(2)

AD 274911

BUSINESS PLANNING MODEL

PROGRAMMER'S GUIDE

DTIC
ELECTE
JAN 25 1994
S A

December 1993

Prepared for:

Advanced Systems Concepts Office
US Army Belvoir RD&E Center
Fort Belvoir, Virginia 22060

By:

BRTRC
8260 Willow Oaks Corporate Drive, Suite 800
Fairfax, Virginia 22031

Under Contract:
DAAK70-92-D-0003, Task Order 0001

Distribution Statement A:
Approved for public release; distribution is unlimited.

TABLE OF CONTENTS

Paragraph		Page
1	PROGRAMMER'S OVERVIEW	1
1.1	INTRODUCTION	1
1.2	PROGRAM CODE FILES	2
1.3	COMPILING AND LINKING IN CLIPPER S87	5
1.4	COMPILING AND LINKING IN CLIPPER 5.2	8
1.5	FILE TYPES AND IDENTIFICATION	10
2	BPM DATA FILES AND STRUCTURES	13
2.1	FACTORS FILE	13
2.2	REVENUE FILE	14
2.3	EXTERNALS FILE	15
2.4	INTERNALS FILE	16
2.5	PRODUCTS FILE	17
2.6	PERSONNEL FILE	17
2.7	WORKYEARS FILE	18
2.8	CAPITAL INVESTMENT FILE	18
2.9	CORE COMPETENCIES FILE	19
2.10	ORGANIZATION FILE	19
2.11	SERIES AND SKILLS FILE	20
2.12	SOURCES FILE	20
2.13	CURRENT TDA FILE	20
2.14	OFFICES FILE	21
2.15	REVENUE TYPES FILE	21
2.16	TOTALS FILE	22
2.17	PESSIMISM FACTORS FILE	22
2.18	REVENUE PESSIMISM FILE	23
2.19	TDA WIF EVALUATION YEAR FILE	24
2.20	TDA WIF ORGANIZATION FILE	25
2.21	BPM HELP FILE	27
3	BPM FUNCTIONS AND PROCEDURES	29
3.1	FUNCTION AND PROCEDURE MAPPING BY PROGRAM CODE FILE	29
3.2	FUNCTION AND PROCEDURE MAPPING BY NAME	60

4 CROSS REFERENCE TO CALLED FUNCTIONS/PROCEDURES 71

5 CROSS REFERENCE TO CALLING FUNCTION/PROCEDURES 121

DTIC QUALITY INSPECTED 5

Accesion For
NTIS CRA&I ☑
DTIC TAB ☐
Unannounced ☐
Justification
By
Distribution /
Availability Codes
Dist | Avail and / or Special
A-1

CHAPTER 1

PROGRAMMER'S OVERVIEW

1.1 INTRODUCTION

This Business Planning Model (BPM) Programmer's Guide is provided to assist future programmers in understanding and properly maintaining the BPM code. If you are interested in how the program appears to the user, you should refer to the BPM User's Guide. If you are interested in the exact workings of the program, you should carefully examine the Methodology Summary, Appendix C, of the User's Guide, and the corresponding documented source code. This Programmer's Guide will help you to identify where in the program source code to find each particular program part, and how those parts fit together into the single BPM program. It assumes that you understand both MS-DOS and the Clipper programming language.

The programming of the BPM was begun in February 1992, and is currently in Version 3.4. The program includes five principal modules:

(1) Data Entry Module - this provides the user the ability to collect, review, edit, and print the business planning data.

(2) Baseline Analysis Module - this provides on-screen analyses which combine information from the various data files, and access to specialized reports.

(3) What If Analysis Module - this provides the user with three "what if" analysis tools which have been specifically tailored to work with this data.

(4) Utilities Module - this provides a number of useful utilities that are occasionally helpful to make the program work smoothly.

(5) System Administrator Module - this provides the System Administrator access to the features he needs to configure the system and maintain the control lists.

The BPM was originally written in Clipper Summer 87 (S87) Version, and linked with the Blinker Version 2.0 dynamic linker. Clipper uses data files which match the structure of DBase III+ data files. Thus, the Clipper data files can be readily used in DBase (III+ and IV), and Foxbase/FoxPro. Because this format is so common, these data files can be readily imported into almost all commercial DBMS, Spreadsheet, Statistics, and Graphics software packages. In keeping with the task order provisions, it has been recompiled and tested in the latest version of Clipper (Version 5.2).

1.2 PROGRAM CODE FILES

The following eighteen (18) source code files (*.prg) comprise the BPM program. Each is listed along with a brief description of what types of routines it contains. Several files were not specifically created for the BPM but have been adapted from previous Clipper programs written by BRTRC. Chapter 3 provides a complete index of the procedures and functions contained in each of these files. Chapters 4 and 5 provide a cross reference of the procedures and functions indicating how the procedures and functions are interrelated.

The first seven files contain the main program and the entry points for the five modules described above.

BPM.PRG — MAIN PROGRAM
This program executes the Business Planning (BPM) model. The main menu is defined here and the five modules are accessed from here. It is kept as small as possible to maximize the performance of the linker.

BPM_EDIT.PRG — DATA ENTRY MODULE
This file contains all the data file edits for the user files. These edit screens are for the factors, revenue, major capital investments, major externals, support contracts, internal transfers, personnel, regular capital investments, and core capabilites.

BPM_ANAL.PRG — BASELINE ANALYSIS MODULE
This file is the entry point for the Baseline Analyses. It contains the analysis screens and the computation routines.

BPM_WHA1.PRG — WHAT IF MODULE, PART 1
This file, along with bpm_wha2.prg, contain the What If routines. There are three types of What Ifs. This file contains the Revenue Pessimism and the Optimize Personnel What Ifs.

BPM_WHA2.PRG — WHAT IF MODULE, PART 2
This file contains the Center TDA What If. It was separated from bpm_wha1.prg to keep the files to a manageable size.

BPM_ASST.PRG — ASSORTED FUNCTIONS MODULE
This file contains the Utility functions and other features for managing the BPM data.

BPM_SYST.PRG SYSTEM ADMINISTRATOR MODULE
This file contains the System Administrator edits and other control features. Only the System Administrator is allowed to execute these functions.

These modules are supported by seven other files which provide general capabilities.

BPM_GENL.PRG GENERAL FUNCTIONS
This file contains the data edit validation functions and the general purpose math and array handling routines which were developed specifically for the BPM.

BPM_OPEN.PRG OPENING ROUTINES
This file contains the opening routines for the BPM. These are called only on startup of the program, so they have been segregated here. The first half of the file includes the file handling and defining functions which are specifically tailored for the BPM. They link closely with the generic file handling functions defined in filehand.prg. The second half of this file contains the Function and Lab selection routines which are initialized at program startup.

BPM_REPT.PRG GENERAL REPORTS
This file contains the general reports and report drivers, which do not otherwise fit into the bpm_rep1.prg, bpm_rep2.prg, bpm_rep3.prg, and bpm_amcc.prg program files. You will find reports in here for the baseline Lab and Center analysis, core capabilities, and average factors.

BPM_REP1.PRG SPECIAL REPORTS MODULE
This file contains the special analysis reports for total expenses, balances, sources and uses, and Tech Support amounts. These were segregated from those in bpm_rept.prg because they are particularly lengthy.

BPM_REP2.PRG GENERIC LISTING REPORTS MODULE
This file contains the Generic Listing Report Writer and all the functions which support the report writer.

BPM_REP3.PRG GENERIC TOTALS REPORTS MODULE
This file contains the Generic Totals Report Writer. Much of this generic report mirrors the procedures used in the Generic Listing Report Writer.

BPM_AMCC.PRG AMC CHARTS MODULE
This file contains the reports for the AMC mandated business planning charts.

The final four program files contain the generic routines used in the BPM, which were developed by BRTRC for use in every Clipper application. They provide most of the user interface capabilities.

BROWSER.PRG BROWSER ROUTINES
This implements a generic data file browser. The programmer can control what fields are displayed and which can be edited. This is used only by the System Administrator.

FILEHAND.PRG FILE HANDLER ROUTINES
These are generic file handling routines used in BPM. This module implements the file handling (defining, opening, closing, indexing, deleting) features of the enhanced programmer interface. This is the non-network version.

TEXTVIEW.PRG TEXT FILE VIEWER ROUTINES
This implements a generic text file viewer. This is used to view the reports which are printed to text files.

UTILITY.PRG GENERAL PURPOSE ROUTINES
This module implements the general purpose utility routines for the enhanced programmer interface. This includes the menuing system, the help system, the windowing system, the report destination system, and many other common routines which are used throughout the BPM.

There are over 400 separate procedures and functions in the BPM program. You should refer to Chapter 3 for a complete listing of the functions contained in each of the program files, and an alphabetical listing of where to find each function. Most of these functions use other functions as building blocks. Therefore, you will need to use the cross references provided in Chapters 4 and 5 to map out the implications of changes in any one program element.

The BPM program files are all ASCII text files. If you want to edit them, you should be careful to use an ASCII text editor. Many good ones are commercially available (These were written using the Norton Editor). You might also consider the 'edit.exe' program included with MS-DOS (Version 5.0 and above).

1.3 COMPILING AND LINKING IN CLIPPER S87

The BPM is compiled as a finished product using the procedures described below. This procedure yields the minimum sized executable program which is important because of run-time memory considerations. Three files (m_bpm.bat, m_bpm.clp, and m_bpm2.clp) contain the compilation instructions, and the fourth (m_bpm.lnk) the linking instructions. As you will see, these files presume that you have set your computer up with the following directory names and contents:

\clipper	Contains the Clipper compiler and libraries, and the object files handles, aa_cdosv, aa_set, aa_files, sysinfo, and tt_clipa. These object files are discussed further in paragraph 1.5.
\clipper\bpm	Contains the program files (*.prg) and these compilation and linking files.
\clipper\blinker	Contains the Blinker linker program.

To compile and link the BPM, all you have to do is execute the batch file m_bpm.bat shown below.

File m_bpm.bat:

```
set lib=\clipper\
\clipper\clipper bpm -m -q -l
\clipper\clipper @m_bpm -m -q -l
\clipper\clipper @m_bpm2 -m -q -l
\clipper\blinker\blinker @m_bpm.lnk
```

This batch file:

(1) Compiles the main program bpm.prg to produce the object file bpm.obj. The parameters -m and -q keep the compiler quiet, and the parameter -l tells Clipper not to include debugger line numbers (which should be included in development and testing runs but adds significantly to the size of the program). The bpm.prg file is compiled separately to keep it as small as possible. This is to take advantage of the dynamic loading capabilities of the Blinker linker.

(2) Compiles together the program files listed in m_bpm.clp (listed below) to produce the object file m_bpm.obj. The Clipper compiler knows when the @filename

syntax is used, that the filename refered to is a .clp file. All this file contains is a list of the .prg files to be compiled together. It is helpful to compile the files together into a single object file because it makes the resulting program smaller. There are two .clp files referenced in this batch file because there is a DOS limit of 256 procedure/ function references in an object file. The BPM has over 400 procedures and functions, hence the need for two files.

(3) Compiles together the program files listed in m_bpm2.clp (listed below) to produce the object file m_bpm2.obj.

(4) Executes the Blinker linker with the linking script contained in m_bpm.lnk (listed below) to produce the executable bpm.exe.

File m_bpm.clp: This file contains a list of the .prg files which are compiled together to produce m_bpm.obj.

```
bpm_edit.prg
bpm_asst.prg
bpm_open.prg
bpm_anal.prg
bpm_syst.prg
bpm_genl.prg
textview.prg
utility.prg
filehand.prg
```

File m_bpm2.clp: This file contains a list of the .prg files which are compiled together to produce m_bpm2.obj.

```
bpm_wha1.prg
bpm_wha2.prg
bpm_rept.prg
bpm_rep1.prg
bpm_rep2.prg
bpm_rep3.prg
bpm_amcc.prg
browser.prg
```

It does not matter in which order the program files are listed in these two .clp files. It also does not matter what program files are included in each of the two .clp files. It is only important that all the program files are included and that neither of the two object files which are created has more than 256 procedures or functions.

The final step in the process of building a new executable file is the linking together of the object files. Blinker 2.0 is the linker used because of its exceptional dynamic loading capabilities.

File m_bpm.lnk: This file contains the linking script used by Blinker to link the bpm object files into an executable. You should refer to the Blinker 2.0 manual for definitions of the script structure and purpose of the special Blinker statements.

```
file bpm.obj
file \clipper\handles.obj
file \clipper\aa_cdosv.obj
file \clipper\aa_set.obj
file \clipper\aa_files.obj
file \clipper\sysinfo.obj
file \clipper\tt_clipa.obj
output bpm.exe
BLINKER EXECUTABLE CLIPPER F55;V20;R16;E0
BLINKER PROCEDURE DEPTH 40
BLINKER OVERLAY OPSIZE 80
BLINKER INCREMENTAL OFF
MAP=bpm.map S
beginarea
  file m_bpm.obj
  file m_bpm2.obj
  allocate \clipper\extend
endarea
lib \clipper\clipper
```

You should feel free to modify these four compilation and linking files to match the directory structures in your computer. Remember to use an ASCII text editor when you edit these files.

1.4 COMPILING AND LINKING IN CLIPPER 5.2

Compiling and linking in Clipper 5.2 requires a different set of compilation and linking files. The scripts shown below assume the default installation of Clipper 5.2, and the placement of the program files and the two files below into the same subdirectory. The default installation creates a number of environment variables and places the Clipper program directories into the DOS path.

These scripts also assume the use of the RTlink linker which is provided with Clipper. At present, RTlink is used with Clipper 5.2 in place of Blinker 2.0 because it has slightly better dynamic loading capabilities with Clipper 5.2 object files. Since the BPM is a large program, memory management is the critical consideration in linker selection. Future releases of Blinker may change this assessment.

The BPM program is compiled and linked by typing 'rmake bpm' from the DOS command line when the current directory is your program directory.

File bpm.rmk: This file is a script for the 'rmake' program provided with Clipper 5.2. Rmake, like all 'make' programs, checks the currency of the specified object files with respect to the corresponding program files and recompiles them only if they are not current. Then, it executes the linker to rebuild the executable if any of the object files are newer.

```
//
//    BPM.rmk
//
//    NOTE:
//       In order to build BPM with debugger (CLD) support you must
//       define the macro name DEBUG.  This can be accomplished as
//       follows:
//
//       RMAKE BPM /dDEBUG
//

// Determine if DEBUGging is enabled
#ifdef DEBUG
   CompOptions := /b /m
#else
   CompOptions := /m
#end

.prg.obj:
    clipper $< $(CompOptions)

bpm.obj:        bpm.prg
bpm_edit.obj:   bpm_edit.prg
bpm_asst.obj:   bpm_asst.prg
bpm_open.obj:   bpm_open.prg
bpm_anal.obj:   bpm_anal.prg
```

```
bpm_syst.obj:   bpm_syst.prg
bpm_genl.obj:   bpm_genl.prg
bpm_wha1.obj:   bpm_wha1.prg
bpm_wha2.obj:   bpm_wha2.prg
bpm_rept.obj:   bpm_rept.prg
bpm.rep1.obj:   bpm_rep1.prg
bpm_rep2.obj:   bpm_rep2.prg
bpm_rep3.obj:   bpm_rep3.prg
bpm_amcc.obj:   bpm_amcc.prg
browser.obj:    browser.prg
textview.obj:   textview.prg
utility.obj:    utility.prg
filehand.obj:   filehand.prg

bpm.exe:      bpm.obj  bpm_edit.obj  bpm_asst.obj  bpm_open.obj  bpm_anal.obj  \
              bpm_syst.obj  bpm_genl.obj  bpm_wha1.obj  bpm_wha2.obj  \
              bpm_rept.obj  bpm.rep1.obj  bpm_rep2.obj  bpm_rep3.obj  \
              bpm_amcc.obj  browser.obj  textview.obj  utility.obj  filehand.obj
   rtlink bpm.lnk
```

The linking script for RTlink is simply a list of the object files to be linked (without the .obj extension), followed by the two needed libraries.

File bpm.lnk

```
file bpm
file bpm_edit
file bpm_asst
file bpm_open
file bpm_anal
file bpm_syst
file bpm_genl
file bpm_wha1
file bpm_wha2
file bpm_rept
file bpm.rep1
file bpm_rep2
file bpm_rep3
file bpm_amcc
file browser
file textview
file utility
file filehand
file \clipper5\obj\tt_clipa
file \clipper5\handles
file \clipper5\aa_cdosv
file \clipper5\aa_files

lib  clipper
lib  extend
```

Again you may need to modify these compilation and linking scripts to match the directory structures in your computer. Remember to use an ASCII text editor when you edit these files.

1.5 FILE TYPES AND IDENTIFICATION

This section provides a brief overview of the types of files that the BPM uses. The programmer needs to be aware of each of them when maintaining and modifying the program.

The following file types exist in the BPM user's directory:

bpm.exe	This is the program executable file.
*.cfg	These are BPM configuration files (bpm.cfg and opt.cfg). They tell the program the current state of certain critical variables (like the current Base Year). They should be delivered with the program executable (bpm.exe).
*.dbf	These are the BPM data files. They are standard DBase III+ format files. They should be retained and backed up regularly.
.nt	These are the BPM index files. These are Clipper generated index files, which are not compatible with DBase or Foxbase index files. They can be deleted if needed, since they are rebuilt each time the program is run (if they are missing or outdated).
*.mem	These are the BPM memory variable files. They are used to store the report definitions for the user-defined listing and totals reports. They can be deleted since they can always be rebuilt from within the program.
*.txt	These are BPM reports which have been printed to files.

The following file types exist in the BPM programmer's directory.

*.prg	These are the BPM program source code files.
*.obj	These are the compiled object files. They are rebuilt each time the program is compiled.
m_bpm.*	These are the BPM compilation and linking files (Clipper S87).
bpm.*	These are the BPM compilation and linking files (Clipper 5.2).

The following object files exist in the Clipper directory. They have been provided along with the 18 BPM source code files.

aa_files.obj	Determines the number of available file handles.
aa_cdosv.obj	Determines the DOS version. It is only needed if the version is less than 3.1, to create the additional file handles. Old versions of DOS limited the number of open file handles to 20.
handles.obj	Creates additional file handles in old versions of DOS.
aa_set.obj	Allows the detection of the state (on or off) of the cursor. It is replaced by a built in function in Clipper 5.2. It is used in the help system to return the cursor to the original state.
sysinfo.obj	Needed by aa_set.obj.
tt_clipa.obj	Provides for shaded windows.

CHAPTER 2

BPM DATA FILES AND STRUCTURES

This chapter documents the twenty one (21) data structures used by the Business Planning Model (BPM). They are listed here, along with their principal indexes, to help you maintain the data files. If you decide to modify these structures, also note the function 'buildfile' in bpm_open.prg and make sure that it also reflects these modifications.

Note that the field 'foe' represents the organization level "Function" in the model. Also note that the field 'year0' is the base year. It represents the year corresponding to the first element of the numerical arrays (e.g., fact_*, rev_*, etc.). For example, if the base year is 1993, then rev_0 is the revenue for 1993, rev_1 is the revenue for 1994, and so on. The 'year0' should be the same for all files and all records in the BPM database.

2.1 FACTORS FILE

The Factors file (FACTORS.DBF) contains the estimating factors used by each Lab/Function to estimate the amounts to charge or 'tax' for specific purposes. Each of the factors represents a charge on Civilian Pay. The file contains one factor per record, and indicates which lab it belongs to. Note that there are three factors in this file which only the System Administrator is supposed to modify. These are the Inflation factors, the G&A factors and the Leave and Benefits factors which apply to the Center as a whole.

```
Structure for database: C:\BPM\FACTORS.DBF
Field  Field Name  Type        Width    Dec
    1  FACTOR      Character       3        <- controlled
    2  LAB         Character       4        <- controlled
    3  FOE         Character       6        <- controlled
    4  YEAR0       Numeric         4
    5  FACT_0      Numeric         6      4
    6  FACT_1      Numeric         6      4
    7  FACT_2      Numeric         6      4
    8  FACT_3      Numeric         6      4
    9  FACT_4      Numeric         6      4
   10  FACT_5      Numeric         6      4
   11  FACT_6      Numeric         6      4
   12  FACT_7      Numeric         6      4
** Total **                      66

Indexes:
ntl:  factor+lab+foe
```

2.2 REVENUE FILE

The Revenue file (REVENUE.DBF) contains the estimates of the anticipated revenues (by source), and associated estimates of the requirements for major contracts, support contracts, internal transfers, and specific capital investments required to support that revenue. The program computes the amount available and the amount of G&A from this data and using estimating factors. The data contained here on major contracts, support contracts, and capital investments are totalled from corresponding records in the Externals file. The data contained here on internal tranfers are totalled from corresponding records in the Internals file.

Structure for database: C:\BPM\REVENUE.DBF

Field	Field Name	Type	Width	Dec	
1	LAB	Character	4		<- controlled
2	FOE	Character	6		<- controlled
3	OFFICE	Character	4		<- controlled
4	GROUP	Character	6		<- controlled
5	SOURCE	Character	10		<- controlled
6	REVID	Numeric	5		<- internal key
7	APPN	Character	5		<- controlled
8	CATEGORY	Character	4		<- controlled
9	MDEP	Character	4		
10	PROG_ELEM	Character	6		
11	PROJ_NO	Character	4		
12	TASK	Character	4		
13	PMS	Character	4		
14	YEAR0	Numeric	4		
15	REV_0	Numeric	6		<- revenue ($K)
16	REV_1	Numeric	6		
17	REV_2	Numeric	6		
18	REV_3	Numeric	6		
19	REV_4	Numeric	6		
20	REV_5	Numeric	6		
21	REV_6	Numeric	6		
22	REV_7	Numeric	6		
23	CONT_0	Numeric	6		<- major cont ($K)
24	CONT_1	Numeric	6		
25	CONT_2	Numeric	6		
26	CONT_3	Numeric	6		
27	CONT_4	Numeric	6		
28	CONT_5	Numeric	6		
29	CONT_6	Numeric	6		
30	CONT_7	Numeric	6		
31	SUPT_0	Numeric	6		<- supt cont ($K)
32	SUPT_1	Numeric	6		
33	SUPT_2	Numeric	6		
34	SUPT_3	Numeric	6		
35	SUPT_4	Numeric	6		
36	SUPT_5	Numeric	6		
37	SUPT_6	Numeric	6		
38	SUPT_7	Numeric	6		
39	CAP_0	Numeric	6		<- cap invest ($K)
40	CAP_1	Numeric	6		
41	CAP_2	Numeric	6		

```
   42  CAP_3        Numeric        6
   43  CAP_4        Numeric        6
   44  CAP_5        Numeric        6
   45  CAP_6        Numeric        6
   46  CAP_7        Numeric        6
   47  INTL_0       Numeric        6       <- intl transfers ($K)
   48  INTL_1       Numeric        6
   49  INTL_2       Numeric        6
   50  INTL_3       Numeric        6
   51  INTL_4       Numeric        6
   52  INTL_5       Numeric        6
   53  INTL_6       Numeric        6
   54  INTL_7       Numeric        6
   55  WKYR_0       Numeric        7      2 <- direct wkyrs
   56  WKYR_1       Numeric        7      2
   57  WKYR_2       Numeric        7      2
   58  WKYR_3       Numeric        7      2
   59  WKYR_4       Numeric        7      2
   60  WKYR_5       Numeric        7      2
   61  WKYR_6       Numeric        7      2
   62  WKYR_7       Numeric        7      2
   63  CARRYOVER    Numeric        6       <- carry in ($K)
   64  CARRY_0      Numeric        6       <- carry out ($K)
   65  CARRY_1      Numeric        6
   66  CARRY_2      Numeric        6
   67  CARRY_3      Numeric        6
   68  CARRY_4      Numeric        6
   69  CARRY_5      Numeric        6
   70  CARRY_6      Numeric        6
   71  CARRY_7      Numeric        6
   72  BASIS        Character     60
** Total **                      481

Indexes:
nt1:  lab+foe+str(revid)
nt2:  revid
```

