

A RAND NOTE

What's on the Year-At-Risk File

**Michael P. Murray, Daniel Relles,
Marygail Brauner, Grace Carter, Leola Cutler,
Deborah Skoller, Warren E. Walker**

March 1989

RAND

The research reported here was sponsored by the United States Air Force under Contract F49620-86-C-0008. Further information may be obtained from the Long Range Planning and Doctrine Division, Directorate of Plans, Hq USAF.

The RAND Publication Series: The Report is the principal publication documenting and transmitting RAND's major research findings and final research results. The RAND Note reports other outputs of sponsored research for general distribution. Publications of The RAND Corporation do not necessarily reflect the opinions or policies of the sponsors of RAND research.

A RAND NOTE

N-2744-AF

What's on the Year-At-Risk File

**Michael P. Murray, Daniel Relles,
Marygail Brauner, Grace Carter, Leola Cutler,
Deborah Skoller, Warren E. Walker**

March 1989

**Prepared for
The United States Air Force**

RAND

PREFACE

This Note was prepared as part of the Enlisted Force Management Project (EFMP). It documents the contents of a SAS¹ data file and the FORTRAN program that generates it, that is the primary source of data for building, testing, and maintaining the models in the Air Force's Enlisted Force Management System (EFMS). The current version of the file, called the Year-at-Risk (YARs) file, contains longitudinal data on every airman who was on regular active duty in the Air Force any time between June 30, 1971, and June 30, 1987.

This Note first documents the computer program that produces the file. It describes the tasks that the program carries out, the subroutines that carry out these tasks, and how to update the program to add data for additional years. It should prove useful to those in the Air Force who are responsible for maintaining and updating the program.

Following the documentation of the computer program is a description of version 5 of the YAR file (YAR5). The Air Force will update it every year with data on the airmen who were in the Air Force during the preceding 12 months. This document describes each of the variables on the YAR file and lists the possible data codes and their meanings.

The RAND Corporation maintains samples from the YAR. The Air Force (AFMPC/DPMDWF) has the entire file. A note describing the program that generates the file and how to maintain and update the program is being prepared as part of the EFMP.

The EFMP is a joint effort of the Air Force (through the Deputy Chief of Staff for Personnel) and The RAND Corporation. RAND's work falls within the Resource Management Program of Project AIR FORCE. The EFMP is part of a larger body of work in the program that is concerned with the effective utilization of human resources in the Air Force.

¹SAS is a software system for data analysis:
SAS Institute Inc.
Box 8000
Cary, North Carolina 26500-8000

ACRONYMS

AFMPC	Air Force Military Personnel Center. The Washington Area Information Management Division (AFMPC/DPMDW) is responsible for implementing the EFMS and maintaining and updating its models and data bases.
AFQT	Air Force Qualifying Test. A test that a prospective enlistee takes before he enlists. The score, which is given as a percentile ranking, is used as an indicator of the airman's "quality."
AFSC	Air Force Specialty Code. This code is used to identify specific jobs. The basic code is five digits. A letter prefix and/or letter suffix (called a "shred") is sometimes added to provide greater detail about the job.
CONUS	Continental United States
DMDC	Defense Manpower Data Center. An agency of the Department of Defense that maintains data bases related to manpower and personnel in the United States armed forces.
EAGL	Enriched Airman Gain/Loss file. A data file that contains longitudinal data on airmen on regular active duty in the Air Force.
EFMS	Enlisted Force Management System.
ETS	End of term of service. The data on which the airman fulfills his obligated active duty.
FIDO	File Item Data Overview file. A data file that contains the standard meanings of most of the codes used on Air Force data files, including the dates during which the codes were valid.
OETS	Original End of Term of Service.
PACE	Processing and Classification of Enlistees file. A data file that contains background information on airmen collected before, during, or shortly after Basic Military Training.
SAS	Statistical Analysis System.
SPD	Separation Program Designator.
TAFMS	Total Active Federal Military Service. This is the number of months of active military service accumulated by the airman up to the reference date.
USAF	United States Air Force.
YAR	Year at Risk. A year of service in the career of an airman. (See Sec. I for a more complete definition.)

CONTENTS

PREFACE	iii
ACRONYMS	v
FIGURE AND TABLES	ix
Section	
I. INTRODUCTION	1
The Year-at-Risk Concept	1
Tasks of the YAR Program	3
Organization of This Note	3
II. SUBROUTINES OF THE YAR PROGRAM	5
Input Subroutines	5
Housekeeping Subroutines	5
Subroutines that Structure an Airman's Terms of Service in YARs	7
Subroutines that Determine Socio-Economic and Military Status during Terms of Service	11
Output Subroutines	14
III. UPDATING THE YAR PROGRAM	15
IV. WHAT'S ON VERSION5 OF THE YAR (YAR5) FILE	17
Organization of Section	17
Pace Data Contained in Each YAR Record	19
Snapshot Data	22
Variables for This YAR	26
Variables for the Term of This YAR	32
Variables for the End of the Term of This YAR	37
Appendix	
A. HOW SNAPSHOTS ARE RECORDED FOR A YAR	43
B. HOW TO RECONSTRUCT THE FORCE AS OF A SPECIFIC JUNE	44
C. SUBROUTINES OF THE YAR PROGRAM	46
D. THE SUBROUTINE STRUCTURE OF THE YAR FORTRAN PROGRAM	49
E. CODE USED TO CONSTRUCT THE SAS VERSION OF THE YAR5 FILE	58
F. RAND RECODES OF TRANSACTION IDS (XTRIND)	62
REFERENCES	77

FIGURE

1. Relative locations of snapshot variables for a YAR that runs from July 1, 1971,
to July 1, 1981. 25

TABLES

1. Parameters requiring updating 16
2. The relationships among FY, YMCWRAT, and SYR. 29
- D.1. Who calls whom among YAR subroutines 50
- D.2. The complexity of YAR FORTRAN subroutines. 56

I. INTRODUCTION

The Year-at-Risk (YAR) file is a longitudinal SAS file containing information about individual airmen. It combines demographic, snapshot, and transaction data from the EAGL file (see Brauner, Walker, Murray, and Davidson, 1989) with historical data on such things as bonuses and unemployment rates (see Walker and McGary, 1989).

The file contains one record for each year at risk in an airman's career. The YARs in the YAR5 file include all those that began after 7106 and before 8707,¹ although years-at-risk beginning between 7107 and 7306 should not be used for loss modeling, since they frequently have missing data. The year-at-risk structure is designed to facilitate analyses of yearly losses in the EFMS's middle-term loss models; certain variables are included to permit analysis of monthly losses in the short-term loss models.

In addition to longitudinal analysis, the file will support some cross-sectional analysis of the enlisted force at the end of each June. For this purpose, there are records on the file called "dummy observations," which give snapshot information not contained in any year-at-risk record along with flags that indicate the type of record (see TRMDUM). A snapshot of the entire force as of a particular June can be obtained by selecting a subset of records from both the years at risk and these dummy observations, although only the snapshot variables can be used without additional manipulation. To obtain additional descriptions of the force as of a particular June (e.g., variables that pertain to the term of enlistment, such as its category, length, ending date, etc.), see App. B.

THE YEAR-AT-RISK CONCEPT

An example might clarify the "year-at-risk" concept. (For a second example, see App. A.) Suppose the terms for an airman were

¹The YAR5 file contains airman transactions for the period July 1, 1971, through June 30, 1987, and snapshots every June 30 from June 30, 1971, through June 30, 1986. In this Note we often code a year and month as a four-digit number, YYMM, where YY are the last two digits of the year and MM is the month. Thus 8707 refers to July 1987. Sometimes we refer to a date in the form yyxxx. In this case yy are the last two digits of the year and xxx is the number of days since the beginning of the year.

first term 7003---7403
second term 7403---7803
third term 7801---8201
fourth term 8105---8505
fifth term 8505---8905

Then the YARs that appear on the YAR5 file would be

YAR	Snapshot Year ²	Term ³
7203---7303	71	1
7303---7403	72	1
7403---7503	73	2
7503---7603	74	2
7603---7703	75	2
7703---7803	76	2
7801---7901	77	3
7901---8001	78	3
8001---8101	79	3
8101---8201	80	3
8105---8205	80	4
8205---8305	81	4
8305---8405	82	4
8405---8505	83	4
8505---8605	84	5
8605---8705	85	5
8707---xxxx ⁴	86	5

²The snapshot that appears is the one for the June preceding the beginning of the YAR.

³The term of a YAR is the term of service the airman was in during the year at risk.

⁴This is not a YAR but a dummy observation used to record the 1986 snapshot information. To create an analysis file that has no dummy observations, see App. A.

TASKS OF THE YAR PROGRAM

The tasks of the YAR FORTRAN program can be divided into five categories:

- Input
- Housekeeping
- Structuring an airman's term of service into YARs
- Determining an airman's socio-economic and military status during his terms of service
- Output.

The ninety-one subroutines of the program are devoted to accomplishing these tasks. Section II treats each of these tasks in turn.

ORGANIZATION OF THIS NOTE

The remainder of this Note describes the YAR FORTRAN program and defines the variables in the YAR5 file. Section II presents the subroutines that make up the YAR program, while Sec. III describes the updating of that program. There are three types of data on the YAR5 file. Section IV defines all variables on the file according to type.

There are six appendixes:

1. *Appendix A (How Snapshots Are Recorded for a YAR).* This appendix contains a second example of the year-at-risk concept and its relationship to an airman's term of service. Actual variables in the file and the values they assume for a particular airman are shown.
2. *Appendix B (How to Reconstruct the Force as of a Specific June).* Each observation in the YAR file contains the snapshot data for the June preceding the beginning of the year at risk. So, for example, if an airman began a second term on 7508, then the snapshot variables for the YAR 7508-7608 would come from the June 1975 snapshot. The "T" variables in the YAR record would contain information on the airman's second term. However, in 7506 when the snapshot was recorded the airman was serving in his first term. If an analyst wishes to reconstruct the force as of a specific June with "T" variables as they were in that June, he must recode some variables. This appendix contains the SAS code to accomplish this revision.

3. *Appendix C (Subroutines of the YAR FORTRAN Program).* This appendix defines the purpose of each of the ninety-one subroutines of the FORTRAN program.
4. *Appendix D (The Subroutine Structure of the YAR FORTRAN Program).* This appendix shows the logic structure of the YAR program.
5. *Appendix E (Code Used to Construct the SAS Version of the YAR5 file).* Because most of the data and statistical analysis using this file are expected to be performed using SAS, the YAR file was converted into a SAS file. Appendix E is a listing of the code that created the SAS file.
6. *Appendix F (RAND Recodes of Transaction IDs).* The variable XTRNID contains the code the Air Force assigned to the transaction that ended the term of the YAR. There are hundreds of these codes. (For example, there are over 200 codes for involuntary discharge.) To assist the analyst, these codes have been grouped into broad categories such as retirement, attrition (quality), etc. These recoded values appear on the file as XCTRN. Appendix F provides the mapping between XTRNID and NCTRN. It also provides a mapping between XTRNID and the end-of-term codes used in the Short-term Aggregate IPM (SAM). SAM is being developed for inclusion in the Air Force's Enlisted Force Management System (EFMS).

II. SUBROUTINES OF THE YAR PROGRAM

INPUT SUBROUTINES

The YAR FORTRAN program uses eight input data files:

1. EAGL file
2. Air Force Specialty Code (AFSC) file
3. Bonus file (which AFSCs got bonuses when)
4. CONUS file (was airman assigned overseas or not)
5. Civilian pay files (estimated civilian pay by AFSC)
6. Military pay file
7. SPD file (separation program designators)
8. Two unemployment rate files

The EAGL file is processed record by record, with YAR records being created for each airman sequentially. The other files are used as tables from which information is read repeatedly as needed. Thirty-seven subroutines associated with these activities are grouped in Appendix C as "file" subroutines. Another seven subroutines associated with these activities are grouped in Appendix C as "input" subroutines. Three of the "input" subroutines initialize the input arrays with -9's.

HOUSEKEEPING SUBROUTINES

The YAR FORTRAN program's first substantive tasks are a series of data cleaning exercises. Six subroutines perform these chores. These subroutines are grouped as "yar" subroutines in Appendix C. The subroutines

- adjust Date of Enlistment to ignore day of month (subroutine ADJDOE)
- impute Date of Enlistment from Total Active Federal Military Service if former is missing (subroutine FIXPER)
- indicate if missing grade, marital status, etc., in snapshot data (subroutine SNAPFX)

- sort transactions (subroutine SRTEX1)
- put extension, loss, and reenlistment transactions in proper sequence (subroutine SRTEX2)
- correct some errors in Total Active Federal Military Service (TAFMS) data (subroutine TFMSFX)

Also among the program's housekeeping chores is to initialize the derived variables for each airman. Subroutine UINIT, grouped among the "yar" subroutines in Appendix C, does the initializing.

The YAR program has frequent need for particular operations. A series of fifteen subroutines, grouped as "util" subroutines in Appendix C, perform these operations. The operations include:

- map yymm and yyxxx to yyxxx (subroutine ADATE)
- map yymm and yyxxx to yymm (subroutine CDATE)
- move characters, but replace ' ' with '-9' (subroutine GETPM9)
- move characters (subroutine GETPUT)
- convert character to integer (subroutine IVALUE)
- map TAFMS into fractional grade (subroutine MDLGRD)
- map yrmo (e.g. 7809) to months beyond Dec 1969 (subroutine M070)
- compare character strings (subroutine QEQUAL)
- map months beyond Dec 1969 to bonus period number (subroutine PERIOD)
- read fixed length record (subroutine GREAD)
- compute for a given date days since Jan 1, 1960 (subroutine \$10160)
- map yymm and yyxxx to yy for last June (subroutine SDATE)
- sort (subroutine SORTCQ)
- read variable length record; return record and length (subroutine VREAD)
- map TAFMS into bonus zones (subroutine ZONE)

SUBROUTINES THAT STRUCTURE AN AIRMAN'S TERMS OF SERVICE IN YARS

The heart of the YAR program lies in its imposition of the Year-at-Risk structure on the longitudinal data for each airman found in the EAGL file. This is done in two steps: (i) a term structure is imposed on an airman's career, and (ii) YARs are devised for each term in an airman's career. (All the subroutines referred to in this subsection are grouped among the "yar" subroutines in Appendix C.)

The EAGL file contains one variable length record for each airman who was in the Air Force during the time covered by the file. These records contain data about the airman at enlistment (data taken from the Air Force's PACE file), snapshot data from each June 30th during the airman's service career covered by the file (data taken from the Defense Manpower Data Center (DMDC) Master Files), and data gathered each time the airman reenlisted, extended, left the service, or returned to the service (data taken from the Air Force's Airman Gain/Loss file).

The YAR program examines the EAGL record for an airman and constructs a series of YAR records and dummy observations for the airman based on the EAGL data. This process is initiated by subroutine PNTCNT. (Devising the term and YAR structures for an airman's terms requires numerous dates and counts. Many subroutines in the program are made more complicated by the frequent need to overcome missing or inconsistent data by allowing for several ways to estimate any given date or count. The program is very "knowledgeable" about the EAGL data. If it were not, many good observations on airmen would be lost. The cost of this knowledge is that the YAR program can be maintained only by someone well versed in both the details of the EAGL data and the analytical uses to which the YAR file is put.)

