-RESTART	
+	<hr/>	
0	1.	A DD-statement (DDNAME=SYS001) must be provided for the checkpoint dataset. It must specify a physical-sequential dataset which may reside on either tape or disk.
0	2.	If the DISP parameter on the SYS001 DD does not specify MOD, then:
0	A.	Each checkpoint will be written over the preceding checkpoint; and, therefore:
0	B.	If a checkpoint cannot be taken successfully, there will be no checkpoint available for restart until another checkpoint is successfully taken.
0	3.	Restart as specified in the JCL manual.

0CHKPT Parameter Card

- CARD COLUMNS FIELD DESCRIPTION
- DOS_CHECKPOINT-RESTART

+

- If checkpoint processing is being performed (i.e.,
the 'CHKPT' parameter is present), the following
additional JCL is needed:

0 // ASSGN SYS001,X'nnn'
// DLBL DTFPHM,'CHECKPOINT',0010,SD
// EXTENT SYS001,WORK02,1,0,20,200

0 If the job must be restarted from a checkpoint, then
the following card replaces the '// EXEC' card
in the run deck. No parameter cards need to be entered:

0 // RSTRT SYS001,nnnn,DTFPHM

0 ...Where 'nnnn' is the Restart or Checkpoint
number the user wishes to restart from.

0Operating Instructions

- PROGRAM FUNCTION: DUPLICATE ELIMINATION PROGRAM NAME: CM30
 0 SYS NR/ DEVICE I/O
 DDNAME ADDRESS FILE DESCRIPTION/LABEL RET. FILE ID DISPOSITIO

+
 T
 A SYS010 I Name and Address File A TAPEIA
 P SYS011 I Name and Address File B TAPEIB
 E SYS012 I Name and Address File C TAPEIC

 o SYS015 O Duplicate Names File TAPEOD DE50
 r SYS016 O Extra file TAPEOE as req'd
 SYS017 O Mail File TAPEOM as req'd
 D
 I SYS018 O Dupe-Elim Summary Stats TAPEOT DE40/DE50
 S SYS019 O State-Counts by File Code TAPEOX Sort/DE41
 K

U
 N
 I
 T
 S

0 SYS NR/
 DDNAME DESCRIPTION OF CARD DISPOSITIO

+
 0 R SYS004 Parameter Cards

P
 0 SYS NR/ FORM CARRIAGE
 DDNAME DESCRIPTION OF REPORT NUMBER TAPE DISPOSITIO

+
 P
 R SYS005 Control Totals As Reqd. As Reqd.

I
 N
 T
 E
 R

0 MESSAGE RESPONSE
 +

C
 O
 N
 S
 O
 L
 E

0Sample OS JCL

```
- //CM30      EXEC   PGM=CM30
//SYS005    DD     SYSOUT=A,DCB=(RECFM=FBA,BLKSIZE=133,LRECL=133)
//SYS010    DD     DSN=INPUT.NAME.ADDRESS.TAPEIA,
//           UNIT=TAPE,VOL=SER=TAPEIA,DISP=OLD,LABEL=(2,BLP),
//           DCB=(DSORG=PS,RECFM=FB,LRECL=rrrr,BLKSIZE=bbbb)
//SYS011    DD     DSN=INPUT.NAME.ADDRESS.TAPEIB,
//           UNIT=TAPE,VOL=SER=TAPEIB,DISP=OLD,LABEL=(2,BLP),
//           DCB=(DSORG=PS,RECFM=FB,LRECL=rrrr,BLKSIZE=bbbb)
//SYS012    DD     DSN=INPUT.NAME.ADDRESS.TAPEIC,
//           UNIT=TAPE,VOL=SER=TAPEIC,DISP=OLD,LABEL=(2,BLP),
//           DCB=(DSORG=PS,RECFM=FB,LRECL=rrrr,BLKSIZE=bbbb)
//SYS015    DD     DSN=OUTPUT.DUP.NAMES.TAPEOD,
//           UNIT=TAPE,VOL=SER=TAPEOD,DISP=OLD,
//           DCB=(DSORG=PS,RECFM=FB,LRECL=rrrr,BLKSIZE=bbbb)
//SYS016    DD     DSN=OUTPUT.EXTRA.FILE.TAPEOE,
//           UNIT=TAPE,VOL=SER=TAPEOE,DISP=OLD,
//           DCB=(DSORG=PS,RECFM=FB,LRECL=rrrr,BLKSIZE=bbbb)
//SYS017    DD     DSN=OUTPUT.MAIL.FILE.TAPEOM,
//           UNIT=TAPE,VOL=SER=TAPEOM,DISP=OLD,
//           DCB=(DSORG=PS,RECFM=FB,LRECL=rrrr,BLKSIZE=bbbb)
//SYS018    DD     DSN=OUTPUT.DUPE.ELIM.SUMMARY.STATS.TAPEOT,
//           UNIT=TAPE,VOL=SER=TAPEOT,DISP=OLD,
//           DCB=(DSORG=PS,RECFM=FB,LRECL=0080,BLKSIZE=bbbb)
//SYS019    DD     DSN=OUTPUT.STATE.COUNTS.BY.FILE.CODE.TAPEOX,
//           UNIT=TAPE,VOL=SER=TAPEOX,DISP=OLD,
//           DCB=(DSORG=PS,RECFM=FB,LRECL=0092,BLKSIZE=bbbb)
//SYSUDUMP  DD     SYSOUT=A
//SYS004    DD     *
```

```
.
.
parameter cards
```

```
/*
```

0Sample DOS JCL

```
- // JOB CM30
  // TLBL TAPEIA          NAME AND ADDRESS FILE A
  // TLBL TAPEIB          NAME AND ADDRESS FILE B
  // TLBL TAPEIC          NAME AND ADDRESS FILE C
  // TLBL TAPEOD          DUPLICATE NAMES FILE
  // TLBL TAPEOE          EXTRA FILE
  // TLBL TAPEOM          MAIL FILE
  // TLBL TAPEOT          DUPLICATE ELIMINATION SUMMARY STATS
  // TLBL TAPEOX          STATE COUNTS BY FILE CODE
  // ASSGN SYS010,X'180'  NAME AND ADDRESS FILE A
  // ASSGN SYS011,X'181'  NAME AND ADDRESS FILE B
  // ASSGN SYS012,X'182'  NAME AND ADDRESS FILE C
  // ASSGN SYS015,X'183'  DUPLICATE NAMES FILE
  // ASSGN SYS016,X'184'  EXTRA FILE
  // ASSGN SYS017,X'185'  MAIL FILE
  // ASSGN SYS018,X'186'  DUPLICATE ELIMINATION SUMMARY STATS
  // ASSGN SYS019,X'187'  STATE COUNTS BY FILE CODE
  // ASSGN SYS004,X'00C'  CARD READER
  // ASSGN SYS005,X'00E'  PRINTER
  // EXEC CM30
      .
      .
parameter cards
      .
      .
/*
```


0

Second Line:

IN Individual Name
PT Professional Title
FN Formatted Name

OHEADER Parameter

-	COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'HEADER'.
0	08 - 17	DATE - May contain a date in the form MM/DD/YYYY. This date will appear on the output reports. If this field is left blank, the current date will be inserted.
0	19 - 58	RUN DESCRIPTION - May contain any run description the user wishes. This run description will appear on the output reports.
0	60 - 65	START POSTAL CODE - May contain a Postal Code, left-justified, at which the user wishes to start processing the input data. All records with a Postal Code lower than the Postal Code in this field will not be processed.
0	67 - 72	STOP POSTAL CODE - May contain a Postal Code, left-justified, after which the user wishes to stop processing. When a Postal Code of higher value than that which is entered here is encountered, the program will act as if it has reached end of file.
0		Note: The START POSTAL CODE and the STOP
+		_____
		POSTAL CODE fields may be both left blank. If they are both blank, all input records will be processed.
0		Column 66 must be either blank or a dash ('-'), to ensure that processing will not continue when the STOP POSTAL CODE is inadvertently entered one column to the left, in columns 66-71 instead of 67-72.

1Group 1 Software -- Consumer Merge/Purge 4.5
Program CM31

0TAPExx Parameter Card

- TAPExx or DISKxx cards must be used to activate each input or output file. Following is a brief description of each of the possible files. See the System Overview section for a detailed description of the TAPExx/DISKxx card fields.

- File_Activators:

+
0 IE - Input Extracted Names File

1

1Group 1 Software -- Consumer Merge/Purge 4.7
Program CM31

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'CONTRL'.
0	08 - 10	PAGE NUMBER POSITION - May contain the position, within the first header line that a five-digit page number will be printed.

1

1Group 1 Software -- Consumer Merge/Purge 4.9
Program CM31

0HEADxx Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 04	PARAMETER KEYWORD - Must contain 'HEAD'.
0		This card can be used to define up to three lines of headings for the report.
0	05 - 06	HEADING LINE DESIGNATOR - Must contain one of the following codes to represent the Heading line designator:
0		1A = left side, heading line 1. 1B = right side, heading line 1. 2A = left side, heading line 2. 2B = right side, heading line 2. 3A = left side, heading line 3. 3B = right side, heading line 3.
0		Left side represents print positions 1-66. Right side represents print positions 67-132.
0	08 - 73	DATA TO PRINT - Must contain the data that is to be printed in the corresponding print positions for each side of each heading line

- Note: When the program encounters the first occurrence of the keyword 'MM/DD/YYYY' in the 'HEADxx' parameter cards, it will replace it with the value specified in columns 8 - 17 of the 'HEADER' parameter. If columns 8 - 17 are left blank, then it will use the current date. If the keyword 'MM/DD/YYYY' is not found in the 'HEADxx' parameter cards, then nothing will be altered.

1Group 1 Software -- Consumer Merge/Purge 4.11
 Program CM31

0CONSTn Parameter Card

- CARD COLUMNS	FIELD DESCRIPTION
- 01 - 05	PARAMETER KEYWORD - Must contain 'CONST'. This parameter is used to define up to 9 constants to the program which can be referenced in the MOVExx, and FORMAT parameter cards as C01, C02, ..., C09.
0 06 - 06	CONSTANT NUMBER - Must contain a numeric value between 1 and 9 inclusive. This number distinguishes the difference among the 9 possible CONSTn parameter cards that may be used in the program.
0 08 - 17	CONSTANT VALUE - Must contain the value, left justified to column 8, of the CONSTn. There is no length implied here except that the maximum size for any CONSTn is 10 positions. The length of the CONSTn is specified when using the MOVExx or FORMAT parameter card(s).

1

1Group 1 Software -- Consumer Merge/Purge 4.13
 Program CM31

0MOVExx Parameter Card

- CARD COLUMNS	FIELD DESCRIPTION
- 01 - 04	PARAMETER KEYWORD - Must contain 'MOVE'.
0 05 - 06	Any value may be entered here.
0 08 - 10	DESTINATION - Must contain the position in the print line to which the data is to be moved.
0 12 - 13	LENGTH - Must contain the length of the data to be moved. This must be a numeric value in the range 01 to 99 inclusive.
0 15 - 17	SOURCE - Must contain one of the following: (1) The starting position on the input record from which data is to be moved. (2) A CONSTANT: C01, C02, ..., C08.
0	Note: if necessary, this field may be extended

+

_____ one position to the left, permitting the entry of a four-digit number in columns 14-17.

0 General: 'MOVExx' parameter cards are used to move data from

+

_____ the input record or constant values defined by the user to the output report. Up to 100 MOVExx parameter cards may be entered and are executed in the order presented.

0FORMAT Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'FORMAT'. The parameter strings together fields and moves the entire string to the Output Area specified. See below for further discussion of this parameter
0	07 - 08	COMMENT - These columns are ignored by the program and may be used for comment.
0	10 - 12	PRINT LOCATION - Must contain the starting position in the print line where the formatted string will be placed. The value entered must be a numeric value between 001 and 132.
0	14 - 15	LENGTH OF RESULTING STRING - Must contain the length of the Area defined in columns 10-12. This must be a non-zero numeric value.
0	17 - 19 24 - 26 31 - 33 38 - 40 45 - 47 52 - 54 59 - 61 66 - 68 73 - 75	LOCATION OF FIELDS TO BE FORMATTED - Must contain the starting positions of 2 to 9 fields in the Input Area that contain the fields to be formatted. The fields are formatted left to right. Columns 17-19 and columns 24-26 must be entered. The first blanks encountered will terminate processing of the card. Note: If necessary, this field may be extended to the left one position, allowing the entry of a 4-digit number in columns 16-19, 23-26, etc.
0	20 - 20 27 - 27 34 - 34 41 - 41 48 - 48 55 - 55 62 - 62 69 - 69 76 - 76	CONSTANT TREATMENT - Must contain one of the following to control the formatting of constants: '*' - constant will not appear if the Preceding field contains all blanks. blank- constant will not appear if the Succeeding field contains all blanks. See example below.
0	21 - 22 28 - 29 35 - 36 42 - 43 49 - 50 56 - 57 63 - 64 70 - 71 77 - 78	LENGTH OF FIELDS TO BE FORMATTED - Must contain the corresponding lengths of the fields defined in columns 17-19, etc.

0FORMAT Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	23 - 23	ZERO SUPPRESSION - Must contain a 'Z' if suppressio
	30 - 30	of leading zeros is desired in the corresponding
	37 - 37	field defined in columns 17-19, etc.
	44 - 44	
	51 - 51	Note that this 'Z' may conflict with a 4-digit
	58 - 58	position, if needed for the following field.
	65 - 65	
	72 - 72	
	79 - 79	

0 General: Following are a number of rules and restrictions that apply t

+

- _____
- the FORMAT parameters:
- 0 (1) FORMAT parameters will be executed in the order presented.
 - 0 (2) The first field to be formatted must not be a constant.
 - 0 (3) There must be at least two fields to be formatted.
 - 0 (4) Two consecutive non-constant fields will automatically have a space placed between them.
 - 0 (5) All leading blanks of non-constant fields will be suppressed and will not appear in the output string.
 - 0 (6) No spaces will appear between constants and variables unless allowed for in defining the constant.

0 Example: FORMAT01 017 25 100 15 C01 02 115 04 C02*01

+

CONSTANT01 ,
CONSTANT02 .

0 If 100 contains SMITH and 115 contains JR -
the output will contain 'SMITH, JR.'

0 If 100 contains SMITH and 115 contains blanks -
the output will contain 'SMITH'.

0UNPKxx Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 04	PARAMETER KEYWORD - Must contain 'UNPK'.
0	05 - 06	Any value may be entered here.
0	08 - 10	PRINT POSITION - Must contain the position in the print line to which the data is to be unpacked.
0	12 - 13	LENGTH IN PRINT LINE - Must contain the length of the unpacked data. This length may not exceed 16.
0	15 - 17	LOCATION OF SOURCE FIELD - Must contain the starting location in the input record of the field that is to be unpacked.
0		Note: If necessary, this field may be extended
+		_____
		one position to the left, permitting the entry of a four-digit number in columns 14-17.
0	19 - 20	LENGTH OF SOURCE FIELD - Must contain the length of the field to be unpacked. This length may not exceed 16.
0		Note: Up to 100 UNPKxx parameter cards may be entered.