2.3 EXTERNALS FILE

The Externals file (EXTERNALS.DBF) contains the specific externals associated with revenue records. They are linked back to the revenue record by the lab+foe+revid. All three kinds of external actions (major contracts, support contracts, and capital investments) are accommodated by this structure and distinguished by the EXTTYPE field.

```
Structure for database: C:\BPM\EXTERNALS.DBF
Field  Field Name  Type        Width    Dec
    1  LAB         Character       4       <- controlled
    2  FOE         Character       6       <- controlled
    3  REVID       Numeric         5       <- internal key
    4  EXTTYPE     Character       3       <- controlled
    5  PRIORITY    Numeric         4
    6  GROUP       Character       6       <- controlled
    7  EXPTO       Character      10       <- controlled
    8  ASS_TYPE    Character      10       <- controlled
```

```
    9  ASSET       Character     30
   10  YEAR0       Numeric        4
   11  CONT_0      Numeric        6        <- external amt ($K)
   12  CONT_1      Numeric        6
   13  CONT_2      Numeric        6
   14  CONT_3      Numeric        6
   15  CONT_4      Numeric        6
   16  CONT_5      Numeric        6
   17  CONT_6      Numeric        6
   18  CONT_7      Numeric        6
   19  BASIS       Character     60
** Total **                     191

Indexes:
ntl:  lab+foe+str(revid)+exttype+str(priority)
```

2.4 INTERNALS FILE

The Internal Transfers file (INTERNALS.DBF) contains the specific internal transfers associated with revenue records. They are linked back to the revenue record by the lab+foe+revid. Note that this file contains only those transfers out of an organization. The anticipated incoming internal transfers are reported as revenues by the receiving organization.

```
Structure for database: C:\BPM\INTERNALS.DBF

Field  Field Name  Type       Width    Dec
    1  LAB         Character      4       <- controlled
    2  FOE         Character      6       <- controlled
    3  REVID       Numeric        5       <- internal key
    4  TOLAB       Character      4       <- controlled
    5  TOFOE       Character      6       <- controlled
    6  OFFICE      Character      4       <- controlled
    7  YEAR0       Numeric        4
    8  INTL_0      Numeric        6       <- intl transfers ($K)
    9  INTL_1      Numeric        6
   10  INTL_2      Numeric        6
   11  INTL_3      Numeric        6
   12  INTL_4      Numeric        6
   13  INTL_5      Numeric        6
   14  INTL_6      Numeric        6
   15  INTL_7      Numeric        6
   16  BASIS       Character     60
** Total **                     142

Indexes:
ntl:  lab+foe+str(revid)
nt2:  tolab+tofoe
```

2.5 PRODUCTS FILE

The Products file (PRODUCTS.DBF) contains the specific work products and projects associated with revenue records. They are linked back to the revenue record by the lab+foe+revid.

```
Structure for database: C:\BPM\PRODUCTS.DBF
Field  Field Name  Type        Width    Dec
    1  LAB         Character       4       <- controlled
    2  FOE         Character       6       <- controlled
    3  REVID       Numeric         5       <- internal key
    4  PRODUCT     Character      20
    5  PROJECT     Character      40
    6  DUEDATE     Date            8
** Total **                      84

Indexes:
ntl:  lab+foe+str(revid)
```

2.6 PERSONNEL FILE

The Personnel file (PERSONEL.DBF) contains the descriptive data for groups of one or more employees who share a common organization, series, and pay source. These records are linked to the specific workyears in the Workyears file by lab+foe+persid.

```
Structure for database: C:\BPM\PERSONEL.DBF
Field  Field Name  Type        Width    Dec
    1  LAB         Character       4       <- controlled
    2  FOE         Character       6       <- controlled
    3  OFFICE      Character       4       <- controlled
    4  PERSID      Numeric         5       <- internal key
    5  PERS_TYPE   Character       4       <- controlled
    6  PAY_SOURCE  Character       1       <- controlled
    7  SERIES      Character       5       <- controlled
    8  SKILL       Character       1
    9  TITLE       Character      20
   10  POSITION    Character       1       <- controlled
   11  AVGBASE     Numeric         6
   12  AVGWKYR     Numeric         6
   13  BASIS       Character      60
** Total **                     124

Indexes:
ntl:  lab_foe+str(persid)
nt2:  persid
```

2.7 WORKYEARS FILE

The Workyears file (WKYRS.DBF) contains the estimates of workyears by type (full time, overtime, part time, or temporary) for the associated personnel record. They are linked to the correct personnel record by lab+foe+persid.

```
Structure for database: C:\BPM\WKYRS.DBF
Field  Field Name  Type        Width    Dec
    1  LAB         Character       4        <- controlled
    2  FOE         Character       6        <- controlled
    3  PERSID      Numeric         5        <- internal key
    4  WKYTYPE     Character       2        <- controlled
    5  YEAR0       Numeric         4
    6  WKYR_0      Numeric         7       2 <- workyears
    7  WKYR_1      Numeric         7       2
    8  WKYR_2      Numeric         7       2
    9  WKYR_3      Numeric         7       2
   10  WKYR_4      Numeric         7       2
   11  WKYR_5      Numeric         7       2
   12  WKYR_6      Numeric         7       2
   13  WKYR_7      Numeric         7       2
** Total **                      78

Indexes:
ntl:  lab+foe+str(persid)+wkytype
```

2.8 CAPITAL INVESTMENT FILE

The Capital Investment file (CAPITAL.DBF) contains only those general capital investments which are not directly charged to specific revenues.

```
Structure for database: C:\BPM\CAPITAL.DBF
Field  Field Name  Type        Width    Dec
    1  LAB         Character       4        <- controlled
    2  FOE         Character       6        <- controlled
    3  OFFICE      Character       4        <- controlled
    4  PRIORITY    Numeric         4
    5  ASS_TYPE    Character      10        <- controlled
    6  ASSET       Character      30
    7  CURVAL      Numeric         6
    8  DEPREC      Numeric         2
    9  YEAR0       Numeric         4
   10  REPL_0      Numeric         6        <- replacement ($K)
   11  REPL_1      Numeric         6
   12  REPL_2      Numeric         6
   13  REPL_3      Numeric         6
   14  REPL_4      Numeric         6
   15  REPL_5      Numeric         6
   16  REPL_6      Numeric         6
   17  REPL_7      Numeric         6
   18  NEWI_0      Numeric         6        <- new invest ($K)
```

```
   19  NEWI_1      Numeric        6
   20  NEWI_2      Numeric        6
   21  NEWI_3      Numeric        6
   22  NEWI_4      Numeric        6
   23  NEWI_5      Numeric        6
   24  NEWI_6      Numeric        6
   25  NEWI_7      Numeric        6
   26  BASIS       Character     60
** Total **                      227

Indexes:
ntl:  lab+foe+str(priority)
```

2.9 CORE COMPETENCIES FILE

The Core Competencies file (CORECOMP.DBF) contains estimates of the percentages of the workyears and revenues that are directed toward each core competency (numbered 1 through 22). There are four percentages (CIV, MIL, OGA, and CON) reported for each combination of lab+foe+appn in the revenue file.

```
Structure for database: C:\BPM\CORECOMP.DBF
Field  Field Name  Type        Width    Dec
    1  LAB         Character       4        <- controlled
    2  FOE         Character       6        <- controlled
    3  APPN        Character       5        <- controlled
    4  CORECOMP    Character       2        <- ' 1' to '22'
    5  CIV_PCT     Numeric         3
    6  MIL_PCT     Numeric         3
    7  OGA_PCT     Numeric         3
    8  CON_PCT     Numeric         3
** Total **                       30

Indexes:
ntl:  lab+foe+appn+corecomp
```

2.10 ORGANIZATION FILE

The Organization file (ORGS.DBF) contains the definitions of the Labs and Functions used in the BPM. Only Labs/Functions included in this file can have data entered for them. This file is maintained by the System Administrator.

```
Structure for database: C:\BPM\ORGS.DBF
Field  Field Name  Type        Width    Dec
    1  LAB         Character       4
    2  LONGLAB     Character      30
    3  FOE         Character       6
    4  LONGFOE     Character      30
    5  TYPEFOE     Character       1        <- controlled
    6  ONTDA       Character       1        <- Y/N
    7  ENTER_REV   Character       1        <- Y/N
```

```
    8  AVGBASE     Numeric        6        <- computed
    9  AVGWKYR     Numeric        6        <- computed
** Total **                      86

Indexes:
nt1:  lab+foe
nt2:  lab (unique)
nt3:  foe (unique)
```

2.11 SERIES AND SKILLS FILE

The Series and Skills file (SKILLS.DBF) contains the definitions of the series to which personnel records can be associated. The corresponding skills and job titles are automatically posted to the personnel record along with the selected series. This file is maintained by the System Administrator.

```
Structure for database: C:\BPM\SKILLS.DBF
Field  Field Name  Type       Width    Dec
    1  SERIES      Character      5
    2  SKILL       Character      1        <- controlled
    3  TITLE       Character     20
** Total **                      27

Indexes:
nt1:  series
```

2.12 SOURCES FILE

The Sources File (SOURCES.DBF) contains the definitions of the revenue groups and sources (i.e., sources of customer funding) to which revenue records can be associated. This file is maintained by the System Administrator.

```
Structure for database: C:\BPM\SOURCES.DBF
Field  Field Name  Type       Width    Dec
    1  GROUP       Character      6
    2  SOURCE      Character     10
** Total **                      17

Indexes:
nt1:  group+source
```

2.13 CURRENT TDA FILE

The Current TDA file (CURR_TDA.DBF) contains the current TDA spaces by Lab and Functions. It is used in the TDA What If, so the FOE of 'ALL' should be used for the labs CED, CSD, and LED. This file is maintained by the System Administrator.

```
Structure for database: C:\BPM\CURR_TDA.DBF
Field  Field Name  Type        Width    Dec
    1  LAB         Character       4
    2  FOE         Character       6
    3  CTDA_DIR    Numeric         4
    4  CTDA_SEC    Numeric         4
    5  CTDA_IND    Numeric         4
    6  CTDA_MGR    Numeric         4
    7  CTDA_TOT    Numeric         4
** Total **                       31

Indexes:
ntl:  lab+foe
```

2.14 OFFICES FILE

The Offices file (OFFICES.DBF) contains the definitions of the Offices included in each Lab. The Office is an optional entry on the revenue, personnel, and capital investment screens. Therefore, this file only needs to be filled for those Labs which want to use the office field. This file is maintained by the System Administrator.

```
Structure for database: C:\BPM\OFFICES.DBF
Field  Field Name  Type        Width    Dec
    1  LAB         Character       4
    2  OFFICE      Character       4
    3  OFF_TITLE   Character      30
** Total **                       39

Indexes:
ntl:  lab+office
```

2.15 REVENUE TYPES FILE

The Revenue Types file (REVTYPES.DBF) contains the definitions of the allowable combinations of revenue identification fields. This control file is not yet implemented in the revenue data entry section, so it is not necessary to fill this file with data. The idea of this file is that the revenues (particularly the direct revenues) could be controlled more precisely to ensure greater data quality. This file is maintained by the System Administrator.

```
Structure for database: C:\BPM\REVTYPES.DBF
Field  Field Name  Type        Width    Dec
    1  APPN        Character       5
    2  CATEGORY    Character       4
    3  MDEP        Character       4
    4  PROG_ELEM   Character       6
    5  PROJ_NO     Character       4
    6  TASK        Character       4
** Total **                       28
```

```
Indexes:
ntl:  appn+category+prog_elem+proj_no+task
```

2.16 TOTALS FILE

The Totals file (TOTALS.DBF) contains computed totals from the Lab and Center baseline analysis recalculation. Funding data in the file is represented in whole dollars. Workyear data is represented in 1/100ths workyears (i.e. 1 whole workyear is represented as 100). The field POOL defines which total is contained on this specific record, and the LAB and FOE fields indicate for which organization these totals apply. The specific LAB and FOE designation of 'ALL' is used to indicate roll-ups of Labs and Functions. All the data contained in this file is computed. The user enters no data directly into this file.

```
Structure for database: C:\BPM\TOTALS.DBF
Field  Field Name  Type        Width    Dec
    1  LAB         Character       4
    2  FOE         Character       6
    3  POOL        Character       3
    4  YEAR0       Numeric         4
    5  PAMT_0      Numeric        10
    6  PAMT_1      Numeric        10
    7  PAMT_2      Numeric        10
    8  PAMT_3      Numeric        10
    9  PAMT_4      Numeric        10
   10  PAMT_5      Numeric        10
   11  PAMT_6      Numeric        10
   12  PAMT_7      Numeric        10
** Total **                       98

Indexes:
ntl:  lab+foe+pool
nt2:  lab+pool
```

2.17 PESSIMISM FACTORS FILE

The Pessimism Factors file (PESSISM.DBF) contains the factors defined by the user in the Revenue Pessimism What If. The first 11 fields are optional and tell the program to which revenue records these factors apply.

```
Structure for database: C:\BPM\PESSISM.DBF
Field  Field Name  Type        Width    Dec
    1  LAB         Character       4
    2  FOE         Character       6
    3  OFFICE      Character       4
    4  GROUP       Character       6
    5  SOURCE      Character      10
    6  APPN        Character       5
    7  CATEGORY    Character       4
    8  MDEP        Character       4
```

```
    9  PROG_ELEM   Character      6
   10  PROJ_NO     Character      4
   11  TASK        Character      4
   12  YEAR0       Numeric        4
   13  FACT_0      Numeric        6       4 <- pessimism factor
   14  FACT_1      Numeric        6       4
   15  FACT_2      Numeric        6       4
   16  FACT_3      Numeric        6       4
   17  FACT_4      Numeric        6       4
   18  FACT_5      Numeric        6       4
   19  FACT_6      Numeric        6       4
   20  FACT_7      Numeric        6       4
** Total **                      110

Indexes:
ntl:  lab
```

2.18 REVENUE PESSIMISM FILE

The Revenue Pessimism file (REVPESS.DBF) is the computed product of the pessimism factors. This file matches the structure of the Revenue file with the exception of the BASIS field. The user enters no data into this, file.

```
Structure for database: C:\BPM\REVPESS.DBF
Field  Field Name  Type        Width    Dec
    1  LAB         Character       4
    2  FOE         Character       6
    3  OFFICE      Character       4
    4  GROUP       Character       6
    5  SOURCE      Character      10
    6  REVID       Numeric         5
    7  APPN        Character       5
    8  CATEGORY    Character       4
    9  MDEP        Character       4
   10  PROG_ELEM   Character       6
   11  PROJ_NO     Character       4
   12  TASK        Character       4
   13  PMS         Character       4
   14  YEAR0       Numeric         4
   15  REV_0       Numeric         6
   16  REV_1       Numeric         6
   17  REV_2       Numeric         6
   18  REV_3       Numeric         6
   19  REV_4       Numeric         6
   20  REV_5       Numeric         6
   21  REV_6       Numeric         6
   22  REV_7       Numeric         6
   23  CONT_0      Numeric         6
   24  CONT_1      Numeric         6
   25  CONT_2      Numeric         6
   26  CONT_3      Numeric         6
   27  CONT_4      Numeric         6
   28  CONT_5      Numeric         6
   29  CONT_6      Numeric         6
```

30	CONT_7	Numeric	6	
31	SUPT_0	Numeric	6	
32	SUPT_1	Numeric	6	
33	SUPT_2	Numeric	6	
34	SUPT_3	Numeric	6	
35	SUPT_4	Numeric	6	
36	SUPT_5	Numeric	6	
37	SUPT_6	Numeric	6	
38	SUPT_7	Numeric	6	
39	CAP_0	Numeric	6	
40	CAP_1	Numeric	6	
41	CAP_2	Numeric	6	
42	CAP_3	Numeric	6	
43	CAP_4	Numeric	6	
44	CAP_5	Numeric	6	
45	CAP_6	Numeric	6	
46	CAP_7	Numeric	6	
47	INTL_0	Numeric	6	
48	INTL_1	Numeric	6	
49	INTL_2	Numeric	6	
50	INTL_3	Numeric	6	
51	INTL_4	Numeric	6	
52	INTL_5	Numeric	6	
53	INTL_6	Numeric	6	
54	INTL_7	Numeric	6	
55	WKYR_0	Numeric	7	2
56	WKYR_1	Numeric	7	2
57	WKYR_2	Numeric	7	2
58	WKYR_3	Numeric	7	2
59	WKYR_4	Numeric	7	2
60	WKYR_5	Numeric	7	2
61	WKYR_6	Numeric	7	2
62	WKYR_7	Numeric	7	2
63	CARRYOVER	Numeric	6	
64	CARRY_0	Numeric	6	
65	CARRY_1	Numeric	6	
66	CARRY_2	Numeric	6	
67	CARRY_3	Numeric	6	
68	CARRY_4	Numeric	6	
69	CARRY_5	Numeric	6	
70	CARRY_6	Numeric	6	
71	CARRY_7	Numeric	6	
** Total **			421	

Indexes: none

2.19 TDA WIF EVALUATION YEAR FILE

The Evaluation Year file (TDA_EVAL) contains the evaluation year-level data for the Center TDA and Optimize Personnel What Ifs. One record is automatically created for each of the 8 years in the planning horizon. Most of the values are computed or are extracted from the baseline data during the totals recalculation process.

```
Structure for database: C:\BPM\TDA_EVAL.DBF
Field  Field Name  Type        Width    Dec
    1  FOR_YEAR    Numeric         4
    2  AVG_BILLMY  Numeric         6
    3  AVG_TOTMY   Numeric         6
    4  GA_RATE     Numeric         5      3
    5  GA_REV      Numeric         9
    6  GA_COSTS    Numeric         9
    7  GA_TEC_SAL  Numeric         9
    8  GA_TGT      Numeric         5      3 <- target G&A
    9  TGT_TDA     Numeric         4
   10  WTDA_TOT    Numeric         4
   11  WKY_TOT     Numeric         6      1
   12  WKY_RED     Numeric         6      1
   13  WKY_CONT    Numeric         6      1
   14  WKY_BILL    Numeric         6      1
   15  WKY_GOVT    Numeric         6      1
   16  EXP_GOVT    Numeric         9
   17  EXP_CONT    Numeric         9
   18  EXP_EXT     Numeric         9
   19  EXP_REDEXT  Numeric         9
   20  EXP_ADJEXT  Numeric         9
   21  EXP_GOVT_N  Numeric         9
   22  TOT_DIR     Numeric         9
   23  TOT_CUST    Numeric         9
   24  TOT_REV     Numeric         9
   25  TOT_OMAD    Numeric         9
   26  TOT_EXP     Numeric         9
   27  TOT_GA      Numeric         9
   28  TOT_NET     Numeric        10
   29  TS_COSTS    Numeric         9
   30  LINE_COSTS  Numeric         9
   31  RAT_SDR_L   Numeric         5      3 <- target ratios
   32  RAT_IDR_L   Numeric         5      3
   33  RAT_MDR_L   Numeric         5      3
   34  RAT_SDR_S   Numeric         5      3
   35  RAT_MDR_S   Numeric         5      3
   36  DPCT_SEC    Numeric         4      2 <- direct percents
   37  DPCT_MGR    Numeric         4      2
   38  DPCT_IND    Numeric         4      2
   39  DPCT_PAE    Numeric         4      2
   40  CFACT_DIR   Numeric         5      3 <- confidence factors
   41  CFACT_CUS   Numeric         5      3
   42  XCHNG       Numeric         5      3 <- govt/contr exchange
** Total **                      284

Indexes:
ntl:  for_year
```

2.20 TDA WIF ORGANIZATION FILE

The TDA What If Organization file (TDA_WIF.DBF) contains the organization-level data for the Center TDA and Optimize Personnel What Ifs. One record is automatically created for each organization in each of the 8 years in the planning horizon. Most of the values are

computed or are extracted from the baseline data during the totals recalculation process.

```
Structure for database: C:\BPM\TDA_WIF.DBF
Field  Field Name   Type        Width    Dec
    1  FOR_YEAR     Numeric         4
    2  LAB          Character       4
    3  FOE          Character       6
    4  TYPEFOE      Character       1
    5  WTDA_DIR     Numeric         4        <- what if TDA
    6  WTDA_SEC     Numeric         4
    7  WTDA_IND     Numeric         4
    8  WTDA_MGR     Numeric         4
    9  WTDA_TOT     Numeric         4
   10  WTDA_OK      Character       1
   11  TOT_DIR      Numeric         9
   12  TOT_CUST     Numeric         9
   13  TOT_REV      Numeric         9
   14  TOT_OMAD     Numeric         9
   15  TOT_EXP      Numeric         9
   16  TOT_GA       Numeric         9
   17  TOT_NET      Numeric        10
   18  SAL_DIR      Numeric         6
   19  SAL_SEC      Numeric         6
   20  SAL_IND      Numeric         6
   21  SAL_MGR      Numeric         6
   22  SAL_CONT     Numeric         6
   23  SFACT_NTS    Numeric         6       4
   24  SFACT_WTS    Numeric         6       4
   25  WKY_TOT      Numeric         6       1
   26  WKY_RED      Numeric         6       1 <- cont wkyr reduction
   27  WKY_CONT     Numeric         6       1
   28  WKY_BILL     Numeric         6       1
   29  WKY_GOVT     Numeric         6       1
   30  WKY_PCT      Numeric         3
   31  EXP_GOVT     Numeric         9
   32  EXP_CONT     Numeric         9
   33  EXP_EXT      Numeric         9
   34  EXP_REDEXT   Numeric         9        <- ext expense reduction
   35  EXP_ADJEXT   Numeric         9
   36  EXP_GOVT_N   Numeric         9
   37  UDATE        Date            8
   38  CTDA_DIR     Numeric         4        <- current TDA
   39  CTDA_SEC     Numeric         4
   40  CTDA_IND     Numeric         4
   41  CTDA_MGR     Numeric         4
   42  CTDA_TOT     Numeric         4
   43  RBFT_TB      Numeric         6        <- revenue by fund type
   44  RBFT_63B     Numeric         6
   45  RBFT_64      Numeric         6
   46  RBFT_65      Numeric         6
   47  RBFT_67      Numeric         6
   48  RBFT_OMA     Numeric         6
   49  RBFT_PROC    Numeric         6
   50  RBFT_DBOF    Numeric         6
   51  RBFT_OTH     Numeric         6
** Total **                       312

Indexes
```

```
nt1:  str(for_year)+wiforg(lab,foe)+foe+lab
nt2:  str(for_year)+lab+foe
```

2.21 BPM HELP FILE

The BPM Help file (BPM_HELP.DBF) contains the text of the help system, broken into 200 character segments per record.

```
Structure for database: C:\BPM\BPM_HELP.DBF
Field  Field Name  Type        Width    Dec
    1  HELP_FOR    Character      20
    2  HELP_TEXT   Character     200
    3  HELP_LINE   Numeric         2
** Total **                      223

Indexes:
nt1:  help_for+str(help_line,2)
```

CHAPTER 3

BPM FUNCTIONS AND PROCEDURES

3.1 FUNCTION AND PROCEDURE MAPPING BY PROGRAM CODE FILE

This paragraph presents a listing of the contents of each of the program code files. Each function and procedure in the file is listed in the order it is found in the file. A brief description of the purpose of the routine is also included. The alphabetical listing of routines in paragraph 3.2 may also be useful in locating a specific procedure or function.

Line	Routine	Description
File BPM AMCC.PRG		
19	amccharts:	This function executes the submenu for AMC Charts.
48	amcc1:	This function produces the tables for the Revenue by Source chart.
74	amcc1a:	This function produces the tables for the OMA Revenue by Source chart.
101	amcc1b:	This function produces the table for the RDTE Revenue by Source chart.
128	amcc1c:	This function produces the table for the PROC Revenue by Source chart.
155	amcc1d:	This function produces the table for the OTHER Revenue by Source chart.
183	amcc1e:	This function produces the table for the DBOF Revenue by Source chart.
210	split_chart:	This function actually prints the Revenue by Source and the Capital Investment by Source tables. The calling routine is responsible for constructing the file 'temp' with the appropriate data in it. This function prints the table. Since the table might be long (i.e. a lot of sources), the report subdivides the output into two tables and accumulates the 'other' entry on the report.
302	amcc2:	This function prints the Revenue by Appropriation chart.
344	amcc5:	This function prints the Revenue by Group chart.
388	amcc3:	This function prints the Workyears by Type chart.
486	amcc4:	This function prints the Cap Invest by Source chart.

Line	Routine	Description
614	amcc_ihoh:	This function prints the In-House/Out-House Expense summary chart. It works directly from the Totals file (unlike all the other AMC charts!) so it must be currently recalculated. It takes all non-TDA, non-OTM org expenses and places them into the totals for the major externals. It puts the chargeable capital investments into the externals.
664	core_tot:	This function pepares a report of revenue being spent by each core capability for a single year. It allocates the revenue by the percentages stated for each core capability. It also produces the text file 'allerror.txt' which tells which of the core capabilities (of those which do not add up to 100%) results in revenue or workyears being improperly allocated in this report.
1156	add_field:	This function adds a field to a structure extended file, and is used when defining a temporary data file structure.
1173	alloc_det:	This function allocates some fraction of the whole costs for a G&A or TS function to a direct function.
1216	ctreport:	This function prints one page of the core capabilities totals report for a specific combination of appropriation and direct/customer.