Identifying the Term Structure of an Airman's Career

Each YAR is a one-year piece of one term of an airman's career. To identify an airman's YARs it is therefore necessary to identify the beginning and ending dates of the airman's terms of service. The YAR program identifies terms by scanning the reenlistment, extension, loss, and gain transactions contained in the airman's EAGL record.

Since each type of transaction (reenlistment, loss, gain, and extension) has different implications for the term structure of an airman's career, the PNTCNT subroutine contains code specific to each transaction type. (Each reenlistment or loss

transaction marks the end of one term of service. Each reenlistment transaction marks the beginning of a new term of service. Each gain transaction marks the beginning of a new term of service. Each extension transaction marks a deferring of the end of a term of service beyond its previously planned end date.)

The EAGL file distinguishes transactions in still another way: the last transaction for an airman (if any) prior to the date of the first snapshot available from the DMDC Master Files is called an "early transaction"; no earlier transactions are included in the EAGL file. The EAGL file stores the early transaction separately from subsequent transactions. The PNTCNT subroutine processes early transactions before other transactions and treats such transactions differently from subsequent transactions.

The differing treatment of early and subsequent transactions is necessary for accurate construction of the term structure of an airman's career. Airmen without early transactions must be either first termers or prior service accessions whose gain transaction will be in the EAGL file. For these airmen, identifying starting points for their term and counting the number of terms they serve is more straightforward than for airmen with early transactions. Airmen with early transactions may be in any term of service, and identifying when their most recent term began, or which term of service they are currently in is particularly difficult.

There are ten tasks carried out by subroutine PNTCNT:

1. identify the starting date for each term of service in an airman's career;
2. identify the ending date (if any) for each term of service in an airman's career;
3. identify the originally contracted ending date for each term of service in an airman's career;
4. determine whether a term of service ended in a loss or a reenlistment;
5. distinguish first term, second term, and career airmen;
6. associate each extension transaction with the term of service in which it occurred;
7. call up subroutine TRMYR to devise the YAR structure of an airman's career based on the term structure identified by subroutine PNTCNT;
8. add dummy observation records for each airman who appears in a snapshot not used in a YAR record (the dummy contains that snapshot's data);

9. call up subroutine MILPAY to create the socio-economic and military variables in the dummy observation records;
10. call up subroutine NXTLV to add to each YAR record the date, if known, on which the airman next leaves the service following the starting date of that YAR.

Since a new term begins when an old term ends, determining the beginning and ending dates for terms depends on the dates reenlistments and losses become official. A slight complication arises when an airman in extension status reenlists and receives credit towards the new obligation for time served in extension status. Careful examination of the initially-contracted date of separation for the new term (OETS) and the recorded length of obligation for the new term exposes these cases. A second complication arises when an airman leaves the service but reenlists shortly thereafter (as permitted by Air Force regulations). The PNTCNT subroutine checks for such cases by calling the subroutine QBAKSN. The beginning dates for terms are determined in subroutine SETBEG using data passed from subroutine PNTCNT.

Many airmen are still in the service when the EAGL file's transaction records end. For all these airmen, the actual end date of the term they are in at the EAGL file's end is unknown. The PNTCNT subroutine sets the ending date for those terms to -9. Unended first terms, unended terms that began with a prior service accession, and unended terms that began with a reenlistment are all treated separately.

Tracking the OETS requires care. Transaction data in the EAGL file reports: (a) the OETS for the new term in reenlistment transactions, and (b) the most recently contracted date of separation in loss transactions. Thus when a loss transaction or reenlistment transaction marks the end of a term of service, determining the OETS for the term ending requires the PNTCNT subroutine to look back to the previous reenlistment or gain transaction (or to PACE data for first termers). If direct observation of OETS proves impossible, other data are used to estimate OETS. The subroutine SETXOE determines OETS from data passed to it from subroutine PNTCNT.

Determining whether an airman is in a first, second, or career term is often not straightforward. The airman's eligibility category at the beginning of the term, the airman's years of service, and the categorization of the preceding term (if available) are used to construct a best guess for terms of service. Subroutine TRMSRV determines which term an airman is in from data passed to it from PNTCNT.

Constructing YARS for Each Term In Airman's Career

The Year-at-Risk concept is defined so that each term an airman serves contains an integer number of years at risk, and each year at risk is twelve months long. The YAR program computes how many years at risk there are in a given term of service each time the PNTCNT subroutine encounters a transaction that ends a term or encounters the end of the EAGL data for a given airman before that airman has left the service. The computation is made by a call to the subroutine TRMYR.

There are ten tasks carried out by subroutine TRMYR:

1. identify the contracted length of service for a term;
2. determine the number of term years in the term;
3. determine which of the term's term years the EAGL file can provide data for;
4. call subroutine PERSON to get for each YAR in the term the person-specific traits of the individual given in the EAGL file;
5. determine for each YAR in the term which snapshot is to be the source of annual data for the YAR;
6. call subroutine SNAPSH to get for each YAR in the term the year-specific data for that YAR as recorded in the EAGL file;
7. call subroutine TRNSAC to get for each YAR in the term the data from the transaction that ended the term as recorded in the EAGL file;
8. call subroutine EXT to get for each YAR in the term the data from any extension transactions that occurred during that YAR;
9. call subroutine TERMS to get for each YAR in the term data specific to the term as a whole;
10. call up subroutine MILPAY to create for each YAR in the term the socio-economic and military variables.

Subroutine TRMYR computes the contracted-for length of term using data passed from PNTCNT. When the length cannot be computed, at least one YAR is created for the term.

Airmen enlist and reenlist for integer numbers of years. However, since subroutine PNTCNT determines independently both a beginning date (TBEGIN) and an

OETS for each term of service, it is possible for these dates to differ by something other than an integer number of years. The YAR program uses OETS as the central date for defining YARs. The YAR file contains TBEGIN and OETS. But subroutine TRMYR, which creates the YARs for each term, proceeds as if TBEGIN were an integer number of years less than OETS, where the integer is the contracted length of the term.

Some airmen on some EAGL files duplicate reenlistment transactions. Subroutine TRMYR catches this and adjusts for it.

Subroutine PNTCNT assigns extension transactions to the term in which they occur. Subroutine PNTCNT, by its call to subroutine EXT, assigns each extension transaction in a given term to a particular YAR in that term.

SUBROUTINES THAT DETERMINE SOCIO-ECONOMIC AND MILITARY STATUS DURING TERMS OF SERVICE

Data on an airman's socio-economic and military status are drawn from the EAGL file and from the supplementary data files. These data are organized on five levels in the YAR file:

1. Person-level variables
2. Snapshot-level variables
3. Transaction-level variables
4. Term-level variables
5. Year-level data

Person-level variables do not change over an airman's career. These data are included in each YAR record. Subroutine PERSON outputs these data for a YAR when called by subroutine TRMYR (for YARs) or subroutine PNTCNT (for dummy observation records).

Each YAR is matched with the DMDC snapshot from the June 30th preceding the start of that YAR. The preceding June is used so that no airmen who share a given YAR will be missing snapshot data for that YAR, even airmen who leave the service early in the YAR. (To use snapshot data from within the YAR year for some and not for others, depending on availability, would risk introducing spurious correlations between snapshot data and whether an airman left the service during the YAR.) Subroutine SNAPSH

outputs these data for a YAR when called by subroutine TRMYR (for YARs) or subroutine PNTCNT (for dummy observation records).

Each term is ended by a particular transaction. EAGL data for the transaction that ends a term is included in each YAR record from that term. Subroutine TRNSAC outputs these data for a YAR when called by subroutine TRMYR (for YARs) or subroutine PNTCNT (for dummy observation records).

Subroutine PNTCNT assembles information about each term of service (e.g., the date the term began, the date the term ended, and the number of extensions in the term). Some of this information is included in each YAR record for the term. Subroutine TERMS outputs these data for a YAR when called by subroutine TRMYR (for YARs) or subroutine PNTCNT (for dummy observation records).

Subroutine TRMYR assembles information about each YAR (e.g., when the YAR begins and ends, and whether the airman extended, reenlisted, or left the service during the YAR). Some of this information is included in the YAR record for the YAR. The variables are preserved in an array that is as long as the number of YARs (plus a dummy record if there is one for the airman) and written out in the program's output file.

Some of the data in the EAGL file can be interpreted only with reference to the date the data were collected. For example, an airman's time in grade, reported in the snapshot data, is specific to the June 30th the snapshot was taken. For another example, an airman's AFSC, whether from the snapshot data or from the transaction data, cannot be interpreted without reference to the date the datum was gathered because the Air Force's AFSC codes have changed over time.

Many variables in the YAR file have a reference date particular to the YAR they pertain to. For example, an airman's grade and AFSC are reported for the beginning of the YAR. Other variables in the YAR file have a reference date that is particular to the term in which the YAR they pertain occurred. Again, an airman's grade and AFSC, this time twelve months prior to the end of the term in which the YAR occurs, are examples.

The YAR program uses the YAR structure of an airman's term built by subroutines PNTCNT and TRMYR to transform data specific to snapshot and transaction dates into data specific to YAR and term dates. Sometimes, the supplemental data files are used in these transformations.

The supplemental data files are also used to create data-specific variables reflecting an airman's economic opportunities. Both YAR-specific and term-specific economic variables are created.

Subroutine MILPAY is the primary manager of these creations of variables specific to dates tied to the YAR representation of an airman's career.

There are ten tasks carried out by subroutine MILPAY:

1. Create YAR-level AFSC and CONUS variables for each YAR in a term;
2. Create YAR-level time in service and grade variables for each YAR in a term;
3. Create YAR-level military and civilian pay variables for each YAR in a term;
4. Create YAR-level unemployment variables for each YAR in a term;
5. Create YAR-level bonus variables for each YAR in a term;
6. Create term-level AFSC and CONUS variables for each YAR in a term;
7. Create term-level bonus variables for each YAR in a term;
8. Create term-level time in service and grade variables for each YAR in a term;
9. Create term-level military and civilian pay variables for each YAR in a term;
10. Create term-level unemployment variables for each YAR in a term.

Subroutine MILPAY fulfills its purposes by calling a sequence of subroutines. Subroutine GRDPNT determines an airman's grade and time in service for a given date, using data passed from subroutine MILPAY. A perversion in the YAR program is that one heavily-used indicator of military and civilian pay opportunities, the ratio of military to civilian pay, is placed in the YAR records deep in the call to subroutine GRDPNT. Subroutine GRDPNT calls subroutine OETS12. Subroutine OETS12 contains a data statement in which the military/civilian pay ratios are stored, and the subroutine puts those data in a COMMON block, OETS2Z, from whence these pay data are written in the program's output file.

Subroutines YPAFSC, AFSCV, TAFSCS, and PREVSK determine a variety of AFSC variables. Subroutine CONUSV determines whether an airman had an overseas assignment on a given date. Subroutines MPAYX, CPAYX, and UNEMPX determine military pay, civilian pay, and unemployment at a given time. Subroutines BONUSZ and BONUSX determine the bonus eligibility of an airman at given times. All these subroutines rely on data passed from subroutine MILPAY and on supplemental data

files: The most structured of these subroutines are grouped as "file" subroutines in Appendix C.

(Subroutines GRDPNT, OETS12, YPAFSC, TAFSCS, and PREVSK are grouped among the "yar" subroutines in Appendix C. Subroutines AFSCV, CONUSV, MPAXY, CPAYX, UNEMPX, BONUSZ, and BONUSX are grouped among the "file" subroutines in Appendix C, as are all the subroutines called by these routines.)

OUTPUT SUBROUTINES

The YAR FORTRAIN program writes an output file that contains the YAR records for each airman, plus a dummy observation record for many airmen. The output data are described in detail in Sec. IV. The subroutines OUNIT, OUTPT, and OUTPUT create the output file; they are grouped as "output" subroutines in Appendix C.

III. UPDATING THE YAR PROGRAM

Updating the YAR file requires altering numerous parameters in the YAR program and providing several new supplemental data files and a new EAGL file as inputs. The supplemental files are listed at the beginning of Sec. II. They are documented in Walker and McGary, 1989.

All parameters in the YAR program that must be changed whenever one changes the EAGL or supplemental files appear in FORTRAN "PARAMETER" statements. A PARAMETER statement defines the values of one or more FORTRAN scalar variables. Its scope is limited to the routine in which the statement appears. The statement

PARAMETER (J1=15,J2=30,J3=45), where

J1 = maximum number of snapshots,

J2 = maximum number of transactions,

J3 = maximum number of year-at-risk records,

appears in many subroutines. The meanings of J1, J2, and J3 are uniform throughout the code.

The meanings of other PARAMETER variables are specific to the routine in which they are named. Each such variable defines the dimensions of externally supplied data. The data are either initialized in the program, or they exist on supplementary files. As the EAGL files are updated, so must these external sources of information. Table 1 lists the parameters besides J1, J2, and J3 that must be changed when input files are changed.

An efficient way to update the YAR file would be to alter only the YAR records of airmen whose EAGL records have changed. When the EAGL is updated, a file of Social Security numbers for the airmen with new (completely new or merely revised) EAGL records can be created. YAR records in the old file should then be divided into two groups, (i) airmen with revised EAGL records, and (ii) airmen with unchanged EAGL records. Passing the new EAGL records through the YAR FORTRAN program will create new YAR records for both the airmen with completely new EAGL records and the airmen with revised EAGL records. These new YAR records should then be merged with the old YAR records for airmen whose EAGL records remained unchanged. As the YAR file grows over time, the savings in processing from this more efficient updating procedure will become considerable. This procedure requires no alterations to the YAR FORTRAN code.

Table 1

PARAMETERS REQUIRING UPDATING

Subroutines	Parameter	YAR5 Value	Meaning
INPUT	JDIMX	2500	Maximum length of EAGL record
	JDIMY	125	Location in EAGL record of first byte of first snapshot record
OUTPUT	J1P	16	Dimension of military/civilian pay ratio vector; should equal J1+1 (same data as in OETS12, below)
RCDTTR	JHSH	2500	Workspace for SPD file lookup; should equal about 4 times size of SPD file
AFSCF1	JHSH	3000	Workspace for AFSC table lookup; should equal about 3 times size of AFSC file
OETS12	J1P	16	Dimension of military/civilian pay ratio vector; should equal J1+1 (same data as in OUTPUT, above)
MPAYF	JHSH	4000	Workspace for MILPAY table lookup; should equal about 3 times size of MILPAY file
BONUSV	JBONP	44	Maximum number of bonus periods
BONUSF	JBON	132	Maximum number of bonus periods times the number of bonus zones (3)
	JBONP	133	JBON+1: amount of workspace needed to supply bonus information
	JHSH	400	Workspace for BONUS table lookup; should equal about 2 times the number of AFSC aggregate groups with a history of bonuses
GPAYV	J1P	16	Dimension of civilian pay ratios vector; should equal J1+1
UNEMPF	JHSH	96	Workspace for UNEMP table lookup; should equal about 2 times size of UNEMP file
PERIOD	JNPD	195	Dimension of vector to map months since Jan 1970 into bonus period
CPAYF1	JHSH	968	Workspace for CIVPAY table lookup, where lookup is based on characters 1-3 and 5 of the AFSC; should equal about 2 times size of CIVPAY file
CPAYF2	JHSH	644	Workspace for aggregated CIVPAY table lookup, where lookup is based on characters 1-3 of the AFSC; should equal about 1.5 times size of CIVPAY file
CONUSF	JHSH	22000	Workspace for CONUS table lookup; should equal about 5 times the number of entries in CONUS file
UNEM2F	JHSH	800	Workspace for UNEM2 table lookup; should equal about 4 times size of UNEMP file
BONUSI	JBONP	133	Amount of workspace needed to supply bonus information
VREAD	JLEN	89	Location in EAGL record of last indicator of whether a given snapshot is present; snapshot indicators are assumed to appear in columns 60 through JLEN+1, and each indicator occupies 2 characters
	JDIM	124	Length in EAGL record of person-level variables

IV. WHAT'S ON VERSIONS OF THE YAR (YAR5) FILE

ORGANIZATION OF SECTION

This section defines each of the variables in the YAR5 file. There are three types of data on the file. Each of the following sections of this document defines all variables of the same type:

1. *PACE Data.* These are variables that originally came from the Processing and Classification of Enlistees (PACE) file, which did not exit before January 1, 1956. They are primarily demographic data, which were obtained when the airman joined the Air Force. These variables begin with the letter "B" (for "beginning"), since they refer to the beginning of the airman's career. They appear on each of the airman's YAR records, even though their values do not change over time.
2. *Snapshot Data.* These data come from the Defense Manpower Data Center (DMDC) Master files. They provide annual "snapshots" of the airman on every June 30 he was in the Air Force, detailing the individual's personal traits and military circumstances on that date. These variables generally begin with the letter "S."
3. *YAR Data.* These data refer to a specific year at risk for the airman. Most YAR variables begin with the letter "Y."
4. *Data on the Term of the YAR.* These variables refer to the term of service the airman was in at the beginning of the YAR (some terms end within a YAR). They generally begin with the letter "T."
5. *Variables for the End of the Term of the YAR.* These variables refer to the airman's military circumstances at the end of term of service the airman was in at the beginning of the YAR. They begin with the letter "X."