0RECLIM Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'RECLIM'. The RECLIM card may optionally be used to limit the records processed for a particular run. A number of records to be skipped may be specified, as well as a number of records to process.
0		The most frequent use of the RECLIM card is to restrict the file to a small number of records, for parameter checkout or calibration. It can also be used, of course, to break what would otherwise be an excessively large job into several smaller independent executions without dividing the input files.
0	08 - 16	RECORDS TO SKIP - May contain the number of records to skip before processing begins. If omitted, processing will begin with the first record. Note: This option will only function as indicated
0		_____
+		when a single input file is used.
+		_____
0	18 - 26	RECORDS TO PROCESS - May contain the number of records to be processed. If omitted, all records will be processed (once the specified number are skipped).
-	Notes:	
+	_____	
0		The RECLIM card is applied to the records as they are read from the input file, before the Start/Stop Postal Code is applied (from the HEADER), before the action of the FLCSEL/FLCREJ cards, and before the application of the NTHSEL card.

0FLCSEL/FLCREJ Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'FLCSEL/FLCREJ'. 0 FLCSEL or FLCREJ parameter cards may optionally be provided to select only certain records, or to reject certain records, based on a substring of the File Code. When a record is read and FLCSE or FLCREJ cards are present, and that record is not selected or is rejected, then that record is bypassed.
0	08 - 08	FILE CODE SUBSTRING POSITION - Must contain the starting position of the substring within the File Code. 0 Each FLCSEL/FLCREJ parameter card must specify a substring position and length, but the positions and lengths need not be the same from card to card.
0	10 - 10	LENGTH OF SUBSTRING - Must contain the length of the substring. 0 The sum of column 8 and column 10 may not exceed 10.
0	12 - 20 22 - 30 32 - 40 42 - 50 52 - 60 62 - 70	SUBSTRING COMPARAND VALUES - Must contain the value that the substring will be compared to for selection or rejection. A blank File Code will be selected/rejected only if columns 12-20 contain blanks. Blanks in any other of these card columns will terminate processing of this parameter card.

0 Notes:
0 If any FLCSEL parameter cards are present, then FLCREJ parameter cards are not permitted, and vice versa.
0 The maximum number of Substring Comparand Values which can be entered in all cards combined is 100.
0 The FLCSEL or FLCREJ cards are applied after the action of the RECLIM card, if present, and before the action of the NTHSEL card.

ONTHSEL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'NTHSEL'.
0		Note: There are two possible ways of specifying an Nth record include or exclude. In the first method presented, the idea is to specify the Nth number and have the program include or exclude each Nth record. This method is particularly suited to sampling operations.
0		The second method is used when a specific number of records must be processed, and when the exact count of the input file is known.
0		Method 1: Sampling
0	08 - 10	NTH NUMBER - Must contain the "Nth" number of records the user wishes to either include or exclude from the records to be printed. This may be a number between 002 and 999 inclusive.
0	12 - 14	INC/EXC - Must contain either 'INC' or 'EXC' to either INclude or EXclude the Nth number of records specified in columns 8-10 of this card.
0		Method 2: Precise number of records
0	08 - 14	FRACTION OF RECORDS - Must contain the fraction of the total input desired, expressed as a seven-digit positive number with a decimal point understood at column 7.
0		Example: Suppose that the input file is known to have exactly 153,714 records, and that our problem is to print exactly 3000 records. Using a calculator, we find that $3000/153714 = 0.0195168$; therefore we enter '0195168' in columns 8-14 and out come the 3000 records.
-	Notes:	
+	_____	
0		The NTHSEL limitation will be applied to the records after the action of the RECLIM or FLCSEL/FLCREJ cards.

0Operating Instructions

- PROGRAM FUNCTION: EXTRACTED FILE PRINT PROGRAM NAME: CM31
0 SYS NR/ DEVICE I/O
DDNAME ADDRESS FILE DESCRIPTION/LABEL RET. FILE ID DISPOSITIO

+
T
A SYS010 I Extracted File TAPEIE
P
E

o
r

D
I
S
K

U
N
I
T
S

0 SYS NR/
DDNAME DESCRIPTION OF CARD DISPOSITIO

+
0 R SYS004 Parameter Cards
P

0 SYS NR/ FORM CARRIAGE
DDNAME DESCRIPTION OF REPORT NUMBER TAPE DISPOSITIO

+
P
R SYS005 Parameter List As Reqd. As Reqd.
I Extracted Record Print
N Control Totals
T
E
R

0 MESSAGE RESPONSE
+

C
O
N
S
O
L
E

1Group 1 Software -- Consumer Merge/Purge 4.27
Program CM31

0Sample OS JCL

```
- //STEP1 EXEC PGM=CM31
//SYS005 DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=133)
//SYS010 DD DSN=EXTRACTD.RECORDS,
// UNIT=TAPE,VOL=SER=TAPEIE,DISP=OLD,LABEL=(2,BLP),
// DCB=(DSORG=PS,RECFM=FB,LRECL=rrrrr,BLKSIZE=bbbbbb)
//SYSUDUMP DD SYSOUT=A
//SYS004 DD *
```

.
.
Parameter Cards

.
.
/*

1Group 1 Software -- Consumer Merge/Purge 4.28
Program CM31

0Sample DOS JCL

```
- // JOB CM31
// TLBL TAPEIE
// ASSGN SYS010,X'180' (Input TAPEIE)
// ASSGN SYS005,X'00E' (Printer)
// ASSGN SYS004,X'00C' (Card reader)
// EXEC CM31
```

.
.
Parameter Cards

.
.
/*
/&

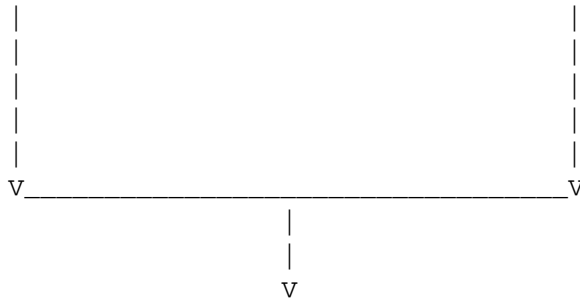
0Narrative

- Narrative_and_Flow:

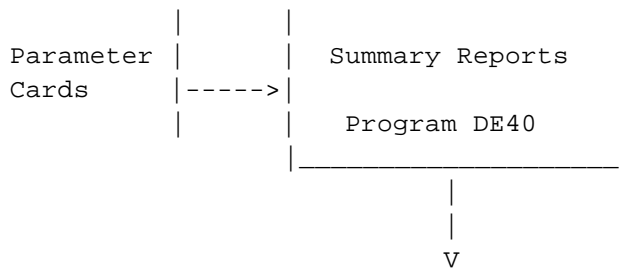
+

```

-      Duplicate-Elimination          List Conversion
      Summary Statistics             Summary Statistics
      File_IT_____                File_IS_(optional)___
+      _____                    _____
  
```



+



+

```

_____
Summary Reports by
File Code and by
List Broker
  
```

0 A Duplicate Elimination Summary report by File Code is produced. If BROKER and BRNAME cards are present (identifying various List Brokers as the source for one or more of the File Codes), then a Summary report in the same format is produced for each List Broker.

0 The report displays for each File Code:
 The File Code, relative priority, special treatment options, and the description
 The number of List Conversion input records and rejects
 The number of input records to the Duplicate Elimination step, the number of drops by category, the number of Single-buyer and Multi-Buyer records retained, and the total number of output records.

1Group 1 Software -- Consumer Merge/Purge 5.3
Program DE40

OHEADER Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'HEADER'.
0	08 - 17	DATE - May contain a date in the form MM/DD/YYYY. This date will appear on the output reports. If this field is left blank, the current date will be inserted.
0	19 - 58	RUN DESCRIPTION - May contain any run description the user wishes. This run description will appear on the output reports.

1

1Group 1 Software -- Consumer Merge/Purge 5.5
Program DE40

OTAPExx Parameter Card

- TAPExx or DISKxx cards must be used to activate each input or output file. Following is a brief description of each of the possible files. See the System Overview section for a detailed description of the TAPExx/DISKxx card fields.

- File_Activators:

+

0 IT - Input Duplicate Elimination Summary Statistics

0 IS - Input List Conversion Summary Statistics (optional)

- Note: If File IS is omitted, then the List Conversion input

+

count (on the report) will be the same as the Duplicate Elimination input, and the List Conversion reject count will be zero.

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'CONTRL'.
0	08 - 08	SUPPRESS PURGE FILES - Must be blank (normal), or contain 'X' to indicate that File Codes which are identified as Purge files are not to be printed on the reports.
0	10 - 10	SUPPRESS FILES WITH NO ACTIVITY - Must be blank (normal), or contain 'X' to indicate that File Codes, for which the Dupe-Elim input count and the Dupe-Elim output count are both zero, are not to be printed on the reports.
0	12 - 12	SEQUENCE OF FILE CODES - Must contain one of the following values to indicate the desired sequence for the reports produced: blank - print the File Codes in the order in which the records appear on the input File IT (normally, this will be the same as the order in which the FLCnnn cards were presented to the duplicate elimination program). 'A' - print the File Codes in alphabetical order by File Code 'P' - list by the relative priority (major), and File Code (minor) 'D' - list by File Code Description (major), relative priority (intermediate), and File Code (minor)
0		

0BROKER Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'BROKER'.
0		BROKER cards may optionally be used to assign various File Codes to a List Broker, with the intent of printing a Summary Report by Broker. Actual report production for a given Broker is triggered by the presence of a BRNAME card for the Broker, and not by the presence of a BROKER card for the Broker. (See also the BRNAME card.)
0	08 - 16	BROKER CODE - Must contain the Broker Code of the List Broker to which the following File Codes are to be assigned.
0	18 - 26	FILE CODE(S) - Must contain one or more File Codes which are to be assigned to the Broker Code entered in columns 8-16. The first blank File Code field encountered terminates processing of the card.
	28 - 36	
	38 - 46	
	48 - 56	
	58 - 66	
0		Each File Code may be assigned only once. File Codes which do not occur in the dupe elimination run may be assigned to Brokers through the BROKER card, but will not appear on the Broker Summary.

OBRNAME Parameter Card

	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'BRNAME'. The BRNAME card may optionally be used to associate a 'Broker name' with a particular Broker Code (see also the BROKER card). The presence of a BRNAME card for a BROKER Code will cause the production of a Summary by File Code of that Broker's files. A BRNAME card for a Broker Code to which no File Codes (which were used) are assigned will have no effect.
0	08 - 16	BROKER CODE - Must contain the Broker Code assigned to the Broker for which a report is to be produced.
0	18 - 37	BROKER NAME - Must contain the name of the List Broker. The value entered here will be printed in the header of the Broker Summary printed as a result of this card.

0FLCSEL/FLCREJ Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'FLCSEL/FLCREJ'. 0 FLCSEL or FLCREJ parameter cards may optionally be provided to select only certain records, or to reject certain records, based on a substring of the File Code. When a record is read and FLCSE or FLCREJ cards are present, and that record is no selected or is rejected, then that record is bypassed.
0	08 - 08	FILE CODE SUBSTRING POSITION - Must contain the starting position of the substring within the File Code. 0 Each FLCSEL/FLCREJ parameter card must specify a substring position and length, but the positions and lengths need not be the same from card to card.
0	10 - 10	LENGTH OF SUBSTRING - Must contain the length of the substring. 0 The sum of column 8 and column 10 may not exceed 10.
0	12 - 20 22 - 30 32 - 40 42 - 50 52 - 60 62 - 70	SUBSTRING COMPARAND VALUES - Must contain the value that the substring will be compared to for selection or rejection. A blank File Code will be selected/rejected only if columns 12-20 contain blanks. Blanks in any other of these card columns will terminate processing of this parameter card.

0 Notes:

+
0 _____
If any FLCSEL parameter cards are present, then FLCREJ parameter cards are not permitted, and vice versa.

+
0 _____
The maximum number of Substring Comparand Values which can be entered in all cards combined is 100.

0Operating Instructions

- PROGRAM FUNCTION: DUPE-ELIM SUMMARY REPORT PROGRAM NAME: DE40
 0 SYS NR/ DEVICE I/O
 DDNAME ADDRESS FILE DESCRIPTION/LABEL RET. FILE ID DISPOSITIO

+
 T
 A SYS010 I Dupe-Elim Summary Stats TAPEIT
 P
 E SYS011 I Conversion Summary Stats TAPEIS
 (optional)

o
 r

D
 I
 S
 K

U
 N
 I
 T
 S

0 SYS NR/
 DDNAME DESCRIPTION OF CARD DISPOSITIO

+
 0 R SYS004 Parameter Cards
 P

0 SYS NR/ FORM CARRIAGE
 DDNAME DESCRIPTION OF REPORT NUMBER TAPE DISPOSITIO

+
 P
 R SYS005 Parameter List As Reqd. As Reqd.
 I Summary Report by File Code
 N Summary Report by List Broker

T
 E
 R

0 MESSAGE RESPONSE
 +

C
 O
 N
 S
 O
 L
 E

1Group 1 Software -- Consumer Merge/Purge 5.15
Program DE40

0Sample OS JCL

```
- //STEP1 EXEC PGM=DE40
//SYS005 DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=133)
//SYS010 DD DSN=DUPELIM.SUMMARY.STATS,
// UNIT=TAPE,VOL=SER=TAPEIT,DISP=OLD,LABEL=(2,BLP),
// DCB=(DSORG=PS,RECFM=FB,LRECL=080,BLKSIZE=bbbb)
//SYS011 DD DSN=CONVERSN.SUMMARY.STATS,
// UNIT=TAPE,VOL=SER=TAPEIS,DISP=OLD,LABEL=(2,BLP),
// DCB=(DSORG=PS,RECFM=FB,LRECL=080,BLKSIZE=bbbb)
//SYSUDUMP DD SYSOUT=A
//SYS004 DD *
```

.
.
Parameter Cards

/*

1Group 1 Software -- Consumer Merge/Purge 5.16
Program DE40

0Sample DOS JCL

```
- // JOB DE40
// TLBL TAPEIT
// TLBL TAPEIS
// ASSGN SYS010,X'180' (Input TAPEIT)
// ASSGN SYS011,X'181' (Input TAPEIS)
// ASSGN SYS005,X'00E' (Printer)
// ASSGN SYS004,X'00C' (Card reader)
// EXEC DE40
```

.
.
Parameter Cards

/*

/&

0Narrative

- In all the above report formats, you have the option of using the State (or Province) abbreviation or the SCF (or FSA) Range in the sortation, and for the report. Using the SCF Range will cause the following:
 - 0 Puerto Rico will appear twice, under 006-007 and again under 009.
 - APO/FPO entries will be printed separately from the state with which they are associated. This will cause Florida and Washington to appear as three entries in total, one on either side of the APO/FPO entry. The APO/FPO entries for New York and California will also appear, but they do not divide the state.
 - 0 Using the State abbreviation will cause the APO/FPO entries to be combined with the appropriate state entry, and will combine the two segments of Puerto Rico. In this case, the report will be printed in sequence by the State abbreviation rather than in ZIP Code (or Postal Code) sequence.
 - 0 In all formats which require a specific sortation, the program will perform a sequence check on the input file, and terminate if an out-of-sequence condition is detected.
 - 0 If your merge/purge includes records from more than one country (that is, from more than one of Australia, Canada, and the United States), and if you select one page per State with the State Abbreviation sortation, then the pages will not be grouped by country; they will appear in the sequence of the State Abbreviation, except that the eight Australian States/Territories will appear as if their abbreviations were A1 through A8. If instead you choose the SCF sortation, then the eight Australian States will appear first, followed by the Canadian Provinces, followed by the US States.
 - 0 Note: The eight Australian States/Territories will always be
+ _____
printed in Post Code sequence regardless of the sortation used, because of the two-character pseudo state abbreviation used in order to make Australian counts fit into the statistics record format.