File BPM_ANAL.PRG

19	baseline:	This function executes the submenu for the Baseline Analyses.
42	view_totals:	This function is the entry point for the Lab/Function analysis, where the user selects the specific Lab/Function to analyze.
66	disp_foe:	This function opens the window and prepares the menu for the Lab/Function baseline analysis.
117	disp_foe_guts:	This function displays the contents of the baseline analysis window, for all five types of analysis windows which are displayed.
256	dotoggle:	This function toggles the Lab/Function baseline analysis between the Function and the Lab (and the reverse). The Lab analysis is indicated by using the FOE of 'ALL'.
277	get_cost_tot:	This function exists to total the costs for a specific lab and foe. The total cost will be the sum of the CPB related costs, the major externals, the chargeable capital investments, the support contractors, and G&A tax (if any). This is used by the Center analysis screen and by various reports to correctly charge the cost of non-TDA organizations as external costs to the Center.
312	anal_details:	This function executes the submenu for the Analyses Details.

Line	Routine	Description
335	disp_expense:	This function displays the elements of the loaded CPB totals which are displayed as a single line on the baseline analysis screen. This matches the reporting function 'l_exp' which should be modified when this function is changed.
430	disp_external:	This function displays the elements of the externals costs which are displayed as a single line on the baseline analysis screen. This matches the reporting function 'l_ext' which should be modified when this function is changed.
495	disp_contract:	This function displays the elements of the support contract totals which are displayed as a single line on the baseline analysis screen. This matches the reporting function 'l_cont' which should be modified when this is changed.
547	disp_capital:	This function displays the elements of the capital investment totals which are displayed as a single line on the baseline analysis screen. This matches the reporting function 'l_cap' which should be modified when this is changed.
601	disp_internal:	This function displays the totals of the internal transfers which are displayed as revenues and expenses on the baseline analysis screen. Internal transfers could be a problem area because an organization may anticipate revenues which another may not be projecting to give up. This matches the reporting function 'l_int' which should be modified when this is changed.
665	disp_wkyrs:	This function displays the elements of the government workyear totals which are displayed as a single line on the baseline analysis screen. This matches the reporting function 'l_wkyrs' which should be modified when this is changed.
750	view_center:	This function opens the window and prepares the menu for the Center/G&A/TS baseline analysis.
802	ctr_toggle:	This routine displays the pull-down menu of toggle options for the Center analysis screen, and lets the user change the type of analysis being performed.
882	recalc_all:	This function is just the user interface for recalculating all Labs at once and then for producing the Center-wide totals. The user can opt to skip either of the recalc phases if he wants to, and then the contents of the screen are redrawn.
944	recomp_center:	This function prepares the Center totals from the Lab totals.
1104	rem_ts_cost:	This function removes the Tech Support cost from the loaded CPB totals for a single Lab/Function. This is important in the Center totals where you do not want to count the contribution toward Tech Support costs in the loaded CPB because you are going to count the actual Tech Support costs incurred by the Tech Support organizations in the totals.
1139	recalc_lab:	This function is the user interface for the Lab recomputation. It asks the user if he wants to perform the recomputation.

Line	Routine	Description
1166	recomp_lab:	This function recomputes the Lab totals in two steps. The first is to process each Function in the Lab one at a time, and the second is to prepare the Lab totals from the Function totals.
1270	recomp_foe:	This function prepares the totals for just one Function.
1532	put_totals:	This function writes a single totals array into the totals file, and marks it for the correct Lab, Function, and pool. It will not write the array if it is all zeros. It always appends a new record in the file for the totals.
1566	post_totals:	This function updates a single totals array in the totals file.
1603	get_totals:	This function retrieves a totals array from the totals file, for a particular Lab, Function, and pool. If the requested totals are not in the file, then the array is set to zero.
1643	compcpb:	This is the revenue-based calculation of EOR's, G&A, and CPB used in the revenue screen and many analyses and reports. Normally, we don't care about the revenue-based expenses, other than the G&A surcharge. Instead, we use the personnel data (and the corresponding function 'compwky') to compute them.
1726	compwky:	This function computes the associated costs of personnel workyears, used in the personnel screen and in various reports and analyses. These are the salary and benefits, as well as the EORs computed by factors. This function is much more important than the function 'compcpb' for this purpose since these costs are based on personnel not revenues.
1808	compass:	This function computes the total capital investment cost from a replacement schedule (which is based on current asset value and depreciation period) and from any new investments.
1865	view_rates:	This function sets up the opens the window and sets up the menu for the G&A Rate Computation baseline analysis. It is designed to display the rates that would be required to exactly fund the G&A Pool. There is a complicated relationship between the rate and the fund pool that it creates, because the rate affects the amount that is available to spend on payroll. It is assumed that the G&A costs are fixed at their current levels. It is also assumed that only the technical mission FOEs will bear the entire cost of the unfunded G&A.
1927	disp_rates:	This function computes and displays the required G&A rates, as well as displaying the other contents of the window.
1993	list_rates:	This function executes the submenu for the G&A rates list.
2014	accept_rates:	This function takes the computed G&A rates and makes them the official G&A rate for the Center. This overwrites the previous G&A rate data so it is potentially dangerous.

Line	Routine	Description

File BPM_ASST.PRG

18 assorted:
This function executes the submenu for the Utilities.

44 doreindex:
This function reindexes the data files. It is possible that the indexes could become corrupted and need to be refreshed (for example, if there is a power outage while in the BPM). Normally, this would not happen, and the program does rebuild the indexes in the startup routines if they are missing or outdated.

78 dopack:
This function packs each of the data files. When records are deleted, they are not physically removed. They still occupy space in the data files. This function will remove the unused space, freeing it up for other purposes.

118 docheck:
This function performs of the database validation check. At the moment, this check only looks at the linkage between the data files and the org file, between the wkyrs file and the personel file, and between selected files and the revenue file. Other linkages which could be checked (such as the data validation codes) are not.

165 checkone:
This function lists out all the records in the passed file which are not correctly linked to the orgs file.

220 doload:
This function loads data from another BPM. This data has been transferred to this PC and is available on a floppy or in some directory on the hard disk. This function gets the path to that new data, determines the type of load and then loads each of the data files.

313 genload:
This function loads data for one file at a time.

338 avgload:
This function loads and averages the factors data.

397 delete_lab:
This function deletes data from the data files for one Lab and/or Function.

439 del_allbut:
This function deletes data from the data files for all but one Lab and/or Function.

482 deleteone:
This function deletes records from a single data file and then packs it to remove the deleted records. Depending upon the value of the parameter, it will delete records which match the Lab and/or Function, or it will delete records which do not match the Lab and/or Function.

518 copydata:
This function creates a copy of the data files (for one Lab and/or Function) on a floppy or a subdirectory of the hard disk. It can be used to transfer data to another BPM or to back up the user's data.

566 copyone:
This function copies data for one file name. It creates the file in the destination path and fills it with the data which matches the specified Lab and/or Function.

Line	Routine	Description
588	toggle_list:	This function toggles the list engine switch 'prswitch'. The switch is used to control whether or not records which do not affect the totals are printed or not.

File BPM_EDIT.PRG

19	data_entry:	This function executes the submenu for user Data Entry edits.
51	view_foe_factors:	This function allows the user to view the factors for a Function. It displays the menu which allows the factors to be edited or printed.
132	addupfactors:	This function adds up the factors and displays the bottom line totals on the factors screen.
158	edit_foe_factors:	This function defines the pulldown menu for editing the foe factors.
205	view_main_factors:	This function allows the user to view the Center-wide factors.
252	edit_main_factors:	This function defines the pulldown menu for editing the Center-wide factors.
280	e_factor:	This function edits a single line of factors. But first, it allows the user to optionally select to estimate the factors.
325	est_factor:	This function asks the user to enter the numerator and denominator of the estimate for the factor. It then puts the ratio into each element of the temporary factors array.
353	load_main_factors:	This function loads the Center factors into the global arrays.
382	load_foe_factors:	This routine loads the factors data for a single Lab/Function into the global arrays for the Function's factors.
428	get_factor:	This function reads a factor array from the factors file.
450	put_factor:	This function stores a factors array into its correct position in the factors file. It only stores non-zero factors. It will open and close the factors file if needed.
486	list_factors:	This function executes the submenu for the factors list.
514	view_revenue:	This function opens the revenue edit window and sets up the horizontal menu.
584	edit_revenue:	This function displays the pull down menu for editing the revenue data.
656	e_revenue:	This function displays and edits the contents of the revenue window. You can only edit the revenue ID data (top half) here. The other items are edited through the edit_revenue pull down menu.

Line	Routine	Description
801	e_carry:	This function edits the carryover amounts on a revenue record. It is needed because there are two places where carryover is edited.
824	del_revenue:	This function is needed (rather than just calling del_rec) because it is possible that this revenue record will have associated external, internal, or products records. If they exist, they must be deleted too.
882	move_revenue:	This function moves a revenue record (and associated external, internal, and product records) to another Lab/Function.
934	e_rev_item:	This function edits a single line of the revenue screen.
969	list_revenue:	This function executes the submenu for the revenue list.
1008	view_major:	This function opens the window for major externals records from the externals file, and sets up the horizontal menu.
1070	e_major:	This function displays and edits the contents of the major externals window.
1144	view_support:	This function opens the window for support contract records from the externals file, and sets up the horizontal menu.
1206	e_support:	This function displays and edits the contents of the support contracts window.
1279	view_capmaj:	This function opens the window for major capital expense records from the externals file, and sets up the horizontal menu.
1342	e_capmaj:	This function displays and edits the contents of the major capital investments window.
1415	view_internal:	This function opens the window to display internal transfer records from the internal file, and sets up the horizontal menu.
1477	e_internal:	This function displays and edits the contents of the internal transfers window.
1550	view_products:	This function opens the window for product/project records from the product file, and sets up the horizontal menu.
1602	e_product:	This function displays and edits the contents of the products window.
1659	view_personnel:	This function opens the personnel data entry window and sets up the horizontal menu.
1723	edit_personnel:	This function displays the pull down menu for editing the personnel data.

Line	Routine	Description
1778	e_personnel:	This function displays and edits the contents of the personnel window. You can only edit the revenue ID data (top half) here. The other items are edited through the edit_personnel pull down menu.
1883	blank_if_cont:	This function makes current field blank if the personnel type is 'CONT'.
1905	set_salary:	This function computes the budget year salary if and only if the execution year salary was zero and has just been edited.
1922	e_per_item:	This function edits a single line of the personnel screen.
1972	get_wkyrs:	This function retrieves a specific record from the wkyrs file. If the record does not exist, then the function returns zero workyears.
1999	del_personnel:	This function is needed (rather than just calling del_rec) because it is possible that this personnel record will have associated workyear records. They must also be deleted.
2031	move_personnel:	This function moves a personnel record (and associated wkyrs records) to another Lab/Function.
2070	list_personnel:	This function executes the submenu for the personnel list.
2099	view_corecap:	This function opens the window to view/enter the core capability percentages for a Lab and Function. It scans the revenue file to determine what the allowable appropriation types are. It also sets up the horizontal menu for this screen.
2210	go_corecap:	This function lets the user select a specific appropriation from the allowable ones for this Lab and Function.
2229	e_corecap:	This function displays and edits the contents of the core capability window.
2316	cctot:	This function computes and displays the total percentages (which should be 100%) of the core competencies.
2341	get_cap:	This function retrieves the percentage which has been entered for a single Lab, Function, Appropriation, and Core Competency. If none exists, it returns zero.
2365	put_cap:	This function stores the percentage which has been entered for this Lab, Function, Appropriation, and Core Competency. It creates a record if one is needed.
2400	view_capital:	This function opens the non-major capital investment window and sets up the horizontal menu.
2459	e_capital:	This function displays and edits the contents of the non-major capital investments window.

Line	Routine	Description
2544	post_repl:	This function posts the computed replacement amounts into the actual replacement amounts in the non-major capital investments window.
2569	list_capital:	This function executes the submenu for the non-major capital investments list.

File BPM_GENL.PRG

40	oksrc:	This is a valid function which ensures that only allowable responses for the customer source are entered. If a bad type is entered, then this creates a pop-up window of the acceptable values. It adds "OTHER" to the source list if the group is not direct. If the source selected is "OTHER" then the user is prompted to enter it directly into the field. Note that this valid is different from the others in that it will accept the field if something is typed in there.
119	okgrp:	This is a valid function which ensures that only allowable responses for the customer group are entered. If a bad group is entered, then this creates a pop-up window of the acceptable values.
172	okapn:	This is a valid function which ensures that only allowable responses for the appropriation codes are entered. If a bad code is entered, then this creates a pop-up window of the acceptable values.
205	okcat:	This is a valid function which ensures that only allowable responses for the category are entered. If a bad category is entered, then this creates a pop-up window of the acceptable values. Note that RDTE gets only the RDTE categories, OMA gets only the OMA categories, and other appropriations must have a blank category.
260	okps:	This is a valid function which ensures that only allowable responses for the TDA category are entered. If a bad category is entered, then this creates a pop-up window of the acceptable values.
288	okext:	This is a valid function which ensures that only allowable responses for the external type are entered. If a bad type is entered, then this creates a pop-up window of the acceptable values.
315	okpers:	This is a valid function which ensures that only allowable responses for the personnel type are entered. If a bad type is entered, then this creates a pop-up window of the acceptable values.
342	okpos:	This is a valid function which ensures that only allowable responses for the position code are entered. If a bad code is entered, then this creates a pop-up window of the acceptable values.

Line	Routine	Description
370	okyn:	This is a valid function which ensures that only allowable responses for the 'yes'/'no' questions are entered. If any other answer is given, then this creates a pop-up window of the acceptable values.
396	okdc:	This is a valid function which ensures that only allowable responses for the 'direct'/'customer' question are entered. If any other response is given, then this creates a pop-up window of the acceptable values.
422	okass:	This is a valid function which ensures that only allowable responses for the asset type are entered. If a bad type is entered, then this creates a pop-up window of the acceptable values.
454	okontda:	This is a valid function which tests the condition of the on_tda flag to make sure that if the Function is not on the TDA, then its Lab begins with an asterisk (*). The asterisk is used extensively in analysis and reports to identify those non-TDA organizations, so it is critical.
477	okorg:	This is a valid function which ensures that only allowable responses for the organization type are entered. If a bad type is entered, then this creates a pop-up window of the acceptable values.
505	okskl:	This is a valid function which ensures that only allowable responses for the skill type are entered. If a bad type is entered, then this creates a pop-up window of the acceptable values.
535	okfact:	This is a valid function which ensures that only allowable responses for the factor type are entered. If a bad type is entered, then this creates a pop-up window of the acceptable values.
571	okwky:	This is a valid function which ensures that only allowable responses for the type of workyears are entered. If a bad type is entered, then this creates a pop-up window of the acceptable values.
599	oklab:	This is a valid function which ensures that only allowable responses for the Laboratory codes are entered. If a bad code is entered, then this creates a pop-up window of the acceptable values.
623	okintlab:	This is a valid function which ensures that only allowable responses for the Laboratory codes are entered. If a bad code is entered, then this creates a pop-up window of the acceptable values. This limits the Labs to internal ones, and so is different from the similar function 'oklab'.
657	okfoe:	This is a valid function which ensures that only allowable responses for the Function (foe) codes are entered. If a bad code is entered, then this creates a pop-up window of the acceptable values.

Line	Routine	Description
688	dircust:	This function determines whether the funds represented by a revenue record are Direct or Customer. It returns the appropriate words for use in an index or printed field.
710	wr_years:	This function writes the sequence of years on the current window, beginning with the base year.
724	years:	This function composes a string of the years, beginning with the base year. Each occupies a seven character space. It is used extensively in the reports and screen painters. Seven characters per year are the most that can be accommodated within the window, given that there are 8 years.
741	wr_uls:	This function prints a five character underline, with two leading spaces, 8 times. These are used to underline the years, and to separate rows of data.
756	wr_duls:	This function prints a five character underline, with two leading spaces, 8 times. This is a double underline using the equals (=) sign, used to indicate totals.
771	wr_arr:	This function writes an array, and its associated message, to the screen in a standard fashion.
790	larray:	This function creates a string of an array which is used for printing or displaying to the screen in a standard fashion.
821	aadd:	This function adds two arrays together, element by element, and stores the result in the first array.
843	asub:	This function subtracts two arrays from each other, element by element, and stores the result in the first array.
865	amult:	This function multiplies each element of an array by the specified number, and stores the result in the array.
887	minmax:	This function ensures that each element of an array is between the low and high values, inclusive. This guarantees that the array will only have acceptable values (which is important if it is going to be written to a file).
907	get_array:	This function transfers data from the current record into an array. It uses the base (common) part of the field name to recognize which data to load.
1031	put_array:	This function transfers data from an array into the current record. It uses the base (common) part of the field name to recognize which data fields to store into.
1100	aaddf:	This function retrieves an array from the current record and adds its contents to another array. It is used when the contents of the current record aren't needed for any other purpose, such as when you need only to sum a group of records.

Line	Routine	Description
1122	avmult:	This function multiplies two arrays together, element by element, and stores the result in another array.
1144	avdiv:	This function divides one array by another, element by element, and stores the result in another array.

File BPM_OI ..N.PRG

24	dolicense:	This procedure displays the general program information window - Who it was written for and who wrote it. It has been activated by the F2 key, so it is required to be a procedure and it must remember where the cursor was when it was called. It waits for any key to close the window.
54	showdemomsg:	This function displays the message below if the program has been started as the demo version. In the main program, if there is the file 'bpm.dem' in the current directory, then this function is called.
74	upgrade:	This function steps through the data files and forces them to have the required structure. This is important whenever program changes have changed the file structures. It is activated only when the user types 'bpm upgrade' at the DOS command line. The only problem with this is that sometimes the changes cannot transfer data (like when a field name changes). *For these problems*, use the function tempupgrade to insert specific, temporary file manipulations.
120	upgrone:	This function upgrades a single file. It takes the original file and renames it to '*.bak' and throws away all old indexes. Then it causes the correct file structure to be created and appends all records from the old data file. Only data with the same field name is brought into the new file. New fields are left empty (blank or zero).
153	tempupgrade:	This function is used to make the specific database changes that must be accommodated for a specific release of the BPM. For example, an empty field may need to be filled or computed from other database values.
172	defineallfiles:	This function defines the files which are specifically used in the BPM. Each file is given an ID number, a name, and one or more indexes (a leading asterisk means a unique index).
207	buildfile:	This function defines the data files and their fields. When called, it will create a data file with the appropriate name and structure.
457	load_orgs:	This function places the organization names (Labs and Functions) into arrays so that the two functions 'sel_foe' and 'sel_lab_foe' can be sped up. Note that the user might enter a specific Lab as a parameter on the DOS command line and so the program will limit the Function choices.

Line	Routine	Description
554	sel_foe:	This function pops up the selection window of the Functions and returns the parameter values for the selected Function.
583	sel_lab_foe:	This function lets the user select the Lab and/or Function to operate with. It always offers all the Labs and Functions to select from. It's used in the Utilities.

File BPM_REPT.PRG

21	list_totals:	This function executes the submenu for the analysis list for the Lab/Function baseline analysis.
56	list_center:	This function executes the submenu for the analysis list for the Center/G&A/TS baseline analysis.
99	list_tot1:	This function prints the listing of the analysis results for all types of baseline analyses. It is generic enough to handle each of the different baseline analysis products. It closely parallels the 'details' functions in the baseline analyses and you will probably want to make changes here whenever there are changes there.
156	l_main:	This function prints the main screen of the baseline analysis.
270	l_exp:	This function prints the elements of the loaded CPB totals from the baseline analysis.
341	l_ext:	This function prints the externals detail screen of the baseline analysis.
389	l_supt:	This function prints the support contractors screen of the baseline analysis.
423	l_cap:	This function prints the capital investment screen of the baseline analysis.
459	l_int:	This function prints the internal transfers screen of the baseline analysis.
505	l_wkyrs:	This function prints the workyears screen of the baseline analysis.
562	intl_diag:	This function prints the diagnosis report for the internal transfers.
681	org_nets:	This function produces a report of the net balance for each Lab/Function.
764	list_corecap:	This function executes the submenu for the core capabilities list.
791	core_qc:	This function prints the core capability QC report.

Line	Routine	Description
852	core_dollars:	This function prints a report of revenue being directed to each core capability in a year. It allocates the revenue by the percentages stated for each core capability. The user can specify which Lab/Function(s) to work with and which year. This excludes all internal transfer revenues.
960	corefilt:	This function allows the user to specify which Lab, Function, Appropriation, and Year to use to filter the core capability reports.
1018	list_avg_fact:	This function prints a report of average factors for the Center or for just one Lab. It averages the factors based on the civilian pay base for each Function being averaged.

File BPM_REP1.PRG

24	expenses:	This function executes the submenu for the total expenses reports in the Center Analysis list.
45	tot_exp1:	This report totals the expenses by Direct/Customer, APPN, and Category for each of the subsets CPB, MS, TT, TNG, Capital Investments, Tech Support (all combined), G&A, and Laboratory Indirect, as well as major externals (including major capital investments), and support contracts. Only the principal technical mission FOEs are included. The calculation is revenue based.
355	retrieve_costs:	This function is used to quickly get the total costs for a group of foes, either G&A, indirect, or tech support. When a parcel of money contributes to the G&A pool, for example, it must be allocated to the basic cost elements of the G&A pool.
415	allocate_costs:	This function takes the amount to be allocated, and divides it among the various expenses in proportion to the ratio of the expense to the total expense. It also accumulates these expenses into the allocated arrays so that the totals for a category can also be printed. It sets a flag so that the totals block will get printed.
483	tot_exp2:	This report totals the expenses by APPN and Category for each of the subsets CPB, MS, TT, TNG, Capital Investments, Tech Support (all combined), G&A, and Laboratory Indirect, as well as major externals (including major capital investments), and support contracts. Only the principal technical mission FOEs are included. The calculation is revenue based.
794	tech_base_ihoh:	This function produces a report of Tech Base in-house expenses and externals and their ratios. It uses the same method for allocating G&A, Tech Support, and Lab Indirect expenses as the previous two expense totals reports.

Line	Routine	Description
1024	source_use:	This report totals the expenses by Direct/Customer, APPN, and Category for each of the subsets CPB, Other (MS, TT, TNG), Capital Investments, OGA, Externals, & Support Contracts. It allocates costs for Tech Support, G&A, and Laboratory Indirect proportionally according to the revenues received. The calculation is revenue based. This report prints for a specified year only.
1478	alloccomp:	This function composes one line for printing in the sources and uses report. If needed, it rounds the numbers before printing them.
1506	allocadd:	This function adds two parallel arrays of 12 elements for the sources and uses report.
1532	allocsu:	This function does the allocation of G&A, TS, and IND costs for the sources and uses report. It works with an array which holds computed values in array elements 7, 8, and 9 respectively. They are transferred to array elements 1 thru 6 based on the ratio of G&A, TS, and IND costs to their own totals.
1564	readjust:	This function rebalances a single array to proportionally accommodate differences between revenue-based and personnel-based calculations.
1618	balances:	This function executes the submenu for the balances reports.
1636	balance1:	This function prints a report which balances the non-OMAD, non-DBOF (i.e., mostly RDTE) revenue against all personnel whose pay source is 'R'.
1679	balance2:	This function prints a report which balances the OMAD revenue against all personnel whose pay source is 'O'.
1719	balance3:	This function prints a report which balances the DBOF revenue against all personnel whose pay source is 'D'.
1759	bal_part1:	This function prints the first half of the balance report, which summarizes the personnel costs.
1873	bal_part2:	This function prints the second half of the balance report, in which the uses of revenues are described.
1958	ts_amounts:	This function executes the submenu for the Tech Support Amount reports.
1977	tsamt:	This function prints a report which shows the amounts contributed to a Tech Support Function by all other organizations. This data is taken from the totals file.