There are for appendixes pertinent to this section:

1. *Appendix A (How Snapshots Are Recorded for a YAR).* This appendix contains a second example of the year-at-risk concept and its relationship to an airman's term of service. Actual variables in the file and the values they assume for a particular airman are shown.

2. *Appendix B (How to Reconstruct the Force as of a Specific June).* Each observation in the YAR file contains the snapshot data for the June preceding the beginning of the year at risk. So, for example, if an airman began a second term on 7508, then the snapshot variables for the YAR 7508-7608 would come from the June 1975 snapshot. The "T" variables in the YAR record would contain information on the airman's second term. However, in 7506 when the snapshot was recorded the airman was serving in his first term. If an analyst wishes to reconstruct the force as of a specific June with "T" variables as they were in that June, he must recode some variables. This appendix contains the SAS code to accomplish this revision.
3. *Appendix E (Code Used to Construct the SAS Version of the YAR5 file).* Because most of the data and statistical analysis using this file are expected to be performed using SAS, the YAR file was converted into a SAS file. Appendix E is a listing of the code that created the SAS file.
4. *Appendix F (RAND Recodes of Transaction IDs).* The variable XTRNID contains the code the Air Force assigned to the transaction that ended the term of the YAR. There are hundreds of these codes. (For example, there are over 200 codes for involuntary discharge.) To assist the analyst, these codes have been grouped into broad categories such as retirement, attrition (quality), etc. These recoded values appear on the file as XCTRN. Appendix F provides the mapping between XTRNID and XCTRN. It also provides a mapping between XTRNID and the end-of-term codes used in the Short-term Aggregate IPM (SAM). SAM is being developed for inclusion in the Air Force's Enlisted Force Management System (EFMS).

PAGE DATA CONTAINED IN EACH YAR RECORD

BID = A number that uniquely identifies the airman. The last two digits are X1, where X is the last digit of the airman's Social Security Number [ID].¹
(Character variable, length 10.)²

BAFQTA = The airman's AFQT group [AFQTGP], based on the airman's AFQT percentile (X).

Code:

- 1 = AFQT Group I (93≤X≤99)
- 2 = AFQT Group II (65≤X≤92)
- 3 = AFQT Group IIIA (50≤X≤64)
- 4 = AFQT Group IIIB (31≤X≤49)
- 5 = AFQT Group IV, V, VI, VII (01≤X≤30)
- 9 = Blank, unknown on AGL, or
airman enlisted before 1/1/56

BBLK = 1 if the airman is Black; = 0 if the airman is not Black;
= -9 if PACE datum is missing.

BDEPEN = The airman's number of dependents at entry [IDEPEN].

Code:

- 0 = Zero dependents
- 1 = One dependent
- 2 = Two dependents
- 3 = 3-10 dependents
- 9 = Unknown, blank on AGL, >10 dependents, or
airman enlisted before 1/1/56

BDOB = The airman's birth date [DOB].

Code: four digits, yymm (e.g., 4706)
= -9 if missing or airman enlisted before
1/1/56

BIAGE = The airman's age (in years) when he enlisted in the Air Force. = -9 if PACE datum is missing.

¹Variables in brackets are defined in Brauner, Murray, Walker, and Davidson, 1989.

²In the following definitions (character variable, length n) will be abbreviated (character n).

BIDOE = The airman's initial date of enlistment [DOE].
This is a five-digit number. The first two digits indicate the year, the last three digits indicate the fraction of the year. For example, if the initial date of enlistment is Jan. 1, 1972, BIDOE = 72000. If it is Dec. 15, 1972, BIDOE = 72955. If this PACE datum is missing, it is a computed variable; -9 if unknown or cannot be computed.

BIEDUC = The airman's education at entry [IEDUC].
Code:
1 = No certificate, diploma, or General Equivalency Degree (G.E.D.)
2 = G.E.D.
3 = Certificate of equivalency
4 = High school diploma
5 = Some college training
6 = Baccalaureate degree or more
-9 = Blank, unknown on AGL, or
airman enlisted before 1/1/56

BITOE = The airman's initial term of enlistment [ITOE].
Code:
1 = Two-year enlistment
2 = Four-year enlistment
3 = Six-year enlistment
-9 = Unknown, blank on AGL, or
airman enlisted before 1/1/56

BLSED = 1 if airman has no certificate, diploma, or G.E.D. (BIEDUC = 1); = 0 otherwise (BIEDUC > 1), unless education unknown (BIEDUC = -9), in which case = -9.

BMALE = 1 if the airman is male; = 0 if the airman is female. = -9 if PACE datum is missing.

BMARRY = The airman's marital status at entry [IMARRY].
Code:
1 = Single, never married
2 = Married (including common law)
3 = Divorced, interlocutory decree, separated or annulled
-9 = Other status, blank on AGL, unknown, or
airman enlisted before 1/1/56

BMRED = 1 if airman has more than a high school diploma
(BIEDUC > 4); = 0 otherwise, unless education unknown
(BIEDUC = -9), in which case = -9.

BNODEP = 1 if airman has no dependents (BDEPEN = 0);
= 1 otherwise (BDEPEN > 0), unless number of dependents
unknown (BDEPEN = -9), in which case = -9.

BSPACE = 1 if PACE data for this airman are available; = 0 if PACE
data are unavailable.

BRACE = The airman's race [RACE].
Code:
1 = Caucasian
2 = Black
3 = Other
-9 = Blank, unknown on AGL, or
airman enlisted before 1/1/56

BSEX = The airman's gender [ISEX].
Code:
1 = Female
2 = Male
-9 = Blank, unknown on AGL, or
airman enlisted before 1/1/56

BSING = 1 if the airman is not married; = 0 if the airman is married;
= -9 if PACE datum is missing.

BSLOWR = 1 if airman is in AFQT group 1 (BAFQTA = 1);
= 2 otherwise (1 < BAFQTA < 6), unless AFQT group unknown
(BAFQTA = -9), in which case = -9.

B1DEP = 1 if airman has one dependent (BDEPEN = 1);
= 1 otherwise (BDEPEN > 1, or BDEPEN = 0), unless
number of dependents unknown (BDEPEN = -9), in which
case = -9.

SNAPSHOT DATA

Snapshots Preceding Year at Risk

Most of the snapshot data pertain to the snapshot for the June preceding the beginning of the year at risk. Prior service (PS) accessions for whom the snapshot preceding the beginning of a year at risk is missing have snapshot data imputed from the snapshot following the beginning of the YAR or, failing that, from the second snapshot preceding the beginning of the YAR. In such cases, SYR is the snapshot year from which the data were taken, STAF, SGRD, and STIG are imputed as of the proper snapshot year, not SYR; and SDOS is -9. Other variables take their values from the snapshot given by SYR.

GOODSNAP = 1 if the snapshot information was found on the EAGL file;
0 if the snapshot information was imputed from other snapshots as for prior service accessions or if this is the second time the snapshot information is recorded for the airman.

SDAF = Duty AFSC [SDAFSC] from the snapshot given by SYR.
(Character 7, -9 is left justified.)

Codes: See Air Force's FIDO File, code AFSC-AMN-DC.
Blank if unknown or before 1980; - 9 if missing.

SDEP = Number of dependents [SDEPN] from the snapshot given by SYR. The Air Force does not include spouse if both are military members. Normally the records of only one of the two military members will show the children. In 1984 the Air Force had 7500 single member parents and 24,000 military couples (thus 48,000 airmen). The analyst should be cautioned that this variable is not reliable, because no effort is made on a yearly basis to update the information.

Code:

- 0 = No dependents
- 1 = 1 dependent
- 2 = 2 dependents
- 3 = 3 dependents
- 4 = 4 dependents
- 5 = 5 dependents
- 6 = 6 dependents
- 7 = 7 dependents
- 8 = 8-15 dependents
- 9 = Unknown or greater than 15 dependents

SDOS = Date of separation in the snapshot given by SYR [SDOSYR, SDOSMO]; - 9 if missing.

SEDU = Education [SEDUC] in the snapshot given by SYR.

Code:

- 00 = Unknown
- 01 = 1-7 years of elementary school completed
- 02 = 8 years of elementary school completed
- 03 = 1 year of high school completed
- 04 = 2 years of high school completed
- 05 = 3 or 4 years of high school completed with no diploma or no G.E.D.
- 06 = High school graduate, diploma or G.E.D.
- 07 = 1 year of college completed
- 08 = 2 years of college completed
- 09 = 3 or 4 years of college completed with no diploma
- 10 = College graduate
- 11 = Masters degree received or other professional degrees beyond college other than a doctorate
- 12 = Doctorate degree received
- 9 = Missing

SGRD = Pay grade [SGRADE] in the snapshot given by SYR.

Code:

- 0 = Enlisted, unknown
- 1 = E-1
- 2 = E-2
- 3 = E-3
- 4 = E-4
- 5 = E-5
- 6 = E-6
- 7 = E-7
- 8 = E-8
- 9 = E-9
- 9 = Missing

SMAR = Marital status [SMARR] in the snapshot given by SYR.

Code:

- 0 = Unknown
- 1 = Single, divorced, interlocutory decree, legally separated, widowed, or marriage annulled
- 2 = Married
- 9 = Missing

SPAF = The primary AFSC [SPAFSC] in the snapshot given by SYR; - 9 if missing. (Character 7, -9 is left justified.)

SPAS = The Personnel Accounting Symbol (PAS) Number [SPASNR] in the snapshot given by SYR. The PAS Number identifies the actual unit to which the individual airman is assigned. PAS Numbers begin with the character "F". The other three positions contain a unit identifier. (We do not have these codes on our copy of the FIDO file.) Blank before 1974 or if PASNR is unknown or classified; -9 if missing. (Character 4, -9 is left justified.)

SPOC = DOD primary occupation group [SPOCGP] in the snapshot given by SYR. Codes are listed in Brauner, Murray, Walker, and Davidson, 1989; -9 if missing.

STAF = Total active federal military service recorded in months [STAFMS] in the snapshot given by SYR; -9 if missing.

STIG = The number of months the airman had served in grade SGRD [STIG] as of the date of the snapshot given by SYR; -9 if missing.

SYR = Snapshot year. The last two digits of the calendar year of the snapshot preceding this YAR. The snapshot is taken in June of the year (e.g., 74 refers to June 30, 1974). The value of SYR is positive if the snapshot is recorded. The value of SYR is negative if the snapshot is unavailable. The value of SYR is -9 if the snapshot is out of range (71-86 inclusive) or if the year cannot be determined.

Snapshot In YAR and Snapshot Following YAR

The YAR5 file contains snapshot information for two snapshots beyond the one for SYR (see Fig. 1 below).

- S1GRD = Pay grade [SGRADE] in the snapshot that occurs within the YAR.
Code: See SGRD
- S2GRD = Pay grade [SGRADE] in the snapshot that follows the YAR
Code: See SGRD
- S1TG = The number of months the airman had served in grade S1GRD [STIG] as of the date of the snapshot that occurs within the YAR; -9 if missing
- S2TG = The number of months the airman had served in grade S2GRD [STIG] as of the date of the snapshot that follows the YAR; -9 missing.
- S1TAF = Total active federal military service recorded in months [STAFMS] in the snapshot that occurs within the YAR; -9 if missing.
- S2TAF = Total active federal military service recorded in months [STAFMS] in the snapshot that follows the YAR; -9 if missing.

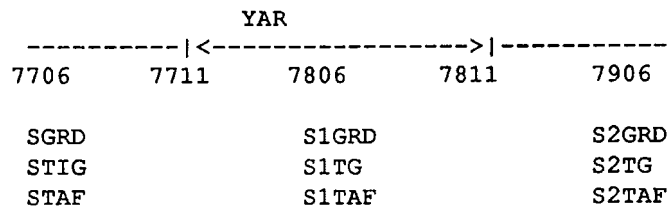


Fig. 1—Relative locations of snapshot variables for a YAR that runs from July 1, 1971, to July 1, 1981

VARIABLES FOR THIS YAR

YBMLT6 = The bonus multiple the airman would obtain six months before the end of this YAR based on YPAF12 and the zone that contains the greater of his actual TAFMS at that time or his TAFMS six months before TXOETS (see p. 37); -9 if this cannot be computed.

YCONUS = The airman's location at the time of the snapshot associated with this YAR.

Code:

0 = CONUS

1 = Non-CONUS

-9 = Before FY80 (we have no data) and missing

YDAF12 = Duty AFSC [SDAFSC] from the SYR snapshot; -9 if blank or missing. (Character 7, -9 is left justified.)
Codes: See Air Force's FIDO File, code AFSC-AMN-DC.

YDOSIN = DOS (date of separation) in effect at the beginning of this YAR; -9 for TRMDUM = 1 or if missing.

YDOSOU = DOS (date of separation) in effect at the end of this YAR or just before reenlistment if airman reenlisted in this YAR; -9 for TRMDUM = 1 or if missing.

YE1BP = The monthly basic pay the airman would be earning at the beginning of this YAR if he were at the modal grade³ with TAFMS at the beginning of the YAR if the military pay schedule were the one that applied at the beginning of the YAR; -9 if cannot be computed.

³If 0 <= TAFMS < 24, modal grade = 3

24 <= TAFMS < 120,
modal grade = $(3*(120 - \text{TAFMS}) + 5*(\text{TAFMS} - 24)) / (120 - 24)$

120 <= TAFMS < 180, modal grade = $(5*(180 - \text{TAFMS}) + 7*(\text{TAFMS} - 120)) / (180 - 120)$

180 <= TAFMS < 312,
modal grade = $(7*(312 - \text{TAFMS}) + 9*(\text{TAFMS} - 180)) / (312 - 180)$

312 <= TAFMS, modal grade = 9

YE1CPG = The annual wage the airman would receive if he were in the civilian sector and doing a similar job, with skill level based on TAFMS at the beginning of the YAR. To calculate skill level⁴ first use YFMS12 to find modal grade;⁵ -9 if cannot be computed.