1Group 1 Software -- Consumer Merge/Purge
Program DE41

6.3

0HEADER Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'HEADER'.
0	08 - 17	DATE - May contain a date in the form MM/DD/YYYY. This date will appear on the output reports. If this field is left blank, the current date will be inserted.
0	19 - 58	RUN DESCRIPTION - May contain any run description the user wishes. This run description will appear on the output reports.

1

1Group 1 Software -- Consumer Merge/Purge
Program DE41

6.5

0TAPExx Parameter Card

- TAPExx or DISKxx cards must be used to activate each input or output file. Following is a brief description of each of the possible files See the System Overview section for a detailed description of the TAPExx/DISKxx card fields.

- File_Activators:

+ _____

0 IX - Input State Counts Statistics file, sorted as required

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'CONTRL'.
0	08 - 08	SUPPRESS PURGE FILES - Must be blank (normal), or contain 'X' to indicate that File Codes which are identified as Purge files are not to be printed on the reports.
0	10 - 10	STATE SEGMENTATION - Must contain one of the following codes to specify how certain states are to be presented. See the introductory description for details, and for how this choice affects the required sortation for the input file. The permissible values are: blank - The sortation is to be by SCF Range, and the states are to be printed in multiple segments where appropriate A - The sortation is to be by State abbreviation, and only one segment is to be printed for each state.
0	12 - 12	REPORT FORMAT - Must contain one of the following values to specify the desired report format. The format selected determines the sortation required for the input file (see the introductory discussion at the beginning of this section). The permissible values are: F - One page per File Code, with the States listed vertically S - One page (or more) per State, with the File Codes listed vertically T - One page only, for all File Codes combined, with the States listed vertically blank - defaults to F, as above.

Program DE41

0FLCSEL/FLCREJ Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'FLCSEL/FLCREJ'. FLCSEL or FLCREJ parameter cards may optionally be provided to select only certain records, or to reject certain records, based on a substring of the File Code. When a record is read and FLCSE or FLCREJ cards are present, and that record is no selected or is rejected, then that record is bypassed.
0	08 - 08	FILE CODE SUBSTRING POSITION - Must contain the starting position of the substring within the File Code. Each FLCSEL/FLCREJ parameter card must specify a substring position and length, but the positions and lengths need not be the same from card to card.
0	10 - 10	LENGTH OF SUBSTRING - Must contain the length of the substring. The sum of column 8 and column 10 may not exceed 10.
0	12 - 20 22 - 30 32 - 40 42 - 50 52 - 60 62 - 70	SUBSTRING COMPARAND VALUES - Must contain the value that the substring will be compared to for selection or rejection. A blank File Code will be selected/rejected only if columns 12-20 contain blanks. Blanks in any other of these card columns will terminate processing of this parameter card.

0 Notes:

+ _____
0 If any FLCSEL parameter cards are present, then FLCREJ parameter cards are not permitted, and vice versa.

+ _____
0 The maximum number of Substring Comparand Values which can be entered in all cards combined is 100.

0Operating Instructions

- PROGRAM FUNCTION: STATE COUNTS REPORT(S) PROGRAM NAME: DE41
0 SYS NR/ DEVICE I/O
DDNAME ADDRESS FILE DESCRIPTION/LABEL RET. FILE ID DISPOSITIO

+
T
A SYS010 I State-Counts Statistics TAPEIX
P
E

o
r

D
I
S
K

U
N
I
T
S

0 SYS NR/
DDNAME DESCRIPTION OF CARD DISPOSITIO

+
0 R SYS004 Parameter Cards
P

0 SYS NR/ FORM CARRIAGE
DDNAME DESCRIPTION OF REPORT NUMBER TAPE DISPOSITIO

+
P
R SYS005 Parameter List As Reqd. As Reqd.
I State Counts by File Code
N or
T File Code counts by State
E
R

0 MESSAGE RESPONSE
+

C
O
N
S
O
L
E

1Group 1 Software -- Consumer Merge/Purge 6.13
Program DE41

0Sample OS JCL

```
- //STEP1 EXEC PGM=DE41
//SYS005 DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=133)
//SYS010 DD DSN=DUPELIM.STATE.COUNT.STATS,
// UNIT=TAPE,VOL=SER=TAPEIX,DISP=OLD,LABEL=(2,BLP),
// DCB=(DSORG=PS,RECFM=FB,LRECL=092,BLKSIZE=bbbb)
//SYSUDUMP DD SYSOUT=A
//SYS004 DD *
```

.
.
Parameter Cards

.
.
/*

1Group 1 Software -- Consumer Merge/Purge 6.14
Program DE41

0Sample DOS JCL

```
- // JOB DE41
// TLBL TAPEIX
// ASSGN SYS010,X'180' (Input TAPEIT)
// ASSGN SYS005,X'00E' (Printer)
// ASSGN SYS004,X'00C' (Card reader)
// EXEC DE41
```

.
.
Parameter Cards

.
.
/*
/&

three such File Codes, and so forth.)

1Group 1 Software -- Consumer Merge/Purge
Program DE50

7.2

0Narrative

- The File Match Analysis produces a page for each File Code, with all of the File Codes listed vertically. For each File Code in the vertical list, the number of matches against the subject (page) File Code is displayed. The number of matches as a percentage of the subject file input count, and as a percentage of the total subject-file matches is computed and printed. A final Summary page lists all the File Codes vertically, with the total number of matches for each displayed.
- 0 The Match Matrix is a square array of numbers, with a column and a row for each File Code. The number of matches between the row and column File Codes is printed at the intersection of the row and column. Row and Column headers are printed on each page to facilitate cutting and pasting the separate pages into a single large square, if desired. File Codes are listed horizontally and vertically on each page to facilitate usage of the report as separate pages, if the square matrix is too large to be assembled usefully into a single page.
- 0 Optionally (if File OM is defined), Multi-buyer records may be written on an output file.

1Group 1 Software -- Consumer Merge/Purge
Program DE50

7.3

0HEADER Parameter Card

- | - | CARD COLUMNS | FIELD DESCRIPTION |
|---|--------------|--|
| - | 01 - 06 | PARAMETER KEYWORD - Must contain 'HEADER'. |
| 0 | 08 - 17 | DATE - May contain a date in the form MM/DD/YYYY. This date will appear on the output reports. If this field is left blank, the current date will be inserted. |
| 0 | 19 - 58 | RUN DESCRIPTION - May contain any run description the user wishes. This run description will appear on the output reports. |

0TAPExx Parameter Card

- TAPExx or DISKxx cards must be used to activate each input or output file. Following is a brief description of each of the possible files. See the System Overview section for a detailed description of the TAPExx/DISKxx card fields.

- File_Activators:

+

-
- 0 IT - Input Dupe-ID Summary Stats
 - 0 ID - Input Duplicate Records file
 - 0 IS - Input Conversion Summary Stats (optional)
 - 0 OM - Output Multi Buyer file (optional)
 - 0 OS - Output Conversion Summary Stats (see below)
 - 0 OT - Output Dupe-ID Summary Stats (see below)

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'CONTRL'. The CONTRL card is required to locate fields in the input Duplicate Names file, and to provide information on the required reports.
0	08 - 10	DUPLICATE CONTROL CHARACTER - Must contain the location in the input Duplicate Names file at which may be found the Duplicate Control character stored by the Duplicate Identification program. The Duplicate Control Character is '0' for the first record in a duplicate group, and is blank for all subsequent records of the same duplicate group. If necessary, this field may be extended one position to the left, permitting the entry of a four-digit location in columns 7-10.
0	12 - 14	SELECTION INDICATOR - May contain the location in the input Duplicate Names file at which may be found the Selection Indicator stored by the Duplicate Identification program. If necessary, this field may be extended one position to the left, permitting the entry of a four-digit location in columns 11-14. Normally, the Selection Indicator location should be specified. The program will use this location to compute the number of drops of each type. If it is omitted, it will assume one record is kept per duplicate group (or none, if a Purge record is in the group), and compute the drop counts on that basis. This will not agree with the actual situation in the Duplicate Identification program if: more than one record was kept; or if the Individual duplicate-control character is specified; or if the PRIORT card was used to drop records without an Individual name, a Business name, or a Business Title.
0	16 - 18	FILE CODE - Must contain the location in the input Duplicate Names file at which may be found the File Code. If necessary, this field may be extended one position to the left, permitting the entry of a four-digit location in columns 15-18.

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	20 - 20	FILE CODE LENGTH - Must contain the length of the File Code in the input Duplicate Names file.
0	22 - 22	SUPPRESS DROP SUMMARY REPORT - Must contain one of the following codes to indicate whether or not the Drop Summary Report is to be produced. X - do not produce the report 1-9 - the number of copies of the Drop Summary Report to be produced blank - defaults to '1'
0	24 - 24	SUPPRESS MULTI-BUYER REPORT - Must contain one of the following codes to indicate whether or not the Multi-Buyer Report is to be produced. X - do not produce the report 1-9 - the number of copies of the Multi-Buyer Report to be produced blank - defaults to '1'
0	26 - 26	SUPPRESS FILE MATCH ANALYSIS - Must contain one of the following codes to indicate whether or not the File Code Match Analysis is to be produced. The Match Analysis is a report basically consisting of one page per File Code, with all the File Codes listed vertically. For each File Code, the number of matches versus the subject File Code (whose page this is) is displayed, along with the percentage that number of matches of the subject file input count, and of the total number of matches for the subject file. The permissible codes are: X - do not produce the report 1-9 - the number of copies of the File Code Match Analysis to be produced blank - defaults to '1'

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	28 - 28	SUPPRESS FILE MATCH MATRIX - Must contain one of the following codes to indicate whether or not the File Code Match Matrix is to be produced. The Match Matrix is basically a rectangular report, with all the File Codes listed both horizontally and vertically. The intersection of each row and column will then contain the number of matches between the "row" File Code and the "column" File Code. The permissible values are:
0		X - Do not produce the report
		1-9 - The number of copies of the File Code Match Matrix to be produced
		blank - defaults to '1'
0		Note: If the number of File Codes is large,
+		_____ then the Match Matrix will be large, and will probably need to be "cut and pasted" in order to be very useful.
0	30 - 30	SUPPRESS PURGE FILES - Must contain one of the following codes to indicate how Purge files are to be treated when printing the reports:
0		blank - Do not suppress the Purge files
		X - Do not print a page for any Purge
+		_____ files in the Match Analysis report, but do print the lines for the Purge
+		_____ files on the pages for the other (non-Purge) File Codes; do print the
+		_____ Purge-file lines, rows, and columns on the other reports.
		Y - do not print a page for any Purge
+		_____ files in the Match Analysis report, and do not print the Purge-file lines
+		_____ on the pages for the other File Codes; do not print either a row or a
+		_____ column for any Purge file on the Match Matrix report; but do print the
+		_____ Purge-file lines on the Drop Summary.
		Z - same as 'Y' above, except that the

Purge-file lines on the Drop Summary
are not printed either.

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	32 - 32	REPORT SEQUENCE - Must contain one of the following codes to control the sequence of the File Codes for all the above reports:
0		blank - print the reports in the same sequence as the input Summary Stats file from the Duplicate Identification program. Normally, this will be the same sequence as the original FLCnnn parameter cards. A - Alphabetically by File Code P - list by the relative priority (major), and File Code (minor) D - list by File Code Description (major), relative priority (intermediate), and File Code (minor)
0	34 - 34	PRINT INACTIVE FILE CODES ON REPORTS - Must contain one of the following codes to indicate whether or not File Codes with no Dupe-ID activity are to be printed on the reports:
0		blank - do not print inactive File Codes X - print inactive File Codes
0	36 - 36	SUPPRESS CONVERSION ANOMALY REPORT - Must contain one of the following codes to indicate whether or not the Conversion Anomaly report is to be produced:
0		blank - allow the Conversion Anomaly Report to be printed, if anomalies are detected X - do not print the Conversion Anomaly Report
0		The Conversion Anomaly report contains a list of all the file codes in which a discrepancy exists between the Conversion Statistics (File IS) output count, and the Dupe-ID Statistics (File IT) input count. If no anomalies are detected, then no report will be produced.

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	38 - 38	SUPPRESS DUPE-ID ANOMALY REPORT - Must contain one of the following codes to indicate whether or not the Dupe-ID Anomaly report is to be produced:
0		blank - allow the Dupe-ID Anomaly Report to be printed, if anomalies are detected
0		X - do not print the Dupe-ID Anomaly Report If discrepancies are detected between the numbers presented in the Summary Statistics (File IT) and the situation as determined by examination of the Duplicates file (File ID), then an Anomaly Report will be produced (unless suppressed) showing the relevant values for each File Code. Normally such anomalies are due to one of the following causes:
0		Presentation of incorrect files Incorrect positions for Dupe-Control character, or Selection Indicator, or both I/O errors on File ID or IT Omission of the Selection Indicator location, when the DIALIT card in program BM30 specified more than one record to be kept per company, or when the Individual Dupe-Control character is specified to this program
0	40 - 40	SUPPRESS ZERO LINES ON MATCH ANALYSIS - Must contain one of the following codes to indicate whether or not to print a line for a file code that has zero matches on the Match Analysis Report blank - print lines that have zero matches
0		X - do not print lines that have zero matches
0	42 - 45	MATCH MATRIX PRINT THRESHOLD ANALYSIS - This field is used to suppress printing of values on the MATCH MATRIX Report below the specified quantity. If not specified, all values will be printed.