Line	Routine	Description

File BPM_REP2.PRG

38	fillnf:	This function fills the standard arrays used by the user- defined listing and totals report writers. It exists to make the definition of these arrays easier for the programmer.
78	list_fac1:	This function constructs the field definitions for the factors file listing report.
96	list_rev1:	This function constructs the field definitions for the revenue file listing report.
130	list_ext1:	This function constructs the field definitions for the externals file listing report. It also prints related data from the revenue file, so it is linked.
176	list_int1:	This function constructs the field definitions for the internal transfers file listing report. It also prints related data from the revenue file, so it is linked.
218	list_prod1:	This function constructs the field definitions for the products file listing report. It also prints related data from the revenue file, so it is linked.
259	list_pers1:	This function constructs the field definitions for the personnel file listing report.
294	list_cap1:	This function constructs the field definitions for the capital investment file listing report.
321	list_core1:	This function constructs the field definitions for the core capabilities file listing report.
344	list_orgs:	This function constructs the field definitions for the organization file listing report.
368	list_skills:	This function constructs the field definitions for the skills/series file listing report.
386	list_off:	This function constructs the field definitions for the offices file listing report.
404	list_sources:	This function constructs the field definitions for the sources file listing report.
421	list_revtypes:	This function constructs the field definitions for the revenue types file listing report.
442	list_ctda:	This function constructs the field definitions for the current TDA file listing report.
464	list_pess_rev:	This function constructs the field definitions for the revenue pessimism file listing report.
497	list_wif1:	This function constructs the field definitions for the What If file listing report.
556	user_defined:	This function is the user interface for the flexible user-defined listing report. It contains the report definition window with the three the three columns for the fields to print, the sort order, and the filter conditions.
884	list_engine:	This function actually prints the user-defined listing report.

Line	Routine	Description
945	show_arr:	This function displays one column of the user-defined report definition in a specified defined window block and clears any unused spaces.
973	show_line_length:	This function computes and displays the width of the resulting report (in characters) based on the currently selected print fields.
1000	compose_line:	This function composes the print string needed to produce the printed line for an array variable. It also accumulates the totals in an array so that they can be printed at the end of the report.
1040	c_o:	This function composes the print string needed to produce the printed line for a scalar variable. It also accumulates the totals so that they can be printed at the end of the report.
1059	anonzero:	This function determines if an array of fields contains any non-zero elements. It is used to determine if the line of text needs to be printed.
1077	getfiltcond:	This function displays a window so the user can enter the operator and the test condition for a single filter condition.
1114	okoper:	This function executes the valid condition for the operator.
1141	savepsf:	This function saves the report definition arrays into a specified memory variable file. It temporarily transfers the contents of the arrays through macro variables because Clipper cannot store arrays in memory (.mem) files.
1184	loadpsf:	This function loads the contents of the report definition arrays from a specified memory variable file. It temporarily transfers the contents of the arrays through macro variables because Clipper cannot store arrays in memory (.mem) files.
1229	rem_old_names:	This function checks the contents of the report definition (.mem) file to ensure that the fields are all currently defined. Only the first 10 characters are checked.
1255	listcapstr:	This function computes the total capital investment and composes the string to write it out.
1281	net_carry:	This function computes the net carryover.
1306	net_revenue:	This function composes a string of the net revenue (with carryover).
1329	net_rev:	This function computes the net revenue (with carryover).

Line	Routine	Description

File BPM_REP3.PRG

43 tot_rev1:
This function constructs the field definitions for the revenue file totals report.

82 tot_ext1:
This function constructs the field definitions for the externals file totals report. It also prints related data from the revenue file, so it is linked.

129 tot_int1:
This function constructs the field definitions for the internals file totals report. It also prints related data from the revenue file, so it is linked.

174 tot_rev2:
This function constructs the field definitions for the revenue file totals report, tailored to print the direct workyears data.

214 tot_carry:
This function constructs the field definitions for the net carryover totals report.

253 tot_netrev:
This function constructs the field definitions for the net revenue totals report.

292 tot_pers1:
This function constructs the field definitions for the personnel file totals report.

331 tot_cap1:
This function constructs the field definitions for the capital investment file listing report.

360 tot_pess_rev:
This function constructs the field definitions for the revenue pessimism file totals report.

390 user_totals:
This function is the user interface for the flexible user-defined totals report. It contains the report definition window with the two columns for the subtotal fields and the filter conditions.

599 total_engine:
This function actually prints the user-defined totals report.

File BPM_SYST.PRG

20 sys_edit:
This function executes the submenu for System Administrator edits and other features for managing the database.

54 view_orgs:
This function is the user interface to view and edit the organization file.

100 e_org:
This function displays and/or edits the contents of the organization window.

164 view_off:
This function is the user interface to view and edit the offices file.

210 e_off:
This function displays and/or edits the contents of the office window.

260 view_skills:
This function is the user interface to view and edit the skills/series file.

Line	Routine	Description
307	e_skill:	This function displays and/or edits the contents of the skills/series window.
357	view_sources:	This function is the user interface to view and edit the sources file.
403	e_source:	This function displays and/or edits the contents of the sources window.
451	view_revtypes:	This function is the user interface to view and edit the revenue types file.
497	e_revtype:	This function displays and/or edits the contents of the revenue types window.
554	view_ctda:	This function is the user interface to view and edit the current TDA file.
602	e_ctda:	This function displays and/or edits the contents of the current TDA window.
661	tot_ctda:	This valid function computes and displays the total TDA.
681	incr_baseyear:	This function is used to increment the baseyear and translate all the data over one year. It should be done only once a year!
776	edit_baseyear:	This function is used to change the base year, without translating any of the data. This just affects the base year (year0) fields in each file.
815	remove_empty:	This function processes the removal of empty records from the data files.
912	removeone:	This function removes empty records from the designated file, based on the passed criteria.
932	remove_unlinked:	This function processes the deletion of records which are not linked to the organizations file, and if appropriate, not linked to either the revenue or personnel files.
964	unlinked:	This function deletes all the records in the specified file which are not correctly linked to the orgs file.
1001	limit_overtime:	This function is used to change the overtime salary limit.

File BPM_WHA1.PRG

31	whatif:	This function executes the submenu for the What If analyses.
122	rev_pessimism:	This function opens the window to display a single pessimism record and sets up the horizontal menu.
173	pessfilt:	This function lets the user specify a Lab and/or Function on which to filter the revenue data.

Line	Routine	Description
197	e_pessimism:	This function displays and edits the contents of the pessimism window.
312	compchg:	This is the valid function which computes and displays the resulting factors as the user enters the percent changes.
340	dopessimism:	This function manages the application of the pessimism factors, displays the results, and defines the menu for allocations and printouts.
486	pess_scan:	This function applies the pessimism factors to the revenue file to produce the revised revenue and costs. In the process it creates the file 'revpess.dbf' which contains the revised data.
641	pess_adj_exp:	This function adjusts the expenses for contractors, externals, and capital investments in the revenue pessimism file. It only works if there was revenue originally, and now that revenue has been driven to zero by the pessimism factors.
666	pess_adj:	This function adjusts the expense elements of the revenue pessimism file proportionally to the extent that the revenue was changed by the factors.
685	pess_show:	This function displays the contents of the 2nd pessimism window (that displays the results of the pessimism factors). As the data is altered by allocations, it is redisplayed.
726	alloc_cost:	This function executes the submenu for the allocation options. Between each call it must redisplay the screen.
761	alloc_wky:	This function allows the users to edit the revised workyears.
791	calloc_wky:	This is the valid function for the workyear editing which redisplays the cost of the workyears and the total costs as the data is edited.
817	alloc_one:	This function edits one of the cost elements of the allocation. It opens a window which displays the baseline data, the impact of the pessimism factors, and the adjusted revised data that the user edits.
885	calloc_one:	This valid function displays the difference between the revised amount and the adjusted revised amount on the line above the entry.
907	e_cpb_wky:	This function edits the average loaded CPB cost per workyear.
940	e_cpb_ga:	This function edits the What If G&A rate.
979	list_pessimism:	This function executes the submenu for the pessimism reports.
1004	list_pess1:	This function reports the same data as the revenue pessimism screen and each of the allocation screens.

Line	Routine	Description
1117	opt_personnel:	This function opens the window to display the personnel optimization menu.
1265	pcttoggle:	This function toggles the flag which controls the viewing of the data by percentages or by $K. If the toggled flag is true, then percentages are viewed.
1278	e_opt_pers:	This function displays and edits the data on the main screen for the optimize personnel algorithm. If new data needs to be loaded and recalculations done, then that is also done here.
1351	opt_disp:	This function displays the contents of a single line on the personnel optimization screen.
1383	opt_fact:	This function displays the pull-down menu for the optimization factors edit screens ('Factors').
1407	opt_fact1:	This function displays the workload planning factors for each Lab/Function, indicating the basis for the factor, and also the lower limit number of personnel which can be assigned to that Function.
1470	opt_fact2:	This function displays and allows the user to edit the factors which tell the algorithm how to estimate the number of contract actions.
1526	opt_view:	This function displays the pull-down menu for the detailed views of the Personnel Optimization. The data displayed on these screens has already been stored in arrays during the computations.
1545	opt_view1:	This function displays the allocation of personnel positions to Lab and Function by total, direct, managers, secretary, and indirect. This data is stored in the array 'staffpos' during the recalculations.
1560	opt_view2:	This function displays in-house costs ($K) by Lab and Function. This data is stored in the array 'ihcosts' during the recalculations.
1572	opt_view3:	This function displays the revenues by fund type for each Lab and Function. This data is stored in the array 'revenues' during the recalculations.
1587	opt_calc:	This function does the Personnel Optimization recalculations. It is a circular calculation, and must be cycled through until it converges.
1756	oneorgcalc:	This function is a general-purpose function to calculate staffing levels for a support organization.
1803	dirorgcalc:	This function is a general-purpose function to calculate staffing levels for a direct organization.

Line	Routine	Description
1843	gasubsidy:	This function figures out where the G&A subsidy is spent. This is non-trivial because the subsidy may large enough to cover the personnel costs or it might not.
1895	reloaddata:	This function rescans the tda_wif file to reload the data arrays used in the Optimize Personnel Algorithm. This is needed each time the year changes.
1979	loadrevs:	This function loads the data for one Lab/Function.
2030	ardiv:	This function creates a distribution of one array and stores it in another. It is used to compute the weighting fractions, as in revenue distributions by fund type.
2052	arsum:	This function adds the first nine elements of an array and stores the sum into the tenth element. It is used to add up the array when the individual elements have been separately entered (as in loading the revenue data).
2071	ardist:	This function takes the total amount of an array (stored in the tenth element) and distributes it to the first nine elements of the array according to the distribution contained in a second array.
2090	arvadd:	This function adds two arrays of ten elements and stores the results in the first array.
2107	arvsub:	This function subracts two arrays of ten elements and stores the results in the first array.
2124	arneg:	This function returns true if any element of an array is negative. It is used to check the data and warn the user if the algorithm produces any strange results because of bad input.
2143	opt_list:	This function displays the pull-down menu for the Personnel Optimization list.
2166	opt_list1:	This function prints the report of staff positions by Lab and Function. It prints the data contained in the array 'staffpos'.
2200	opt_list2:	This function prints the report of in house costs by Lab and Function. It prints the data contained in the array 'ihcosts'.
2229	opt_list3:	This function prints the report of revenues by Lab & Function. It prints the data contained in the array 'revenues'.
2258	opt_save:	This function stores the personnel allocation computed by the Personnel Optimization algorithm into the tda_wif file for this year.

Line	Routine	Description

File BPM_WHA2.PRG

30 tda_whatif:
This function opens the window for the TDA What If evaluation summary data for a single year, and sets up the submenu.

95 e_tda_what:
This function displays and edits the data for a single evaluation year.

170 tda_eval_comp:
This function fills all the computed fields for a single evaluation year. It assumes that the totalled fields have already been filled from the organization-level data.

244 e_tda_tgt:
This function allows the user to enter the target TDA totals for all the years in one window.

276 view_tda_org:
This function opens the organization-level window. It starts by displaying the current organization's data, and then sets up the horizontal menu.

340 tgt_toggle:
This function presents and executes the pull down submenu for 'Use' at the organization-level menu.

385 eval_toggle:
This function presents and executes the pull down submenu for 'Use' at the Center-level menu.

423 showrecmsg:
This function displays the status of the recalc flag and the type of G&A rate being used, on the bottom border of the current window.

443 e_tda_org:
This function displays and edits a single record of the organization-level data. It works closely with the related valid function 'tda_tot' to display the data on the screen.

535 show_overall:
This function causes the current evaluation year to be retotaled and then displays certain measures on the bottom line of the current window.

566 tda_tot:
This function displays the results of the line calculations, which are performed in the function 'tda_tot_comp'.

617 tda_tot_comp:
This function performs the line calculations that fill in all of the computed fields for one organization in the evaluation year.

721 wkyrs_adjust:
This function calculates the adjusted total required workyears due to the government/contractor exchange ratio. The problem is that as contractors are added/subtracted, then the total required workyears will change (if the exchange ratio is other than 1.0).

Line	Routine	Description
745	recalc_tda:	This function recalculates the totals from the baseline BPM data and stores the results into the 'tda_wif' file by organization. This same function is used by both the Center TDA and the Optimize Personnel What Ifs. As an option, the user can choose to prepare the totals from the revenue pessimism file rather than from the baseline revenue data.
1145	rerecalc_tda:	This function retotals the data for a specific Lab. This is called only for the technical Labs which are represented by single records in the database, rather than at the Function level.
1224	recalc_year:	This function retotals the separate organization records to produce the TDA What If totals for a specific year. It calls the function 'recalc_yr' to actually do it, but this routine provides a nicer user interface.
1253	recalc_yr:	This function retotals the separate organization records to produce the TDA What If totals for a specific year. This needs to be redone when the database is retotaled, when certain factors have been changed, and whenever you want to see the yearly critical measures.
1336	list_tda_wif:	This function executes the submenu for the TDA What If Center-level reports.
1364	list_tda_org:	This function executes the submenu for the organization-level reports.
1390	list_tda1:	This function prints the TDA What If Summary Report. It lists selected critical measures by organization for just the current year.
1424	list_tda2:	This function prints a report of critical measures by year for the Center.
1477	one_measure:	This function prints one line of the Critical Measures by Year (Center-level) report.
1498	list_tda3:	This function prints a report of last edit dates by Lab, Function, and evaluation year.
1545	list_tda4:	This function prints a report of critical measures by year for a specific organization.
1601	org_measure:	This function prints one line of the Critical Measures by Year (organization-level) report.
1622	list_tda5:	This function prints the report 'TDA by Organization by Year', which just shows the total allocated workyears by Lab and Function by evaluation year.
1670	project_tda:	This function projects What If data for all organizations, and for the evaluation year, to all subsequent evaluation years. Then it recalculates all affected years.
1742	doproj_org_tda:	This function asks the user if organization level data should be projected to subsequent years.

Line	Routine	Description
1764	proj_org_tda:	This function projects the data for a particular Lab and Function from the current evaluation year to all subsequent evaluation years. The data which gets projected includes only that which the user modifies, and the allocation of workyears is also placed in the 'current' fields for subsequent years.
1821	wiforg:	This function produces a one character code for organizations that puts the organizations in proper order by type. It is part of the index for the 'tda_wif' file.
1848	compare_wif:	This function displays critical measures for organizations of a similar type.
1899	te_tot:	This function displays the results of the calculations for a single organization. It is used as a valid function during editing on the table edit screen.
1931	te_tda_org:	This function is the table edit for the TDA Wif by organization. In it, a window pops up with all the organizations listed. The user highlights one line and selects it to edit it.

File BROWSER.PRG

20	dobrowse:	This is the user interface that calls the browse function. It manages the window and calls the browse function. It will only browse the current file.
59	fillfld:	This function fills the arrays used by browse. This is used when the programmer wants to customize the browser.
83	browse:	This is the top-level browse function that the program should call (rather than dbedit). It sets up the various variables and arrays that are used by the keystroke handler 'xbrowse'.
170	xbrowse:	This function is a keystroke handler for dbedit. It allows the program to control what data can be edited. It returns ret_val to dbedit telling it what to do. You will need to read the Clipper manual to understand this.
285	get_pic:	This function creates a picture string to use for a field. This is only needed if the programmer has failed to provide an array of pictures to use (and so he wants the default). It handles character strings (truncating them to 78 characters in a scrolling field) and numbers (with and without decimals).
317	clear_gets:	This procedure clears the current get. It is activated when you are in the browse mode and are editing a field, and then you press the up or down arrow keys to go to the next field. It must be a separate procedure because it is activated by the set key command
334	xseek:	This function is a keystroke handler for dbedit. It allows dbedit to seek to the first record corresponding to the key which is pressed. It returns ret_val to dbedit telling it what to do. You will need to read the Clipper manual to understand this.

Line	Routine	Description

File FILEHAND.PRG

18 addnewfile:
This function adds a new file name to the list of available files for the application. If the file doesn't exist, it is built (using buildfile). It then checks each of the indexes for this files (using chkidx).

66 getfileid:
This function returns the position in the file name array of this file name.

81 openfile:
This function opens a specific file name, and all of its indexes, and makes it the current work area.

131 closefile:
This function closes a file (work area) that may not be the current file (work area). It optionally selects a work area to be current (otherwise, you take your chances as to which work area will be the current one).

156 del_record:
This is the user interface to delete a single record. It confirms the users choice to delete, finds out what record is to be made current next, and then deletes the record.

184 whatsnext:
This function returns an adjacent record, in index order, to the current record. The default is the next record, but if that fails, then return the previous record.

209 findkey:
This function determines which of the indexes that the file may contain is the index that is desired. Use of this feature frees the programmer from having to know much about the order of his indexes.

242 align:
This function positions a file, which is not the current file (work area), to a specific indexed position. It is used instead of setting a relation because it can be much faster.

264 getdatetime:
This function determines the date and time of a file name.

285 chkidx:
This function checks the index existence, date, and key expression. If there are any problems with this index, then it is rebuilt. The database in question must already be in use for this procedure.

333 doindex:
This function performs the actual indexing operation.

359 doskip:
This function skips a file in either direction, but requires that the file pointer remain within a specific range in the current index. If you try to skip outside that range, then the computer beeps.

406 addrec:
This function controls the adding of a new record to a file. Before it will append a blank record, it first checks to see if there are any deleted records at the bottom of the file. If so, then a deleted record is recalled and reused.

Line	Routine	Description
431	del_rec:	This procedure deletes the current record, and positions it at the bottom of the file (in index order) by filling the first field in the index expression with chr(255) or 9999.
458	zaprecord:	This function "erases" an entire record.

File TEXTVIEW.PRG

19	doviewtext:	This procedure is the user interface to the text viewer. It lists the text files available in the local directory. When the user selects a file, then he can view it, print it, or delete it.
87	scrollfile:	This function opens a scrolling window to view the contents of an ASCII file. It is not limited by file size, unlike the memoedit function. The window takes over the whole screen.
189	writescreen:	This function displays the entire screen.
209	scrollfwd:	This function scrolls the screen one line up, and then displays the last line of text.
227	scrollbkwd:	This function scrolls the screen down one line, and then displays the last line of text.
245	initscreen:	This function reads from a file, forward from the beginning of the file, filling the array 'lines' with lines of data. It fills the parallel array 'starts' with the starting points of each line of text. It returns .T. if the read is ok. It revises the value of endpos to be the character position of the next line in the file after all the reads.
287	readscrfwd:	This function reads from a file, forward from the old ending position, for nlines-1 of data, one line at a time. It returns .T. if the read is ok. It revises the value of endpos to be the character position of the next line in the file after all the reads.
320	readlinefwd:	This function reads from a file, forward from the old ending position, putting the new line into last element of the array lines, and scrolling the elements of the array up one. It returns .T. if the read is ok. It revises the value of endpos to be the character position of the next line in the file after the read.
357	readfwd:	This function reads from a file, forward from the start position, and parses this into the next line for text display. It returns .T. if the read is ok. It revises the value of endpos to be the character position of the next line in the file.
395	readscrbkwd:	This function reads from a file, backward from the old starting position, for nlines-1 of data, one line at a time. It returns .T. if the read is ok. It revises the value of endpos to be the character position of the next line in the file after all the reads.

Line	Routine	Description
428	readlinebkwd:	This function reads from a file, backward from the first starting position, putting the new line into first element of the array lines, and scrolling the elements of the array down one. It returns .T. if the read is ok. It revises the value of endpos to be the character position of the line that was pitched.
466	readbkwd:	This function reads from a file, backward from the end position, and parses this into the next line for text display. It returns .T. if the read is ok. It revises the value of start to be the first character position of this line in the file.

File UTILITY.PRG

22	go_general:	This function is used to execute a generic GoTo capability within a file. It is usually used when the whole file can be accessed (use the function 'go_spec' to display just a range of records). It sets up a dbedit window.
70	go_spec:	This function has the same use as 'go_general', except that it restricts the choices to a range of records defined by the current index key and a passed key segment. It works by reading the field expressions into an array, which is slower than dbedit but has other advantages.
118	isinfile:	This function is used as a general purpose valid statement when the possible answers are contained in a specific file. This can be used even if only some of the records in that file can satisfy the valid condition.
191	nofunction:	This function is merely a place-holder or stub for a feature which is not yet implemented.
204	win_print:	This function prints the contents of the current window. This is like a print screen, except that everything outside the window is omitted.
236	nonzero:	This function determines if there are any non-zero elements in an array.
254	tabisdown:	This procedure places a down-arrow keystroke on the keyboard buffer whenever the tab key is pressed.
267	shtabisup:	This procedure places an up-arrow keystroke on the keyboard buffer whenever the shift-tab key is pressed.
280	checklevel:	This executes a specified function if the user's permission level is less than or equal to the required permission level for the function call. It returns whatever is returned by the function. If the user does not have sufficient permission, this function warns the user and returns false. The user's permlevel is set in the main program.

Line	Routine	Description
302	popq:	This executes a specified function if the passed string contains a question mark ('?'). It is used in 'valid' clauses to conditionally execute the 'valid' function.
320	orblank:	This executes a specified function if the passed string is not empty. It is used in 'valid' clauses to conditionally execute the 'valid' function.
338	showeditmsg:	This function puts one of several status messages at the lower right of the screen.
367	leftjust:	This function left-justifies a message on the screen.
382	pad:	This function is used to force a string to a specific length, by adding spaces to the end or by truncating it. It is needed extensively in report formating.
397	choose:	This function produces a basic pop-up selection window. It is used for menus and pop-up selection lists.
459	fillpu:	This function is used to simplify the definition of the parameters to be passed to 'choosemenu'.
476	escape:	This function puts the <Escape> key into the keyboard, so that the user can have a 'Done' prompt on the 'choosemenu' list that will exit the menu.
490	ahandler:	This is a keystroke handler that is used by 'achoice' to intercept keystrokes and let the programmer control what goes on. It's function is to implement the help system so that it responds to each individual prompt. You need to read the Clipper documentation on this, since it is not self-evident (see achoice and dbedit).
529	choosemenu:	This is a principal part of the menu system. It displays a pop-up window with menu choices. The user selects any choice by highlighting it and pressing <enter>. The program then executes a function associated with that menu choice. Control then returns to the menu. The menu is exited by pressing the <escape> key or by calling the escape function.
590	cent:	This function displays a text string centered within a left and right boundary.
609	warnem:	This function produces a standard one or two line warning message window. It displays an OK prompt and waits for the user to respond.
647	genmsgwindow:	This function opens a generic message window. It starts at the top (which is passed) and extends down as far as required by the length of the message. The window is centered on the screen with the passed width. Hard carriage returns can be coded into the message by using a semicolon.
692	askok:	This function provides the trivial prompt 'Ok' for genmsgwindow.

Line	Routine	Description
707	amaxlen:	This function determines the maximum length of array elements, particularly for string array elements.
725	alength:	This function determines the number of elements in an array which are defined.
744	optselect:	This function is used to prompt the user when there are up to six choices. This is typically used in 'Yes/No' type questions.
809	setdest:	This function lets the user specify whether to send reports to the printer or to a file. It checks the printer to see that it is on-line.
881	closepage:	This function writes a page eject character (12) to the printer or output file. It is used frequently in reports to start a new page and to terminate a report.
895	clnupdest:	This function is the terminator for the setdest routine. It turns off the printing and closes the alternate file and sets the console back on. You should run this everytime setdest executes successfully to get back on track.
912	tlbr:	This function makes it easier to assign the top, left, bottom, and right corners of the window.
930	openwindow:	This function opens a window, using the local variables t,l,b,r defined in the calling routine. It returns the buffer character string of the screen contents before the window was opened. If width and/or height is provided, then the window will be centered in width and/or height.
969	closewindow:	This function is the companion to openwindow. Use it to close the window. Don't forget that it also needs t,l,b,r which should be defined as local variables in the calling routine.
984	norec:	This function checks to see that there are records in the database before it executes the specified. Use it whenever the specified function is unstable when the database is empty.
1002	fillpvp:	This function loads the arrays used by getthese. The arrays are defined in the calling routine.
1024	getthese:	This is a generalized gets window. It stacks a number of gets vertically and aligns them neatly.
1102	showmemory:	This procedure is used with a hot-key to let the user find out how much system RAM he has left. It can be important in Clipper applications because RAM is not freed up cleanly.
1127	fillpmu:	This function fills the arrays needed to operate the horizontal ('doit') menu.