YE1MP = The monthly basic military compensation the airman would be earning at the beginning of this YAR if he were at the modal grade with TAFMS at the beginning of the YAR if the military pay schedule were the one that applied at the beginning of the YAR. [In the FORTRAN code this variable is YTIMPL.] = -9 if cannot be computed.

YFMS12 = TAFMS at beginning of this YAR; -9 if cannot be computed.

YGRD12 = Grade at beginning of this YAR.

Code:

- 1 = E-1
- 2 = E-2
- 3 = E-3
- 4 = E-4
- 5 = E-5
- 6 = E-6
- 7 = E-7
- 8 = E-8
- 9 = E-9
- 9 = Missing

⁴If the airman's grade was 1, 2, or 3, his equivalent civilian skill level was assumed to be 3. If his grade was 5, 7, or 9, skill levels of 5, 7, and 9, respectively, were assigned to the airman. If his grade was 4, his civilian wage is the average of the civilian wages assigned to skill levels 3 and 5; if his grade was 6, his civilian wage is the average of the civilian wages assigned to skill levels 5 and 7; and if his grade was 8, his civilian wage is the average of the civilian wages assigned to skill levels 7 and 9. These data are based on civilian earnings by specialty for 1972 and 1978. No updating of the data has taken place since 1978.

⁵If $0 \leq \text{TAFMS} < 24$, modal grade = 3
 $24 \leq \text{TAFMS} < 120$,
modal grade = $(3*(120 - \text{TAFMS}) + 5*(\text{TAFMS} - 24)) / (120 - 24)$
 $120 \leq \text{TAFMS} < 180$, modal grade = $(5*(180 - \text{TAFMS}) + 7*(\text{TAFMS} - 120)) / (180 - 120)$
 $180 \leq \text{TAFMS} < 312$, modal grade = $(7*(312 - \text{TAFMS}) + 9*(\text{TAFMS} - 180)) / (312 - 180)$
 $312 \leq \text{TAFMS}$, modal grade = 9

YLGP12 = Loss group for the primary AFSC derived from YPAF12 for this YAR; -9 if cannot be computed. (Character 7, -9 is left justified.) This variable has not been used in the modeling and is not being updated.

Codes (lowercase letters are replaced by numbers in the file):

AMBIG = This AFSC cannot be tracked to an ultimate AFSC because of a split less than 90%-10%.⁶

DELETD = This AFSC was deleted before 30 April 1985.

99nTm = Temporary condition AFSC—e.g., student or patient. (Treated as one group—possibly missing—for analysis purposes, without regard to n and m.)

nnnBm = This small AFSC has been assigned to the "base case" for analysis purposes.

nnnXm = AFSC not grouped, or all shreds grouped together (e.g., 202X0A, 202X0B grouped together to 202X0).

nnnXm? = Some, but not all, shreds grouped together.

nnnYm = Shreds and non-shred grouped together (e.g., 113X0, 113X0A, 113X0B grouped to 113Y0)

nnnYm? = Nonshred and some, but not all, shreds grouped together

nnnXX = This AFSC grouped with all AFSCs beginning nnn.

nnnXY = This AFSC grouped with some, but not all, AFSCs beginning nnn.

nnnZm = This small AFSC combined with nnnXm.

nnZZZ = Only the first two digits of the ultimate AFSC are known with certainty for this AFSC.

nnnZZ = Only the first three digits of the ultimate AFSC are known with certainty for this AFSC.

nnnC9 = This AFSC grouped with all 9-level and 0-level cappers beginning with nnn.

blank = This AFSC not assigned to any loss group.

YMCWRAT = The ratio of military to civilian compensation for the fiscal year. The values of YMCWRAT in YAR5 are given in 1. The variable SYR is used as input to the table. (In the Table FORTRAN code this variable is TNBFRB.) -9 if cannot be computed.

⁶For a discussion of ultimate AFSCs, see Carter et al., 1987, App. A.

Table 2
THE RELATIONSHIPS AMONG FY,
YMCWRAT, AND SYR

FY	YMCWRAT	SYR
72	100.0	71
73	100.2	72
74	102.0	73
75	101.1	74
76	97.4	75
77	95.4	76
78	95.7	77
79	93.9	78
80	93.2	79
81	95.4	80
82	99.8	81
83	96.0	82
84	94.6	83
85	93.6	84
86	92.3	85
87	91.2	86

SOURCE: AFMPC/DPMDW from data
supplied by AF/DPXA.

- YNEXT = Number of extensions in this YAR; -9 if TRMDUM =1.
- YNTYR = The year of term for this YAR (that is, YNTYR = 1 for the first year of the term, YNTYR = 2 for the second year of the term, etc.; YNTYR = 1 for dummy YARs); -9 when the year of term cannot be ascertained.
- YNXTLS = Date of airman's loss from Air Force; -9 if still in service at 8707 or if this is a dummy record.⁷
- YPAF12 = Primary AFSC [SPAFSC] from the SYR snapshot. If this snapshot is unavailable an imputation from other snapshots is used based on first available SPAFSC(J-1), SPAFSC(J-2), SPAFSC(J-3), SPAFSC(J+1), SPAFSC(J+2),

⁷If this airman left the Air Force and rejoined it as a PS accession, this variable will change value when the airman returns to the force.

SPAFSC(J+3), where J=SYR for this YAR. (Character 7, -9 is left justified.)

Codes: See Air Force's FIDO File, code AFSC-AMN-DC;
= -9 if unknown.

- YRISKB = The date this YAR began (yyymm) or xx07 if this is a dummy YAR, where xx = SYR for this year; = -9 if the YAR began before 7107 and the term ended after 7107, or if the date cannot be calculated. The highest value of YRISKB in the YAR5 file is 8607.
- YTRMYR = Indexes the YARs on the file for each airman. For example YTRMYR = 3 means this is the third YAR for this airman.
- YT1BP = The airman's monthly basic pay 12 months before the end of this YAR. This variable is calculated using YYOS12, YGRD12, and the snapshot year that occurs during this YAR; -9 if cannot be computed.
- YT1CP = The annual wage the airman would earn in the civilian sector if he were doing a similar job 12 months before the end of this YAR.⁸ Skill level⁹ is based on YGRD12 and the snapshot year is the year that occurs during the YAR; = -9 if cannot be computed.
- YT1MP = The airman's monthly regular military compensation (basic pay, subsistence allowance, allowance for quarters, and tax advantage) 12 months before the end of this YAR. This variable is calculated using YYOS12, YGRD12, and the

⁸These data are based on civilian earnings by specialty for 1972 and 1978. No updating of the data has taken place since 1978.

⁹If the airman's grade was 1, 2, or 3, his equivalent civilian skill level was assumed to be 3. If his grade was 5, 7, or 9, skill levels of 5, 7, and 9, respectively, were assigned to the airman. If his grade was 4, his civilian wage is the average of the civilian wages assigned to skill levels 3 and 5; if his grade was 6, his civilian wage is the average of the civilian wages assigned to skill levels 5 and 7; and if his grade was 8, his civilian wage is the average of the civilian wages assigned to skill levels 7 and 9.

snapshot year that occurs during this YAR; = -9 if cannot be computed.

- YT0UEM = Unemployment rate at the end of this YAR by sex, race, and age; age based on TAFMS at the end of current YAR + 12 months;¹⁰ unemployment rate based on first month of current YRISK+12 months; = -9 if cannot be computed or after June 30, 1982.
- YT1UEM = Unemployment rate by sex, race, and age for this YAR; age based on TAFMS at end of this YAR; unemployment rate based on first month of YAR. If this YAR has TRMDUM=1, then YT1UEM gives the unemployment rate for the next year; = -9 if cannot be computed or after June 30, 1982.
- YT2UEM = Unemployment rate by sex, race, and age for previous YAR; age based on TAFMS 12 months before end of this YAR; unemployment rate based on first month of previous YAR; = -9 if cannot be computed or after June 30, 1982.
- YUAF12 = Ultimate Primary AFSC¹¹ derived from YPAF12 for this YAR; -9 if YPAF12 is missing or if the ultimate AFSC cannot be tracked. For nonprior service airmen in their first year of service, the value is -9. (Character 7, -9 is left justified.)
- YUDF12 = Ultimate Duty AFSC derived from YDAF12 for this term year; -9 if YDAF12 is missing or if the ultimate AFSC cannot be tracked. (Character 7, -9 is left justified.)
- YYOS12 = YFMS12/12. Number of full years of service at beginning of this YAR; -9 if cannot be computed.

¹⁰If 0 <= TAFMS <= 72, age group = 20-24
72 < TAFMS <= 144, age group = 25-34
144 < TAFMS, age group = 35-44

¹¹For a discussion of ultimate AFSCs, see Carter et al., 1987, App. A.

MUNEMP = Unemployment rate for 20- to 24-year-olds in the month ending the YAR (YRISKB + 100).¹²

MAUNEMP = Average unemployment rate for 20- to 24-year-olds for the 12 months preceding the end of the YAR (from YRISKB -1 month through YRISKB + 100).

VARIABLES FOR THE TERM OF THIS YAR

GOODTYR = 0 if this is a record belonging to an airmen who left to go to OCS or if TBDOBS=1 or if TRMDUM=1 or if TBEGIN=TBEND and TENDLS=0; = 1 otherwise.¹³

TABMLT = The average bonus multiple over the months of the TXOETS year based on TPAF12; -9 if cannot be computed.

TBDOBS = 1 if TBEGIN of the term of the YAR is unknown or TBEND is during a term year that begins before 7107. In this case the data may be unreliable; = -9 if this is a dummy observation used to record the last snapshot information; = 0 otherwise.

TBEGIN = Date on which the term of the YAR began. The first term begins on the initial date of enlistment. The next term begins when the current term ends (see TBEND). If TBEGIN is not an integer number of years from the original ETS of the term, TBEGIN for the first YAR in the term may not coincide with YRISKB for that year, since the latter presumes integer years of enlistment; -9 if cannot determine when term began. The highest value of TBEGIN in the YAR5 file is 8706.

TBEND = Date on which the term of the YAR ended. For terms ending in a loss, TBEND is equal to [TEDCSA] for the transaction

¹²Adding 100 to YRISKB adds one year.

¹³Appendix A shows how to use this variable to select all records that are not dummy records.

that ends the term; for reenlistments, BEND is equal to the new date of separation less the new term of enlistment times 100. Where these calculations lead to obviously incorrect results, TBEND and TBEGIN are based on Date of Last Enlistment drawn from the EAGL snapshots [SDOLE]. If EAGL data indicate that the airman was no longer in the Air Force but the transaction ending the term was missing, TBEND was imputed to be December following last snapshot year; -9 if the term had not ended before 8706. The highest value of TBEND in the YAR5 file is 8706.

- TBML12 = The bonus multiple during the month of TXOETS, based on TPAF12; = -9 if cannot be computed.
- TBRECD = The bonus multiple the airman was eligible to receive on his date of previous reenlistment (based on his Duty AFSC in the first June following TBEGIN); = 0 for first term; = -9 if cannot be computed. If TBEGIN \leq 7906 then TBRECD = -9.
- TDAP12 = Duty AFSC 12 months before XOETS; -9 if missing. (Character 7, -9 is left justified.)
- TENDLS = 1 if the term of the YAR ended in a loss; = 0 if the term ended in a reenlistment; = -9 when term has not ended or if TRMDUM=1.
- TFMS12 = Months of total active federal military service the airman had or would have completed by 12 months before his TXOETS; -9 if TXOETS cannot be determined.
- TGRD12 = Airman's grade 12 months before his TXOETS. When this is not known it is estimated as either 02SGRD (the second snapshot before TXOETS) or XGRADE, depending on whether 02SYR or XTDCSA is closer in time to (TXOETS-12); -9 when not available and unable to estimate.
- TLAFSC = Primary AFSC in last year of previous term, or the best estimate thereof; = -9 when unable to estimate. (Character 7, -9 is left justified.)
- TLASYR = The last two digits of the snapshot year from which TLAFSC is drawn; -9 when TLAFSC is -9.
- TLGP12 = Loss group of AFSC12 (12 months before TXOETS); -9 when unavailable. (Character 7, -9 is left justified.)

TMAUNEMP = Average unemployment rate for 20- to 24-year-olds for the 12 months preceding the end of the term. (TMAUNEMP=MAUNEMP in the ETS year.)

TMBMLT = The maximum bonus multiple over the months of the TXOETS year based on TPAF12; = -9 when unable to compute.

TMCWRAT = The ratio of military to civilian compensation for the fiscal year that ends this term. (TMCWRAT=YMCWRAT in the ETS year.)

TMUNEMP = Unemployment rate for 20- to 24-year-olds in the ETS month. (TMUNEMP=MUNEMP in the ETS year.)

TNEXT = Total number of extensions in the term of the YAR. If term begins before 7107 there may be a miscounting of the number of extensions because of unseen extensions preceding the start of the EAGL file data; -9 if unavailable.

TNTS = 1 if the term of the YAR is the airman's first term of service; = 2 if the term is the airman's second term of service; = 3 if the term is the airman's third or more term of service; = -9 if cannot determine airman's term. (Values of 1 or 2 are based on [TECAT] for the transaction ending the term of the YAR; the value 3 is derived from years of service and initial date of enlistment, or from TECAT for terms ending in a loss, or from the value of TNTS for previous terms.)

TNTYR = The number of years actually served by the airman in this term. If the term had not ended by June 1987, TNTYR is number of YARs in the YAR file for this term. This variable = 1 in the dummy record. For example, if the airman left after serving three years in a term, TNTYR=3 for each YAR in the term. If the airman extended in a term and actually served seven years before reenlisting or leaving then TNTYR=7 for each YAR in the term. If TBEGIN < 0 then TNTYR < 0.

TNWBON = The bonus multiple the airman would have had if he had reenlisted at the TXOETS for this term but had the primary AFSC he has in the second snapshot after this TXOETS, using TAFMS at TXOETS (TFMS12 + 12); -9 if unable to compute.

TOLBP = The airman's monthly base pay based on year, grade, and TAFMS reported in the snapshot preceding the start of this term;

- = -9 for first term. If this YAR has TRMDUM =1 then this variable is calculated using the information on the snapshot.
- TOLCP = The annual wage the airman would have earned in the civilian sector at the time of the snapshot preceding the start of this term if he were doing similar work to that recorded in the snapshot and had the skill level computed from SGRD;¹⁴ = -9 for first term. If this YAR has TRMDUM =1 then this variable is calculated using the information on the snapshot.¹⁵
- TOLMP = The airman's monthly regular military compensation based on year, grade, and TAFMS reported in the snapshot preceding the start of this term; = -9 for first term. If this YAR has TRMDUM =1 then this variable is calculated using the information on the snapshot.
- TOLUEM = The unemployment rate for the group defined by the airman's race, sex, and age group¹⁶ in the June preceding the beginning of this term as a function of his TAFMS on the snapshot corresponding to that June; = -9 for first term and after June 1982. If this YAR has TRMDUM =1 then this variable is calculated using the information on the snapshot.
- TOTSPD = Indicates that a transaction occurred during the term whose transaction code does not appear in App. D. The value is -9 or the value of the offending transaction code [TTRNID or ETRNID]. (Character 3, -9 is left justified.)