0MULTIB Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'MULTIB'. The MULTIB parameter card may be used to change the File Code of Multi-Buyer records on the output Mail file(s) to the same value for all, regardless of the original File Code, without affecting any
0		_____
+		of the reports.
0	08 - 16	MULTI-BUYER FILE CODE - Must contain the File Code to be assigned to the survivor(s) of a multi-buyer duplicate set.
0	Note:	For the Duplicate Elimination program, an output File
+	_____	Code location must be specified in the CONTRL card if
+		_____
		the File Code specified here is to appear in the output Mail files (OM and/or OE).

0MLTLIM Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'MLTLIM'. The MLTLIM parameter card may optionally be provided to limit the identification of Multi-Buyer records.
0		Normally, if a duplicate-group contains records from a single File Code, then the group is identified as a Single-Buyer group, any drops are recorded as Single-Buyer drops, and any survivors are identified as Single-Buyer survivors. If more than one File Code is present in the group, it is considered a Multi-Buyer group, any drops are counted as Multi-Buyer drops, and the survivors are marked as Multi-Buyer survivors.
0		By means of this card, the File Code comparison, for the purpose of Multi-Buyer identification only, may be restricted to a substring of the File Code proper.
0	08 - 08	POSITION OF SUBSTRING - Must contain the starting position of the relevant substring, within the File Code as stored by the FLCODE parameter.
0	10 - 10	LENGTH OF SUBSTRING - Must contain the length of the substring.

0 Notes: The sum of column 8 and column 10 may not exceed 10.

0 Note: If a MLTLIM card was provided to the Duplicate Identification

+

Program, then the same MLTLIM must be provided to this program if the Multi-Buyer levels of the duplicate groups are to be computed correctly. The specification of a File Code as a non-Multi-Buyer contributor in the FLCnnn cards (for the Duplicate Identification Program) will be recognized by this program from the records in the Summary Statistics file.

0PRIORT Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'PRIORT'. The PRIORT parameter card may be used to specify the algorithm to be used to select the surviving record of a duplicate set.
0	08 - 08	RECORD SELECTION ALGORITHM - Must contain one of the following codes to specify the algorithm which is to be used to determine which record of each duplicate-group is to survive (that is, which will be written to File OM) P - Priority by Position; the last File Code defined by a File Code card will have the highest priority, the next-to-last File Code will have the next-highest priority, and so forth. R - Random; the survivor will be selected on a statistically random basis. S - Semi-random priority; the surviving record will be the first record encountered in a duplicate set. If there is a bias introduced in the sortation of the input files, then this bias will be reflected in the record selection. M - Mixed priority; the last n (where n is + the value entered in columns 10-12) File Code parameter cards have priority (the last having the highest priority, the next to last having the next to highest priority and so forth) and the rest of the File Code parameters have random priority
0	10 - 12	EXTENT OF MIXED PRIORITY - Must be blank if column 8 is not 'M'. If column 8 does contain 'M', then this field must contain a number between 001 and the value one less than the number of non-Purge File Codes defined by the File Code parameter cards. The number entered specifies how many of the defined File Codes (beginning with the last-defined) will be treated as 'P' above, with the remaining File Codes being treated as 'R' above.

0SEQCHK Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'SEQCHK'. The SEQCHK card is used to define the sequence of the input file, by specifying the fields which make up the sequencing key.
0		The presence of this card causes a sequence check operation to be performed on the defined fields. If a sequence error is detected, it will be handled as specified by one of the SEQERR card options ('E' if no SEQERR card is provided). This sequence checking occurs immediately following the the read of the input file and before the effect of any other parameter card.
0	08 - 10 15 - 17 22 - 24 29 - 31 36 - 38 43 - 45 50 - 52 57 - 59 64 - 66 71 - 73	SEQUENCE FIELD SEGMENT LOCATIONS - Must contain the starting positions of the fields that define the sequence of the input name and address file. If necessary, these fields may be extended to the left one position, allowing the entry of a 4-digit number in columns 7-10, 14-17, etc.
0	11 - 11 18 - 18 25 - 25 32 - 32 39 - 39 46 - 46	PACKED SEQUENCE FIELD - Must be blank, or contain 'P' to indicate that this sequence field segment is in packed form, and should be unpacked prior to use. If entered as 'P', then the field will occupy (2n-1) characters in the total merge contro field, where n is the length specified.
+		-
0	53 - 53 60 - 60 67 - 67 74 - 74 12 - 13 19 - 20 26 - 27 33 - 34 40 - 41 47 - 48 54 - 55 61 - 62 68 - 69 75 - 76	SEQUENCE FIELD SEGMENT LENGTHS - Must contain the length of each field defined in the previous parameter card field. The total of all field lengths must not exceed 99.

0SEQERR Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'SEQERR'. The SEQERR card may optionally be supplied to specify the treatment of sequence errors in the input file as defined in the SEQCHK card. If omitted, the default treatment will be that described under 'E' below.
0	08 - 08	SEQUENCE ERROR TREATMENT - Must contain one of the following values to specify the desired treatment of an out-of-sequence condition. B - Bypass the offending record, and continue sequence checking C - Continue processing of the offending record, and continue sequence checking E - Bypass the offending record, and do not read any more records from the input file. That is, bring the execution to a graceful close with what has been read so far. I - Continue processing of the offending record, and abandon any further sequence checking of the input file. O - Ask the operator for one of the above treatment options, and honor it.
0		Note: If the SEQERR card is omitted, the default treatment option is E. This is recommended because a sequence error is almost certain to indicate either an error in the sortation, or an error in the definition of the sequence control field.
0		In all cases, the sequence error will be displayed on the console, and the occurrence of a sequence error will be noted on the print output (SYS005). If option E is selected, via SEQERR card or by the operator, a note will be displayed on the console and the print output that processing was terminated by a sequence error condition.

0ANALIM Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'ANALIM'.
0		The ANALIM card may optionally be provided to limit the number of lines on each page of the File Match Analysis, or to cause selected File Codes (which have a specified relationship to the "page" File Code) to be printed first on the Match Analysis pages.
0		Either or both of two relationships may be defined. One is that a specified substring of the File Code or File Code Description be equal, and the other is that the Matches, as a percentage of the ("page" File Code) input, exceed a specified value.
0	08 - 08	FUNCTION OF ANALIM CARD - Must contain one of the following codes to indicate which of the above two functions is desired. The codes are:
0		S - change the sequence of the File Codes on each page of the File Match Analysis, but do not suppress any of the lines
		X - Suppress those lines on the File Match Analysis which do not have the indicated relationship to the page File Code
0	10 - 14	IDENTIFICATION RELATIONSHIP - These columns must either be blank, or must define a relationship based on a substring of either the File Code or the Description (of each File Code) which must be equal in order for an identification relationship to exist between two File Codes. If column 10 is blank, then the other columns are not examined.
0	10 - 10	SOURCE OF SUBSTRING - Must contain one of the following codes to indicate whether the File Code or the Description is to be the source of the substring which is to be checked:
0		F - the File Code is the source
		D - the Description is the source
		blank - no identification relationship is to be established
0	12 - 12	POSITION OF SUBSTRING - Must contain the position of the substring within the source field.
0		If necessary (when using the Description), this field may be extended one column to the left, permitting the entry of a two-digit position in columns 11-12.

0ANALIM Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	14 - 14	LENGTH OF SUBSTRING - Must contain the length of the substring to be compared. The sum of this field and the substring position may not exceed the length of the source field by more than one. The length of the source field is either 9 (for the File Code) or 20 (Description), depending on column 10.
0		If necessary (when using the Description), this field may be extended one column to the left, permitting the entry of a two-digit length in columns 13-14.
0	16 - 17	PERCENTAGE RELATIONSHIP - May contain the minimum percentage (Matches as a percentage of "page" File input) which must be present in order to establish a percentage relationship. If the number of matches for a File Code, expressed as a percentage of the "page" File Code's input, is greater than or equal to this value, then a percentage is established.

- Notes:

- 0 Column 10 and columns 16-17 may not both be blank.
- 0 If column 8 is 'X', then only the File Codes with either or both of the specified relationships (to the "page" File Code) will be printed on each page of the Match Analysis.
- 0 If column 8 is 'S', then those File Codes which have an identification relationship will be printed first, followed by those File Codes with a percentage relationship (but not both relationships), followed by the remaining File Codes.

0FLCSEL/FLCREJ Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'FLCSEL/FLCREJ'. 0 FLCSEL or FLCREJ parameter cards may optionally be provided to select only certain records, or to reject certain records, based on a substring of the File Code. When a record is read and FLCSE or FLCREJ cards are present, and that record is not selected or is rejected, then that record is bypassed.
0	08 - 08	FILE CODE SUBSTRING POSITION - Must contain the starting position of the substring within the File Code. 0 Each FLCSEL/FLCREJ parameter card must specify a substring position and length, but the positions and lengths need not be the same from card to card.
0	10 - 10	LENGTH OF SUBSTRING - Must contain the length of the substring. 0 The sum of column 8 and column 10 may not exceed 10.
0	12 - 20 22 - 30 32 - 40 42 - 50 52 - 60 62 - 70	SUBSTRING COMPARAND VALUES - Must contain the value that the substring will be compared to for selection or rejection. A blank File Code will be selected/rejected only if columns 12-20 contain blanks. Blanks in any other of these card columns will terminate processing of this parameter card.

0 Notes:

0 If any FLCSEL parameter cards are present, then FLCREJ parameter cards are not permitted, and vice versa.

0 The maximum number of Substring Comparand Values which can be entered in all cards combined is 100.

0 Care must be taken to ensure that the FLCSEL or FLCREJ cards are such that whole duplicate groups are included or excluded, or else the match analysis will be distorted. Normally, these cards are only used in DE50 when the Duplicate Identification Program was restricted (via the CMPLIM card or the no-driver flag), to accomplish more than one dupe-id in a single pass.

0CHCKPT Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'CHCKPT'.
0	08 - 14	CHECKPOINT COUNT - The checkpoint parameter card activates the IBM checkpoint facility. If this option is used, checkpoints will be taken for every 'n' input records processed. This field must contain a number which will be used as 'n'. The value must be numeric and greater than one, entered with leading zeroes as required.
0	68 - 71	(DOS only) DEVICE - May contain the device to which the checkpoint file is to be assigned. See the TAPExx card for a list of available device types. The default is 2314.
-	OS_CHECKPOINT-RESTART	
0	1.	A DD-statement (DDNAME=SYS001) must be provided for the checkpoint dataset. It must specify a physical-sequential dataset which may reside on either tape or disk.
0	2.	If the DISP parameter on the SYS001 DD does not specify MOD, then:
0	A.	Each checkpoint will be written over the preceding checkpoint; and, therefore:
0	B.	If a checkpoint cannot be taken successfully, there will be no checkpoint available for restart until another checkpoint is successfully taken.
0	3.	Restart as specified in the JCL manual.

0CHKPT Parameter Card

- CARD COLUMNS FIELD DESCRIPTION
- DOS_CHECKPOINT-RESTART

+

- If checkpoint processing is being performed (i.e.,
the 'CHKPT' parameter is present), the following
additional JCL is needed:

0 // ASSGN SYS001,X'nnn'
// DLBL DTFPHM,'CHECKPOINT',0010,SD
// EXTENT SYS001,WORK02,1,0,20,200

0 If the job must be restarted from a checkpoint, then
the following card replaces the '// EXEC' card
in the run deck. No parameter cards need to be entered:

0 // RSTRT SYS001,nnnn,DTFPHM

0 ...Where 'nnnn' is the Restart or Checkpoint
number the user wishes to restart from.

0Operating Instructions

PROGRAM FUNCTION: MATRIX REPORTS		PROGRAM NAME: DE50			
SYS NR/	DEVICE	I/O	FILE DESCRIPTION/LABEL	RET. FILE ID	DISPOSITIO
DDNAME	ADDRESS				
A	SYS010	I	Dupe-ID Summary Stats	TAPEIT	
P	SYS011	I	Duplicate Names File	TAPEID	
E	SYS012	I	Conversion Statisticks (opt)	TAPEIS	
o	SYS015	O	Multi-Buyer File (opt)	TAPEOM	
D	SYS016	O	Comversion Summary Stats	TAPEOS	DE40
I	SYS017	O	Dupe-ID Summary Stats	TAPEOT	DE40

U
N
I
T
S

SYS NR/	DDNAME	DESCRIPTION OF CARD	DISPOSITIO
R	SYS004	Parameter Cards	

SYS NR/	DDNAME	DESCRIPTION OF REPORT	FORM NUMBER	CARRIAGE TAPE	DISPOSITIO
R	SYS005	Parameter List	As Reqd.	As Reqd.	
I		Multi-Buyer Counts by level			
N		Match Analysis (File Code vs File Code)			

MESSAGE	RESPONSE
C	
O	
N	
S	
O	
L	
E	

0Sample OS JCL

```
- //REPORTS EXEC PGM=DE50
//SYS005 DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=133)
//SYS010 DD DSN=INPUT.DUPEID.SUMMARY.STATS,DISP=OLD,
// UNIT=TAPE,VOL=SER=TAPEIT,LABEL=(2,BLP),
// DCB=(DSORG=PS,RECFM=FB,LRECL=80,BLKSIZE=bbbb)
//SYS012 DD DSN=INPUT.CONVERT.SUMMARY.STATS,DISP=OLD,
// UNIT=TAPE,VOL=SER=TAPEIS,LABEL=(2,BLP),
// DCB=(DSORG=PS,RECFM=FB,LRECL=80,BLKSIZE=bbbb)
//SYS011 DD DSN=INPUT.DUPE.NAMES.FILE,
// UNIT=TAPE,VOL=SER=TAPEID,DISP=OLD,LABEL=(2,BLP),
// DCB=(DSORG=PS,RECFM=FB,LRECL=rrrr,BLKSIZE=bbbb)
//SYS015 DD DSN=OUTPUT.MULTI.BUYER.FILE,
// UNIT=TAPE,VOL=(,,30),DISP=(,CATLG),
// DCB=(DSORG=PS,RECFM=FB,LRECL=rrrr,BLKSIZE=bbbb)
//SYSUDUMP DD SYSOUT=A
//SYS004 DD *
```

```
.
.
Parameter Cards
```

```
.
.
/*
```

0Sample DOS JCL

```
- // JOB DE50
  // TLBL TAPEIT
  // TLBL TAPEID
  // TLBL TAPEIS
  // TLBL TAPEOM
  // ASSGN SYS010,X'180'      Input TAPEIT
  // ASSGN SYS011,X'181'      Input TAPEID
  // ASSGN SYS012,X'182'      Input TAPEIS
  // ASSGN SYS015,X'183'      Output TAPEOM
  // ASSGN SYS005,X'00E'      Printer
  // ASSGN SYS004,X'00C'      Card Reader
  // EXEC DE50
```

.
.