Line	Routine	Description
1148	doit:	This function is a generalized horizontal light bar menu with user defined functions, which are executed when a menu item is selected.
1190	lmenu:	This function is the actual light bar menu drawing routine.
1230	dosvers:	This function returns the DOS version as a string. This is needed to identify old versions of DOS.
1246	help:	This procedure displays and optionally edits the on-line context sensitive help text.
1338	xmemo:	This function is a keystroke handler for the Clipper memoedit function. See the manual for details. It only checks for the escape key if the text has been edited. If so, then it allows the user to go back to editing, or to lose his changes.
1370	ispath:	This function is used in valids to ensure that when the user provides you a path name to a disk or a subdirectory, that it is OK to use.

3.2 FUNCTION AND PROCEDURE MAPPING BY NAME

This paragraph presents a mapping of each function and procedure in the BPM by name. They are listed alphabetically, along with the procedure and line number they appear in. This will be very useful in locating a specific piece of code (there are 417 procedures and functions in the BPM).

Routine	Program File	Line
aadd	BPM_GENL.PRG	830
aaddf	BPM_GENL.PRG	1114
accept_rates	BPM_ANAL.PRG	2023
addnewfile	FILEHAND.PRG	35
addupfactors	BPM_EDIT.PRG	140
add_field	BPM_AMCC.PRG	1165
add_rec	FILEHAND.PRG	418
alength	UTILITY.PRG	733
align	FILEHAND.PRG	254
allocadd	BPM_REP1.PRG	1515
allocate_costs	BPM_REP1.PRG	437
alloccomp	BPM_REP1.PRG	1488
allocsu	BPM_REP1.PRG	1544
alloc_cost	BPM_WHA1.PRG	736
alloc_det	BPM_AMCC.PRG	1187
alloc_one	BPM_WHA1.PRG	834
alloc_wky	BPM_WHA1.PRG	768
amaxlen	UTILITY.PRG	715
amcc1	BPM_AMCC.PRG	57
amcc1a	BPM_AMCC.PRG	83
amcc1b	BPM_AMCC.PRG	110
amcc1c	BPM_AMCC.PRG	137
amcc1d	BPM_AMCC.PRG	164
amcc1e	BPM_AMCC.PRG	192
amcc2	BPM_AMCC.PRG	313
amcc3	BPM_AMCC.PRG	403
amcc4	BPM_AMCC.PRG	501
amcc5	BPM_AMCC.PRG	355
amccharts	BPM_AMCC.PRG	28
amcc_ihoh	BPM_AMCC.PRG	632
amult	BPM_GENL.PRG	874
anal_details	BPM_ANAL.PRG	320
anonzero	BPM_REP2.PRG	1069
ardist	BPM_WHA1.PRG	2081
ardiv	BPM_WHA1.PRG	2039
arneg	BPM_WHA1.PRG	2132
arsum	BPM_WHA1.PRG	2061

Routine	Program File	Line
arvadd	BPM_WHA1.PRG	2098
arvsub	BPM_WHA1.PRG	2115
askok	UTILITY.PRG	700
assorted	BPM_ASST.PRG	27
asub	BPM_GENL.PRG	852
avdiv	BPM_GENL.PRG	1153
avgload	BPM_ASST.PRG	349
avmult	BPM_GENL.PRG	1131
a_handler	UTILITY.PRG	507
balance1	BPM_REP1.PRG	1647
balance2	BPM_REP1.PRG	1688
balance3	BPM_REP1.PRG	1728
balances	BPM_REP1.PRG	1625
bal_part1	BPM_REP1.PRG	1791
bal_part2	BPM_REP1.PRG	1894
baseline	BPM_ANAL.PRG	27
blank_if_cont	BPM_EDIT.PRG	1892
browse	BROWSER.PRG	116
buildfile	BPM_OPEN.PRG	226
calloc_one	BPM_WHA1.PRG	894
calloc_wky	BPM_WHA1.PRG	800
cctot	BPM_EDIT.PRG	2324
cent	UTILITY.PRG	601
checklevel	UTILITY.PRG	293
checkone	BPM_ASST.PRG	178
chkidx	FILEHAND.PRG	304
choose	UTILITY.PRG	421
choosemenu	UTILITY.PRG	554
clear_gets	BROWSER.PRG	329
clnupdest	UTILITY.PRG	905
closefile	FILEHAND.PRG	144
closepage	UTILITY.PRG	890
closewindow	UTILITY.PRG	978
compare_wif	BPM_WHA2.PRG	1860
compass	BPM_ANAL.PRG	1825
compchg	BPM_WHA1.PRG	321
compcpb	BPM_ANAL.PRG	1679
compose_line	BPM_REP2.PRG	1015
compwky	BPM_ANAL.PRG	1761
copydata	BPM_ASST.PRG	533
copyone	BPM_ASST.PRG	577
corefilt	BPM_REPT.PRG	975
core_dollars	BPM_REPT.PRG	871
core_qc	BPM_REPT.PRG	799
core_tot	BPM_AMCC.PRG	684
ctreport	BPM_AMCC.PRG	1227

Routine	Program File	Line
ctr_toggle	BPM_ANAL.PRG	816
c_o	BPM_REP2.PRG	1053
data_entry	BPM_EDIT.PRG	28
defineallfiles	BPM_OPEN.PRG	181
deleteone	BPM_ASST.PRG	496
delete_lab	BPM_ASST.PRG	409
del_allbut	BPM_ASST.PRG	451
del_personnel	BPM_EDIT.PRG	2009
del_rec	FILEHAND.PRG	444
del_record	FILEHAND.PRG	170
del_revenue	BPM_EDIT.PRG	836
dircust	BPM_GENL.PRG	697
dirorgcalc	BPM_WHA1.PRG	1814
disp_capital	BPM_ANAL.PRG	566
disp_contract	BPM_ANAL.PRG	514
disp_expense	BPM_ANAL.PRG	363
disp_external	BPM_ANAL.PRG	448
disp_foe	BPM_ANAL.PRG	81
disp_foe_guts	BPM_ANAL.PRG	144
disp_internal	BPM_ANAL.PRG	617
disp_rates	BPM_ANAL.PRG	1943
disp_wkyrs	BPM_ANAL.PRG	689
dobrowse	BROWSER.PRG	31
docheck	BPM_ASST.PRG	131
doindex	FILEHAND.PRG	345
doit	UTILITY.PRG	1166
dolicense	BPM_OPEN.PRG	36
doload	BPM_ASST.PRG	237
dopack	BPM_ASST.PRG	89
dopessimism	BPM_WHA1.PRG	369
doproj_org_tda	BPM_WHA2.PRG	1751
doreindex	BPM_ASST.PRG	57
doskip	FILEHAND.PRG	374
dosvers	UTILITY.PRG	1238
dotoggle	BPM_ANAL.PRG	267
doviewtext	TEXTVIEW.PRG	38
edit_baseyear	BPM_SYST.PRG	787
edit_foe_factors	BPM_EDIT.PRG	168
edit_main_factors	BPM_EDIT.PRG	262
edit_personnel	BPM_EDIT.PRG	1740
edit_revenue	BPM_EDIT.PRG	604
escape	UTILITY.PRG	485
est_factor	BPM_EDIT.PRG	334
eval_toggle	BPM_WHA2.PRG	393
expenses	BPM_REP1.PRG	33
e_capital	BPM_EDIT.PRG	2474

Routine	Program File	Line
e_capmaj	BPM_EDIT.PRG	1357
e_carry	BPM_EDIT.PRG	811
e_corecap	BPM_EDIT.PRG	2238
e_cpb_ga	BPM_WHA1.PRG	947
e_cpb_wky	BPM_WHA1.PRG	914
e_ctda	BPM_SYST.PRG	614
e_factor	BPM_EDIT.PRG	295
e_internal	BPM_EDIT.PRG	1492
e_major	BPM_EDIT.PRG	1085
e_off	BPM_SYST.PRG	221
e_opt_pers	BPM_WHA1.PRG	1288
e_org	BPM_SYST.PRG	112
e_personnel	BPM_EDIT.PRG	1799
e_per_item	BPM_EDIT.PRG	1934
e_pessimism	BPM_WHA1.PRG	216
e_product	BPM_EDIT.PRG	1616
e_revenue	BPM_EDIT.PRG	679
e_revtype	BPM_SYST.PRG	508
e_rev_item	BPM_EDIT.PRG	946
e_skill	BPM_SYST.PRG	318
e_source	BPM_SYST.PRG	414
e_support	BPM_EDIT.PRG	1221
e_tda_org	BPM_WHA2.PRG	452
e_tda_tgt	BPM_WHA2.PRG	254
e_tda_what	BPM_WHA2.PRG	103
fillfld	BROWSER.PRG	73
fillnf	BPM_REP2.PRG	55
fillpmu	UTILITY.PRG	1138
fillpu	UTILITY.PRG	469
fillpvp	UTILITY.PRG	1015
findkey	FILEHAND.PRG	221
gasubsidy	BPM_WHA1.PRG	1857
genload	BPM_ASST.PRG	322
genmsgwindow	UTILITY.PRG	665
getdatetime	FILEHAND.PRG	276
getfileid	FILEHAND.PRG	76
getfiltcond	BPM_REP2.PRG	1089
getthese	UTILITY.PRG	1051
get_array	BPM_GENL.PRG	920
get_cap	BPM_EDIT.PRG	2351
get_cost_tot	BPM_ANAL.PRG	295
get_factor	BPM_EDIT.PRG	437
get_pic	BROWSER.PRG	298
get_totals	BPM_ANAL.PRG	1616
get_wkyrs	BPM_EDIT.PRG	1984
go_corecap	BPM_EDIT.PRG	2218

Routine	Program File	Line
go_general	UTILITY.PRG	38
go_spec	UTILITY.PRG	87
help	UTILITY.PRG	1265
incr_baseyear	BPM_SYST.PRG	693
initscreen	TEXTVIEW.PRG	264
intl_diag	BPM_REPT.PRG	571
isinfile	UTILITY.PRG	140
ispath	UTILITY.PRG	1381
larray	BPM_GENL.PRG	801
leftjust	UTILITY.PRG	375
limit_overtime	BPM_SYST.PRG	1008
listcapstr	BPM_REP2.PRG	1266
list_avg_fact	BPM_REPT.PRG	1029
list_cap1	BPM_REP2.PRG	302
list_capital	BPM_EDIT.PRG	2580
list_center	BPM_REPT.PRG	71
list_core1	BPM_REP2.PRG	329
list_corecap	BPM_REPT.PRG	775
list_ctda	BPM_REP2.PRG	450
list_engine	BPM_REP2.PRG	894
list_ext1	BPM_REP2.PRG	141
list_fac1	BPM_REP2.PRG	86
list_factors	BPM_EDIT.PRG	496
list_int1	BPM_REP2.PRG	187
list_off	BPM_REP2.PRG	394
list_orgs	BPM_REP2.PRG	352
list_pers1	BPM_REP2.PRG	267
list_personnel	BPM_EDIT.PRG	2080
list_pess1	BPM_WHA1.PRG	1013
list_pessimism	BPM_WHA1.PRG	989
list_pess_rev	BPM_REP2.PRG	472
list_prod1	BPM_REP2.PRG	229
list_rates	BPM_ANAL.PRG	2002
list_rev1	BPM_REP2.PRG	105
list_revenue	BPM_EDIT.PRG	980
list_revtypes	BPM_REP2.PRG	429
list_skills	BPM_REP2.PRG	376
list_sources	BPM_REP2.PRG	412
list_tda1	BPM_WHA2.PRG	1399
list_tda2	BPM_WHA2.PRG	1432
list_tda3	BPM_WHA2.PRG	1508
list_tda4	BPM_WHA2.PRG	1553
list_tda5	BPM_WHA2.PRG	1633
list_tda_org	BPM_WHA2.PRG	1375
list_tda_wif	BPM_WHA2.PRG	1347
list_tot1	BPM_REPT.PRG	114

Routine	Program File	Line
list_totals	BPM_REPT.PRG	35
list_wifl	BPM_REP2.PRG	507
loadpsf	BPM_REP2.PRG	1205
loadrevs	BPM_WHA1.PRG	1989
load_foe_factors	BPM_EDIT.PRG	392
load_main_factors	BPM_EDIT.PRG	364
load_orgs	BPM_OPEN.PRG	483
l_cap	BPM_REPT.PRG	435
l_exp	BPM_REPT.PRG	291
l_ext	BPM_REPT.PRG	353
l_int	BPM_REPT.PRG	470
l_main	BPM_REPT.PRG	176
l_menu	UTILITY.PRG	1200
l_supt	BPM_REPT.PRG	401
l_wkyrs	BPM_REPT.PRG	521
minmax	BPM_GENL.PRG	898
move_personnel	BPM_EDIT.PRG	2040
move_revenue	BPM_EDIT.PRG	892
net_carry	BPM_REP2.PRG	1291
net_rev	BPM_REP2.PRG	1337
net_revenue	BPM_REP2.PRG	1317
nofunction	UTILITY.PRG	199
nonzero	UTILITY.PRG	244
norec	UTILITY.PRG	993
okapn	BPM_GENL.PRG	184
okass	BPM_GENL.PRG	432
okcat	BPM_GENL.PRG	219
okdc	BPM_GENL.PRG	406
okext	BPM_GENL.PRG	298
okfact	BPM_GENL.PRG	545
okfoe	BPM_GENL.PRG	667
okgrp	BPM_GENL.PRG	136
okintlab	BPM_GENL.PRG	634
oklab	BPM_GENL.PRG	609
okontda	BPM_GENL.PRG	466
okoper	BPM_REP2.PRG	1121
okorg	BPM_GENL.PRG	487
okpers	BPM_GENL.PRG	325
okpos	BPM_GENL.PRG	352
okps	BPM_GENL.PRG	270
okskl	BPM_GENL.PRG	515
oksrc	BPM_GENL.PRG	56
okwky	BPM_GENL.PRG	581
okyn	BPM_GENL.PRG	380
oneorgcalc	BPM_WHA1.PRG	1771
one_measure	BPM_WHA2.PRG	1486

Routine	Program File	Line
openfile	FILEHAND.PRG	94
openwindow	UTILITY.PRG	943
optselect	UTILITY.PRG	767
opt_calc	BPM_WHA1.PRG	1595
opt_disp	BPM_WHA1.PRG	1362
opt_fact	BPM_WHA1.PRG	1393
opt_fact1	BPM_WHA1.PRG	1416
opt_fact2	BPM_WHA1.PRG	1478
opt_list	BPM_WHA1.PRG	2150
opt_list1	BPM_WHA1.PRG	2173
opt_list2	BPM_WHA1.PRG	2207
opt_list3	BPM_WHA1.PRG	2236
opt_personnel	BPM_WHA1.PRG	1126
opt_save	BPM_WHA1.PRG	2266
opt_view	BPM_WHA1.PRG	1535
opt_view1	BPM_WHA1.PRG	1554
opt_view2	BPM_WHA1.PRG	1567
opt_view3	BPM_WHA1.PRG	1580
orblank	UTILITY.PRG	330
org_measure	BPM_WHA2.PRG	1610
org_nets	BPM_REPT.PRG	703
pad	UTILITY.PRG	392
pcttoggle	BPM_WHA1.PRG	1273
pessfilt	BPM_WHA1.PRG	181
pess_adj	BPM_WHA1.PRG	676
pess_adj_exp	BPM_WHA1.PRG	653
pess_scan	BPM_WHA1.PRG	505
pess_show	BPM_WHA1.PRG	694
popq	UTILITY.PRG	312
post_repl	BPM_EDIT.PRG	2553
post_totals	BPM_ANAL.PRG	1578
project_tda	BPM_WHA2.PRG	1683
proj_org_tda	BPM_WHA2.PRG	1784
put_array	BPM_GENL.PRG	1044
put_cap	BPM_EDIT.PRG	2376
put_factor	BPM_EDIT.PRG	461
put_totals	BPM_ANAL.PRG	1547
readbkwd	TEXTVIEW.PRG	483
readfwd	TEXTVIEW.PRG	373
readjust	BPM_REP1.PRG	1579
readlinebkwd	TEXTVIEW.PRG	447
readlinefwd	TEXTVIEW.PRG	339
readscrbkwd	TEXTVIEW.PRG	411
readscrfwd	TEXTVIEW.PRG	303
recalc_all	BPM_ANAL.PRG	899
recalc_lab	BPM_ANAL.PRG	1151

Routine	Program File	Line
recalc_tda	BPM_WHA2.PRG	784
recalc_year	BPM_WHA2.PRG	1234
recalc_yr	BPM_WHA2.PRG	1267
recomp_center	BPM_ANAL.PRG	952
recomp_foe	BPM_ANAL.PRG	1299
recomp_lab	BPM_ANAL.PRG	1180
reloaddata	BPM_WHA1.PRG	1903
removeone	BPM_SYST.PRG	920
remove_empty	BPM_SYST.PRG	823
remove_unlinked	BPM_SYST.PRG	942
rem_old_names	BPM_REP2.PRG	1241
rem_ts_cost	BPM_ANAL.PRG	1117
rerecalc_tda	BPM_WHA2.PRG	1174
retrieve_costs	BPM_REP1.PRG	375
rev_pessimism	BPM_WHA1.PRG	134
savepsf	BPM_REP2.PRG	1162
scrollbkwd	TEXTVIEW.PRG	237
scrollfile	TEXTVIEW.PRG	110
scrollfwd	TEXTVIEW.PRG	219
sel_foe	BPM_OPEN.PRG	566
sel_lab_foe	BPM_OPEN.PRG	593
setdest	UTILITY.PRG	826
set_salary	BPM_EDIT.PRG	1914
showdemomsg	BPM_OPEN.PRG	64
showeditmsg	UTILITY.PRG	346
showmemory	UTILITY.PRG	1111
showrecmsg	BPM_WHA2.PRG	433
show_arr	BPM_REP2.PRG	956
show_line_length	BPM_REP2.PRG	985
show_overall	BPM_WHA2.PRG	545
shtabisup	UTILITY.PRG	275
source_use	BPM_REP1.PRG	1048
split_chart	BPM_AMCC.PRG	231
sys_edit	BPM_SYST.PRG	30
tabisdown	UTILITY.PRG	262
tda_eval_comp	BPM_WHA2.PRG	179
tda_tot	BPM_WHA2.PRG	578
tda_tot_comp	BPM_WHA2.PRG	627
tda_whatif	BPM_WHA2.PRG	50
tech_base_ihoh	BPM_REP1.PRG	847
tempupgrade	BPM_OPEN.PRG	163
te_tda_org	BPM_WHA2.PRG	1949
te_tot	BPM_WHA2.PRG	1912
tgt_toggle	BPM_WHA2.PRG	349
tlbr	UTILITY.PRG	921
toggle_list	BPM_ASST.PRG	597

Routine	Program File	Line
total_engine	BPM_REP3.PRG	614
tot_cap1	BPM_REP3.PRG	340
tot_carry	BPM_REP3.PRG	224
tot_ctda	BPM_SYST.PRG	668
tot_exp1	BPM_REP1.PRG	104
tot_exp2	BPM_REP1.PRG	542
tot_ext1	BPM_REP3.PRG	93
tot_int1	BPM_REP3.PRG	140
tot_netrev	BPM_REP3.PRG	263
tot_pers1	BPM_REP3.PRG	302
tot_pess_rev	BPM_REP3.PRG	368
tot_rev1	BPM_REP3.PRG	53
tot_rev2	BPM_REP3.PRG	185
tsamt	BPM_REP1.PRG	1987
ts_amounts	BPM_REP1.PRG	1966
unlinked	BPM_SYST.PRG	975
upgrade	BPM_OPEN.PRG	88
upgrone	BPM_OPEN.PRG	132
user_defined	BPM_REP2.PRG	618
user_totals	BPM_REP3.PRG	443
view_capital	BPM_EDIT.PRG	2413
view_capmaj	BPM_EDIT.PRG	1296
view_center	BPM_ANAL.PRG	766
view_corecap	BPM_EDIT.PRG	2116
view_ctda	BPM_SYST.PRG	567
view_foe_factors	BPM_EDIT.PRG	68
view_internal	BPM_EDIT.PRG	1431
view_main_factors	BPM_EDIT.PRG	219
view_major	BPM_EDIT.PRG	1024
view_off	BPM_SYST.PRG	176
view_orgs	BPM_SYST.PRG	66
view_personnel	BPM_EDIT.PRG	1672
view_products	BPM_EDIT.PRG	1564
view_rates	BPM_ANAL.PRG	1884
view_revenue	BPM_EDIT.PRG	530
view_revtypes	BPM_SYST.PRG	463
view_skills	BPM_SYST.PRG	272
view_sources	BPM_SYST.PRG	369
view_support	BPM_EDIT.PRG	1160
view_tda_org	BPM_WHA2.PRG	293
view_totals	BPM_ANAL.PRG	52
warnem	UTILITY.PRG	624
whatif	BPM_WHA1.PRG	38
whatsnext	FILEHAND.PRG	195
wiforg	BPM_WHA2.PRG	1831
win_print	UTILITY.PRG	215

Routine	Program File	Line
wkyrs_adjust	BPM_WHA2.PRG	.737
writescreen	TEXTVIEW.PRG	198
wr_arr	BPM_GENL.PRG	783
wr_duls	BPM_GENL.PRG	765
wr_uls	BPM_GENL.PRG	750
wr_years	BPM_GENL.PRG	718
xbrowse	BROWSER.PRG	187
xmemo	UTILITY.PRG	1350
xseek	BROWSER.PRG	349
years	BPM_GENL.PRG	735
zaprecord	FILEHAND.PRG	471

CHAPTER 4

CROSS REFERENCE TO CALLED FUNCTIONS/PROCEDURES

The following table is a cross references between BPM functions/procedures and the functions and procedures that they call. This is an important tool for programmers who want to understand or modify the BPM program files. Only routines that are non-standard Clipper are included. This list is created a program and should be more comprehensive and more current than the documented code files.