¹⁴If the airman's grade was 1, 2, or 3, his equivalent civilian skill level was assumed to be 3. If his grade was 5, 7, or 9, skill levels of 5, 7, and 9, respectively, were assigned to the airman. If his grade was 4, his civilian wage is the average of the civilian wages assigned to skill levels 3 and 5; if his grade was 6, his civilian wage is the average of the civilian wages assigned to skill levels 5 and 7; and if his grade was 8, his civilian wage is the average of the civilian wages assigned to skill levels 7 and 9. These data are based on civilian earnings by specialty for 1972 and 1978. No updating of the data has taken place since 1978.

¹⁵The data are based on civilian earnings by specialty for 1972 and 1978. No updating of the data has taken place since 1978.

¹⁶If 0 <= TAFMS <= 72, age group = 20-24
72 < TAFMS <= 144, age group = 25-34
144 < TAFMS, age group = 35-44

Examples:

Code	Meaning
050	4-year enlistment, guaranteed AFSC, base of choice
055	6-year enlistment, guaranteed AFSC, accelerated promotion guaranteed retraining
059	6-year enlistment, guaranteed AFSC, accelerated promotion, enlistment bonus
061	enlisted to go to OTS/prep school

- TPAF12 = Primary AFSC 12 months before TXOETS; -9 if information is missing. (Character 7, -9 is left justified.)
- TPRVBM = The bonus multiple the airman was offered the last time he reenlisted; -9 if missing based on TLAFC and the month this term began.
- TPSA = 1 if term began as a prior service accession; = 0 otherwise.
- TQWBAK = 0 if most recent loss was not a quick return (return within 6 months); date of most recent loss if it was a quick return [TEDCSA]; -9 otherwise.
- TQWLST = TQWBAK from previous term.
- TRMDUM = 1 if this is not a year at risk but a dummy observation used to record the last snapshot information; = 0 otherwise.
- TTIUEM = The unemployment rate for the group defined by the airman's race, sex, and age¹⁷ as a function of TFMS12 + 12. TTIUEM = -9 when TXOETS > 82.
- TUAFPR = Ultimate Primary AFSC [ULTAFSC]¹⁸ in last year of previous term; = -9 for first term. (Character 7, -9 is left justified.)
- TUAF12 = Ultimate Primary AFSC [ULTAFSC] 12 months before TXOETS; -9 if missing. (Character 7, -9 is left justified.)

¹⁷If 0 ≤ TAFMS ≤ 72, age group = 20-24
72 < TAFMS ≤ 144, age group = 25-34
144 < TAFMS, age group = 35-44

¹⁸For a discussion of ultimate AFSCs, see Carter et al., 1987, App. A.

TUCAF = Ultimate Control AFSC at end of term [ULTAFSC] based on (TCAFSC) of the transaction that ended this term; -9 if Control AFSC is missing or if unable to be converted. (Character 7, -9 is left justified.)

TUDF12 = Ultimate Duty AFSC [ULTAFSC] 12 months before TXOETS; -9 if Duty AFSC is missing or if unable to be converted. (Character 7, -9 is left justified.)

TXOETS = Date on which the term of the YAR was first scheduled to expire; -9 if missing. This date was the original ETS for the term when the term began.

TXTOE = The term of enlistment for the term of the YAR; -9 if unable to determine correct value.

TYOS12 = Number of full years of total active federal military service the airman has or would have completed by 12 months before his TXOETS; -9 if TFMS12 = -9.

VARIABLES FOR THE END OF THE TERM OF THIS YAR

XCAFSC = The airman's Control AFSC at the end of the term of this YAR [TCAFSC or ECAFSC]; -9 if term had not ended by 8706. (Character 6, -9 is left justified.)
Codes: See Air Force's FIDO file, code AFSC-AMN-DC

XCAREA = The airman's major command area at the end of the term of this YAR [TCAREA or ECAREA];¹⁹ -9 if the term had not ended by 8706.
Code:
0 = Not applicable
1 = CONUS
2 = Overseas
3 = CONUS/oversea area (leave area only)
9 = Unknown

¹⁹This variable is of questionable use. For analytic purposes use YCONUS.

XCHAR = The character of discharge at the end of the term of this YAR [TCHAR or ECHAR].

Code:

- 0 = No change or not applicable (1/11/67-12/31/72);
no entry (1/1/73-)
- 1 = Honorable (7/1/69-)
- 2 = General (6/1/65-12/31/72);
under honorable conditions (1/1/73-)
- 3 = Under conditions other than honorable (6/1/65-)
- 4 = Undesirable (6/1/65-12/31/72);
under other than honorable conditions (1/1/73-)
- 5 = Bad conduct (6/1/65-12/31/72);
under other than honorable conditions (1/1/73-)
- 6 = Dishonorable (6/1/65-)
- 7 = Applies only to officers (7/1/73-)
- 8 = Undetermined (deferred by special directive)
(7/1/69-)
- 9 = Unknown
- 9 = Not applicable (1/1/73-)

XCOH = The last two digits of the calendar year associated with the June 30 preceding the airman's entry into the Air Force. (E.g., airmen entering the service on July 5, 1968, or January 20, 1969, would have XCOH equal to 68); = -9 if BIDOE is -9.

XCTRN = Recoded transaction ID for the transaction that ended the term of this YAR. (The cross-walk between XTRNID and XCTRN is given in App. D.); = -9 if XTRNID is -9.

XDOS = Date of separation on the transaction that ended the term of this YAR [TDOS]. For losses, this is the date of separation at the end of the term of this YAR; for reenlistments this is the new DOS established by the reenlistment; = -9 if the term had not ended by 8706.

XECAT = The category of enlistment from the transaction that ended this YAR [TECAT or EECAT]. TENTS is a much more reliable variable for determining an airman's category of enlistment.

Code:

- 1 = FIRST TERM AIRMAN (3/1/76-): An individual who is serving on his:
 - initial term of enlistment, or

- first tour of extended active duty,²⁰ or
 - first enlistment in the USAF with previous active federal military service totaling <24 months
- = **FIRST TERM AIRMAN (1/1/72-3/1/76):** An individual who is serving on his:
- initial term of enlistment, or
 - first tour of extended active duty with no previous extended active service²¹
- 2 = **SECOND TERM AIRMAN (3/1/76-):** An individual who:
- is serving on his second term of enlistment and/or
 - is serving an extended active duty tour,²² or
 - is serving a first enlistment in the USAF with ≥24 months previous active federal military service, or
 - has one term of service totaling ≥24 months with other military departments, or
 - is a former nonregular airman (AFR or ANG) who enlists into the regular AF during an extended active duty, or
 - is a former nonregular airman (AFR or ANG) who enlists into regular AF subsequent to previous service on extended active duty totaling ≥24 months
- = **SECOND TERM AIRMAN (8/1/72-3/1/76):** An individual who:
- is serving on his second term of enlistment, and/or
 - is serving on an extended active duty tour,²³ or
 - is serving on a first enlistment and has more than six months active military service, or
 - is an inductee who enlisted from inducted status during or at the completion of his 2-year term of involuntary service, or
 - has one term of service with some other military department, or
 - is a former nonregular airman (AFR or ANG) who enlisted during or subsequent to previous service on extended active duty
- 4 = **CAREER AIRMAN (12/1/77-):** An individual who is serving on his:
- third or subsequent term of enlistment, other than ADT in any component of military service, or
 - second term of enlistment who has extended his enlistment for two or more years

²⁰Extension of initial enlistment is for <24 months.

²¹Extension of initial enlistment is for ≥24 months (may result from several short extensions).

²²Extension of initial enlistment is for ≥24 months (may result from several short extensions).

²³Extension of initial enlistment is for ≥24 months (may result from several short extensions).

- = CAREER AIRMAN (8/1/72-12/1/77): An individual who is serving on his
 - third or subsequent term of enlistment, and/or
 - second term of enlistment with an extended active duty tour
- = CAREER AIRMAN (1/1/72-8/1/72): An individual who:
 - is serving on his second or subsequent term of enlistment, and/or
 - is serving on an extended active duty tour,²⁴
 - has one or more terms of service with other military departments, or
 - is a former nonregular airman (AFR or ANG) who enlisted during or subsequent to active prior service on extended active duty, or
 - is an inductee who enlisted from inducted status during or at the completion of his 2-year term of involuntary service
- 5 = E-9 or E-9 selectee with HYT waived (5/1/78-)
- 9 = Unknown (1/1/72-)
- 9 = Missing

XELIG = The eligibility for reenlistment from the transaction that ended this YAR [TELIG or EELIG]; -9 if term had not ended before 8606 or if missing. (Character 2)

Many of the codes refer to sensitive information about the individual. Therefore, the codes are not listed here. For those who need access to them, they are available in listings of the FIDO file.

Generally (but not always), the following coding conventions have been used:

- 9 = Not assigned or not applicable
- 1 = Individual eligible to reenlist
- 2,3,4 = Individual not eligible to reenlist
- 9 = Unknown

XEXTAF1 = The airman's Control AFSC at the time of the first extension in this YAR [1CAFS]. (Character 6, -9 is left justified.)

Codes: See Air Force's FIDO file, code AFSC-AMN-DC

²⁴Extension of initial enlistment is for ≥ 24 months (may result from several short extensions).

XEXTAF2 = The airman's Control AFSC at the time of the last extension in this YAR [TCAFSC]. (Character 6, -9 is left justified.)

Codes: See Air Force's FIDO file, code AFSC-AMN-DC

XEXTED1 = The effective date of the first extension in this YAR [TEDSCA]. An extension with an effective date of X is recorded as having occurred in this YAR if $YRISKB < X \leq YRISKB + 100$.²⁵ However, if $X = TBEGIN$ (the effective date is the same as the start of the term of this YAR), then the extension is recorded as having occurred in the YAR for which $YRISKB = X$; = -9 if there are no extensions in this YAR.

XXTDOS1 = The transaction date of separation for the first extension in this YAR [TDOS]; -9 if EXTED2 is -9.

XEXTED2 = The effective date of the last extension in this YAR [TEDCSA]. An extension with an effective date of X is recorded as having occurred in this YAR if $YRISKB < X \leq YRISKB + 100$.²⁶ However, if $X = TBEGIN$ (the effective date is the same as the start of the term of this YAR), then the extension is recorded as having occurred in the YAR for which $YRISKB = X$; = -9 if there are no extensions in this YAR.

XXTDOS2 = The transaction date of separation for the last extension in this YAR [TDOS]; -9 if XEXTED2 is -9.

XGRADE = The airman's grade at the end of the term of this YAR [TGRADE or EGRADE].

Code:

- 1 = E-1
- 2 = E-2
- 3 = E-3
- 4 = E-4
- 5 = E-5
- 6 = E-6
- 7 = E-7
- 8 = E-8
- 9 = E-9
- 9 = Missing

²⁵If X falls within this YAR. (Adding 100 to YRISKB adds one year.)

²⁶If X falls within this YAR. (Adding 100 to YRISKB adds one year.)

XTDCSA = The effective date for force accountability of the transaction ending the term of this YAR [TEDCSA or EEDCSA]; -9 if term had not ended by 8706.

XTOE = The term of enlistment from the transaction that ended this YAR [TTOE or ETOE]. (For terms ending in a reenlistment this is the term of service of the new term begun by the reenlistment); = -9 if term had not ended by 8706.

XTRNID = The transaction code for the transaction that ended the term of this YAR [TTRNID or ETRNID]; -9 if term had not ended by 8706. (Character 3, -9 is left justified.)

Many of the codes are For Official Use Only.
They are available in listings of the Air Force's FIDO file.

Appendix A

HOW SNAPSHOTS ARE RECORDED FOR A YAR

To clarify how snapshots are recorded for a YAR, another example is useful. Suppose the airman is a prior service accession. Then an example of his record might be

first term	7108---7508
second term	7504---7804
	leaves 7804
third term	8104---8504
fourth term	8411---8811

YAR	SYR	GOODSNAP	GOODTYR	YNTYR	TNTYR	YTRMYR
7108-7208	-71 ¹	0	1	1	4	1
7208-7308	72	1	1	2	4	2
7308-7408	73	1	1	3	4	3
7408-7508	74	1	1	4	4	4
7504-7604	74	0	1	1	3	5
7604-7704	75	1	1	2	3	6
7704-7804	76	1	1	3	3	7
7707-XXXX	77	1	0	1	1	8
8104-8204	81 ²	0	1	1	4	9
8204-8304	81	1	1	2	4	10
8304-8404	82	1	1	3	4	11
8404-8504	83	1	1	4	4	12
8411-8511	83	1	1	1	3	13
8511-8611	84	1	1	2	3	14
8611-8711	85	1	1	3	3	15
8707-XXXX	86	1	0	1	1	16

The SAS statement

If GOODTYR=1; will select all observations that are not dummy records.

¹The first snapshot has a negative value because there was no information for this airman in 7106.

²For a prior service accession, the first YAR has SYR=the year of the June following reentry. The snapshot information is imputed from available information. If SYR is positive, the information is taken from the EAGL snapshot.

Appendix B

HOW TO RECONSTRUCT THE FORCE AS OF A SPECIFIC JUNE

Because information in the T variables is for the term in which the YAR lies, some recoding of variables is necessary if it is desired to view the force as of a specific June. The following SAS code resets the variables TXOETS, TNTS, TXTOE, and TENDLS if the term began after the snapshot date or the record is a dummy. When the term began after the snapshot date, then the values of these variables on the YAR file refer to the next term rather than the term the airman is in at the June of the snapshot. These variables are unreliable for the records that are not associated with a YAR. The values are set by carrying forward the values from an earlier record that describes the correct term for the snapshot. If such a record does not exist, then the variables are set to missing values. The code drops any YAR with GOODSNAPE = 0. An example will illustrate the code.

```
first term      6904---7204
second term     7204---7504
third term      7504---7804
leaves in 7712
```

YAR	SYR	TXOETS	RECODED TXOETS	TNTS	RECODED TNTS	TENDLS	RECODED TENDLS
7204-7304	71	7504		2	.	.	.
7304-7404	72	7504	7504	2	2	0	0
7404-7504	73	7504	7504	2	2	0	0
7504-7604	74	7804	7504	3	2	0	0
7604-7704	75	7804	7804	3	3	1	1
7704-7804	76	7804	7804	3	3	1	1
7707-XXXX	77	7804	7804	3	3	-9	1

```
DATA SSTRMYR;
  SET INP.SSTRMYR ;
  RETAIN LASTB2 LASTLOSS;
  IF YNXTLS NE -9 AND GOODSNAPE +GOODTYR >= 1
```



```
      THEN DO;
        LASTLOSS=YNXTLS;
        LASTB2=BID;
      END ;
    IF YNXTLS EQ -9 AND TRMDUM EQ 1 AND BID EQ LASTB2
      THEN YNXTLS=LASTLOSS;

    IF GOODSNAP = 1 ;
    GOODREC=1;
    RETAIN LASTX LASTNTS LASTTOE LASTOUT LASTBID LASTYR;
    IF TBEND GT (SYR+1)*100+6 AND TBEND LE YRISKB + 100
      THEN DO;
        LASTBID=BID;
        LASTYR=SYR;
        LASTX=TXOETS;
        LASTNTS=TNTS;
        LASTTOE=TXTOE;
        LASTOUT=TENDLS;
      END;
    ELSE IF (TBEGIN GT SYR*100+6 OR TRMDUM = 1) AND BID EQ LASTBID AND
      LASTYR=SYR-1 THEN DO;
      TXOETS=LASTX;
      TNTS=LASTNTS;
      TXTOE=LASTTOE;
      TENDLS=LASTOUT;
    END;
    ELSE IF TBEGIN GT SYR*100+6 THEN DO;
      GOODREC=0;
      TXOETS=.;
      TNTS=.;
      TXTOE=.;
      TENDLS=.;
    END;

  PROC PRINT ;
  /*
  //
```