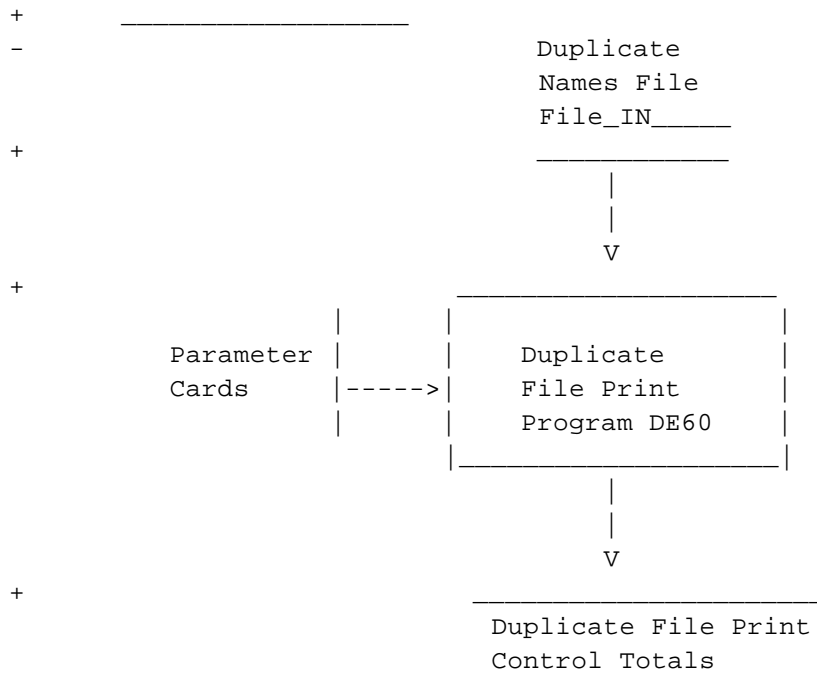
Parameter Cards

.
.

/*
/&

0Narrative

- Narrative_and_Flow:



- DE60 can be used to print the Duplicate file in a one-line-per-record format, with the contents of each line fully specifiable via parameter card. Each Duplicate Group is double spaced from the preceding one, and the records within a single group are single-spaced.

0 Nth-group selection and selection by File Code are both available. Custom headers may be constructed for use by the program during the file print. The size of the report can be limited to a specific number of pages if desired.

0 By omitting the Duplicate Control character location, the program may also be used to print the clean file (or any other file) with the same degree of control.

1Group 1 Software -- Consumer Merge/Purge 8.3
Program DE60

OHEADER Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'HEADER'.
0	08 - 17	DATE - May contain a date in the form MM/DD/YYYY. This date will appear on the output reports. If this field is left blank, the current date will be inserted.
0	19 - 58	RUN DESCRIPTION - May contain any run description the user wishes. This run description will appear on the output reports.

1

1Group 1 Software -- Consumer Merge/Purge 8.5
Program DE60

OTAPExx Parameter Card

- TAPExx or DISKxx cards must be used to activate each input or output file. Following is a brief description of each of the possible files See the System Overview section for a detailed description of the TAPExx/DISKxx card fields.

- File_Activators:

+
0 IN - Input Duplicate Names File
0 OP - Output Print File (optional)

- Notes:

+
0 If File OP is activated, then the print lines constructed by the MOVE, UNPK, and FORMAT cards will be written to the file defined (SYS015). No headers or control totals will be written to the tape or disk file, and the printer (SYS005) will contain only the parameter list and control totals. The record size for File OP must be 133. This file can be used as one of the two inputs to program DE61 (which prints the difference
+ between two duplicate files), for calibration of the Merge/Purge.

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'CONTRL'.
0	08 - 10	FILE CODE LOCATION - Must contain the starting location in the input records of the file code. Note: If necessary, this field may be extended
0		_____
+		one position to the left, allowing the entry of a four-digit location in columns 7-10.
0	12 - 12	FILE CODE LENGTH - Must contain the length of the file code.
0	14 - 16	DUPLICATE CONTROL LOCATION - Must the location of the duplicate control character. If no duplicate control location is specified, then the report will be single spaced. If duplicate control location is specified, then the first record of a duplicate group will be double-spaced. This allows for easy identification of duplicate groups.
0		Note: If necessary, this field may be extended
0		_____
+		one position to the left, allowing the entry of a four-digit location in columns 13-16.
0	18 - 18	DUPLICATE CONTROL CHARACTER VALUE - If columns 14-16 are not blank, then this field must contain the value of the duplicate control character that identifies the beginning of a duplicate group. If the duplicate file was created using program BM30 or CM30, then this column should contain a zero ('0').
0	20 - 22	PAGE LIMIT - May contain a numeric value to limit the number of pages printed. Note: If necessary, this field may be extended
0		_____
+		one position to the left, allowing for the entry of a four-digit number in columns 19-22.
0	24 - 26	PAGE NUMBER POSITION - May contain the position, within the first header line that a five-digit page number will be printed.

1Group 1 Software -- Consumer Merge/Purge
Program DE60

8.8

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	28 - 28	PRINT SINGLE RECORD DUPLICATE GROUPS - Must contain one of the following codes to indicate whether or not single-record duplicate groups are to be printed:
0		blank - do not print single-record duplicate group
+		_____
		X - Print single-record duplicate groups
0		Note: Single-record duplicate groups may be
+		_____
		present if: a) the Individual duplicate-control
+		_____
		character is specified, rather than the Business
+		_____
		duplicate-control character, on duplicate files produced by BM30; or b) the duplicate groups have been reduced from their original content due to the action of FLCSEL or FLCREJ parameter cards.

1Group 1 Software -- Consumer Merge/Purge
Program DE60

8.9

0HEADxx Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 04	PARAMETER KEYWORD - Must contain 'HEAD'. This card can be used to define up to three lines of headings for the report.
0		
0	05 - 06	HEADING LINE DESIGNATOR - Must contain one of the following codes to represent the Heading line designator:
0		1A = left side, heading line 1. 1B = right side, heading line 1. 2A = left side, heading line 2. 2B = right side, heading line 2. 3A = left side, heading line 3. 3B = right side, heading line 3.
0		Left side represents print positions 1-66. Right side represents print positions 67-132.
0	08 - 73	DATA TO PRINT - Must contain the data that is to be printed in the corresponding print positions for each side of each heading line
-		Note: When the program encounters the first occurrence of the keyword 'MM/DD/YYYY' in the 'HEADxx' parameter cards, it will replace it with the value specified in columns 8 - 17 of the 'HEADER' parameter. If columns 8 - 17 are left blank, then it will use the current date. If the keyword 'MM/DD/YYYY' is not found in the 'HEADxx' parameter cards, then nothing will be altered.

1Group 1 Software -- Consumer Merge/Purge 8.11
 Program DE60

0CONSTn Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 05	PARAMETER KEYWORD - Must contain 'CONST'.
0		This parameter is used to define up to 9 constants to the program which can be referenced in the MOVExx, and FORMAT parameter cards as C01, C02, ..., C09.
0	06 - 06	CONSTANT NUMBER - Must contain a numeric value between 1 and 9 inclusive. This number distinguishes the difference among the 9 possible CONSTn parameter cards that may be used in the program.
0	08 - 17	CONSTANT VALUE - Must contain the value, left justified to column 8, of the CONSTn. There is no length implied here except that the maximum size for any CONSTn is 10 positions. The length of the CONSTn is specified when using the MOVExx or FORMAT parameter card(s).

1

1Group 1 Software -- Consumer Merge/Purge 8.13
 Program DE60

0MOVExx Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 04	PARAMETER KEYWORD - Must contain 'MOVE'.
0	05 - 06	Any value may be entered here.
0	08 - 10	DESTINATION - Must contain the position in the print line to which the data is to be moved.
0	12 - 13	LENGTH - Must contain the length of the data to be moved. This must be a numeric value in the range 01 to 99 inclusive.
0	15 - 17	SOURCE - Must contain one of the following:
0		(1) The starting position on the input record from which data is to be moved.
0		(2) A CONSTANT: C01, C02, ..., C08.
0		Note: if necessary, this field may be extended

+

_____ one position to the left, permitting the entry of a four-digit number in columns 14-17.

0 General: 'MOVExx' parameter cards are used to move data from

+

_____ the input record or constant values defined by the user to the output report. Up to 100 MOVExx parameter cards may be entered and are executed in the order presented.

0FORMAT Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'FORMAT'. The parameter strings together fields and moves the entire string to the Output Area specified. See below for further discussion of this parameter
0	07 - 08	COMMENT - These columns are ignored by the program and may be used for comment.
0	10 - 12	PRINT LOCATION - Must contain the starting position in the print line where the formatted string will be placed. The value entered must be a numeric value between 001 and 132.
0	14 - 15	LENGTH OF RESULTING STRING - Must contain the length of the Area defined in columns 10-12. This must be a non-zero numeric value.
0	17 - 19 24 - 26 31 - 33 38 - 40 45 - 47 52 - 54 59 - 61 66 - 68 73 - 75	LOCATION OF FIELDS TO BE FORMATTED - Must contain the starting positions of 2 to 9 fields in the Input Area that contain the fields to be formatted. The fields are formatted left to right. Columns 17-19 and columns 24-26 must be entered. The first blanks encountered will terminate processing of the card. Note: If necessary, this field may be extended to the left one position, allowing the entry of a 4-digit number in columns 16-19, 23-26, etc.
0	20 - 20 27 - 27 34 - 34 41 - 41 48 - 48 55 - 55 62 - 62 69 - 69 76 - 76	CONSTANT TREATMENT - Must contain one of the following to control the formatting of constants: '*' - constant will not appear if the Preceding field contains all blanks. blank- constant will not appear if the Succeeding field contains all blanks. See example below.
0	21 - 22 28 - 29 35 - 36 42 - 43 49 - 50 56 - 57 63 - 64 70 - 71 77 - 78	LENGTH OF FIELDS TO BE FORMATTED - Must contain the corresponding lengths of the fields defined in columns 17-19, etc.

0FORMAT Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	23 - 23	ZERO SUPPRESSION - Must contain a 'Z' if suppressio
	30 - 30	of leading zeros is desired in the corresponding
	37 - 37	field defined in columns 17-19, etc.
	44 - 44	
	51 - 51	Note that this 'Z' may conflict with a 4-digit
	58 - 58	position, if needed for the following field.
	65 - 65	
	72 - 72	
	79 - 79	

0 General: Following are a number of rules and restrictions that apply t

+

- _____
- the FORMAT parameters:
- 0 (1) FORMAT parameters will be executed in the order presented.
 - 0 (2) The first field to be formatted must not be a constant.
 - 0 (3) There must be at least two fields to be formatted.
 - 0 (4) Two consecutive non-constant fields will automatically have a space placed between them.
 - 0 (5) All leading blanks of non-constant fields will be suppressed and will not appear in the output string.
 - 0 (6) No spaces will appear between constants and variables unless allowed for in defining the constant.

0 Example: FORMAT01 017 25 100 15 C01 02 115 04 C02*01

+

CONSTANT01 ,
CONSTANT02 .

0 If 100 contains SMITH and 115 contains JR -
the output will contain 'SMITH, JR.'

0 If 100 contains SMITH and 115 contains blanks -
the output will contain 'SMITH'.

0UNPKxx Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 04	PARAMETER KEYWORD - Must contain 'UNPK'.
0	05 - 06	Any value may be entered here.
0	08 - 10	PRINT POSITION - Must contain the position in the print line to which the data is to be unpacked.
0	12 - 13	LENGTH IN PRINT LINE - Must contain the length of the unpacked data. This length may not exceed 16.
0	15 - 17	LOCATION OF SOURCE FIELD - Must contain the starting location in the input record of the field that is to be unpacked.
0		Note: If necessary, this field may be extended
+		_____
		one position to the left, permitting the entry of a four-digit number in columns 14-17.
0	19 - 20	LENGTH OF SOURCE FIELD - Must contain the length of the field to be unpacked. This length may not exceed 16.
0		Note: Up to 100 UNPKxx parameter cards may be entered.

0RECLIM Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'RECLIM'. The RECLIM card may optionally be used to limit the records processed for a particular run. A number of records to be skipped may be specified, as well as a number of records to process.
0		The most frequent use of the RECLIM card is to restrict the file to a small number of records, for parameter checkout or calibration. It can also be used, of course, to break what would otherwise be an excessively large job into several smaller independent executions without dividing the input files.
0	08 - 16	RECORDS TO SKIP - May contain the number of records to skip before processing begins. If omitted, processing will begin with the first record. Note: This option will only function as indicated
0		_____
+		when a single input file is used.
+		_____
0	18 - 26	RECORDS TO PROCESS - May contain the number of records to be processed. If omitted, all records will be processed (once the specified number are skipped).
-	Notes:	
0		The RECLIM card is applied to the records as they are read from the input file, before the action of the FLCSEL/FLCREJ cards or the NTHSEL card. Therefore, RECLIM may be used to focus on a particular segment of the entire file (such as the last fifth, etc.).

0FLCSEL/FLCREJ Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'FLCSEL/FLCREJ'. 0 FLCSEL or FLCREJ parameter cards may optionally be provided to select only certain records, or to reject certain records, based on a substring of the File Code. When a record is read and FLCSEL or FLCREJ cards are present, and that record is not selected or is rejected, then that record is bypassed.
0	08 - 08	FILE CODE SUBSTRING POSITION - Must contain the starting position of the substring within the File Code. 0 Each FLCSEL/FLCREJ parameter card must specify a substring position and length, but the positions and lengths need not be the same from card to card.
0	10 - 10	LENGTH OF SUBSTRING - Must contain the length of the substring. 0 The sum of column 8 and column 10 may not exceed 10.
0	12 - 20 22 - 30 32 - 40 42 - 50 52 - 60 62 - 70	SUBSTRING COMPARAND VALUES - Must contain the value that the substring will be compared to for selection or rejection. A blank File Code will be selected/rejected only if columns 12-20 contain blanks. Blanks in any other of these card columns will terminate processing of this parameter card.

0 Notes:

+
0 _____
0 If any FLCSEL parameter cards are present, then FLCREJ parameter cards are not permitted, and vice versa.

+
0 _____
0 The maximum number of Substring Comparand Values which can be entered in all cards combined is 100.