Calling Routine	Called Routine
aaddf	aadd
aaddf	get_array
accept_rates	optselect
accept_rates	put_factor
accept_rates	warnem
addnewfile	buildfile
addnewfile	chkidx
addnewfile	optselect
addupfactors	aadd
addupfactors	wr_arr
add_rec	zaprecord
allocate_costs	aadd
allocate_costs	avdiv
allocate_costs	avmult
allocate_costs	larray
allocate_costs	nonzero
alloccomp	allocadd
alloc_cost	alloc_one
alloc_cost	alloc_wky
alloc_cost	choose
alloc_cost	e_cpb_ga
alloc_cost	e_cpb_wky
alloc_cost	fillpu
alloc_cost	pess_show
alloc_one	amult
alloc_one	asub
alloc_one	avdiv
alloc_one	calloc_one
alloc_one	closewindow
alloc_one	openwindow
alloc_one	optselect

Calling Routine	Called Routine
alloc_one	showeditmsg
alloc_one	tlbr
alloc_one	wr_arr
alloc_one	wr_uls
alloc_one	wr_years
alloc_wky	aadd
alloc_wky	asub
alloc_wky	avdiv
alloc_wky	avmult
alloc_wky	calloc_wky
alloc_wky	optselect
alloc_wky	showeditmsg
amcc1	clnupdest
amcc1	closefile
amcc1	closepage
amcc1	openfile
amcc1	setdest
amcc1	split_chart
amcc1a	clnupdest
amcc1a	closefile
amcc1a	closepage
amcc1a	openfile
amcc1a	setdest
amcc1a	split_chart
amcc1b	clnupdest
amcc1b	closefile
amcc1b	closepage
amcc1b	openfile
amcc1b	setdest
amcc1b	split_chart
amcc1c	clnupdest
amcc1c	closefile
amcc1c	closepage
amcc1c	openfile
amcc1c	setdest
amcc1c	split_chart
amcc1d	clnupdest
amcc1d	closefile
amcc1d	closepage
amcc1d	openfile
amcc1d	setdest
amcc1d	split_chart
amcc1e	clnupdest
amcc1e	closefile
amcc1e	closepage
amcc1e	openfile

Calling Routine	Called Routine
amcc1e	setdest
amcc1e	split_chart
amcc2	aadd
amcc2	aaddf
amcc2	clnupdest
amcc2	closefile
amcc2	closepage
amcc2	larray
amcc2	openfile
amcc2	pad
amcc2	setdest
amcc2	years
amcc3	aadd
amcc3	clnupdest
amcc3	closefile
amcc3	closepage
amcc3	get_wkyrs
amcc3	larray
amcc3	openfile
amcc3	setdest
amcc3	years
amcc4	aadd
amcc4	aaddf
amcc4	avdiv
amcc4	avmult
amcc4	clnupdest
amcc4	closefile
amcc4	closepage
amcc4	get_array
amcc4	openfile
amcc4	put_array
amcc4	setdest
amcc4	split_chart
amcc5	aadd
amcc5	aaddf
amcc5	clnupdest
amcc5	closefile
amcc5	closepage
amcc5	larray
amcc5	openfile
amcc5	pad
amcc5	setdest
amcc5	years
amccharts	amcc1
amccharts	amcc1a
amccharts	amcc1b

Calling Routine	Called Routine
amccharts	amcc1c
amccharts	amcc1d
amccharts	amcc1e
amccharts	amcc2
amccharts	amcc3
amccharts	amcc4
amccharts	amcc5
amccharts	amcc_ihoh
amccharts	choosemenu
amccharts	core_tot
amccharts	fillpu
amccharts	showeditmsg
amcc_ihoh	aadd
amcc_ihoh	amult
amcc_ihoh	clnupdest
amcc_ihoh	closepage
amcc_ihoh	get_cost_tot
amcc_ihoh	get_totals
amcc_ihoh	larray
amcc_ihoh	setdest
amcc_ihoh	years
anal_details	choosemenu
anal_details	disp_capital
anal_details	disp_contract
anal_details	disp_expense
anal_details	disp_external
anal_details	disp_internal
anal_details	disp_wkyrs
anal_details	escape
anal_details	fillpu
anal_details	showeditmsg
anonzero	get_array
anonzero	nonzero
assorted	choosemenu
assorted	copydata
assorted	delete_lab
assorted	del_allbut
assorted	docheck
assorted	doload
assorted	doreindex
assorted	doviewtext
assorted	escape
assorted	fillpu
assorted	remove_unlinked
assorted	toggle_list
avgload	aadd

Calling Routine	Called Routine
avgload	avdiv
avgload	avmult
avgload	cent
avgload	closefile
avgload	get_array
avgload	openfile
avgload	put_array
avgload	warnem
a_handler	help
balance1	asub
balance1	bal_part1
balance1	bal_part2
balance1	cent
balance1	clnupdest
balance1	closefile
balance1	closepage
balance1	closewindow
balance1	larray
balance1	openfile
balance1	openwindow
balance1	setdest
balance2	asub
balance2	bal_part1
balance2	bal_part2
balance2	cent
balance2	clnupdest
balance2	closefile
balance2	closepage
balance2	closewindow
balance2	larray
balance2	openfile
balance2	openwindow
balance2	setdest
balance3	asub
balance3	bal_part1
balance3	bal_part2
balance3	cent
balance3	clnupdest
balance3	closefile
balance3	closepage
balance3	closewindow
balance3	larray
balance3	openfile
balance3	openwindow
balance3	setdest
balances	balance1

Calling Routine	Called Routine
balances	balance2
balances	balance3
balances	choosemenu
balances	fillpu
balances	showeditmsg
bal_part1	compwky
bal_part1	get_wkyrs
bal_part1	larray
bal_part1	load_foe_factors
bal_part1	years
bal_part2	aadd
bal_part2	align
bal_part2	compcpb
bal_part2	get_array
bal_part2	larray
bal_part2	load_foe_factors
bal_part2	years
baseline	choosemenu
baseline	escape
baseline	fillpu
baseline	showeditmsg
baseline	view_center
baseline	view_rates
baseline	view_totals
bpm (main)	assorted
bpm (main)	baseline
bpm (main)	blimempak
bpm (main)	bliovlclr
bpm (main)	cent
bpm (main)	checklevel
bpm (main)	choose
bpm (main)	closewindow
bpm (main)	data_entry
bpm (main)	defineallfiles
bpm (main)	dopack
bpm (main)	dosvers
bpm (main)	files
bpm (main)	getthese
bpm (main)	handles
bpm (main)	load_main_factors
bpm (main)	load_orgs
bpm (main)	openwindow
bpm (main)	optselect
bpm (main)	pad
bpm (main)	showdemomsg
bpm (main)	showeditmsg

Calling Routine	Called Routine
bpm (main)	sys_edit
bpm (main)	upgrade
bpm (main)	warnem
bpm (main)	whatif
browse	afields
browse	dbedit
browse	get_pic
browse	indexkey
browse	lastrec
checklevel	warnem
checkone	closefile
checkone	openfile
chkidx	doindex
chkidx	getdatetime
chkidx	indexkey
choose	achoice
choose	alength
choose	amaxlen
choose	cent
choose	closewindow
choose	openwindow
choose	showeditmsg
choosemenu	achoice
choosemenu	amaxlen
choosemenu	blimempak
choosemenu	bliovlclr
choosemenu	cent
choosemenu	closewindow
choosemenu	openwindow
choosemenu	pad
choosemenu	showeditmsg
closewindow	restscreen
compare_wif	closewindow
compare_wif	openwindow
compare_wif	showeditmsg
compare_wif	wiforg
compass	amult
compose_line	aadd
compose_line	get_array
copydata	cent
copydata	closewindow
copydata	copyone
copydata	getthese
copydata	ispath
copydata	openwindow
copydata	optselect

Calling Routine	Called Routine
copydata	pad
copydata	sel_lab_foe
copydata	warnem
copyone	cent
copyone	closefile
copyone	openfile
corefilt	cent
corefilt	closewindow
corefilt	okapn
corefilt	okfoe
corefilt	oklab
corefilt	openwindow
corefilt	popq
corefilt	showeditmsg
corefilt	tlbr
core_dollars	align
core_dollars	cent
core_dollars	clnupdest
core_dollars	closefile
core_dollars	closepage
core_dollars	closewindow
core_dollars	corefilt
core_dollars	indexord
core_dollars	openfile
core_dollars	openwindow
core_dollars	pad
core_dollars	setdest
core_qc	cent
core_qc	clnupdest
core_qc	closefile
core_qc	closepage
core_qc	closewindow
core_qc	openfile
core_qc	openwindow
core_qc	setdest
core_tot	add_field
core_tot	align
core_tot	alloc_det
core_tot	asub
core_tot	cent
core_tot	clnupdest
core_tot	closefile
core_tot	closepage
core_tot	closewindow
core_tot	ctreport
core_tot	dircust

Calling Routine	Called Routine
core_tot	getthese
core_tot	get_cap
core_tot	get_totals
core_tot	indexord
core_tot	openfile
core_tot	openwindow
core_tot	pad
core_tot	setdest
ctreport	cent
ctreport	closefile
ctreport	closepage
ctreport	openfile
ctreport	pad
ctr_toggle	choose
ctr_toggle	disp_foe_guts
ctr_toggle	pad
ctr_toggle	showeditmsg
data_entry	choosemenu
data_entry	escape
data_entry	fillpu
data_entry	showeditmsg
data_entry	view_capital
data_entry	view_corecap
data_entry	view_foe_factors
data_entry	view_personnel
data_entry	view_revenue
defineallfiles	addnewfile
defineallfiles	wiforg
deleteone	cent
deleteone	closefile
deleteone	openfile
delete_lab	cent
delete_lab	closewindow
delete_lab	deleteone
delete_lab	openwindow
delete_lab	optselect
delete_lab	sel_lab_foe
del_allbut	cent
del_allbut	closewindow
del_allbut	deleteone
del_allbut	openwindow
del_allbut	optselect
del_allbut	sel_lab_foe
del_personnel	del_rec
del_personnel	optselect
del_personnel	whatsnext

Calling Routine	Called Routine
del_rec	indexkey
del_record	del_rec
del_record	optselect
del_record	whatsnext
del_revenue	closefile
del_revenue	del_rec
del_revenue	del_record
del_revenue	openfile
del_revenue	optselect
del_revenue	whatsnext
dirorgcalc	arsum
dirorgcalc	arvadd
disp_capital	aadd
disp_capital	asub
disp_capital	closewindow
disp_capital	get_totals
disp_capital	openwindow
disp_capital	pad
disp_capital	showeditmsg
disp_capital	tlbr
disp_capital	wr_arr
disp_capital	wr_duls
disp_capital	wr_uls
disp_capital	wr_years
disp_contract	asub
disp_contract	closewindow
disp_contract	get_totals
disp_contract	openwindow
disp_contract	pad
disp_contract	showeditmsg
disp_contract	tlbr
disp_contract	wr_arr
disp_contract	wr_duls
disp_contract	wr_uls
disp_contract	wr_years
disp_expense	closewindow
disp_expense	get_totals
disp_expense	openwindow
disp_expense	pad
disp_expense	showeditmsg
disp_expense	tlbr
disp_expense	wr_arr
disp_expense	wr_duls
disp_expense	wr_uls
disp_expense	wr_years
disp_external	aadd

Calling Routine	Called Routine
disp_external	closewindow
disp_external	get_cost_tot
disp_external	get_totals
disp_external	openwindow
disp_external	pad
disp_external	showeditmsg
disp_external	tlbr
disp_external	wr_arr
disp_external	wr_duls
disp_external	wr_uls
disp_external	wr_years
disp_foe	anal_details
disp_foe	closefile
disp_foe	closewindow
disp_foe	disp_foe_guts
disp_foe	doit
disp_foe	dotoggle
disp_foe	fillpmu
disp_foe	list_totals
disp_foe	openfile
disp_foe	openwindow
disp_foe	recalc_lab
disp_foe	showeditmsg
disp_foe	tlbr
disp_foe	view_foe_factors
disp_foe_guts	aadd
disp_foe_guts	asub
disp_foe_guts	get_cost_tot
disp_foe_guts	get_totals
disp_foe_guts	pad
disp_foe_guts	wr_arr
disp_foe_guts	wr_duls
disp_foe_guts	wr_uls
disp_foe_guts	wr_years
disp_internal	asub
disp_internal	closewindow
disp_internal	get_totals
disp_internal	openwindow
disp_internal	pad
disp_internal	showeditmsg
disp_internal	tlbr
disp_internal	wr_arr
disp_internal	wr_duls
disp_internal	wr_uls
disp_internal	wr_years
disp_rates	aadd

Calling Routine	Called Routine
disp_rates	asub
disp_rates	avdiv
disp_rates	avmult
disp_rates	get_cost_tot
disp_rates	get_totals
disp_rates	wr_arr
disp_rates	wr_duls
disp_rates	wr_uls
disp_rates	wr_years
disp_wkyrs	aadd
disp_wkyrs	closewindow
disp_wkyrs	get_totals
disp_wkyrs	openwindow
disp_wkyrs	pad
disp_wkyrs	showeditmsg
disp_wkyrs	tlbr
disp_wkyrs	wr_arr
disp_wkyrs	wr_duls
disp_wkyrs	wr_uls
disp_wkyrs	wr_years
dobrowse	browse
dobrowse	cent
dobrowse	closewindow
dobrowse	get_set
dobrowse	openwindow
dobrowse	showeditmsg
docheck	cent
docheck	checkone
docheck	clnupdest
docheck	closefile
docheck	closepage
docheck	closewindow
docheck	openfile
docheck	openwindow
docheck	optselect
docheck	setdest
doindex	cent
doindex	closewindow
doindex	openwindow
doit	doskip
doit	l_menu
doit	restscreen
dolicense	cent
dolicense	closewindow
dolicense	openwindow
doload	aadd

Calling Routine	Called Routine
doload	amult
doload	asub
doload	avgload
doload	cent
doload	closefile
doload	closewindow
doload	genload
doload	getthese
doload	get_wkyrs
doload	ispath
doload	openfile
doload	openwindow
doload	optselect
doload	pad
doload	sel_lab_foe
doload	warnem
dopack	cent
dopack	closefile
dopack	closewindow
dopack	openfile
dopack	openwindow
dopack	optselect
dopessimism	aadd
dopessimism	alloc_cost
dopessimism	asub
dopessimism	avdiv
dopessimism	closefile
dopessimism	closewindow
dopessimism	doit
dopessimism	fillpmu
dopessimism	get_cost_tot
dopessimism	get_totals
dopessimism	list_pessimism
dopessimism	openfile
dopessimism	openwindow
dopessimism	optselect
dopessimism	pess_scan
dopessimism	pess_show
dopessimism	showeditmsg
dopessimism	tlbr
dopessimism	wr_arr
dopessimism	wr_uls
dopessimism	wr_years
doproj_org_tda	cent
doproj_org_tda	closewindow
doproj_org_tda	openwindow

Calling Routine	Called Routine
doproj_org_tda	optselect
doproj_org_tda	proj_org_tda
doreindex	cent
doreindex	closefile
doreindex	closewindow
doreindex	openfile
doreindex	openwindow
doreindex	optselect
doskip	if
doskip	indexkey
dosvers	cdosvers
dotoggle	disp_foe_guts
doviewtext	asort
doviewtext	choose
doviewtext	optselect
doviewtext	pad
doviewtext	scrollfile
doviewtext	showeditmsg
doviewtext	warnem
edit_baseyear	cent
edit_baseyear	closefile
edit_baseyear	closewindow
edit_baseyear	getthese
edit_baseyear	openfile
edit_baseyear	openwindow
edit_foe_factors	addupfactors
edit_foe_factors	choose
edit_foe_factors	e_factor
edit_foe_factors	fillpu
edit_foe_factors	warnem
edit_main_factors	choose
edit_main_factors	e_factor
edit_main_factors	fillpu
edit_personnel	choose
edit_personnel	e_personnel
edit_personnel	e_per_item
edit_personnel	fillpu
edit_personnel	get_wkyrs
edit_personnel	showeditmsg
edit_personnel	warnem
edit_revenue	choose
edit_revenue	e_carry
edit_revenue	e_revenue
edit_revenue	e_rev_item
edit_revenue	fillpu
edit_revenue	get_array

Calling Routine	Called Routine
edit_revenue	showeditmsg
edit_revenue	view_capmaj
edit_revenue	view_internal
edit_revenue	view_major
edit_revenue	view_products
edit_revenue	view_support
edit_revenue	warnem
est_factor	fillpvp
est_factor	getthese
est_factor	warnem
eval_toggle	choose
eval_toggle	optselect
eval_toggle	recalc_year
eval_toggle	showeditmsg
eval_toggle	showrecmsg
expenses	choosemenu
expenses	fillpu
expenses	showeditmsg
expenses	source_use
expenses	tech_base_ihoh
expenses	tot_exp1
expenses	tot_exp2
e_capital	add_rec
e_capital	compass
e_capital	get_array
e_capital	isinfile
e_capital	minmax
e_capital	okass
e_capital	orblank
e_capital	put_array
e_capital	showeditmsg
e_capital	wr_duls
e_capital	wr_uls
e_capital	wr_years
e_capmaj	add_rec
e_capmaj	get_array
e_capmaj	minmax
e_capmaj	okass
e_capmaj	put_array
e_capmaj	showeditmsg
e_capmaj	wr_uls
e_capmaj	wr_years
e_carry	compcpb
e_carry	e_rev_item
e_corecap	cctot
e_corecap	get_cap

Calling Routine	Called Routine
e_corecap	put_cap
e_corecap	showeditmsg
e_corecap	warnem
e_cpb_ga	avdiv
e_cpb_ga	closewindow
e_cpb_ga	openwindow
e_cpb_ga	showeditmsg
e_cpb_ga	tlbr
e_cpb_ga	warnem
e_cpb_ga	wr_uls
e_cpb_ga	wr_years
e_cpb_wky	closewindow
e_cpb_wky	openwindow
e_cpb_wky	showeditmsg
e_cpb_wky	tlbr
e_cpb_wky	wr_uls
e_cpb_wky	wr_years
e_ctda	add_rec
e_ctda	showeditmsg
e_ctda	tot_ctda
e_factor	est_factor
e_factor	minmax
e_factor	optselect
e_factor	put_factor
e_factor	showeditmsg
e_internal	add_rec
e_internal	get_array
e_internal	isinfile
e_internal	minmax
e_internal	okintlab
e_internal	orblank
e_internal	put_array
e_internal	showeditmsg
e_internal	wr_uls
e_internal	wr_years
e_major	add_rec
e_major	get_array
e_major	minmax
e_major	okgrp
e_major	oksrc
e_major	put_array
e_major	showeditmsg
e_major	wr_uls
e_major	wr_years
e_off	add_rec
e_off	oklab

Calling Routine	Called Routine
e_off	showeditmsg
e_opt_pers	arvadd
e_opt_pers	opt_calc
e_opt_pers	opt_disp
e_opt_pers	reloaddata
e_opt_pers	showeditmsg
e_org	add_rec
e_org	okontda
e_org	okorg
e_org	showeditmsg
e_personnel	add_rec
e_personnel	align
e_personnel	blank_if_cont
e_personnel	compwky
e_personnel	findkey
e_personnel	get_wkyrs
e_personnel	isinfile
e_personnel	okpers
e_personnel	okpos
e_personnel	okps
e_personnel	orblank
e_personnel	set_salary
e_personnel	showeditmsg
e_personnel	wr_arr
e_personnel	wr_duls
e_personnel	wr_uls
e_personnel	wr_years
e_per_item	add_rec
e_per_item	compwky
e_per_item	del_rec
e_per_item	minmax
e_per_item	nonzero
e_per_item	put_array
e_per_item	showeditmsg
e_pessimism	add_rec
e_pessimism	amult
e_pessimism	compchg
e_pessimism	get_array
e_pessimism	isinfile
e_pessimism	okapn
e_pessimism	okcat
e_pessimism	okfoe
e_pessimism	okgrp
e_pessimism	oklab
e_pessimism	oksrc
e_pessimism	popq

Calling Routine	Called Routine
e_pessimism	put_array
e_pessimism	showeditmsg
e_pessimism	wr_uls
e_pessimism	wr_years
e_product	add_rec
e_product	showeditmsg
e_revenue	add_rec
e_revenue	closefile
e_revenue	compcpb
e_revenue	findkey
e_revenue	get_array
e_revenue	isinfile
e_revenue	okapn
e_revenue	okcat
e_revenue	okgrp
e_revenue	oksrc
e_revenue	openfile
e_revenue	optselect
e_revenue	orblank
e_revenue	showeditmsg
e_revenue	wr_arr
e_revenue	wr_duls
e_revenue	wr_uls
e_revenue	wr_years
e_revtype	add_rec
e_revtype	okapn
e_revtype	okcat
e_revtype	showeditmsg
e_rev_item	compcpb
e_rev_item	minmax
e_rev_item	put_array
e_rev_item	showeditmsg
e_skill	add_rec
e_skill	okskl
e_skill	showeditmsg
e_source	add_rec
e_source	okgrp
e_source	showeditmsg
e_support	add_rec
e_support	get_array
e_support	minmax
e_support	oksrc
e_support	put_array
e_support	showeditmsg
e_support	wr_uls
e_support	wr_years

Calling Routine	Called Routine
e_tda_org	showeditmsg
e_tda_org	show_overall
e_tda_org	tda_tot
e_tda_tgt	fillpvp
e_tda_tgt	getthese
e_tda_what	recalc_year
e_tda_what	showeditmsg
e_tda_what	tda_eval_comp
fillfld	ltrim
findkey	indexkey
findkey	warnem
gasubsidy	ardist
gasubsidy	ardiv
gasubsidy	arvadd
genload	cent
genload	closefile
genload	openfile
genload	warnem
genmsgwindow	askok
genmsgwindow	cent
genmsgwindow	closewindow
genmsgwindow	openwindow
getfiltcond	cent
getfiltcond	closewindow
getfiltcond	okoper
getfiltcond	openwindow
getthese	amaxlen
getthese	cent
getthese	closewindow
getthese	get_set
getthese	openwindow
getthese	pad
get_cost_tot	aadd
get_cost_tot	get_totals
get_factor	get_array
get_factor	openfile
get_pic	ltrim
get_totals	amult
get_totals	closefile
get_totals	get_array
get_totals	openfile
get_wkyrs	get_array
go_corecap	choose
go_corecap	showeditmsg
go_general	cent
go_general	closewindow

Calling Routine	Called Routine
go_general	dbedit
go_general	openwindow
go_general	showeditmsg
go_general	tlbr
go_general	warnem
go_spec	choose
go_spec	indexkey
go_spec	showeditmsg
go_spec	warnem
help	cent
help	closefile
help	closewindow
help	get_set
help	openfile
help	openwindow
help	pad
incr_baseyear	cent
incr_baseyear	closefile
incr_baseyear	closewindow
incr_baseyear	openfile
incr_baseyear	openwindow
incr_baseyear	optselect
incr_baseyear	warnem
initscreen	readfwd
intl_diag	aadd
intl_diag	clnupdest
intl_diag	closefile
intl_diag	closepage
intl_diag	get_array
intl_diag	larray
intl_diag	l_int
intl_diag	openfile
intl_diag	pad
intl_diag	setdest
intl_diag	years
isinfile	choose
isinfile	closefile
isinfile	go_general
isinfile	indexkey
isinfile	openfile
isinfile	showeditmsg
isinfile	warnem
ispath	pad
ispath	warnem
larray	amult
limit_overtime	getthese

Calling Routine	Called Routine
listcapstr	aadd
listcapstr	get_array
list_avg_fact	aadd
list_avg_fact	amult
list_avg_fact	avdiv
list_avg_fact	avmult
list_avg_fact	cent
list_avg_fact	clnupdest
list_avg_fact	closefile
list_avg_fact	closepage
list_avg_fact	closewindow
list_avg_fact	get_wkyrs
list_avg_fact	larray
list_avg_fact	load_foe_factors
list_avg_fact	openfile
list_avg_fact	openwindow
list_avg_fact	optselect
list_avg_fact	sel_lab_foe
list_avg_fact	setdest
list_avg_fact	years
list_cap1	fillnf
list_cap1	okass
list_cap1	okfoe
list_cap1	oklab
list_cap1	user_defined
list_capital	choosemenu
list_capital	doviewtext
list_capital	fillpu
list_capital	list_cap1
list_capital	showeditmsg
list_capital	tot_cap1
list_capital	win_print
list_center	amccharts
list_center	balances
list_center	choosemenu
list_center	doviewtext
list_center	expenses
list_center	fillpu
list_center	intl_diag
list_center	list_tot1
list_center	org_nets
list_center	pad
list_center	showeditmsg
list_center	tot_cap1
list_center	tot_ext1
list_center	tot_int1

Calling Routine	Called Routine
list_center	tot_pers1
list_center	tot_rev1
list_center	tot_rev2
list_center	ts_amounts
list_center	win_print
list_core1	fillnf
list_core1	okapn
list_core1	okfoe
list_core1	oklab
list_core1	user_defined
list_corecap	choosemenu
list_corecap	core_dollars
list_corecap	core_qc
list_corecap	doviewtext
list_corecap	fillpu
list_corecap	list_core1
list_corecap	showeditmsg
list_corecap	win_print
list_ctda	fillnf
list_ctda	okfoe
list_ctda	oklab
list_ctda	user_defined
list_engine	cent
list_engine	clnupdest
list_engine	closefile
list_engine	closepage
list_engine	closewindow
list_engine	indexord
list_engine	openfile
list_engine	openwindow
list_engine	setdest
list_engine	time
list_ext1	closefile
list_ext1	dircust
list_ext1	fillnf
list_ext1	okapn
list_ext1	okass
list_ext1	okcat
list_ext1	okdc
list_ext1	okext
list_ext1	okfoe
list_ext1	okgrp
list_ext1	oklab
list_ext1	oksrc
list_ext1	openfile
list_ext1	user_defined

Calling Routine	Called Routine
list_fac1	fillnf
list_fac1	okfact
list_fac1	okfoe
list_fac1	oklab
list_fac1	user_defined
list_factors	choosemenu
list_factors	doviewtext
list_factors	fillpu
list_factors	list_avg_fact
list_factors	list_fac1
list_factors	showeditmsg
list_factors	win_print
list_int1	closefile
list_int1	dircust
list_int1	fillnf
list_int1	okapn
list_int1	okcat
list_int1	okdc
list_int1	okfoe
list_int1	okgrp
list_int1	oklab
list_int1	oksrc
list_int1	openfile
list_int1	user_defined
list_off	fillnf
list_off	oklab
list_off	user_defined
list_orgs	fillnf
list_orgs	okfoe
list_orgs	oklab
list_orgs	okorg
list_orgs	okyn
list_orgs	user_defined
list_pers1	fillnf
list_pers1	isinfile
list_pers1	okfoe
list_pers1	oklab
list_pers1	okpers
list_pers1	okpos
list_pers1	okps
list_pers1	okskl
list_pers1	okwky
list_pers1	user_defined
list_personnel	choosemenu
list_personnel	doviewtext
list_personnel	fillpu