Appendix C

SUBROUTINES OF THE YAR PROGRAM

SUBR GROUP	SUBR NAME	PURPOSE
yar	adjdoe	adjust doe to ignore day of month in raw date from 1980 onward
yar	ext	assign each extension to a termyear; output ext. variables
yar	fixper	impute doe from stafms if it is missing
yar	grdpnt	sets tafms, yos, and grade variables to the jsyr snapshot year
yar	lastsn	have we written the june snapshot preceding start of last term?
yar	milpay	create the pay, skill, grade, and service variables
yar	nxtlv	determine when an airmen next leaves the force
yar	oets12	set cohort, time-in-service, and grade pointers
yar	person	output person-level variables to a term/year record
yar	pntcnt	find begin & end transactions; call appropriate routines
yar	prevsk	check for quickly-returning losses
yar	setbeg	set begin date for most recently registered term (eg. 7606)
yar	setxoe	determine original separation date (XOETS) for a term of service
yar	snapfx	indicate if missing grade, married, etc., in snapshot data
yar	snapsh	output snapshot-level variables to a term/year record
yar	srtext1	sort transactions
yar	srtext2	put extension, loss, and reup transactions in proper sequence
yar	tafscs	create term-level afsc variables
yar	terms	output term-related variables
yar	tfmsfx	correct some stafms errors
yar	trmsrv	determine airman's term of service (1st, 2nd, career)
yar	trmyr	determine the number of termyears in a term
yar	trmyr1	create variable tgrd, which is passed up to subroutine trmyrs
yar	trmyrs	put the variable tgrd into common block grdzz
yar	trnsac	output transaction-level variables
yar	uinit	initialize a multitude of common blocks
yar	ypafsc	find afsc 12-months prior to end of termyear
util	adate	map yymm and yyxxx to yyxxx
util	cdate	map yymm and yyxxx to yymm
util	getpm9	move characters, but replace ' ' with '-9'
util	getput	move characters
util	ivalue	character to integer conversion
util	mdlgrd	map tafms into fractional grade
util	mo70	map yrmo (e.g. 7809) to months beyond dec 1969
util	oequal	compare character strings
util	period	map months beyond dec 1969 to bonus period number
util	qread	read fixed length record
util	s10160	calculate days since jan 1, 1960

SUBR GROUP	SUBR NAME	PURPOSE
util	sdate	map yymm and yyxxx to yy for last june
util	sortcq	sort
util	vread	read variable blocksize record; return record and length
util	zone	map tafms into bonus zones 1,2,3; 1.5 and 2.5 where not 1-1
file	afscf1	afsc supplementary file: table lookup
file	afscil	afsc supplementary file: input
file	afscv	afsc supplementary file: define lookup variables
file	afscx1	afsc supplementary file: control
file	bonusf	bonus supplementary file: table lookup
file	bonusi	bonus supplementary file: input
file	bonusv	bonus supplementary file: define lookup variables
file	bonusx	bonus supplementary file: control
file	bonusz	bonus supplementary file: control
file	conusf	conus supplementary file: table lookup
file	conusv	conus supplementary file: define lookup variables
file	cpay2i	civilian pay supplementary files: input
file	cpay3i	civilian pay supplementary files: input
file	cpayf1	civilian pay supplementary files: table lookup
file	cpayf2	civilian pay supplementary files: table lookup
file	cpayf3	civilian pay supplementary files: table lookup
file	cpayv	civilian pay supplementary files: define lookup variables
file	cpayv1	civilian pay supplementary files: define lookup variables
file	cpayvx	civilian pay supplementary files: define lookup variables
file	cpayx	civilian pay supplementary files: control
file	dates	afsc supplementary file: dates for table lookup
file	fmtrd	supplementary files: formatted input
file	link	supplementary files: table generation and lookup
file	linkfi	supplementary files: table lookup
file	linkhb	supplementary files: table building
file	linkhl	supplementary files: table lookup
file	linkin	supplementary files: table input
file	mpayf	military pay supplementary file: table lookup
file	mpayv	military pay supplementary file: define lookup variables
file	mpayx	military pay supplementary file: control
file	rcdttr	spds supplementary file: control
file	unem2f	second unemployment supplementary file: table lookup
file	unem2v	second unemployment supplementary file: define lookup variables
file	unem2x	second unemployment supplementary file: control
file	unempf	first unemployment supplementary file: table lookup
file	unempv	first unemployment supplementary file: define lookup variables
file	unempx	first unemployment supplementary file: control
input	initp	initialize person scalars with data
input	initp0	initialize person scalars with -9's
input	inits	initialize snapshot vectors with data
input	inits0	initialize snapshot vectors with -9's

SUBR	SUBR	
GROUP	NAME	PURPOSE
input	initt	initialize transaction vectors with data
input	initt0	initialize transaction vectors with -9's
input	input	input
output	ouinit	initialize output vectors with -9's
output	outpt	initialize output record with data
output	output	output
debug	ftninp	select cases
debug	out	write subroutine name

Appendix D

THE SUBROUTINE STRUCTURE OF THE YAR FORTRAN PROGRAM

Two useful ways to represent the subroutine structure of the YAR FORTRAN Program are by which routine calls which other routines and by the level of complexity of individual routines. This appendix presents a listing of "who calls whom" and a listing of subroutines arrayed by their complexity. (By "complexity" we mean the extent to which a subroutine calls other subroutines that, in turn, call other subroutines. In this sense the least complex subroutines are those that call no other subroutines. The next least complex are those that call only subroutines that in turn call no other subroutines. By repeated application of this notion, one arrives at a hierarchy of complexity.) Table D.1 contains the list of who calls whom. Table D.2 contains the list of subroutines arranged according to their degree of complexity. The levels of complexity are indicated by alphabetic codes from n-z, with n indicating the most complex subroutines and z the least complex subroutines (see Table D.2).

Table D.1

WHO CALLS WHOM AMONG YAR SUBROUTINES

Calling Routine		Called Routine	
Name	Complexity	Name	Complexity
adjdoe	y	out	z
afscf1	v	link	w
		getput	z
		out	z
afscf2	v	link	w
		getput	z
		out	z
afscv	u	afscf1	v
		getput	z
		out	z
afscx1	y	out	z
bonusf	v	link	w
		getput	z
		out	z
bonusv	u	bonusf	v
		out	z
bonusx	t	bonusv	u
		getput	z
		out	z
bonusz	t	bonusv	u
		out	z
conusf	v	link	w
		getput	z
conusv	u	conusf	v
cpayf1	v	link	w
		getput	z
		out	z

Table D.1—continued

Calling Routine		Called Routine	
Name	Complexity	Name	Complexity
cpayf2	v	link	w
		getput	z
		out	z
cpayf3	v	link	w
		out	z
cpayv	s	cpayvx	t
cpayv1	u	cpayf1	v
		cpayf2	v
		cpayf3	v
		out	z
cpayvx	t	capyv1	u
		out	z
cpayx	r	cpayv	s
		out	z
ext	y	out	z
fixper	w	rcdttr	x
		out	z
grdpnt	w	trmyrs	x
		oets12	y
		out	z
initp0	y	getput	z
		out	z
inits	y	getput	z
		out	z
initt	y	getput	z
		out	z
inpsch	y	qread	z

Table D.1—continued

Calling Routine		Called Routine	
Name	Complexity	Name	Complexity
input	x	initp	y
		inits	y
		initt	y
		ftninp	z
		getput	z
		initp0	z
		inits0	z
		initt0	z
		out	z
		vread	z
link	w	linkfi	x
		linkin	z
linkfi	x	linkhb	y
		linkhl	z
linkhb	y	getput	z
main	n	pntcnt	o
		fixper	w
		srtext2	w
		input	x
		output	x
		adjoe	y
		snapfx	y
		srtext1	y
		tfmsfx	y
		uinit	y
milpay	q	ouinit	z
		cpayx	r
		bonusz	s
		bonusx	t
		mpayx	t
		prevsk	t
		unempx	t
		ypafsc	t
		afscv	u
		conusv	u
		grdpnt	w
		tafscs	y
		out	z

Table D.1—continued

Calling Routine		Called Routine	
Name	Complexity	Name	Complexity
mpayf	v	link out	w z
mpayv	u	mpayf out	v z
mpayx	t	mpayv out	u z
nxtlv	y	out	z
oets12	y	out	z
outpt	y	getput out	z z
output	x	outpt getput out	y z z
person	y	out	z
pntent	o	trmyr milpay qbaksn trnsac rcdttr ext lastsn nxtlv person setbeg setxoe snapsh terms trmsrv lastsn out	p q w w x y y y y y y y y z z
prevsk	t	afscv afscx1 getput out	u y z z

Table D.1—continued

Calling Routine		Called Routine	
Name	Complexity	Name	Complexity
qbaksn	w	redttr out	x z
redttr	x	inpsch out	y z
setbeg	y	out	z
setxoe	y	out	z
snapfx	y	out	z
snapsh	y	out	z
srtext1	y	getput sortcq	z z
srtext2	w	redttr out	x z
tafscs	y	out	z
terms	y	out	z
tfmsfx	y	out	z
trmsrv	y	out	z
trmyr	v	milpay trnsac ext person snapsh terms out	q w y y y y z
trmyr1	y	out	z
trmyrs	x	trmyr1 out	y z

Table D.1—continued

Calling Routine		Called Routine	
Name	Complexity	Name	Complexity
trnsac	w	rcdttr out	x z
unit	y	out	z
unempf	v	link out	w z
unempv	u	unempf out	v z
unempx	t	unempv out	u z
ypafsc	t	afscv afscx1 out	u y z
FUNCTIONS:			
mdlgrd	y	out	z
mo70	y	out	z
period	y	out	z
s10160	y	out	z
zone	y	out	z

Table D.2

THE COMPLEXITY OF YAR FORTRAN SUBROUTINES

Level z (Call no subroutines)		
getput		
initp0		
inits0		
initt0		
lastsn		
linkhl		
linkin		
ouinit		
out		
qread		
sortcq		
vread		
Level y (Call only Level z subroutines)		
adjdoe		
afscx1		
ext		
fixper		
initp		
inits		
initt		
inpsch		
linkhb		
nxtlv		
oets12		
outpt		
person		
setbeg		
setxoe		
snapfx		
snapsh		
srtex1		
srtex2		
tafscs		
terms		
tfmsfx		
trmsrv		
trmyr1		
uinit		
mdlgrd	(a function)	
mo70	(a function)	
period	(a function)	
s10160	(a function)	
zone	(a function)	
Level x (Call only Level z or y subroutines)		
input		
linkfi		
output		
rcdttr		
trmyrs		

Table D.2—continued

Level w (Call only Level z, y, or x subroutines)
fixper
grdpnt
link
qbaksn
srtex2
trnsac
Level v (Call only Level z,y,x, or w subroutines)
afscf1
afscf2
bonusf
conusf
cpayf1
cpayf2
cpayf3
mpayf
unempf
Level u (Call only Level z,y,x,w, or v subroutines)
afscv
bonusv
conusv
cpayv1
mpayv
unempv
Level t (Call only Level z,y,x,w,v, or u subroutines)
bonusx
bonusz
cpayvx
mpayx
prevsk
unempx
ypafsc
Level s (Call only Level z,y,x,w,v,u, or t subroutines)
cpayv
Level r (Call only Level z,y,x,w,v,u,t, or s subroutines)
cpayx
Level q (Call only Level z,y,x,w,v,u,t,s, or r subroutines)
milpay
Level p (Call only Level z,y,x,w,v,u,t,s,r, or q subroutines)
trmyr
Level o (Call only Level z,y,x,w,v,u,t,s,r,q, or p subroutines)
pntcnt
Level n (Call anything)
main

Appendix E

CODE USED TO CONSTRUCT THE SAS VERSION OF THE YAR5 FILE

Because most of the data and statistical analysis on the Enlisted Force Management Project was performed using SAS, the YAR5 file was converted into a SAS file. Listed below is the SAS program that was used to create this SAS file. Note that the program does more than just convert the YAR file into a SAS file. It creates two new variables (GOODSNAP and GOODTYR) and sets missing values to -9.

```
options noovp nocenter;

/* may 1988      */

data sasdata1.yar87064 (drop=holdssan count);

retain holdssan bid;

length
bid $ 10 holdssan $ 10
bafqta bblk bdepen biage bieduc bitoe blsed bmale bmarry
bmred bnoded bpace brace bsex bsing bslowr bldep goodsnap
sdep sedu sgrd smar syr slgrd s2grd yconus ygrdl2
ynext yntyr ytrmyr yyos12 goodtyr tbdobs tendls tgrdl2 tlasyr
tnext tnts tntyr tpsa trmdum txtoe tyos12 xcarea xchar
xcoh xctrn xecat xgrade xtoe 2
bdob sdos staf stig sltg s2tg sltaf s2taf
ydosin ydosou yfms12 ynxtls yriskb tbegin tbend tfms12 tqwbak
tqwlst txoets xdos xexted1 xexted2 xxtdos1 xxtdos2 xtdcsa 3
default=4;
infile inpl;
input
@ 1 ssan $char10. /* 1 10 10 */
@ 113 bafqta rb4. /* 113 116 4 */
@ 133 bblk rb4. /* 133 136 4 */
@ 105 bdepen rb4. /* 105 108 4 */
@ 97 bdob rb4. /* 97 100 4 */
@ 141 biage rb4. /* 141 144 4 */
@ 93 bidoe rb4. /* 93 96 4 */
@ 81 bieduc rb4. /* 81 84 4 */
@ 109 bitoe rb4. /* 109 112 4 */
@ 121 blsed rb4. /* 121 124 4 */
@ 137 bmale rb4. /* 137 140 4 */
@ 101 bmarry rb4. /* 101 104 4 */
```