0 FLCSEL or FLCREJ cards are applied to the input records after the action of the RECLIM card, if present, and before the application of an NTHSEL specification if one is present.

0 FLCSEL or FLCREJ cards may have the effect of reducing a well-formed duplicate group to a single record. Such records will not be printed unless CONTRL card column 28 is X.

ONTHSEL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'NTHSEL'.
0		Note: There are two possible ways of specifying an Nth record include or exclude. In the first method presented, the idea is to specify the Nth number and have the program include or exclude each Nth record. This method is particularly suited to sampling operations.
0		The second method is used when a specific number of records must be processed, and when the exact count of the input file is known.
0		Method 1: Sampling
0	08 - 10	NTH NUMBER - Must contain the "Nth" number of records the user wishes to either include or exclude from the records to be printed. This may be a number between 002 and 999 inclusive.
0	12 - 14	INC/EXC - Must contain either 'INC' or 'EXC' to either INclude or EXclude the Nth number of records specified in columns 8-10 of this card.
0		Method 2: Precise number of records
0	08 - 14	FRACTION OF RECORDS - Must contain the fraction of the total input desired, expressed as a seven-digit positive number with a decimal point understood at column 7.
0		Example: Suppose that the input file is known to have exactly 153,714 records, and that our problem is to print exactly 3000 records. Using a calculator, we find that $3000/153714 = 0.0195168$; therefore we enter '0195168' in columns 8-14 and out come the 3000 records.
0	Note:	The Nth selection will be applied to duplicate sets when
+	_____	a duplicate control character location is specified in the CONTRL parameter card, or to the individual records if the duplicate control location is not provided.

0Operating Instructions

- PROGRAM FUNCTION: DUPLICATE FILE PRINT PROGRAM NAME: DE60
 0 SYS NR/ DEVICE I/O
 DDNAME ADDRESS FILE DESCRIPTION/LABEL RET. FILE ID DISPOSITIO

+
 T
 A SYS010 I Duplicate File TAPEIN
 P
 E SYS015 O Duplicate Record Print TAPEOP DE61
 (on tape or disk)

o
 r

D
 I
 S
 K

U
 N
 I
 T
 S

0 SYS NR/
 DDNAME DESCRIPTION OF CARD DISPOSITIO

+
 0 R SYS004 Parameter Cards
 P

0 SYS NR/ FORM CARRIAGE
 DDNAME DESCRIPTION OF REPORT NUMBER TAPE DISPOSITIO

+
 P
 R SYS005 Parameter List As Reqd. As Reqd.
 I Duplicate Record Print
 N Control Totals

T
 E
 R

0 MESSAGE RESPONSE

+
 C
 O
 N
 S
 O
 L
 E

1Group 1 Software -- Consumer Merge/Purge
Program DE60

8.27

0Sample OS JCL

```
- //STEP1 EXEC PGM=DE60
//SYS005 DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=133)
//SYS010 DD DSN=DUPLICAT.RECORDS,
// UNIT=TAPE,VOL=SER=TAPEIN,DISP=OLD,LABEL=(2,BLP),
// DCB=(DSORG=PS,RECFM=FB,LRECL=rrrr,BLKSIZE=bbbb)
//SYSUDUMP DD SYSOUT=A
//SYS004 DD *
```

.
.
Parameter Cards

.
.
/*

1Group 1 Software -- Consumer Merge/Purge
Program DE60

8.28

0Sample DOS JCL

```
- // JOB DE60
// TLBL TAPEIN
// ASSGN SYS010,X'180' (Input TAPEIN)
// ASSGN SYS005,X'00E' (Printer)
// ASSGN SYS004,X'00C' (Card reader)
// EXEC DE60
```

.
.
Parameter Cards

.
.
/*
/&

start of a new group). Two files whose carriage-control position
does not indicate the start of duplicate groups may also be compared,
+ —
on a record-to-record basis.

1Group 1 Software -- Consumer Merge/Purge 9.2
Program DE61

0Narrative

- The records read from the IA file can be restricted to only a portion of the file via the RECLIM card. Additionally, the output may be limited by selecting only every Nth unmatched group, and/or by specifying a maximum number of pages.

1Group 1 Software -- Consumer Merge/Purge 9.3
Program DE61

0HEADER Parameter Card

- | - | CARD COLUMNS | FIELD DESCRIPTION |
|---|--------------|--|
| - | 01 - 06 | PARAMETER KEYWORD - Must contain 'HEADER'. |
| 0 | 08 - 17 | DATE - May contain a date in the form MM/DD/YYYY. This date will appear on the output reports. If this field is left blank, the current date will be inserted. |
| 0 | 19 - 58 | RUN DESCRIPTION - May contain any run description the user wishes. This run description will appear on the output reports. |

1

1Group 1 Software -- Consumer Merge/Purge 9.5
Program DE61

0TAPExx Parameter Card

- TAPExx or DISKxx cards must be used to activate each input or output file. Following is a brief description of each of the possible files. See the System Overview section for a detailed description of the TAPExx/DISKxx card fields.

- File_Activators:

+

- | | |
|---|--|
| 0 | IA - Input First Duplicate Print File |
| | IB - Input Second Duplicate Print File |

- Notes:

+

- | | |
|---|---|
| 0 | If File IB is not activated, then the file IA records will all be printed, using the original carriage-control (and the headers defined by the HEADnx cards). |
|---|---|

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'CONTRL'.
0		The CONTRL card is used to specify those fields needed to control the matching, and to control the print output. All "positions" referenced are positions in the print line (that is, they do not account for the carriage-control position at the front of each record), which corresponds to the wa MOVES, UNPKs, and FORMATs were specified in progra DE60.
0	08 - 10	SYNCHRONIZATION STRING POSITION - May contain the position of a "synchronization string" within the input records. If present, then the input files will be sequence-checked on this string, and the memory for the file match will be synchronized with the File IA records on this string.
0		If this field is omitted, then no synchronization of the input files (other than equal duplicate groups) will be possible, and it is possible that the match memory will become filled with unmatched duplicate groups, which will then result in all subsequent File IA groups being treated as unmatched.
0		In general, whenever a possible synchronization string (the high-order field of the original sortation for the duplicate-elimination input) is present in the duplicate-print file, it should be specified here.
0	12 - 13	SYNCHRONIZATION STRING LENGTH - If columns 8-10 are not blank, then this field must contain the length of the synchronization string. The sum of this length and the position specified in columns 8-10 may not exceed 133.

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	15 - 17	COMPARISON SUBSTRING POSITION - May contain the starting position of a "comparison substring" with the input records.
0		If no comparison substring is specified, then the record comparison will be performed on positions 1 through 132, inclusive. In some circumstances this will result in "undesirable" differences being detected and printed. For instance, if match scores, driver indicators, priority numbers, or other elements are stored by the dupe-identification program, and selected for print in DE60, then a minor variation in the set-up for duplicate elimination can result in a difference in the print lines even though the actual records matched remain the same.
0		In general, best results will be obtained if a comparison substring is specified which identifies the individual records (the name, address, etc.), but does not include scores, indicators, and so forth, which were stored by the dupe-identification program.
0	19 - 21	COMPARISON SUBSTRING LENGTH - If columns 15-17 are not blank, then this field must contain the length of the comparison substring. The sum of this field and columns 15-17 may not exceed 133.
0	23 - 25	PAGE LIMIT - May contain the maximum number of pages which are to be printed. No duplicate group will be displayed if it starts on a page higher than the number entered here.
0		If this number is omitted, then a very large report can be produced if the two files differ to a great extent.
0		If necessary, this field may be extended one position to the left, allowing the entry of a four-digit limit in columns 22-25.
0	27 - 29	POSITION FOR PAGE NUMBER - May contain the print position (within the header defined by the HEAD1x cards) at which will be placed a page number. If this field is omitted, no page number will be printed.

1Group 1 Software -- Consumer Merge/Purge 9.9
Program DE61

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	31 - 31	NOT A DUPLICATE FILE - Must contain one of the following codes to indicate whether matching is to be performed on duplicate groups or on single records:
0		blank - duplicate groups are to be matched
		X - single records are to be matched

1

1Group 1 Software -- Consumer Merge/Purge 9.11
Program DE61

0HEADxx Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 04	PARAMETER KEYWORD - Must contain 'HEAD'. This card can be used to define up to three lines of headings for the report.
0	05 - 06	HEADING LINE DESIGNATOR - Must contain one of the following codes to represent the Heading line designator:
0		1A = left side, heading line 1. 1B = right side, heading line 1. 2A = left side, heading line 2. 2B = right side, heading line 2. 3A = left side, heading line 3. 3B = right side, heading line 3.
0		Left side represents print positions 1-66. Right side represents print positions 67-132.
0	08 - 73	DATA TO PRINT - Must contain the data that is to be printed in the corresponding print positions for each side of each heading line
-	Note:	When the program encounters the first occurrence of the keyword 'MM/DD/YYYY' in the 'HEADxx' parameter cards, it will replace it with the value specified in columns 8 - 17 of the 'HEADER' parameter. If columns 8 - 17 are left blank, then it will use the current date. If the keyword 'MM/DD/YYYY' is not found in the 'HEADxx' parameter cards, then nothing will be altered.

0RECLIM Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'RECLIM'. The RECLIM card may optionally be used to limit the records processed for a particular run. A number of records to be skipped may be specified, as well as a number of records to process.
0		The most frequent use of the RECLIM card is to restrict the file to a small number of records, for parameter checkout or calibration. It can also be used, of course, to break what would otherwise be an excessively large job into several smaller independent executions without dividing the input files.
0	08 - 16	RECORDS TO SKIP - May contain the number of records to skip before processing begins. If omitted, processing will begin with the first record. Note: This option will only function as indicated + + + when a single input file is used.
0	18 - 26	RECORDS TO PROCESS - May contain the number of records to be processed. If omitted, all records will be processed (once the specified number are skipped).
-	Notes:	
0		The RECLIM card is applied to the File IA records as they are read. It may result in the first and/or last duplicate group printed being truncated.
0		If no synchronization field is specified in the CONTRL card, then the first field of this card should be omitted (or else, all groups will appear to be unmatched).

ONTHSEL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'NTHSEL'.
0		Note: There are two possible ways of specifying an Nth record include or exclude. In the first method presented, the idea is to specify the Nth number and have the program include or exclude each Nth record. This method is particularly suited to sampling operations.
0		The second method is used when a specific number of records must be processed, and when the exact count of the input file is known.
0		Method 1: Sampling
0	08 - 10	NTH NUMBER - Must contain the "Nth" number of records the user wishes to either include or exclude from the records to be printed. This may be a number between 002 and 999 inclusive.
0	12 - 14	INC/EXC - Must contain either 'INC' or 'EXC' to either INclude or EXclude the Nth number of records specified in columns 8-10 of this card.
0		Method 2: Precise number of records
0	08 - 14	FRACTION OF RECORDS - Must contain the fraction of the total input desired, expressed as a seven-digit positive number with a decimal point understood at column 7.
0		Example: Suppose that the input file is known to have exactly 153,714 records, and that our problem is to print exactly 3000 records. Using a calculator, we find that $3000/153714 = 0.0195168$; therefore we enter '0195168' in columns 8-14 and out come the 3000 records.
-	Notes:	
+	_____	
0		The Nth selection will be applied to duplicate groups which have been determined to be unmatched, and therefore selected for printing.

0Operating Instructions

- PROGRAM FUNCTION: DUPLICATE FILE PRINT PROGRAM NAME: DE61
0 SYS NR/ DEVICE I/O
DDNAME ADDRESS FILE DESCRIPTION/LABEL RET. FILE ID DISPOSITIO
+
T
A SYS010 I First Duplicate File TAPEIA
P SYS011 I Second Duplicate File TAPEIB
E

o
r

D
I
S
K

U
N
I
T
S

0 SYS NR/
DDNAME DESCRIPTION OF CARD DISPOSITIO
+
0 R SYS004 Parameter Cards
P

0 SYS NR/ FORM CARRIAGE
DDNAME DESCRIPTION OF REPORT NUMBER TAPE DISPOSITIO
+
P

R SYS005 Parameter List As Reqd. As Reqd.
I Unmatched Duplicate Groups
N Control Totals

T
E
R

0 MESSAGE RESPONSE
+
C
O
N
S
O
L
E

1Group 1 Software -- Consumer Merge/Purge
Program DE61

9.19

0Sample OS JCL

```
- //STEP1 EXEC PGM=DE61
//SYS005 DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=133)
//SYS010 DD DSN=FIRST.DUPE.PRINT.FILE,
// UNIT=TAPE,VOL=SER=TAPEIA,DISP=OLD,LABEL=(2,BLP),
// DCB=(DSORG=PS,RECFM=FBA,LRECL=0133,BLKSIZE=bbbb)
//SYS011 DD DSN=SECOND.DUPE.PRINT.FILE,
// UNIT=TAPE,VOL=SER=TAPEIB,DISP=OLD,LABEL=(2,BLP),
// DCB=(DSORG=PS,RECFM=FBA,LRECL=0133,BLKSIZE=bbbb)
//SYSUDUMP DD SYSOUT=A
//SYS004 DD *
```

.
.
Parameter Cards

/*

1Group 1 Software -- Consumer Merge/Purge
Program DE61

9.20

0Sample DOS JCL

```
- // JOB DE61
// TLBL TAPEIA
// TLBL TAPEIB
// ASSGN SYS010,X'180' (Input TAPEIA)
// ASSGN SYS011,X'181' (Input TAPEIB)
// ASSGN SYS005,X'00E' (Printer)
// ASSGN SYS004,X'00C' (Card reader)
// EXEC DE61
```

.
.
Parameter Cards

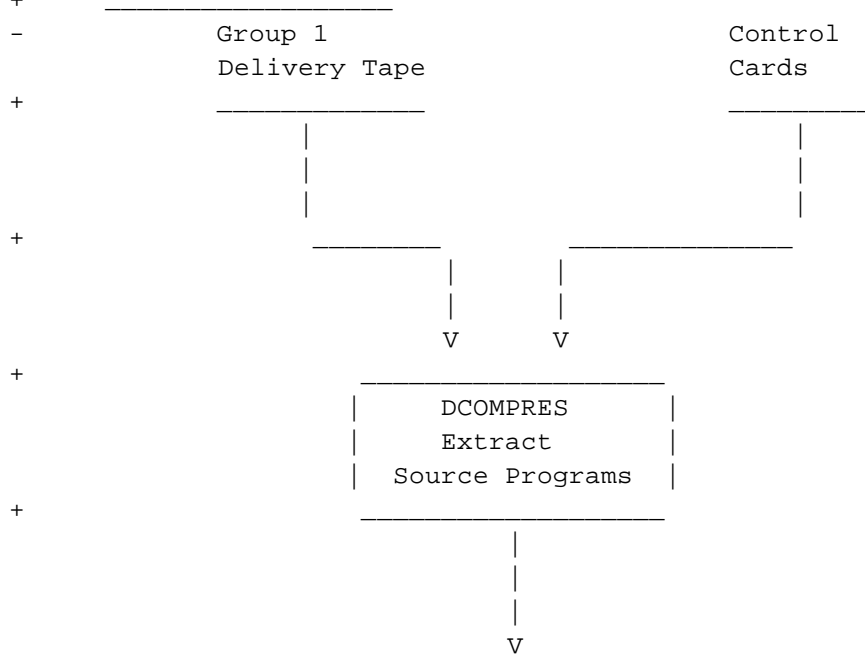
/*

/&

DCOMPRES Program

Extract Programs from Delivery tape

Narrative_and_flow:



Source programs and JCL/Control cards mixed on Tape, Disk, or Card Punch

Extract_Source_Program(s)

0 DCOMPRES reads and tables a series of INSERT and/or COPY control statements. If all Control cards are valid, it then reads the Group 1 Delivery Tape and honors the control statements in order of occurrence. Each INSERT statement causes a single card to be inserted into the output stream, and each COPY statement causes an entire source program (or text file) to be inserted into the output stream from the Delivery Tape. The output may be directed either to the card punch (by default), or to a Tape or Disk file (by use of a TAPEPN parameter card anywhere in the control card stream). OS users, of course, control the device type through the JCL provided. Whether or not a TAPEPN card is provided, the DDname, or logical assignment, of the output stream is SYS003.