Calling Routine	Called Routine
list_personnel	list_pers1
list_personnel	showeditmsg
list_personnel	tot_pers1
list_personnel	win_print
list_pess1	aadd
list_pess1	asub
list_pess1	avmult
list_pess1	clnupdest
list_pess1	closepage
list_pess1	larray
list_pess1	pad
list_pess1	setdest
list_pess1	years
list_pessimism	choosemenu
list_pessimism	doviewtext
list_pessimism	fillpu
list_pessimism	list_pess1
list_pessimism	list_pess_rev
list_pessimism	showeditmsg
list_pessimism	tot_pess_rev
list_pessimism	win_print
list_pess_rev	fillnf
list_pess_rev	okapn
list_pess_rev	okcat
list_pess_rev	okfoe
list_pess_rev	okgrp
list_pess_rev	oklab
list_pess_rev	oksrc
list_pess_rev	user_defined
list_prod1	closefile
list_prod1	dircust
list_prod1	fillnf
list_prod1	okapn
list_prod1	okcat
list_prod1	okdc
list_prod1	okfoe
list_prod1	okgrp
list_prod1	oklab
list_prod1	oksrc
list_prod1	openfile
list_prod1	user_defined
list_rates	choosemenu
list_rates	doviewtext
list_rates	fillpu
list_rates	showeditmsg
list_rates	win_print

Calling Routine	Called Routine
list_rev1	dircust
list_rev1	fillnf
list_rev1	okapn
list_rev1	okcat
list_rev1	okdc
list_rev1	okfoe
list_rev1	okgrp
list_rev1	oklab
list_rev1	oksrc
list_rev1	user_defined
list_revenue	choosemenu
list_revenue	doviewtext
list_revenue	fillpu
list_revenue	list_ext1
list_revenue	list_int1
list_revenue	list_prod1
list_revenue	list_rev1
list_revenue	showeditmsg
list_revenue	tot_carry
list_revenue	tot_ext1
list_revenue	tot_int1
list_revenue	tot_netrev
list_revenue	tot_rev1
list_revenue	tot_rev2
list_revenue	win_print
list_revtypes	fillnf
list_revtypes	okapn
list_revtypes	okcat
list_revtypes	user_defined
list_skills	fillnf
list_skills	isinfile
list_skills	okskl
list_skills	user_defined
list_sources	fillnf
list_sources	okgrp
list_sources	oksrc
list_sources	user_defined
list_tda1	clnupdest
list_tda1	closepage
list_tda1	openfile
list_tda1	setdest
list_tda1	wiforg
list_tda2	clnupdest
list_tda2	closepage
list_tda2	one_measure
list_tda2	setdest

Calling Routine	Called Routine
list_tda3	clnupdest
list_tda3	closepage
list_tda3	openfile
list_tda3	setdest
list_tda3	wiforg
list_tda4	clnupdest
list_tda4	closepage
list_tda4	org_measure
list_tda4	setdest
list_tda5	clnupdest
list_tda5	closepage
list_tda5	openfile
list_tda5	setdest
list_tda5	wiforg
list_tda_org	choosemenu
list_tda_org	doviewtext
list_tda_org	fillpu
list_tda_org	list_tda4
list_tda_org	list_wif1
list_tda_org	showeditmsg
list_tda_org	win_print
list_tda_wif	choosemenu
list_tda_wif	doviewtext
list_tda_wif	fillpu
list_tda_wif	list_tda1
list_tda_wif	list_tda2
list_tda_wif	list_tda3
list_tda_wif	list_tda5
list_tda_wif	list_wif1
list_tda_wif	showeditmsg
list_tda_wif	win_print
list_tot1	cent
list_tot1	clnupdest
list_tot1	closepage
list_tot1	closewindow
list_tot1	l_cap
list_tot1	l_exp
list_tot1	l_ext
list_tot1	l_int
list_tot1	l_main
list_tot1	l_supt
list_tot1	l_wkyrs
list_tot1	openwindow
list_tot1	pad
list_tot1	setdest
list_totals	choosemenu

Calling Routine	Called Routine
list_totals	doviewtext
list_totals	fillpu
list_totals	intl_diag
list_totals	list_totl
list_totals	showeditmsg
list_totals	tot_capl
list_totals	tot_extl
list_totals	tot_intl
list_totals	tot_persl
list_totals	tot_revl
list_totals	tot_rev2
list_totals	win_print
list_wifl	closefile
list_wifl	fillnf
list_wifl	okfoe
list_wifl	oklab
list_wifl	openfile
list_wifl	user_defined
loadrevs	arsum
loadrevs	arvadd
loadrevs	pad
load_foe_factors	closefile
load_foe_factors	get_factor
load_foe_factors	openfile
load_foe_factors	pad
load_main_factors	amult
load_main_factors	closefile
load_main_factors	get_factor
load_main_factors	openfile
load_orgs	closefile
load_orgs	findkey
load_orgs	openfile
l_cap	aadd
l_cap	asub
l_cap	get_totals
l_cap	larray
l_cap	years
l_exp	get_totals
l_exp	larray
l_exp	years
l_ext	aadd
l_ext	amult
l_ext	get_cost_tot
l_ext	get_totals
l_ext	larray
l_ext	years

Calling Routine	Called Routine
l_int	asub
l_int	get_totals
l_int	larray
l_int	years
l_main	aadd
l_main	amult
l_main	asub
l_main	get_cost_tot
l_main	get_totals
l_main	larray
l_main	pad
l_main	years
l_menu	pad
l_supt	asub
l_supt	get_totals
l_supt	larray
l_supt	years
l_wkyrs	aadd
l_wkyrs	get_totals
l_wkyrs	larray
l_wkyrs	years
move_personnel	fillpvp
move_personnel	findkey
move_personnel	getthese
move_personnel	isinfile
move_personnel	oklab
move_personnel	whatsnext
move_revenue	closefile
move_revenue	fillpvp
move_revenue	findkey
move_revenue	getthese
move_revenue	isinfile
move_revenue	oklab
move_revenue	openfile
move_revenue	whatsnext
net_carry	aadd
net_carry	asub
net_carry	get_array
net_rev	aadd
net_rev	asub
net_rev	get_array
net_revenue	aadd
net_revenue	net_rev
nofunction	warnem
norec	warnem
okapn	choose

Calling Routine	Called Routine
okapn	showeditmsg
okass	choose
okass	showeditmsg
okcat	choose
okcat	showeditmsg
okdc	choose
okdc	showeditmsg
okext	choose
okext	showeditmsg
okfact	choose
okfact	showeditmsg
okfoe	choose
okfoe	showeditmsg
okgrp	choose
okgrp	showeditmsg
okintlab	choose
okintlab	showeditmsg
oklab	choose
oklab	showeditmsg
okontda	warnem
okoper	choose
okoper	showeditmsg
okorg	choose
okorg	showeditmsg
okpers	choose
okpers	showeditmsg
okpos	choose
okpos	showeditmsg
okps	choose
okps	showeditmsg
okskl	choose
okskl	showeditmsg
oksrc	choose
oksrc	closefile
oksrc	openfile
oksrc	pad
oksrc	showeditmsg
oksrc	warnem
okwky	choose
okwky	showeditmsg
okyn	choose
okyn	showeditmsg
one_measure	pad
openfile	findkey
openfile	getfileid
openwindow	shadowwin

Calling Routine	Called Routine
optselect	cent
optselect	closewindow
optselect	openwindow
opt_calc	ardist
opt_calc	ardiv
opt_calc	arneg
opt_calc	arsum
opt_calc	arvadd
opt_calc	arvsub
opt_calc	dirorgcalc
opt_calc	e_opt_pers
opt_calc	gasubsidy
opt_calc	oneorgcalc
opt_calc	warnem
opt_disp	pad
opt_fact	choosemenu
opt_fact	fillpu
opt_fact	opt_calc
opt_fact	opt_fact1
opt_fact	opt_fact2
opt_fact	reloaddata
opt_fact	showeditmsg
opt_fact1	closewindow
opt_fact1	openwindow
opt_fact1	showeditmsg
opt_fact1	tlbr
opt_fact2	closewindow
opt_fact2	openwindow
opt_fact2	showeditmsg
opt_fact2	tlbr
opt_list	choosemenu
opt_list	doviewtext
opt_list	fillpu
opt_list	list_wif1
opt_list	opt_list1
opt_list	opt_list2
opt_list	opt_list3
opt_list	showeditmsg
opt_list	win_print
opt_list1	clnupdest
opt_list1	closepage
opt_list1	setdest
opt_list2	clnupdest
opt_list2	closepage
opt_list2	setdest
opt_list3	clnupdest

Calling Routine	Called Routine
opt_list3	closepage
opt_list3	setdest
opt_personnel	cent
opt_personnel	closefile
opt_personnel	closewindow
opt_personnel	doit
opt_personnel	e_opt_pers
opt_personnel	fillpmu
opt_personnel	go_general
opt_personnel	openfile
opt_personnel	openwindow
opt_personnel	opt_fact
opt_personnel	opt_list
opt_personnel	opt_save
opt_personnel	opt_view
opt_personnel	pcttoggle
opt_personnel	recalc_tda
opt_personnel	showeditmsg
opt_personnel	tlbr
opt_save	cent
opt_save	closewindow
opt_save	openwindow
opt_save	optselect
opt_save	recalc_year
opt_view	choosemenu
opt_view	fillpu
opt_view	opt_view1
opt_view	opt_view2
opt_view	opt_view3
opt_view	showeditmsg
opt_view1	choose
opt_view2	choose
opt_view3	choose
org_measure	pad
org_measure	wiforg
org_nets	aadd
org_nets	asub
org_nets	cent
org_nets	clnupdest
org_nets	closefile
org_nets	closepage
org_nets	closewindow
org_nets	get_totals
org_nets	larray
org_nets	openfile
org_nets	openwindow

Calling Routine	Called Routine
org_nets	pad
org_nets	setdest
org_nets	years
pessfilt	optselect
pessfilt	sel_lab_foe
pess_adj	avmult
pess_adj	get_array
pess_adj	put_array
pess_adj_exp	get_array
pess_adj_exp	put_array
pess_scan	aadd
pess_scan	aaddf
pess_scan	amult
pess_scan	asub
pess_scan	avmult
pess_scan	cent
pess_scan	closewindow
pess_scan	get_array
pess_scan	openwindow
pess_scan	optselect
pess_scan	pess_adj
pess_scan	pess_adj_exp
pess_scan	put_array
pess_show	aadd
pess_show	asub
pess_show	avmult
pess_show	wr_arr
pess_show	wr_uls
post_repl	compass
post_repl	get_array
post_repl	minmax
post_repl	optselect
post_repl	put_array
post_totals	amult
post_totals	closefile
post_totals	minmax
post_totals	openfile
post_totals	pad
post_totals	put_array
post_totals	put_totals
project_tda	cent
project_tda	closewindow
project_tda	openfile
project_tda	openwindow
project_tda	optselect
project_tda	proj_org_tda

Calling Routine	Called Routine
project_tda	recalc_year
project_tda	wiforg
proj_org_tda	recalc_yr
proj_org_tda	tda_tot_comp
proj_org_tda	warnem
proj_org_tda	wiforg
put_cap	add_rec
put_cap	del_rec
put_factor	add_rec
put_factor	closefile
put_factor	del_rec
put_factor	nonzero
put_factor	openfile
put_factor	pad
put_factor	put_array
put_totals	amult
put_totals	minmax
put_totals	nonzero
put_totals	put_array
readlinebkwd	readbkwd
readlinefwd	readfwd
readscrbkwd	readlinebkwd
readscrfwd	readlinefwd
recalc_all	closefile
recalc_all	disp_foe_guts
recalc_all	openfile
recalc_all	optselect
recalc_all	recomp_center
recalc_all	recomp_lab
recalc_lab	disp_foe_guts
recalc_lab	optselect
recalc_lab	recomp_lab
recalc_tda	aadd
recalc_tda	aaddf
recalc_tda	amult
recalc_tda	avdiv
recalc_tda	avmult
recalc_tda	blimempak
recalc_tda	bliovlclr
recalc_tda	cent
recalc_tda	closefile
recalc_tda	closewindow
recalc_tda	get_array
recalc_tda	get_cost_tot
recalc_tda	get_wkyrs
recalc_tda	load_foe_factors

Calling Routine	Called Routine
recalc_tda	net_rev
recalc_tda	openfile
recalc_tda	openwindow
recalc_tda	optselect
recalc_tda	recalc_year
recalc_tda	rerecalc_tda
recalc_year	cent
recalc_year	closewindow
recalc_year	openwindow
recalc_year	recalc_yr
recalc_year	tda_tot_comp
recalc_yr	tda_eval_comp
recalc_yr	tda_tot_comp
recomp_center	aaddf
recomp_center	cent
recomp_center	closefile
recomp_center	closewindow
recomp_center	openfile
recomp_center	openwindow
recomp_center	post_totals
recomp_center	rem_ts_cost
recomp_foe	aadd
recomp_foe	aaddf
recomp_foe	amult
recomp_foe	asub
recomp_foe	avdiv
recomp_foe	compcpb
recomp_foe	compwky
recomp_foe	get_array
recomp_foe	get_wkyrs
recomp_foe	load_foe_factors
recomp_foe	minmax
recomp_foe	put_factor
recomp_foe	put_totals
recomp_lab	aaddf
recomp_lab	asub
recomp_lab	cent
recomp_lab	closefile
recomp_lab	closewindow
recomp_lab	get_totals
recomp_lab	openfile
recomp_lab	openwindow
recomp_lab	post_totals
recomp_lab	put_totals
recomp_lab	recomp_foe
reloaddata	arsum

Calling Routine	Called Routine
reloaddata	closefile
reloaddata	findkey
reloaddata	get_cost_tot
reloaddata	loadrevs
reloaddata	openfile
removeone	cent
removeone	closefile
removeone	openfile
remove_empty	anonzero
remove_empty	cent
remove_empty	closefile
remove_empty	closewindow
remove_empty	openfile
remove_empty	openwindow
remove_empty	optselect
remove_empty	removeone
remove_unlinked	cent
remove_unlinked	closefile
remove_unlinked	closewindow
remove_unlinked	openfile
remove_unlinked	openwindow
remove_unlinked	optselect
remove_unlinked	unlinked
rem_ts_cost	aadd
rem_ts_cost	asub
rem_ts_cost	get_totals
rem_ts_cost	pad
rem_ts_cost	post_totals
rerecalc_tda	cent
rerecalc_tda	pad
retrieve_costs	aadd
retrieve_costs	amult
retrieve_costs	get_cost_tot
retrieve_costs	get_totals
retrieve_costs	pad
rev_pessimism	cent
rev_pessimism	closefile
rev_pessimism	closewindow
rev_pessimism	del_record
rev_pessimism	doit
rev_pessimism	dopessimism
rev_pessimism	e_pessimism
rev_pessimism	fillpmu
rev_pessimism	go_general
rev_pessimism	norec
rev_pessimism	openfile

Calling Routine	Called Routine
rev_pessimism	openwindow
rev_pessimism	pessfilt
rev_pessimism	showeditmsg
rev_pessimism	tlbr
scrollfile	initscreen
scrollfile	readlinebkwd
scrollfile	readlinefwd
scrollfile	readscrbkwd
scrollfile	readscrfwd
scrollfile	restscreen
scrollfile	scrollbkwd
scrollfile	scrollfwd
scrollfile	writescreen
sel_foe	choose
sel_lab_foe	choose
setdest	getthese
setdest	optselect
setdest	warnem
showdemomsg	genmsgwindow
showeditmsg	leftjust
showmemory	blimempak
showmemory	bliovlclr
showmemory	cent
showmemory	closewindow
showmemory	openwindow
show_arr	pad
show_overall	recalc_yr
source_use	allocadd
source_use	alloccomp
source_use	allocsu
source_use	cent
source_use	clnupdest
source_use	closefile
source_use	closepage
source_use	closewindow
source_use	compcpb
source_use	dircust
source_use	fillpvp
source_use	getthese
source_use	get_array
source_use	load_foe_factors
source_use	openfile
source_use	openwindow
source_use	optselect
source_use	readjust
source_use	retrieve_costs

Calling Routine	Called Routine
source_use	setdest
split_chart	aadd
split_chart	aaddf
split_chart	getthese
split_chart	get_array
split_chart	larray
split_chart	pad
split_chart	years
sys_edit	choosemenu
sys_edit	edit_baseyear
sys_edit	escape
sys_edit	fillpu
sys_edit	incr_baseyear
sys_edit	limit_overtime
sys_edit	remove_empty
sys_edit	showeditmsg
sys_edit	view_ctda
sys_edit	view_main_factors
sys_edit	view_off
sys_edit	view_orgs
sys_edit	view_revtypes
sys_edit	view_skills
sys_edit	view_sources
tda_tot	tda_tot_comp
tda_tot_comp	wkyrs_adjust
tda_whatif	cent
tda_whatif	closefile
tda_whatif	closewindow
tda_whatif	doit
tda_whatif	eval_toggle
tda_whatif	e_tda_tgt
tda_whatif	e_tda_what
tda_whatif	fillpmu
tda_whatif	go_general
tda_whatif	list_tda_wif
tda_whatif	norec
tda_whatif	openfile
tda_whatif	openwindow
tda_whatif	project_tda
tda_whatif	recalc_tda
tda_whatif	showeditmsg
tda_whatif	showrecmsg
tda_whatif	tlbr
tda_whatif	view_tda_org
tech_base_ihoh	aadd
tech_base_ihoh	allocate_costs

Calling Routine	Called Routine
tech_base_ihoh	avdiv
tech_base_ihoh	cent
tech_base_ihoh	clnupdest
tech_base_ihoh	closefile
tech_base_ihoh	closepage
tech_base_ihoh	closewindow
tech_base_ihoh	compcpb
tech_base_ihoh	get_array
tech_base_ihoh	larray
tech_base_ihoh	load_foe_factors
tech_base_ihoh	openfile
tech_base_ihoh	openwindow
tech_base_ihoh	retrieve_costs
tech_base_ihoh	setdest
tech_base_ihoh	years
tempupgrade	cent
tempupgrade	closewindow
tempupgrade	openwindow
te_tda_org	achoice
te_tda_org	closewindow
te_tda_org	openwindow
te_tda_org	optselect
te_tda_org	recalc_yr
te_tda_org	showeditmsg
te_tda_org	showrecmsg
te_tda_org	show_overall
te_tda_org	tda_tot_comp
te_tda_org	te_tot
te_tda_org	tlbr
te_tda_org	wiforg
te_tda_org	wkyrs_adjust
te_tot	tda_tot_comp
tgt_toggle	choose
tgt_toggle	optselect
tgt_toggle	showeditmsg
tgt_toggle	showrecmsg
tgt_toggle	show_overall
tgt_toggle	tda_tot_comp
toggle_list	optselect
toggle_list	warnem
total_engine	aadd
total_engine	aaddf
total_engine	cent
total_engine	clnupdest
total_engine	closefile
total_engine	closepage

Calling Routine	Called Routine
total_engine	closewindow
total_engine	indexord
total_engine	larray
total_engine	nonzero
total_engine	openfile
total_engine	openwindow
total_engine	pad
total_engine	setdest
total_engine	time
total_engine	years
tot_capl	closefile
tot_capl	fillnf
tot_capl	okass
tot_capl	okfoe
tot_capl	oklab
tot_capl	openfile
tot_capl	user_totals
tot_carry	closefile
tot_carry	dircust
tot_carry	fillnf
tot_carry	okapn
tot_carry	okcat
tot_carry	okdc
tot_carry	okfoe
tot_carry	okgrp
tot_carry	oklab
tot_carry	oksrc
tot_carry	openfile
tot_carry	user_totals
tot_expl	aadd
tot_expl	allocate_costs
tot_expl	cent
tot_expl	clnupdest
tot_expl	closefile
tot_expl	closepage
tot_expl	closewindow
tot_expl	compcpb
tot_expl	dircust
tot_expl	get_array
tot_expl	larray
tot_expl	load_foe_factors
tot_expl	openfile
tot_expl	openwindow
tot_expl	pad
tot_expl	retrieve_costs
tot_expl	setdest

Calling Routine	Called Routine
tot_exp1	years
tot_exp2	aadd
tot_exp2	allocate_costs
tot_exp2	cent
tot_exp2	clnupdest
tot_exp2	closefile
tot_exp2	closepage
tot_exp2	closewindow
tot_exp2	compcpb
tot_exp2	get_array
tot_exp2	larray
tot_exp2	load_foe_factors
tot_exp2	openfile
tot_exp2	openwindow
tot_exp2	pad
tot_exp2	retrieve_costs
tot_exp2	setdest
tot_exp2	years
tot_ext1	closefile
tot_ext1	dircust
tot_ext1	fillnf
tot_ext1	okapn
tot_ext1	okass
tot_ext1	okcat
tot_ext1	okdc
tot_ext1	okext
tot_ext1	okfoe
tot_ext1	okgrp
tot_ext1	oklab
tot_ext1	oksrc
tot_ext1	openfile
tot_ext1	user_totals
tot_int1	closefile
tot_int1	dircust
tot_int1	fillnf
tot_int1	okapn
tot_int1	okcat
tot_int1	okdc
tot_int1	okfoe
tot_int1	okgrp
tot_int1	oklab
tot_int1	oksrc
tot_int1	openfile
tot_int1	user_totals
tot_netrev	closefile
tot_netrev	dircust

Calling Routine	Called Routine
tot_netrev	fillnf
tot_netrev	okapn
tot_netrev	okcat
tot_netrev	okdc
tot_netrev	okfoe
tot_netrev	okgrp
tot_netrev	oklab
tot_netrev	oksrc
tot_netrev	openfile
tot_netrev	user_totals
tot_pers1	closefile
tot_pers1	fillnf
tot_pers1	isinfile
tot_pers1	okfoe
tot_pers1	oklab
tot_pers1	okpers
tot_pers1	okpos
tot_pers1	okps
tot_pers1	okskl
tot_pers1	okwky
tot_pers1	openfile
tot_pers1	user_totals
tot_pess_rev	fillnf
tot_pess_rev	okapn
tot_pess_rev	okcat
tot_pess_rev	okfoe
tot_pess_rev	okgrp
tot_pess_rev	oklab
tot_pess_rev	oksrc
tot_pess_rev	user_totals
tot_rev1	closefile
tot_rev1	dircust
tot_rev1	fillnf
tot_rev1	okapn
tot_rev1	okcat
tot_rev1	okdc
tot_rev1	okfoe
tot_rev1	okgrp
tot_rev1	oklab
tot_rev1	oksrc
tot_rev1	openfile
tot_rev1	user_totals
tot_rev2	closefile
tot_rev2	dircust
tot_rev2	fillnf
tot_rev2	okapn

Calling Routine	Called Routine
tot_rev2	okcat
tot_rev2	okdc
tot_rev2	okfoe
tot_rev2	okgrp
tot_rev2	oklab
tot_rev2	oksrc
tot_rev2	openfile
tot_rev2	user_totals
tsamt	aadd
tsamt	amult
tsamt	cent
tsamt	clnupdest
tsamt	closepage
tsamt	closewindow
tsamt	get_array
tsamt	larray
tsamt	openwindow
tsamt	setdest
tsamt	years
ts_amounts	choosemenu
ts_amounts	fillpu
ts_amounts	showeditmsg
ts_amounts	tsamt
unlinked	closefile
unlinked	openfile
upgrade	tempupgrade
upgrade	upgrone
upgrade	warnem
upgrone	buildfile
upgrone	cent
upgrone	closewindow
upgrone	openwindow
user_defined	anonzero
user_defined	cent
user_defined	choose
user_defined	closewindow
user_defined	compose_line
user_defined	c_o
user_defined	dtos
user_defined	getfiltcond
user_defined	larray
user_defined	list_engine
user_defined	loadpsf
user_defined	openwindow
user_defined	optselect
user_defined	pad