@ 125	bmred	rb4.	/* 125 128 4 */
@ 149	bnodep	rb4.	/* 149 152 4 */
@ 117	bpace	rb4.	/* 117 120 4 */
@ 85	brace	rb4.	/* 85 88 4 */
@ 89	bsex	rb4.	/* 89 92 4 */
@ 145	bsing	rb4.	/* 145 148 4 */
@ 129	bslowr	rb4.	/* 129 132 4 */
@ 153	bldep	rb4.	/* 153 156 4 */
@ 18	sdaf	\$char7.	/* 18 24 7 */
@ 173	sdep	rb4.	/* 173 176 4 */
@ 325	sdos	rb4.	/* 325 328 4 */
@ 169	sedu	rb4.	/* 169 172 4 */
@ 181	sgrd	rb4.	/* 181 184 4 */
@ 177	smar	rb4.	/* 177 180 4 */
@ 11	spaf	\$char7.	/* 11 17 7 */
@ 48	spas	\$char4.	/* 48 51 4 */
@ 189	spoc	rb4.	/* 189 192 4 */
@ 165	staf	rb4.	/* 165 168 4 */
@ 185	stig	rb4.	/* 185 188 4 */
@ 161	syr	rb4.	/* 161 164 4 */
@ 193	slgrd	rb4.	/* 193 196 4 */
@ 197	sltg	rb4.	/* 197 200 4 */
@ 209	sltaf	rb4.	/* 209 212 4 */
@ 201	s2grd	rb4.	/* 201 204 4 */
@ 205	s2tg	rb4.	/* 205 208 4 */
@ 593	s2taf	rb4.	/* 593 596 4 */
@ 461	ybm1t6	rb4.	/* 461 464 4 */
@ 621	yconus	rb4.	/* 621 624 4 */
@ 354	ydafl2	\$char7.	/* 354 360 7 */
@ 269	ydosin	rb4.	/* 269 272 4 */
@ 273	ydosou	rb4.	/* 273 276 4 */
@ 437	yelbp	rb4.	/* 437 440 4 */
@ 489	yelcpq	rb4.	/* 489 492 4 */
@ 445	yelmp	rb4.	/* 445 448 4 */
@ 425	yfms12	rb4.	/* 425 428 4 */
@ 433	ygrd12	rb4.	/* 433 436 4 */
@ 347	ylgp12	\$char7.	/* 347 353 7 */
@ 601	ymcwrat	rb4.	/* 601 604 4 */
@ 265	ynext	rb4.	/* 265 268 4 */
@ 625	yntyr	rb4.	/* 625 628 4 */
@ 329	ynxtls	rb4.	/* 329 332 4 */
@ 340	ypaf12	\$char7.	/* 340 346 7 */
@ 629	yriskb	rb4.	/* 629 632 4 */
@ 213	ytrmyr	rb4.	/* 213 216 4 */
@ 453	yt1bp	rb4.	/* 453 456 4 */
@ 465	yt1cp	rb4.	/* 465 468 4 */
@ 441	yt1mp	rb4.	/* 441 444 4 */
@ 469	yt0uem	rb4.	/* 469 472 4 */
@ 497	yt1uem	rb4.	/* 497 500 4 */
@ 501	yt2uem	rb4.	/* 501 504 4 */

@ 333	yuaf12	\$char7.	/* 333 339	7 */
@ 361	yudf12	\$char7.	/* 361 367	7 */
@ 429	yyos12	rb4.	/* 429 432	4 */
@ 577	tabmlt	rb4.	/* 577 580	4 */
@ 301	tbdobs	rb4.	/* 301 304	4 */
@ 281	tbegln	rb4.	/* 281 284	4 */
@ 285	tbend	rb4.	/* 285 288	4 */
@ 585	ttml12	rb4.	/* 585 588	4 */
@ 617	tbrecd	rb4.	/* 617 620	4 */
@ 410	tdaf12	\$char7.	/* 410 416	7 */
@ 293	tendls	rb4.	/* 293 296	4 */
@ 533	tetmp	rb4.	/* 533 536	4 */
@ 505	tfms12	rb4.	/* 505 508	4 */
@ 513	tgrdl2	rb4.	/* 513 516	4 */
@ 375	tlafsc	\$char7.	/* 375 381	7 */
@ 517	tlasyr	rb4.	/* 517 520	4 */
@ 403	tlgpl2	\$char7.	/* 403 409	7 */
@ 581	tmbmlt	rb4.	/* 581 584	4 */
@ 297	tnext	rb4.	/* 297 300	4 */
@ 277	tnts	rb4.	/* 277 280	4 */
@ 633	tntyr	rb4.	/* 633 636	4 */
@ 613	tnwbom	rb4.	/* 613 616	4 */
@ 521	tolbp	rb4.	/* 521 524	4 */
@ 545	tolcp	rb4.	/* 545 548	4 */
@ 525	tolmp	rb4.	/* 525 528	4 */
@ 565	toluem	rb4.	/* 565 568	4 */
@ 637	totspd	\$char3.	/* 637 639	3 */
@ 396	tpaf12	\$char7.	/* 396 402	7 */
@ 589	tprvbm	rb4.	/* 589 592	4 */
@ 313	tpsa	rb4.	/* 313 316	4 */
@ 305	tqwbak	rb4.	/* 305 308	4 */
@ 309	tqwlst	rb4.	/* 309 312	4 */
@ 317	trmdum	rb4.	/* 317 320	4 */
@ 561	ttludem	rb4.	/* 561 564	4 */
@ 382	tuafr	\$char7.	/* 382 388	7 */
@ 389	tuaf12	\$char7.	/* 389 395	7 */
@ 368	tucaf	\$char7.	/* 368 374	7 */
@ 417	tudf12	\$char7.	/* 417 423	7 */
@ 289	txoets	rb4.	/* 289 292	4 */
@ 321	txtoe	rb4.	/* 321 324	4 */
@ 509	tyos12	rb4.	/* 509 512	4 */
@ 30	xcafsc	\$char6.	/* 30 35	6 */
@ 241	xcarea	rb4.	/* 241 244	4 */
@ 229	xchar	rb4.	/* 229 232	4 */
@ 157	xcoh	rb4.	/* 157 160	4 */
@ 245	xctrn	rb4.	/* 245 248	4 */
@ 225	xdos	rb4.	/* 225 228	4 */
@ 233	xecat	rb4.	/* 233 236	4 */
@ 28	xelig	\$char2.	/* 28 29	2 */
@ 36	xextaf1	\$char6.	/* 36 41	6 */


```

@ 42  xextaf2  $char6.          /* 42  47  6 */
@ 249  xexted1  rb4.           /* 249 252 4 */
@ 253  xxtdos1  rb4.           /* 253 256 4 */
@ 257  xexted2  rb4.           /* 257 260 4 */
@ 261  xxtdos2  rb4.           /* 261 264 4 */
@ 221  xgrade   rb4.           /* 221 224 4 */
@ 217  xtdcsa   rb4.           /* 217 220 4 */
@ 237  xtoe     rb4.           /* 237 240 4 */
@ 25   xtrnid   $char3.        /* 25   27  3 */
@ 597  goodsnap rb4.           /* 597 268 4 */
@ 573  goodtyr  rb4.           /* 573 244 4 */
@ 609  maunemp  rb4.           /* 609 280 4 */
@ 605  munemp   rb4.           /* 605 276 4 */
@ 553  tmcwrat  rb4.           /* 553 556 4 */
@ 557  tmunemp  rb4.           /* 557 560 4 */
@ 569  tmaunemp rb4. ;        /* 569 572 4 */

/* ===== */
/* the following variables have been omitted from the sas file */
/* ===== */
* @ 493  yecpg   rb4.          /* 493 496 4 */
* @ 477  yocpg   rb4.          /* 477 480 4 */
* @ 485  yocpgl  rb4.          /* 485 488 4 */
* @ 481  yocpgn  rb4.          /* 481 484 4 */
* @ 473  ytlcpn  rb4.          /* 473 476 4 */
* @ 449  ytlmpn  rb4.          /* 449 452 4 */
* @ 549  tolcp1  rb4.          /* 549 552 4 */
* @ 529  tolmp1  rb4.          /* 529 532 4 */
* @ 457  yetbp   rb4.          /* 457 460 4 */
* @ 537  tetbp   rb4.          /* 537 540 4 */
* @ 541  te0cpg  rb4. ;        /* 541 544 4 */

if ssan /= holdssan then do;
  holdssan = ssan;
  count+1;
  bid=input (put (count,z9.),$9.)||trim(substr(ssan,9,1));
end;

proc freq data=sasdata1.yar87064 ;
tables
  bdepen biage bidoe bieduc bitoe bmale bmarry bpace brace bsex
  sdos sedu sgrd staf stig syr
  ybmlt6 ydosin ydosou yelbp yelcpg yelmp yfms12 ygrd12
  ylgp12 ymcwrat yntyrr ynxtls yriskb ytrmyr ytlbp ytlcp ytlmp
  ytluem yyos12 tmcwrat tmunemp tmaunemp
  tabmlt tbdobs tbegin tbend tbml12 tbrecd tendls tfms12 tgrd12
  tlasyr tmbmlt tprvbm tnext tnts tntyrr tnwbon tolbp tolcp tolmp
  toluem tpsa tqwbak tqwlst trmdum tlluem txoets txtoe tyos12
  xcch xctrn xdos xexted1 xtrnid xxtdos1 xgrade xtdcsa xtoe
  goodtyr goodsnap goodsnap*syr tendls*trmdum yconus*syr;

```

Appendix F

RAND RECODES OF TRANSACTION IDs (XTRIND)

RAND CODES (XCTRN)

0 = Not used in time period
1 = Attrition-Marginal performer
2 = Attrition-Miscellaneous reasons
3 = Attrition-Hardship
4 = Attrition-Pregnancy
5 = Attrition-Death
6 = Attrition-Early Retirement
7 = Attrition-Disability
8 = Attrition-Unsuitability/Misconduct/Quality (AFM 39-12)
9 = Attrition-Other
10 = OTS/AECP

11 = Reenlistment, le 3 months PETS
12 = Reenlistment, gt 3 months PETS
13 = Reenlistment, PETS
14 = Reenlistment
15 = Reenlistment, PETS (Reservist)
16 = Reenlistment, ETS (Reservist)
17 = Reenlistment, le 3 mo post DOS

21 = Extension
22 = Extension, ge 24 months
23 = Extension, lt 24 months

30 = Retirement

41 = Prior service gain

81 = Early Release

91 = ETS
92 = PETS
93 = Special

99 = Other

CODES USED IN SHORT-TERM AGGREGATE IPM (SAM)

ANG = To Air National Guard
 CDFR = Correction of "dropped from rolls"
 EXT = Extension transaction
 GAIN = Gain transaction
 LATR = Loss due to attrition
 LETS = Loss at ETS
 LOTS = Loss to Officer Training School (OTS)
 LRET = Loss to retirement
 LSPC = Special loss (PETS and early release)
 NPS = Nonprior service accession
 RDFR = Returned "dropped from rolls" (usually to discharge)
 REUP = Reenlistment
 XXXX = Not recoded

XTRNID (SPD)	XCTRN Time Period			SAM Code
	≤ 7406	7407-8409	≥ 8410	
B30	0	41	41	GAIN
B31	0	41	41	GAIN
CBL	0	9	9	LATR
DFS	0	8	8	LATR
EKD	0	8	8	LATR
EKF	0	8	8	LATR
FBL	0	9	9	LATR
FHC	0	99	99	LATR
FLB	0	9	9	LATR
FTY	0	99	99	LATR
GDA	0	8	8	LATR
GDG	0	9	9	LATR
GDJ	0	99	99	LATR
GDK	0	8	8	LATR
GFC	0	8	8	LATR
GFJ	0	99	99	LATR
GFK	0	8	8	LATR
GFL	0	99	99	LATR
GFT	0	9	9	LATR
GFV	0	8	8	LATR
GFX	0	8	8	LATR
GHF	0	99	99	LATR
GHJ	0	1	8	LATR
GKA	0	8	8	LATR
GKB	0	8	8	LATR
GKC	0	8	8	LATR
GKD	0	8	8	LATR

XTRNID (SPD)	XCTR Time Period			SAM Code
	≤ 7406	7407-8409	≥ 8410	
GKE	0	8	8	LATR
GKF	0	8	8	LATR
GKG	0	8	8	LATR
GKH	0	8	8	LATR
GKJ	0	8	8	LATR
GKK	0	8	8	LATR
GKL	0	8	8	LATR
GKM	0	8	8	LATR
GKN	0	8	8	LATR
GKQ	0	8	8	LATR
GLB	0	8	8	LATR
GLC	0	8	8	LATR
GLF	0	8	8	LATR
GLG	0	8	8	LATR
GLH	0	8	8	LATR
GLJ	0	8	8	LATR
GLK	0	8	8	LATR
GLL	0	8	8	LATR
GMB	0	8	8	LATR
GMC	0	8	8	LATR
GMD	0	8	8	LATR
GME	0	8	8	LATR
GMG	0	8	8	LATR
GMH	0	8	8	LATR
GMJ	0	8	8	LATR
GMK	0	99	99	LATR
GML	0	8	8	LATR
GMM	0	8	8	LATR
GMN	0	8	8	LATR
GMP	0	8	8	LATR
GNE	0	99	99	LATR
GPB	0	8	8	LATR
GPC	0	8	8	LATR
GPD	0	8	8	LATR
GRA	0	8	8	LATR
GRB	0	8	8	LATR
GRC	0	8	8	LATR
GSG	0	99	99	LATR
GSL	0	99	99	LATR
GSM	0	99	99	LATR
G34	0	8	8	LATR
G35	0	8	8	LATR
G36	0	8	8	LATR
G37	0	8	8	LATR
G39	0	8	8	LATR

XTRNID (SPD)	XCTR Time Period			SAM Code
	≤ 7406	7407-8409	≥ 8410	
G40	0	8	8	LATR
G41	0	8	8	LATR
G42	0	8	8	LATR
G43	0	99	99	LATR
G44	0	99	99	LATR
G99	0	99	99	LATR
HDA	0	8	8	LATR
HDG	0	9	9	LATR
HDK	0	8	8	LATR
HFC	0	8	8	LATR
HFJ	0	99	99	LATR
HFJ	0	99	99	LATR
HFK	0	99	99	LATR
HFT	0	9	9	LATR
HFV	0	8	8	LATR
HFX	0	8	8	LATR
HHJ	0	1	8	LATR
HHN	0	99	99	LATR
HKA	0	8	8	LATR
HKB	0	8	8	LATR
HKC	0	8	8	LATR
HKD	0	8	8	LATR
HKE	0	8	8	LATR
HKF	0	8	8	LATR
HKG	0	8	8	LATR
HKH	0	8	8	LATR
HKJ	0	8	8	LATR
HKK	0	8	8	LATR
HKL	0	8	8	LATR
HKM	0	8	8	LATR
HKN	0	8	8	LATR
HKQ	0	8	8	LATR
HLB	0	8	8	LATR
HLC	0	8	8	LATR
HLF	0	8	8	LATR
HLG	0	8	8	LATR
HLH	0	8	8	LATR
HLJ	0	8	8	LATR
HLK	0	8	8	LATR
HLL	0	8	8	LATR
HMB	0	8	8	LATR
HMC	0	8	8	LATR
HMD	0	8	8	LATR
HMF	0	8	8	LATR
HMG	0	8	8	LATR

XTRNID (SPD)	XCTRN Time Period			SAM Code
	≤ 7406	7407-8409	≥ 8410	
HMH	0	8	8	LATR
HMJ	0	8	8	LATR
HMK	0	99	99	LATR
HML	0	8	8	LATR
HMM	0	8	8	LATR
HMN	0	8	8	LATR
HMP	0	8	8	LATR
HNF	0	99	99	LATR
HPB	0	8	8	LATR
HPC	0	8	8	LATR
HPD	0	8	8	LATR
HRA	0	8	8	LATR
HRB	0	8	8	LATR
HRC	0	8	8	LATR
HSG	0	99	99	LATR
HSK	0	99	99	LATR
HSL	0	99	99	LATR
HSM	0	99	99	LATR
H34	0	8	8	LATR
H35	0	8	8	LATR
H36	0	8	8	LATR
H37	0	8	8	LATR
H39	0	8	8	LATR
H40	0	8	8	LATR
H41	0	8	8	LATR
H42	0	8	8	LATR
H43	0	99	99	LATR
H44	0	99	99	LATR
JBB	0	9	9	LATR
JBC	0	9	9	LATR
JBK	0	99	99	LETS
JBM	0	92	92	LETS
JBP	0	99	99	LATR
JDA	0	8	8	LATR
JDF	0	4	4	LATR
JDG	0	9	9	LATR
JDK	0	8	8	LATR
JDR	0	81	81	LSPC
JEB	0	99	99	LOTS
JED	0	92	92	LETS
JEM	0	1	1	LATR
JET	0	1	1	LATR
JFB	0	9	9	LATR
JFC	0	8	8	LATR
JFE	0	9	9	LATR