0 Since the files to be decompressed are always the third, fourth, or fifth files on the Delivery Tape, OS users will need to specify (3,NL), (4,NL), or (5,NL) in the JCL, and DOS and OS/3 users will need to position the tape by MTC commands or the equivalent (and place an 'X' in column 26 of the TAPEIN card to specify no rewind).

0 In DOS shops which have ACLR (Access Control, Logging & Reporting), reading a no-label tape with no-rewind specified will be found to

present a problem. In this case, proceed as follows:

1Group 1 Software -- Consumer Merge/Purge 10.2
DCOMPRES Program

- Copy the third (fourth or fifth) file from the delivery tape to another no-label tape (as the first file), using DITTO or the equivalent.
0 Then use the new no-label tape as input to the DCOMPRES program specifying no labels but not specifying the no-rewind option.

+
1Group 1 Software -- Consumer Merge/Purge 10.3
DCOMPRES Program

OHEADER Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'HEADER'.
0	08 - 17	DATE - May contain a date in the form MM/DD/YYYY. This date will appear on the output reports. If this field is left blank, the current date will be inserted.
0	18 - 43	RUN DESCRIPTION - May contain any run description the user wishes. This run description will appear on the output reports.

1
1Group 1 Software -- Consumer Merge/Purge 10.5
DCOMPRES Program

OTAPExx Parameter Card

- TAPExx or DISKxx cards must be used to activate each input or output file. Following is a brief description of each of the possible files. See the System Overview section for a detailed description of the TAPExx/DISKxx card fields.

- File_Activators:

+
0 IN - The Group 1 Delivery Tape. The record size and block size should be specified as shown on the listing which accompanies the delivery tape.

0 PN - The output stream of control/source or text card-images. The record size is verified and must be 80.

- Note: If File PN is omitted, then the output stream will be

+
directed to a card punch, using the same (SYS003) assignment.

1Group 1 Software -- Consumer Merge/Purge 10.7

DCOMPRES Program

0CONTRL Parameter Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'CONTRL'.
0	08 - 10	SEQUENCE NUMBER TREATMENT - May contain a value to indicate the desired treatment of the sequence number field (columns 73-80) of the files to be extracted, as follows: 0 TXT - the file is a text file (a book), and the sequence number field is to be left as is nnn - (a numeric value) the sequence number field is to be incremented by this value, and the number for the first card is to be this value blank - the default treatment is '100'.

1

1Group 1 Software -- Consumer Merge/Purge 10.9

DCOMPRES Program

0INSERT Control Card

-	CARD COLUMNS	FIELD DESCRIPTION
-	01 - 06	PARAMETER KEYWORD - Must contain 'INSERT'.
0	08 - 71	TEXT OF CARD - Must contain the text which is to make up the output card-image, starting at column 1.
0	72 - 72	CONTINUATION COLUMN - Must contain the character which is to be placed in the output card-image in column 72 (normally the continuation column).

0COPY Control Card

- 01 - 04 PARAMETER KEYWORD - Must contain 'COPY'.
0 06 - 08 FILE NUMBER - Must contain the File number
 of the program or text file to be copied. Note
 that this is not the number of the file on
+ _____ the tape which contains all the programs (3), or
 all the text files (4), but the number of the
 program within file 3 on the tape, or the number
 of the book within file 4.
0 The listing accompanying the Delivery Tape will
 show the positions (file numbers) of the source
 or text files within each of the physical files
 on the tape. If the listing has been lost or
 destroyed, a new copy may be obtained by printing
 the first (physical) file on the tape. It consist
 of 80-byte card images blocked 8000.
0 10 - 17 FILE NAME - Must contain the name of the file to
 be copied, as it appears on the printed listing.
 The File name is checked against the File Number,
 and a mismatch condition will cause execution to
 terminate.
0 Note: As many COPY statements as desired may be provided, interspersed
+ _____ with INSERT statements if desired. However, the File Numbers of
 multiple files to be copied must be presented in ascending
 sequence, and no File Number may be repeated.

0Operating Instructions

- PROGRAM FUNCTION: Extract Source or Text PROGRAM NAME: DCOMPRES
0 SYS NR/ DEVICE I/O
DDNAME ADDRESS FILE DESCRIPTION/LABEL RET. FILE ID DISPOSITIO

+
T
A SYS010 I Group 1 Delivery Tape, TAPEIN
P File 3 or 4
E
SYS003 O Output stream TAPEPN
o
r

D
I
S
K

U
N
I
T
S

0 SYS NR/
DDNAME DESCRIPTION OF CARD DISPOSITIO

+
0 R SYS004 Control Cards
P SYS003 Output Stream if no TAPEPN

0 SYS NR/ FORM CARRIAGE
DDNAME DESCRIPTION OF REPORT NUMBER TAPE DISPOSITIO

+
P
R SYS005 Parameter List and Execution
I Log
N
T
E
R

0 MESSAGE RESPONSE
+

C
O
N
S
O
L
E

1Group 1 Software -- Consumer Merge/Purge
DCOMPRES Program

10.14

0Operating Instructions

- PROGRAM FUNCTION: Extract Source or Text PROGRAM NAME: DCOMPRES
- Additional Operating Instructions:
- 0 Mount files as indicated, put proper forms in the printer, put JCL and parameter cards in the card reader, and execute the program.
- 0 The parameters are read, edited, and listed on the printer. If errors are detected in the parameters, the parameters with errors are flagged, the tape files will not be opened, and the program will be terminated. If the parameters are error-free, the tape files will be opened and processing will continue until all Control Statements have been honored.

1Group 1 Software -- Consumer Merge/Purge
DCOMPRES Program

10.15

0Sample DOS JCL

- The following illustrates the use of the DCOMPRES program to create an unblocked tape that can serve as a SYSIN stream to assemble program MW210, the 14th program on a standard Group 1 delivery tape.
- 0

```
// JOB MAKE TAPE FOR MW210 ASSEMBLY
// ASSGN SYS003,X'280'           UNBLOCKED TAPE
// ASSGN SYS004,X'00C'           READER
// ASSGN SYS005,X'00E'           PRINTER
// ASSGN SYS010,X'281'           DELIVERY TAPE
// MTC REW,SYS010
// MTC FSF,SYS010,2
// EXEC DCOMPRES
HEADER          MAKE ASMBLY TAPE - MW210
CONTRL 100
TAPEIN320048000VN 2 TAPE X
TAPEPN 0080 080 N 2 TAPE
INSERT // JOB ASSEMBLE MW210
INSERT // OPTION CATAL
INSERT PHASE MW210,*
INSERT // EXEC ASSEMBLY
COPY 014 MW210
INSERT /*
INSERT // EXEC LNKEDT
INSERT /*
INSERT /&
/*
/&
```

0Sample OS JCL

- The following illustrates the use of the DCOMPRES program to submit a job that will replace programs CR03, CR04, and M\$05 (programs 3, 4, and 6 within physical file 3) in a source library. The programs are extracted by DCOMPRES, and embedded in the JCL stream as in-line files which will be stored as members by the job submitted.

```
0 //DCOMPRES EXEC PGM=DCOMPRES
//STEPLIB DD as appropriate
//SYS003 DD SYSOUT=(A,INTRDR),
// DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//SYS004 DD *
HEADER LOAD THREE PROGRAMS
CONTRL 010
TAPEIN320048000VN 2 TAPE X
INSERT //COPY3 JOB as appropriate
INSERT //COPY PROC P=XXX
INSERT //*
INSERT //COPYPGM EXEC PGM=IEBGENER
INSERT //SYSPRINT DD SYSOUT=A
INSERT //SYSIN DD DUMMY
INSERT //SYSUT2 DD DSN=GROUP1.SOURCE(&P), X
INSERT // DISP=OLD
INSERT // PEND
INSERT //*
INSERT //CR03 EXEC COPY,P=CR03
INSERT //COPYPGM.SYSUT1 DD *
COPY 003 CR03
INSERT //CR04 EXEC COPY,P=CR04
INSERT //COPYPGM.SYSUT1 DD *
COPY 004 CR04
INSERT //M$05 EXEC COPY,P=M$05
INSERT //COPYPGM.SYSUT1 DD *
COPY 006 M$05
//SYS005 DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=133)
//SYS010 DD UNIT=2400-3,VOL=SER=CMnnnn,LABEL=(3,NL),
// DSN=PROGRAM.FILE,DISP=(OLD,PASS),
// DCB=(DSORG=PS,RECFM=VB,LRECL=8000,BLKSIZE=32004)
//SYSUDUMP DD SYSOUT=A
//
```

0Sample OS JCL

- The following is a simpler example, in which DCOMPRES is used to copy program IX01, the 17th program in the third physical file, directly into a source library.

```
0 //DCOMPRES EXEC PGM=DCOMPRES
//STEPLIB DD as appropriate
//SYS003 DD DSN=GROUP1.SOURCE(IX01),DISP=OLD
//SYS004 DD *
HEADER COPY IX01
CONTRL 100
TAPEIN320048000VN 2 TAPE X
DISKPN 4240 080 S 2
COPY 017 IX01
//SYS005 DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=133)
//SYS010 DD UNIT=2400-3,VOL=SER=CMnnnn,LABEL=(3,NL),
// DSN=PROGRAM.FILE,DISP=(OLD,PASS),
// DCB=(DSORG=PS,RECFM=VB,LRECL=8000,BLKSIZE=32004)
//SYSUDUMP DD SYSOUT=A
//
```


1Group 1 Software -- Consumer Merge/Purge 11.1

File Layouts and Sample Runs

0Extract Area Format

- File Name: Consumer Merge/Purge Extract Area Format
0 Field Description/Contents Format Length Positions
+

0General Information:

0	Input Tape ID (A/B/C)	C	1	001 - 001
	File Code	C	9	002 - 010
	ZIP Code	C	6	011 - 016
	(for internal use)	X	8	017 - 024
	Uniqueness Number	B	4	025 - 028
	Priority Number:	B	2	029 - 030
	0 = Purge File Record			
	1-9998 = Priority (1 = Highest)			
	9999 = Unknown File Code			
	Randomization Number	B	2	031 - 032
	Special Treatment Indicator:	C	1	033 - 033
	blank = Normal			
	P = Purge File			
	S = No Multi-Buyer Contribution			
	Driver/No-Driver Switch	C	1	034 - 034
	Dupe-Group Driver Indicator:	C	1	035 - 035
	X = Dupe-Group driving record			
	blank = Not the driving record			
	Selection Indicator:	C	1	036 - 036
	blank = not involved in duplication			
	S = Single-buyer survivor			
	M = Multi-buyer survivor			
	0 = Purge-file drop			
	1 = Single-buyer drop			
	9 = Multi-buyer drop			
	Nr of records in duplicate group	B	2	037 - 038
	Degree of Surname/Address match	X	2	039 - 040
	Degree of Individual match	X	1	041 - 041
	Multi-Buyer Level	B	1	042 - 042
	Indicator Bits	B	1	043 - 043
	(reserved)	X	5	044 - 048

File Layouts and Sample Runs

0Extract Area Format

0	Field Description/Contents	Format	Length	Positions
+-----+-----+-----+-----				
-Typed-Field Information:				
	Street Address:		31	049 - 079
	1-byte length of house number	X		
	10-byte house number	C		
	2-byte directional	C		
	14-byte street name	C		
	4-byte suffix	C		
	Apartment Number	C	12	080 - 091
	4-byte type designator			
	8-byte number			
	PO Box	C	10	092 - 101
	Rural Route/Box:	C	13	102 - 114
	3-byte rural route number			
	10-byte box number			
	Individual Name:	C	31	115 - 145
	13-byte surname			
	3-byte first name			
	3-byte middle name			
	3-byte maturity title			
	1-byte gender code (from first, middle, and maturity title)			
	A - ambiguous			
	B - blank			
	C - company			
	F - female			
	I - initials only			
	M - male			
	U - unrecognized			
	7-byte title of respect			
	1-byte sex code (combination of title and gender code)			
	A - ambiguous			
	B - blank			
	C - company			
	D - dual sex (MR & MRS, etc.)			
	F - female			
	I - initials only			
	M - male			
	U - unrecognized			
	X - title/gender conflict (MR RUTH, for example)			
	Formatted Name	C	30	146 - 175
	First initial as entered	C	1	176 - 176
	Second initial as entered	C	1	177 - 177
	Business Title	C	15	178 - 192
	Bit Indicators to expedite comparisons	X	6	193 - 198
	Surname length	X	1	199 - 199
	(for internal use)	X	2	200 - 201

	Formatted name length	X	1	202 - 202
	(reserved)	X	6	203 - 208
0	Record Area			
0	Name-and-Address Record:	X		209 - nnn

1Group 1 Software -- Consumer Merge/Purge 11.3

File Layouts and Sample Runs

0Duplicate Elimination Summary Statistics TAPEOT SYS018
- File Name: Duplicate Elimination File-Code Statistics
0 Block Size: Record Size: 80 Mode: FB
0 Label Records: Recording Medium: sequential
0 Data Field Field
Field Description/Contents Format Length Positions
+
0 Record ID ('DE ') C 3 001 - 003
0 Priority 9 4 004 - 007
0 File Code C 9 008 - 016
0 Special Handling Flags C 4 017 - 020
0 Description C 20 021 - 040
0 Dupe-Elim Input 9 8 041 - 048
0 Purge Drops 9 8 049 - 056
0 Multi-Buyer Drops 9 8 057 - 064
0 Single-Buyer Drops 9 8 065 - 072
0 Multi-Buyer records out 9 8 073 - 080
1