XTRNID (SPD)	XCTR Time Period			SAM Code
	≤ 7406	7407-8409	≥ 8410	
JFF	0	9	9	LATR
JFK	0	8	8	LATR
JFL	0	7	7	LATR
JFM	0	7	7	LATR
JFP	0	7	7	LATR
JFR	0	7	7	LATR
JFT	0	9	9	LATR
JFU	0	9	9	LATR
JFV	0	8	8	LATR
JFX	0	8	8	LATR
JGA	0	8	8	LATR
JGH	0	1	1	LATR
JGU	0	99	99	LOTS
JGZ	0	1	1	LATR
JHD	0	9	9	LATR
JHF	0	99	99	LATR
JHJ	0	1	8	LATR
JHM	0	9	9	LATR
JJB	0	99	99	LATR
JJC	0	9	9	LATR
JJD	0	9	9	LATR
JKA	0	8	8	LATR
JKB	0	8	8	LATR
JKC	0	8	8	LATR
JKD	0	8	8	LATR
JKE	0	8	8	LATR
JKF	0	8	8	LATR
JKG	0	8	8	LATR
JKH	0	8	8	LATR
JKJ	0	8	8	LATR
JKK	0	8	8	LATR
JKL	0	8	8	LATR
JKM	0	8	8	LATR
JKN	0	8	8	LATR
JKQ	0	8	8	LATR
JMB	0	8	8	LATR
JMC	0	8	8	LATR
JMD	0	8	8	LATR
JMF	0	8	8	LATR
JMG	0	8	8	LATR
JMH	0	8	8	LATR
JMJ	0	8	8	LATR
JML	0	8	8	LATR
JMM	0	9	9	LATR

XTRNID (SPD)	XCTRN Time Period			SAM Code
	≤ 7406	7407-8409	≥ 8410	
JMN	0	99	99	LATR
JMP	0	8	8	LATR
JNF	0	99	99	LATR
JPB	0	8	8	LATR
JPC	0	8	8	LATR
JPD	0	8	8	LATR
JTS	0	99	99	LATR
JTW	0	99	99	LATR
JTX	0	99	99	LATR
J10	0	9	9	LATR
J11	0	8	8	LATR
J15	0	92	92	LETS
J22	0	81	81	LSPC
J23	0	81	81	LSPC
J24	0	99	99	XXXX
J33	0	9	9	LATR
J34	0	8	8	LATR
J35	0	8	8	LATR
J36	0	8	8	LATR
J37	0	8	8	LATR
J38	0	8	8	LATR
J39	0	8	8	LATR
J40	0	8	8	LATR
J41	0	8	8	LATR
J42	0	8	8	LATR
J43	0	99	99	LATR
J44	0	99	99	LATR
KBD	0	30	30	LRET
KBJ	0	9	9	LATR
KBK	91	91	91	LETS
KCE	0	92	92	LETS
KCF	92	92	92	LETS
KCG	0	9	9	LATR
KCK	0	9	9	LATR
KCL	0	99	99	LATR
KCM	0	9	9	LATR
KCQ	0	9	9	LATR
KDB	0	3	3	LATR
KDC	0	9	9	LATR
KDF	0	4	4	LATR
KDH	0	99	99	LATR
KDM	0	91	91	LETS
KDQ	0	9	9	LATR
KDR	0	81	81	LSPC
KEA	0	91	91	LETS

XTRNID (SPD)	XCTRN Time Period			SAM Code
	≤ 7406	7407-8409	≥ 8410	
KEB	0	91	91	LETS
KEC	0	91	91	LETS
KFB	0	9	9	LATR
KFC	0	9	9	LATR
KFN	0	7	7	LATR
KFS	0	8	8	LATR
KFU	0	9	9	LATR
KFX	0	8	8	LATR
KGL	0	10	10	LOTS
KGM	0	10	10	LOTS
KGN	0	9	9	LATR
KGU	0	99	99	LOTS
KGW	0	9	9	LATR
KGX	0	93	93	LETS
KGY	0	99	99	LOTS
KHD	0	9	9	LATR
KMD	0	99	99	LSPC
KND	0	2	2	LATR
KNL	0	8	8	LATR
KTP	0	99	99	XXXX
KTS	0	99	99	LATR
K13	0	91	91	LETS
K14	0	92	92	LETS
K16	0	92	92	LETS
K17	0	9	9	LATR
K18	0	91	91	LETS
K19	0	91	91	LETS
K20	0	93	93	LSPC
K21	0	93	93	LSPC
K22	0	81	81	LSPC
K23	0	81	81	LSPC
K24	0	99	99	XXXX
LAD	0	9	9	LATR
LBB	0	99	99	XXXX
LBK	0	99	99	LETS
LBM	0	92	92	LETS
LCC	0	99	99	XXXX
LDR	0	81	81	LSPC
LED	0	92	92	LETS
LFC	0	99	99	XXXX
LEJ	0	99	99	LATR
LEK	0	99	99	LRET
LEF	0	30	30	LRET
LGM	0	10	10	LATR
LHD	0	9	9	LATR

XTRNID (SPD)	XCTRN Time Period			SAM Code
	≤ 7406	7407-8409	≥ 8410	
LTL	0	99	99	LATR
LTM	0	99	99	LATR
L15	0	92	92	LETS
L22	0	81	81	LSPC
L23	0	81	81	LSPC
L24	0	99	99	XXXX
L33	0	9	9	LATR
MBJ	0	9	9	LATR
MBK	0	91	91	LETS
MBN	0	91	91	LETS
MCE	0	92	92	LETS
MCF	0	92	92	LETS
MCK	0	9	9	LATR
MDF	0	4	4	LATR
MDG	0	99	99	XXXX
MDM	0	91	91	LETS
MDR	0	81	81	LSPC
MEA	0	91	91	LETS
MEB	0	91	91	LETS
MEC	0	91	91	LETS
MFK	0	99	99	XXXX
MGU	0	99	99	LOTS
MGX	0	93	93	LETS
MND	0	2	2	LATR
MTB	0	0	99	XXXX
MTM	0	99	99	XXXX
MTN	0	99	99	XXXX
M12	0	91	91	LETS
M13	0	91	91	LETS
M14	0	92	92	LETS
M16	0	92	92	LETS
M18	0	91	91	LETS
M19	0	91	91	LETS
M20	0	93	93	LSPC
M21	0	93	93	LSPC
M22	0	81	81	LSPC
M23	0	81	81	LSPC
M24	0	99	99	XXXX
PGU	0	9	9	LATR
RBC	30	30	30	LRET
RBD	30	30	30	LRET
R12	0	30	30	LRET
R99	0	99	99	XXXX
SBB	0	99	99	LRET
SBC	0	99	99	LRET

XTRNID (SPD)	XCTRN Time Period			SAM Code
	≤ 7406	7407-8409	≥ 8410	
SBD	0	99	99	LRET
SFJ	6	6	6	LATR
SFK	0	6	6	LATR
TCC	0	91	91	LETS
TEE	0	91	91	LETS
T43	0	91	91	LETS
VBK	0	30	30	LRET
VFJ	0	30	30	LRET
VFK	0	30	30	LRET
YDN	0	9	9	LATR
YFB	0	9	9	LATR
001	41	41	41	GAIN
002	41	41	41	GAIN
003	41	41	41	GAIN
004	99	99	99	NPS
005	41	41	41	GAIN
006	41	41	41	GAIN
007	99	99	99	NPS
008	99	99	99	LOTS
009	99	99	99	LOTS
010	41	41	41	GAIN
011	17	17	17	REUP
012	99	99	99	NPS
013	99	99	99	NPS
014	99	99	99	NPS
015	99	99	99	NPS
016	99	99	99	NPS
017	99	99	99	NPS
018	99	99	99	NPS
019	99	99	99	NPS
020	99	99	99	RDFR
021	99	99	99	RDFR
022	99	99	99	RDFR
023	99	99	99	CDFR
024	99	99	99	RDFR
025	99	99	99	RDFR
026	41	41	41	GAIN
030	99	99	99	CDFR
031	99	99	99	CORR
040	0	0	41	GAIN
041	0	0	41	GAIN
042	0	0	41	GAIN
043	0	0	41	GAIN
044	0	0	41	GAIN
050	0	99	99	NPS

XTRNID (SPD)	XCTR Time Period			SAM Code
	≤ 7406	7407-8409	≥ 8410	
051	0	99	0	NPS
052	0	99	99	NPS
053	0	99	0	NPS
054	0	99	99	NPS
055	0	99	99	NPS
056	0	99	99	NPS
057	0	99	99	NPS
058	0	99	99	NPS
059	0	99	99	NPS
060	0	99	99	NPS
061	0	99	99	LOTS
062	41	41	41	GAIN
063	41	41	41	GAIN
064	0	99	99	XXXX
065	0	99	99	XXXX
127	99	99	99	ANG
128	99	99	99	ANG
131	99	99	99	ANG
132	99	99	99	ANG
135	99	99	99	ANG
136	99	99	99	ANG
137	99	99	99	ANG
138	99	99	99	ANG
139	99	99	99	ANG
140	99	99	99	ANG
141	99	99	99	ANG
142	99	99	99	ANG
143	99	99	99	ANG
144	99	99	99	ANG
145	99	99	99	ANG
146	99	99	99	ANG
147	99	99	99	ANG
148	99	99	99	ANG
149	99	99	99	ANG
150	99	99	99	ANG
151	99	99	99	ANG
152	99	99	99	ANG
153	99	99	99	ANG
155	99	99	99	ANG
156	99	99	99	ANG
157	0	99	99	ANG
158	0	99	99	LOTS
203	91	0	0	LETS
205	91	0	0	LETS
21G	99	99	99	LSPC

XTRNID (SPD)	XCTR Time Period			SAM Code
	≤ 7406	7407-8409	≥ 8410	
21L	2	0	0	LATR
21M	99	99	99	LSPC
21N	99	99	99	LOTS
21P	10	0	0	LATR
21Q	99	99	99	LOTS
21R	99	0	0	LSPC
21U	9	0	0	LATR
21W	9	0	0	LATR
211	93	0	0	LETS
213	30	0	0	LRET
214	93	93	93	LETS
215	9	0	0	LATR
217	9	0	0	LATR
220	8	0	0	LATR
221	4	0	0	LATR
225	9	0	0	LATR
227	3	0	0	LATR
228	9	0	0	LATR
229	9	0	0	LATR
230	30	30	30	LRET
231	30	0	0	LRET
240	99	99	99	LATR
241	99	99	99	LATR
242	8	0	0	LATR
246	8	0	0	LATR
247	8	0	0	LATR
248	8	0	0	LATR
250	99	99	99	LATR
251	99	99	99	LATR
253	8	0	0	LATR
254	99	99	99	LATR
256	99	99	99	LATR
257	8	0	0	LATR
258	8	0	0	LATR
260	8	0	0	LATR
261	8	0	0	LATR
262	8	0	0	LATR
263	8	0	0	LATR
264	8	0	0	LATR
265	8	0	0	LATR
269	6	0	0	LATR
270	6	0	0	LATR
271	6	0	0	LATR
272	6	0	0	LATR
273	7	0	0	LATR

XTRNID (SPD)	XCTR Time Period			SAM Code
	≤ 7406	7407-8409	≥ 8410	
274	7	0	0	LATR
277	7	0	0	LATR
278	7	0	0	LATR
279	9	0	0	LATR
28B	8	0	0	LATR
28E	8	0	0	LATR
28F	8	0	0	LATR
28G	8	0	0	LATR
28H	8	0	0	LATR
28I	8	0	0	LATR
280	8	0	0	LATR
281	8	0	0	LATR
282	8	0	0	LATR
283	8	0	0	LATR
284	8	0	0	LATR
288	9	0	0	LATR
289	8	0	0	LATR
290	9	0	0	LATR
291	8	0	0	LATR
292	9	0	0	LATR
301	9	0	0	LATR
314	9	0	0	LATR
317	9	0	0	LATR
318	9	0	0	LATR
319	9	0	0	LATR
320	92	0	0	LETS
361	8	0	0	LATR
362	8	0	0	LATR
363	99	99	99	LATR
370	99	99	99	LRET
375	8	0	0	LATR
377	9	0	0	LATR
384	8	0	0	LATR
386	8	0	0	LATR
388	8	0	0	LATR
41A	92	0	0	LETS
41B	99	99	99	LSPC
41E	9	0	0	LATR
41F	99	99	99	LSPC
41G	1	0	0	LATR
411	92	0	0	LETS
413	92	0	0	LETS
417	9	0	0	LATR
418	9	0	0	LATR
419	9	0	0	LATR

XTRNID (SPD)	XCTR Time Period			SAM Code
	≤ 7406	7407-8409	≥ 8410	
420	99	99	99	LSPC
421	91	0	0	LETS
422	91	0	0	LETS
423	91	0	0	LETS
424	91	0	0	LETS
448	99	99	99	LATR
46A	8	0	0	LATR
46B	8	0	0	LATR
46C	8	0	0	LATR
46D	8	0	0	LATR
466	8	0	0	LATR
469	99	99	99	LATR
474	5	5	5	LATR
475	0	0	0	XXXX
488	99	99	99	LATR
489	8	0	0	LATR
490	8	8	8	LATR
491	0	9	9	LATR
492	9	9	9	LATR
493	0	0	0	XXXX
494	0	9	9	LATR
495	0	9	9	LATR
496	0	9	9	LATR
497	0	9	9	LATR
498	0	9	9	LATR
703	1	0	0	LATR
710	81	0	0	LSPC
711	81	81	81	LSPC
712	81	81	81	LSPC
713	81	81	81	LSPC
715	93	93	93	LETS
716	93	93	93	LETS
717	81	81	81	LSPC
720	9	0	0	LATR
721	8	0	0	LATR
722	8	0	0	LATR
723	8	0	0	LATR
730	92	0	0	LETS
731	99	99	99	LSPC
732	99	99	99	LSPC
743	99	99	99	LSPC
753	99	99	99	LSPC
772	8	0	0	LATR
780	0	8	8	LATR

XTRNID (SPD)	XCTR Time Period			SAM Code
	≤ 7406	7407-8409	≥ 8410	
781	0	92	92	LETS
782	0	92	92	LETS
783	0	81	81	LSPC
900	11	11	11	REUP
901	15	15	15	REUP
902	12	12	12	REUP
903	13	13	13	REUP
904	16	16	16	REUP
905	14	14	14	REUP
906	21	21	21	EXT
907	21	21	21	EXT
909	22	22	22	REUP
911	23	23	23	EXT
912	16	16	16	REUP

REFERENCES

- Brauner, Marygail, Michael Murray, Warren Walker, and Elizabeth Davidson, *What's On the Enriched Airman Gain/Loss File*, The RAND Corporation, N-2610-AF, March 1989.
- Carter, Grace M., Michael Murray, R. Yilmaz Arguden, Marygail Brauner, Harvey Greenberg, Deborah Skoller, Allan Abrahamse, *Middle-Term Loss Prediction Models for the Air Force's Enlisted Force Management System: Specification and Estimation*, The RAND Corporation, R-3482-AF, December 1987.
- Walker, Warren, and Dan McGary, *Supplementary Historical Data Files for the Enlisted Force Management Project*, The RAND Corporation, N-2844-AF, March 1989.

RAND/N-2744-AF