1Group 1 Software -- Consumer Merge/Purge 11.5

File Layouts and Sample Runs

0Duplicate Elimination State-Count by File Code TAPEOX SYS019
- File Name: Duplicate Elimination State-Count Statistics
0 Block Size: Record Size: 92 Mode: FB
0 Label Records: Recording Medium: sequential
0 Data Field Field
Field Description/Contents Format Length Positions
+
0 Low 3-digit ZIP Code C 3 001 - 003
0 High 3-digit ZIP Code C 3 004 - 006
0 State/Province Standard Abbreviation C 2 007 - 008
0 State/Province Name C 19 009 - 027
0 File Code C 9 028 - 036
0 Priority C 4 037 - 040
0 Special Handling Flags C 4 041 - 044
0 Description C 20 045 - 064
0 Dupe-Elim input B 4 065 - 068
0 Purge Drops B 4 069 - 072
0 Multi-Buyer Drops B 4 073 - 076
0 Single-Buyer Drops B 4 077 - 080
0 Single-Buyer records out B 4 081 - 084
0 Multi-Buyer records out B 4 085 - 088
0 Total Output records B 4 089 - 092

1Group 1 Software -- Consumer Merge/Purge 11.7

File Layouts and Sample Runs

0List Conversion Summary Statistics

- File Name: List Conversion Summary Statistics
0 Block Size: Record Size: 80 Mode: FB
0 Label Records: Recording Medium: sequential
0 Data Field Field
Field Description/Contents Format Length Positions
+
0 Record ID ('LC ') C 3 001 - 003
0 File Code C 9 004 - 012
0 Description C 20 013 - 032
0 Conversion Input 9 8 033 - 040
0 Rejects 9 8 041 - 048
0 Conversion Output 9 8 049 - 056
0 (reserved) - 24 057 - 080

1
1Group 1 Software -- Consumer Merge/Purge 11.9
File Layouts and Sample Runs

0Sample Runs

- Following is an Consumer Merge/Purge run. Additional File
Layout specifications may be placed in front of this page if
desired.

1
1Group 1 Software -- Consumer Merge/Purge 11.11
File Layouts and Sample Runs

+
- Replace this page with Sample Consumer Merge/Purge run.

-	Index	
+	-----	
0		
-	-- A --	
	Accumulation of information.....	3.79
	ADDRDF Card.....	3.31
	ALGSEL Card.....	3.37
	ANALIM Card.....	7.23
-	-- B --	
	BRKFLD Card.....	3.15
	BRNAME Card.....	5.10
	BROKER Card.....	5.9
-	-- C --	
	Canadian addresses.....	1.10
	CHANGE Card.....	3.71
	Checkpoint	
	CHKPT Card.....	3.99, 7.27
	Restart.....	3.100, 7.28
	clean records	
	splitting between Files OM and OE.....	3.23
	storing Multi-Buyer level.....	3.58
	storing records in group.....	3.58
	storing relative priority code in.....	3.57
	storing selection indicator in.....	3.57
	storing uniqueness code in.....	3.57
	CLNREC Card.....	3.57
	CMMODS Card.....	2.24
	CMPFLD Card.....	3.35
	CMPLIM Card.....	3.77
	Comparison of records	
	ADDR Algorithm.....	3.48
	APNR Algorithm.....	3.47
	general control of.....	3.37
	HHLD Algorithm.....	3.49
	HSNR Algorithm.....	3.43
	INDV Algorithm.....	3.42
	POBX Algorithm.....	3.45
	RESI Algorithm.....	3.51
	restricting.....	3.67, 3.77
	RRBX Algorithm.....	3.46
	special fields.....	3.35
	STNM Algorithm.....	3.44
	SURN Algorithm.....	3.41
	TOTL Algorithm.....	3.50
	ZIPC Algorithm.....	3.40
	CONSTn Card.....	4.11, 8.11
	Consumer Merge/Purge	
	Introduction.....	1.0
	Organization of the manual.....	1.1

Program Descriptions.....	1.7
Theory and Practice.....	1.3

-Index

-CONTRL Card

CM30.....	3.19
CM31.....	4.7
DCOMPRES.....	10.7
DE40.....	5.7
DE41.....	6.7
DE50.....	7.7
DE60.....	8.7
DE61.....	9.7

Conversion Summary Statistics

used in DE40.....	5.5
used in DE50.....	7.5

COPY Control Card

DCOMPRES.....	10.11
---------------	-------

-

-- D --

DCOMPRES Program.....	10.1
-----------------------	------

degree of match

storing in duplicate records.....	3.54
-----------------------------------	------

Delivery Tape

Extracting Programs from.....	10.1
-------------------------------	------

Direct Access Master File

VSAM.....	2.7
-----------	-----

Discussions

Auxiliary Parameter File.....	2.26
Disk Cards.....	2.3
exit routines	
EXITOP.....	2.19
TAPE/DISK exits.....	2.7
user exits.....	3.87

Parameter Cards

auxiliary file.....	2.26
grouping.....	2.26
Tape Cards.....	2.3
VSAM Master Files.....	2.7

DISKIP.....	2.26
-------------	------

DISKxx Card (see also TAPE/DISK Card).....	2.3
--	-----

DUPCTL Card.....	3.53
------------------	------

duplicate control character

storing.....	3.53
used in DE50.....	7.7
used in DE60.....	8.7

Duplicate Elimination

duplicate file match (DE61).....	9.1
duplicate file print (DE60).....	8.1
extracted file print (CM31).....	4.1
program (CM30).....	3.1
State counts by File Code (DE41).....	6.1
summary report (DE40).....	5.1

duplicate records
posting one record from another.....3.79
storing degree of equality in.....3.54
storing duplicate-control character in.....3.53

-Index

- storing Multi-Buyer level.....3.56
- storing records in group.....3.56
- storing relative priority code in.....3.55
- storing selection indicator in.....3.55
- storing uniqueness code in.....3.55
- DUPREC Card.....3.55

-- E --

Exit Routines

- activation of.....3.87
- EXITOP Card.....2.19

Extract Area

- displaying.....4.1
- Format.....11.1

Extracted File Format

- creation of.....3.21
- for Extra File.....3.24
- use of as input.....3.21

-- F --

File Activators

- CM30.....3.9
- CM31.....4.5
- DCOMPRES.....10.5
- DE40.....5.5
- DE41.....6.5
- DE50.....7.5
- DE60.....8.5
- DE61.....9.5

File Code

- assigning Priority to.....3.59, 3.65, 7.17
- Assigning to List Brokers.....5.9
- changing during dupe-elim.....3.25, 3.71
- defining Purge File Codes.....3.66
- definition and control of.....3.65
- duplicate elimination summary by (DE40).....5.1
- entering a description of, for reports.....3.68
- ignoring specific.....3.68
- location of in Name/Address records.....3.27
- placing in output records.....3.25
- sequence of on reports.....3.24, 5.7, 7.10, 7.23
- State counts by (DE41).....6.1
- to control RESIDENT treatment.....3.68
- treatment of unknown.....3.24

File Code vs File Code

- Match Analysis.....7.1

File layout

- Conversion Summary Statistics.....11.7
- Extract Area.....11.1

File-Code Statistics.....	11.3
State-Count Statistics.....	11.5

1Group 1 Software -- Consumer Merge/Purge	12.4
-Index	
-File Priority	
assignment of.....	3.59, 3.65
File-Code Statistics	
layout.....	11.3
used in DE40.....	5.5
used in DE50.....	7.5
FLCnmm Card.....	3.65
FLCODE Card.....	3.27
FLCSEL/FLCREJ Card.....	4.21, 5.11, 6.9, 7.25, 8.21
footers for reports	
user-specified.....	2.17
FORMAT Card.....	4.15, 8.15
-	-- G --
General Information.....	2.1
-	-- H --
HEADER Card.....	5.3, 6.3, 7.3, 8.3, 9.3, 10.3
HEADER Parameter	
Start/Stop Postal Code.....	3.7, 4.3
headings for reports	
user-specified.....	2.17
HEADxx Card.....	4.9, 8.9, 9.11
Household duplicate elimination	
versus Individual.....	3.42
-	-- I --
Indexed Master File	
VSAM.....	2.7
Individual duplicate elimination	
Company name option.....	3.43
specification of.....	3.43
versus Household.....	3.42
INSERT Control Card	
DCOMPRES.....	10.9
INTRPT Card.....	3.97
IOMODS Card.....	2.23
-	-- K --
Key Code (see File Code).....	3.65
-	-- L --
List Broker	
assigning File Codes to.....	5.9
associating name with code.....	5.10
report for.....	5.10
List Code (see File Code).....	3.65

-Index

- -- M --

Match Analysis.....7.1

Match Matrix.....7.1

Memory Load

 controlling size of.....3.19

Memory requirements.....2.27

 Adjustment of.....2.29

MERGES Card.....3.11

MLTLIM Card.....3.75, 7.15

MOVExx Card.....4.13, 8.13

Multi-Buyer level

 counts by.....7.1

 in clean records.....3.58

 in duplicate records.....3.56

Multi-Buyer records

 defined.....3.75, 7.15

 elimination of.....3.23

 identification of.....3.66, 3.75, 7.15

 isolated on a file.....3.23, 7.2

Multi-Buyer Report.....7.1

MULTIB Card.....3.73, 7.13

- -- N --

NAMEDF Card.....3.33

Names

 elimination of duplicates.....3.23

Multi-Buyer level

 storing in clean records.....3.58

 storing in duplicate records.....3.56

records in group

 storing in clean records.....3.58

 storing in duplicate records.....3.56

relative priority

 storing in clean records.....3.57

 storing in duplicate records.....3.55

selection indicator

 storing in clean records.....3.57

 storing in duplicate records.....3.55

Sequence of

 definition (MERGES).....3.11

 definition (SEQCHK).....7.19

 error handling (SEQERR).....3.13, 7.21

 numbering (SEQUN).....3.85

uniqueness code

 storing in clean records.....3.57

 storing in duplicate records.....3.55

Nth record selection.....4.23, 8.23, 9.15

NTHSEL Card.....4.23, 8.23, 9.15

- -- O --

Operating instructions

CM30.....3.101

-Index

-	CM31.....	4.25
	DCOMPRES.....	10.13
	DE40.....	5.13
	DE41.....	6.11
	DE50.....	7.29
	DE60.....	8.25
	DE61.....	9.17
	Operator interruptible.....	3.97
-		
	-- P --	
	PAGESZ Card.....	2.15
	Parameter Cards	
	auxiliary file.....	2.26
	grouping.....	2.26
	Parameter List	
	EXITOP.....	2.19
	for TAPE/DISK Exits.....	2.7
	user exits.....	3.89
	Partition sizes.....	2.27
	PEXITS Card.....	3.87
	POST Card.....	3.79
	Priority	
	assigning to File Codes.....	3.65
	assignment algorithm selection.....	3.59
	effect of on Purge File records.....	3.59
	for RESIDENT names.....	3.61
	modification of for missing elements.....	3.62
	PRIORT Card.....	3.59, 7.17
	Programs	
	memory requirements.....	2.27
	names of.....	2.24
	Purge File Code.....	3.66
	purge-by-priority.....	3.59
-		
	-- R --	
	RECLIM Card.....	3.17, 4.19, 8.19, 9.13
	records in group	
	in clean records.....	3.58
	in duplicate records.....	3.56
	Region sizes.....	2.27
	Rejection	
	by File Code.....	4.21, 5.11, 6.9, 7.25, 8.21
	by Nth record.....	4.23, 8.23, 9.15
	by record count.....	3.17, 4.19, 8.19, 9.13
	Relative priority code	
	storing in clean records.....	3.57
	storing in duplicate records.....	3.55

Renaming programs.....	2.24
Renaming subroutines.....	2.23

-Index

-report footers

- user-specified.....2.17

report headings

- user-specified.....2.17

RESIDENT records

- priority of.....3.61

RESIDENT treatment

- based on the name in the record.....3.41
- by File Code.....3.68
- comparison of RESIDENT names.....3.51
- for all records.....3.41

- -- S --

Sample Exit Function B.....3.91

Sample Exit Function D.....3.92

Sample Exit Function L.....3.93

Sample Exit Function X.....3.94

Sample Exit Routine.....3.90

Sample JCL

- CM30.....3.103
- CM31.....4.27
- DCOMPRES.....10.15
- DE40.....5.15
- DE41.....6.13
- DE50.....7.31
- DE60.....8.27
- DE61.....9.19

Sample runs.....11.9

Selection

- by File Code.....4.21, 5.11, 6.9, 7.25, 8.21
- by Nth record.....4.23, 8.23, 9.15
- by record count.....3.17, 4.19, 8.19, 9.13

Selection indicator

- storing in clean records.....3.57
- storing in duplicate records.....3.55

SEQCHK Card.....7.19

SEQERR Card.....3.13, 7.21

Sequence

- of input Names
 - definition (MERGES).....3.11
 - definition (SEQCHK).....7.19
 - error handling (SEQERR).....3.13, 7.21

Sequence Number

- on output files.....3.85

SEQUNx Card.....3.85

SIZE parameter on EXEC.....2.27

Source Code (see File Code).....3.65

State Counts
 by File Code (DE41).....6.1
State-Count Statistics
 layout.....11.5

- Index
- used in DE41.....6.5
- Subroutines
 - names of.....2.23, 2.24
 - relocation (old DOS).....2.31
 - Static Linkedit (OS/3).....2.32
- Summary Report
 - by File Code.....5.1
 - by List Broker.....5.1, 5.9
- Suppression File (see Purge File).....3.66
- SYNCHK Card.....2.11
- System Flowchart.....1.11
- System Overview.....2.1
-
- T --
- TAPE/DISK Card.....3.9, 4.5, 5.5, 6.5, 7.5, 8.5, 9.5, 10.5
- Field Description.....2.3
- TAPEIP.....2.26
-
- U --
- UFTnx Card.....2.17
- UHDnx Card.....2.17
- Uniqueness code
 - storing in clean records.....3.57
 - storing in duplicate records.....3.55
- UNPKxx Card.....4.17, 8.17
- User Exit Routines
 - activation of.....3.87
- User Report Footers
 - defining.....2.17
- User Report Headings
 - defining.....2.17
-
- V --
- Verification
 - of file content.....2.11
- VSAM Master Files.....2.7
-
- Z --
- ZPCODE Card.....3.29