Calling Routine	Called Routine
user_defined	popq
user_defined	rem_old_names
user_defined	savepsf
user_defined	show_arr
user_defined	show_line_length
user_defined	tlbr
user_defined	warnem
user_defined	years
user_totals	cent
user_totals	choose
user_totals	closewindow
user_totals	getfiltcond
user_totals	loadpsf
user_totals	openwindow
user_totals	optselect
user_totals	pad
user_totals	popq
user_totals	rem_old_names
user_totals	savepsf
user_totals	show_arr
user_totals	tlbr
user_totals	total_engine
user_totals	warnem
view_capital	cent
view_capital	closefile
view_capital	closewindow
view_capital	del_record
view_capital	doit
view_capital	e_capital
view_capital	fillpmu
view_capital	go_spec
view_capital	list_capital
view_capital	norec
view_capital	openfile
view_capital	openwindow
view_capital	post_repl
view_capital	sel_foe
view_capital	showeditmsg
view_capital	tlbr
view_capmaj	aaddf
view_capmaj	cent
view_capmaj	closefile
view_capmaj	closewindow
view_capmaj	del_record
view_capmaj	doit
view_capmaj	e_capmaj

Calling Routine	Called Routine
view_capmaj	fillpmu
view_capmaj	go_spec
view_capmaj	minmax
view_capmaj	norec
view_capmaj	openfile
view_capmaj	openwindow
view_capmaj	put_array
view_capmaj	showeditmsg
view_capmaj	tlbr
view_center	anal_details
view_center	closefile
view_center	closewindow
view_center	ctr_toggle
view_center	disp_foe_guts
view_center	doit
view_center	fillpmu
view_center	list_center
view_center	openfile
view_center	openwindow
view_center	recalc_all
view_center	showeditmsg
view_center	tlbr
view_corecap	closefile
view_corecap	closewindow
view_corecap	doit
view_corecap	e_corecap
view_corecap	fillpmu
view_corecap	go_corecap
view_corecap	list_corecap
view_corecap	openfile
view_corecap	openwindow
view_corecap	pad
view_corecap	sel_foe
view_corecap	showeditmsg
view_corecap	tlbr
view_corecap	warnem
view_ctda	cent
view_ctda	closefile
view_ctda	closewindow
view_ctda	del_record
view_ctda	doit
view_ctda	e_ctda
view_ctda	fillpmu
view_ctda	go_general
view_ctda	list_ctda
view_ctda	norec

Calling Routine	Called Routine
view_ctda	openfile
view_ctda	openwindow
view_ctda	showeditmsg
view_ctda	tlbr
view_foe_factors	addupfactors
view_foe_factors	closefile
view_foe_factors	closewindow
view_foe_factors	doit
view_foe_factors	edit_foe_factors
view_foe_factors	fillpmu
view_foe_factors	list_factors
view_foe_factors	load_foe_factors
view_foe_factors	openfile
view_foe_factors	openwindow
view_foe_factors	sel_foe
view_foe_factors	showeditmsg
view_foe_factors	tlbr
view_foe_factors	warnem
view_foe_factors	wr_arr
view_foe_factors	wr_uls
view_foe_factors	wr_years
view_internal	aaddf
view_internal	cent
view_internal	closefile
view_internal	closewindow
view_internal	del_record
view_internal	doit
view_internal	e_internal
view_internal	fillpmu
view_internal	go_spec
view_internal	minmax
view_internal	norec
view_internal	openfile
view_internal	openwindow
view_internal	put_array
view_internal	showeditmsg
view_internal	tlbr
view_main_factors	closefile
view_main_factors	closewindow
view_main_factors	doit
view_main_factors	edit_main_factors
view_main_factors	fillpmu
view_main_factors	list_factors
view_main_factors	openfile
view_main_factors	openwindow
view_main_factors	showeditmsg

Calling Routine	Called Routine
view_main_factors	tlbr
view_main_factors	wr_arr
view_main_factors	wr_uls
view_main_factors	wr_years
view_major	aaddf
view_major	cent
view_major	closefile
view_major	closewindow
view_major	del_record
view_major	doit
view_major	e_major
view_major	fillpmu
view_major	go_spec
view_major	minmax
view_major	norec
view_major	openfile
view_major	openwindow
view_major	put_array
view_major	showeditmsg
view_major	tlbr
view_off	cent
view_off	closefile
view_off	closewindow
view_off	del_record
view_off	doit
view_off	e_off
view_off	fillpmu
view_off	go_general
view_off	list_off
view_off	norec
view_off	openfile
view_off	openwindow
view_off	showeditmsg
view_off	tlbr
view_orgs	cent
view_orgs	closefile
view_orgs	closewindow
view_orgs	del_record
view_orgs	doit
view_orgs	e_org
view_orgs	fillpmu
view_orgs	go_general
view_orgs	list_orgs
view_orgs	norec
view_orgs	openfile
view_orgs	openwindow

Calling Routine	Called Routine
view_orgs	showeditmsg
view_orgs	tlbr
view_personnel	cent
view_personnel	checklevel
view_personnel	closefile
view_personnel	closewindow
view_personnel	del_personnel
view_personnel	doit
view_personnel	edit_personnel
view_personnel	e_personnel
view_personnel	fillpmu
view_personnel	go_spec
view_personnel	list_personnel
view_personnel	load_foe_factors
view_personnel	move_personnel
view_personnel	norec
view_personnel	openfile
view_personnel	openwindow
view_personnel	sel_foe
view_personnel	showeditmsg
view_personnel	tlbr
view_products	cent
view_products	closefile
view_products	closewindow
view_products	del_record
view_products	doit
view_products	e_product
view_products	fillpmu
view_products	go_spec
view_products	norec
view_products	openfile
view_products	openwindow
view_products	showeditmsg
view_products	tlbr
view_rates	accept_rates
view_rates	checklevel
view_rates	closefile
view_rates	closewindow
view_rates	disp_rates
view_rates	doit
view_rates	fillpmu
view_rates	genmsgwindow
view_rates	list_rates
view_rates	openfile
view_rates	openwindow
view_rates	tlbr

Calling Routine	Called Routine
view_revenue	cent
view_revenue	checklevel
view_revenue	closefile
view_revenue	closewindow
view_revenue	del_revenue
view_revenue	doit
view_revenue	edit_revenue
view_revenue	e_revenue
view_revenue	fillpmu
view_revenue	go_spec
view_revenue	list_revenue
view_revenue	load_foe_factors
view_revenue	move_revenue
view_revenue	norec
view_revenue	openfile
view_revenue	openwindow
view_revenue	sel_foe
view_revenue	showeditmsg
view_revenue	tlbr
view_revtypes	cent
view_revtypes	closefile
view_revtypes	closewindow
view_revtypes	del_record
view_revtypes	doit
view_revtypes	e_revtype
view_revtypes	fillpmu
view_revtypes	go_general
view_revtypes	list_revtypes
view_revtypes	norec
view_revtypes	openfile
view_revtypes	openwindow
view_revtypes	showeditmsg
view_revtypes	tlbr
view_skills	cent
view_skills	closefile
view_skills	closewindow
view_skills	del_record
view_skills	doit
view_skills	e_skill
view_skills	fillpmu
view_skills	go_general
view_skills	list_skills
view_skills	norec
view_skills	openfile
view_skills	openwindow
view_skills	showeditmsg

Calling Routine	Called Routine
view_skills	tlbr
view_sources	cent
view_sources	closefile
view_sources	closewindow
view_sources	del_record
view_sources	doit
view_sources	e_source
view_sources	fillpmu
view_sources	go_general
view_sources	list_sources
view_sources	norec
view_sources	openfile
view_sources	openwindow
view_sources	showeditmsg
view_sources	tlbr
view_support	aaddf
view_support	cent
view_support	closefile
view_support	closewindow
view_support	del_record
view_support	doit
view_support	e_support
view_support	fillpmu
view_support	go_spec
view_support	minmax
view_support	norec
view_support	openfile
view_support	openwindow
view_support	put_array
view_support	showeditmsg
view_support	tlbr
view_tda_org	closewindow
view_tda_org	compare_wif
view_tda_org	doit
view_tda_org	doproj_org_tda
view_tda_org	e_tda_org
view_tda_org	fillpmu
view_tda_org	go_spec
view_tda_org	list_tda_org
view_tda_org	norec
view_tda_org	openfile
view_tda_org	openwindow
view_tda_org	recalc_yr
view_tda_org	showeditmsg
view_tda_org	showrecmsg
view_tda_org	te_tda_org

Calling Routine	Called Routine
view_tda_org	tgt_toggle
view_tda_org	tlbr
view_tda_org	wiforg
view_totals	disp_foe
view_totals	load_foe_factors
view_totals	sel_foe
warnem	cent
warnem	closewindow
warnem	openwindow
whatif	cent
whatif	choosemenu
whatif	closefile
whatif	closewindow
whatif	escape
whatif	fillpu
whatif	genmsgwindow
whatif	openfile
whatif	openwindow
whatif	optselect
whatif	opt_personnel
whatif	rev_pessimism
whatif	showeditmsg
whatif	tda_whatif
whatsnext	doskip
win_print	clnupdest
win_print	closepage
win_print	setdest
writescreen	pad
wr_arr	larray
wr_years	years
xbrowse	lastrec
xmemo	optselect

CHAPTER 5

CROSS REFERENCE TO CALLING FUNCTION/PROCEDURES

The following table is a cross references between BPM functions/procedures and the functions and procedures which call them. This is exactly the reverse of Appendix B. This is an important tool for programmers who want to understand or modify the BPM program files. Only routines that are non-standard Clipper are included. This list is created a program and should be more comprehensive and more current than the documented code files.

Called Routine	Calling Routine
aadd	aaddf
aadd	addupfactors
aadd	allocate_costs
aadd	alloc_wky
aadd	amcc2
aadd	amcc3
aadd	amcc4
aadd	amcc5
aadd	amcc_ihoh
aadd	avgload
aadd	bal_part2
aadd	compose_line
aadd	disp_capital
aadd	disp_external
aadd	disp_foe_guts
aadd	disp_rates
aadd	disp_wkyrs
aadd	doload
aadd	dopessimism
aadd	get_cost_tot
aadd	intl_diag
aadd	listcapstr
aadd	list_avg_fact
aadd	list_pess1
aadd	l_cap
aadd	l_ext
aadd	l_main
aadd	l_wkyrs
aadd	net_carry
aadd	net_rev
aadd	net_revenue
aadd	org_nets

Called Routine	Calling Routine
aadd	pess_scan
aadd	pess_show
aadd	recalc_tda
aadd	recomp_foe
aadd	rem_ts_cost
aadd	retrieve_costs
aadd	split_chart
aadd	tech_base_ihoh
aadd	total_engine
aadd	tot_exp1
aadd	tot_exp2
aadd	tsamt
aaddf	amcc2
aaddf	amcc4
aaddf	amcc5
aaddf	pess_scan
aaddf	recalc_tda
aaddf	recomp_center
aaddf	recomp_foe
aaddf	recomp_lab
aaddf	split_chart
aaddf	total_engine
aaddf	view_capmaj
aaddf	view_internal
aaddf	view_major
aaddf	view_support
accept_rates	view_rates
achoice	choose
achoice	choosemenu
achoice	te_tda_org
addnewfile	defineallfiles
addupfactors	edit_foe_factors
addupfactors	view_foe_factors
add_field	core_tot
add_rec	e_capital
add_rec	e_capmaj
add_rec	e_ctda
add_rec	e_internal
add_rec	e_major
add_rec	e_off
add_rec	e_org
add_rec	e_personnel
add_rec	e_per_item
add_rec	e_pessimism
add_rec	e_product
add_rec	e_revenue

Called Routine	Calling Routine
add_rec	e_revtype
add_rec	e_skill
add_rec	e_source
add_rec	e_support
add_rec	put_cap
add_rec	put_factor
afields	browse
alength	choose
align	bal_part2
align	core_dollars
align	core_tot
align	e_personnel
allocadd	alloccomp
allocadd	source_use
allocate_costs	tech_base_ihoh
allocate_costs	tot_exp1
allocate_costs	tot_exp2
alloccomp	source_use
allocsu	source_use
alloc_cost	dopessimism
alloc_det	core_tot
alloc_one	alloc_cost
alloc_wky	alloc_cost
amaxlen	choose
amaxlen	choosemenu
amaxlen	getthese
amcc1	amccharts
amcc1a	amccharts
amcc1b	amccharts
amcc1c	amccharts
amcc1d	amccharts
amcc1e	amccharts
amcc2	amccharts
amcc3	amccharts
amcc4	amccharts
amcc5	amccharts
amccharts	list_center
amcc_ihoh	amccharts
amult	alloc_one
amult	amcc_ihoh
amult	compass
amult	doload
amult	e_pessimism
amult	get_totals
amult	larray
amult	list_avg_fact

Called Routine	Calling Routine
amult	load_main_factors
amult	l_ext
amult	l_main
amult	pess_scan
amult	post_totals
amult	put_totals
amult	recalc_tda
amult	recomp_foe
amult	retrieve_costs
amult	tsamt
anal_details	disp_foe
anal_details	view_center
anonzero	remove_empty
anonzero	user_defined
ardist	gasubsidy
ardist	opt_calc
ardiv	gasubsidy
ardiv	opt_calc
arneg	opt_calc
arsum	dirorgcalc
arsum	loadrevs
arsum	opt_calc
arsum	reloaddata
arvadd	dirorgcalc
arvadd	e_opt_pers
arvadd	gasubsidy
arvadd	loadrevs
arvadd	opt_calc
arvsub	opt_calc
askok	genmsgwindow
asort	doviewtext
assorted	bpm (main)
asub	alloc_one
asub	alloc_wky
asub	balance1
asub	balance2
asub	balance3
asub	core_tot
asub	disp_capital
asub	disp_contract
asub	disp_foe_guts
asub	disp_internal
asub	disp_rates
asub	doload
asub	dopessimism
asub	list_pess1

Called Routine	Calling Routine
asub	l_cap
asub	l_int
asub	l_main
asub	l_supt
asub	net_carry
asub	net_rev
asub	org_nets
asub	pess_scan
asub	pess_show
asub	recomp_foe
asub	recomp_lab
asub	rem_ts_cost
avdiv	allocate_costs
avdiv	alloc_one
avdiv	alloc_wky
avdiv	amcc4
avdiv	avgload
avdiv	disp_rates
avdiv	dopessimism
avdiv	e_cpb_ga
avdiv	list_avg_fact
avdiv	recalc_tda
avdiv	recomp_foe
avdiv	tech_base_ihoh
avgload	doload
avmult	allocate_costs
avmult	alloc_wky
avmult	amcc4
avmult	avgload
avmult	disp_rates
avmult	list_avg_fact
avmult	list_pess1
avmult	pess_adj
avmult	pess_scan
avmult	pess_show
avmult	recalc_tda
balance1	balances
balance2	balances
balance3	balances
balances	list_center
bal_part1	balance1
bal_part1	balance2
bal_part1	balance3
bal_part2	balance1
bal_part2	balance2
bal_part2	balance3

Called Routine	Calling Routine
baseline	bpm (main)
blank_if_cont	e_personnel
blimempak	bpm (main)
blimempak	choosemenu
blimempak	recalc_tda
blimempak	showmemory
bliovlclr	bpm (main)
bliovlclr	choosemenu
bliovlclr	recalc_tda
bliovlclr	showmemory
browse	dobrowse
buildfile	addnewfile
buildfile	upgrone
calloc_one	alloc_one
calloc_wky	alloc_wky
cctot	e_corecap
cdosvers	dosvers
cent	avgload
cent	balance1
cent	balance2
cent	balance3
cent	bpm (main)
cent	choose
cent	choosemenu
cent	copydata
cent	copyone
cent	corefilt
cent	core_dollars
cent	core_qc
cent	core_tot
cent	ctreport
cent	deleteone
cent	delete_lab
cent	del_allbut
cent	dobrowse
cent	docheck
cent	doindex
cent	dolicense
cent	doload
cent	dopack
cent	doproj_org_tda
cent	doreindex
cent	edit_baseyear
cent	genload
cent	genmsgwindow
cent	getfiltcond

Called Routine	Calling Routine
cent	getthese
cent	go_general
cent	help
cent	incr_baseyear
cent	list_avg_fact
cent	list_engine
cent	list_tot1
cent	optselect
cent	opt_personnel
cent	opt_save
cent	org_nets
cent	pess_scan
cent	project_tda
cent	recalc_tda
cent	recalc_year
cent	recomp_center
cent	recomp_lab
cent	removeone
cent	remove_empty
cent	remove_unlinked
cent	rerecalc_tda
cent	rev_pessimism
cent	showmemory
cent	source_use
cent	tda_whatif
cent	tech_base_ihoh
cent	tempupgrade
cent	total_engine
cent	tot_exp1
cent	tot_exp2
cent	tsamt
cent	upgrone
cent	user_defined
cent	user_totals
cent	view_capital
cent	view_capmaj
cent	view_ctda
cent	view_internal
cent	view_major
cent	view_off
cent	view_orgs
cent	view_personnel
cent	view_products
cent	view_revenue
cent	view_revtypes
cent	view_skills

Called Routine	Calling Routine
cent	view_sources
cent	view_support
cent	warnem
cent	whatif
checklevel	bpm (main)
checklevel	view_personnel
checklevel	view_rates
checklevel	view_revenue
checkone	docheck
chkidx	addnewfile
choose	alloc_cost
choose	bpm (main)
choose	ctr_toggle
choose	doviewtext
choose	edit_foe_factors
choose	edit_main_factors
choose	edit_personnel
choose	edit_revenue
choose	eval_toggle
choose	go_corecap
choose	go_spec
choose	isinfile
choose	okapn
choose	okass
choose	okcat
choose	okdc
choose	okext
choose	okfact
choose	okfoe
choose	okgrp
choose	okintlab
choose	oklab
choose	okoper
choose	okorg
choose	okpers
choose	okpos
choose	okps
choose	okskl
choose	oksrc
choose	okwky
choose	okyn
choose	opt_view1
choose	opt_view2
choose	opt_view3
choose	sel_foe
choose	sel_lab_foe

Called Routine	Calling Routine
choose	tgt_toggle
choose	user_defined
choose	user_totals
choosemenu	amccharts
choosemenu	anal_details
choosemenu	assorted
choosemenu	balances
choosemenu	baseline
choosemenu	data_entry
choosemenu	expenses
choosemenu	list_capital
choosemenu	list_center
choosemenu	list_corecap
choosemenu	list_factors
choosemenu	list_personnel
choosemenu	list_pessimism
choosemenu	list_rates
choosemenu	list_revenue
choosemenu	list_tda_org
choosemenu	list_tda_wif
choosemenu	list_totals
choosemenu	opt_fact
choosemenu	opt_list
choosemenu	opt_view
choosemenu	sys_edit
choosemenu	ts_amounts
choosemenu	whatif
clnupdest	amcc1
clnupdest	amcc1a
clnupdest	amcc1b
clnupdest	amcc1c
clnupdest	amcc1d
clnupdest	amcc1e
clnupdest	amcc2
clnupdest	amcc3
clnupdest	amcc4
clnupdest	amcc5
clnupdest	amcc_ihoh
clnupdest	balance1
clnupdest	balance2
clnupdest	balance3
clnupdest	core_dollars
clnupdest	core_qc
clnupdest	core_tot
clnupdest	docheck
clnupdest	intl_diag

Called Routine	Calling Routine
clnupdest	list_avg_fact
clnupdest	list_engine
clnupdest	list_pess1
clnupdest	list_tda1
clnupdest	list_tda2
clnupdest	list_tda3
clnupdest	list_tda4
clnupdest	list_tda5
clnupdest	list_tot1
clnupdest	opt_list1
clnupdest	opt_list2
clnupdest	opt_list3
clnupdest	org_nets
clnupdest	source_use
clnupdest	tech_base_ihoh
clnupdest	total_engine
clnupdest	tot_exp1
clnupdest	tot_exp2
clnupdest	tsamt
clnupdest	win_print
closefile	amcc1
closefile	amcc1a
closefile	amcc1b
closefile	amcc1c
closefile	amcc1d
closefile	amcc1e
closefile	amcc2
closefile	amcc3
closefile	amcc4
closefile	amcc5
closefile	avgload
closefile	balance1
closefile	balance2
closefile	balance3
closefile	checkone
closefile	copyone
closefile	core_dollars
closefile	core_qc
closefile	core_tot
closefile	ctreport
closefile	deleteone
closefile	del_revenue
closefile	disp_foe
closefile	docheck
closefile	doload
closefile	dopack

Called Routine	Calling Routine
closefile	dopessimism
closefile	doreindex
closefile	edit_baseyear
closefile	e_revenue
closefile	genload
closefile	get_totals
closefile	help
closefile	incr_baseyear
closefile	intl_diag
closefile	isinfile
closefile	list_avg_fact
closefile	list_engine
closefile	list_ext1
closefile	list_int1
closefile	list_prod1
closefile	list_wif1
closefile	load_foe_factors
closefile	load_main_factors
closefile	load_orgs
closefile	move_revenue
closefile	oksrc
closefile	opt_personnel
closefile	org_nets
closefile	post_totals
closefile	put_factor
closefile	recalc_all
closefile	recalc_tda
closefile	recomp_center
closefile	recomp_lab
closefile	reloaddata
closefile	removeone
closefile	remove_empty
closefile	remove_unlinked
closefile	rev_pessimism
closefile	source_use
closefile	tda_whatif
closefile	tech_base_ihoh
closefile	total_engine
closefile	tot_cap1
closefile	tot_carry
closefile	tot_exp1
closefile	tot_exp2
closefile	tot_ext1
closefile	tot_int1
closefile	tot_netrev
closefile	tot_pers1

Called Routine	Calling Routine
closefile	tot_rev1
closefile	tot_rev2
closefile	unlinked
closefile	view_capital
closefile	view_capmaj
closefile	view_center
closefile	view_corecap
closefile	view_ctda
closefile	view_foe_factors
closefile	view_internal
closefile	view_main_factors
closefile	view_major
closefile	view_off
closefile	view_orgs
closefile	view_personnel
closefile	view_products
closefile	view_rates
closefile	view_revenue
closefile	view_revtypes
closefile	view_skills
closefile	view_sources
closefile	view_support
closefile	whatif
closepage	amcc1
closepage	amcc1a
closepage	amcc1b
closepage	amcc1c
closepage	amcc1d
closepage	amcc1e
closepage	amcc2
closepage	amcc3
closepage	amcc4
closepage	amcc5
closepage	amcc_ihoh
closepage	balance1
closepage	balance2
closepage	balance3
closepage	core_dollars
closepage	core_qc
closepage	core_tot
closepage	ctreport
closepage	docheck
closepage	intl_diag
closepage	list_avg_fact
closepage	list_engine
closepage	list_pess1

Called Routine	Calling Routine
closepage	list_tda1
closepage	list_tda2
closepage	list_tda3
closepage	list_tda4
closepage	list_tda5
closepage	list_tot1
closepage	opt_list1
closepage	opt_list2
closepage	opt_list3
closepage	org_nets
closepage	source_use
closepage	tech_base_ihoh
closepage	total_engine
closepage	tot_exp1
closepage	tot_exp2
closepage	tsamt
closepage	win_print
closewindow	alloc_one
closewindow	balance1
closewindow	balance2
closewindow	balance3
closewindow	bpm (main)
closewindow	choose
closewindow	choosemenu
closewindow	compare_wif
closewindow	copydata
closewindow	corefilt
closewindow	core_dollars
closewindow	core_qc
closewindow	core_tot
closewindow	delete_lab
closewindow	del_allbut
closewindow	disp_capital
closewindow	disp_contract
closewindow	disp_expense
closewindow	disp_external
closewindow	disp_foe
closewindow	disp_internal
closewindow	disp_wkyrs
closewindow	dobrowse
closewindow	docheck
closewindow	doindex
closewindow	dolicense
closewindow	doload
closewindow	dopack
closewindow	dopessimism

Called Routine	Calling Routine
closewindow	doproj_org_tda
closewindow	doreindex
closewindow	edit_baseyear
closewindow	e_cpb_ga
closewindow	e_cpb_wky
closewindow	genmsgwindow
closewindow	getfiltcond
closewindow	getthese
closewindow	go_general
closewindow	help
closewindow	incr_baseyear
closewindow	list_avg_fact
closewindow	list_engine
closewindow	list_tot1
closewindow	optselect
closewindow	opt_fact1
closewindow	opt_fact2
closewindow	opt_personnel
closewindow	opt_save
closewindow	org_nets
closewindow	pess_scan
closewindow	project_tda
closewindow	recalc_tda
closewindow	recalc_year
closewindow	recomp_center
closewindow	recomp_lab
closewindow	remove_empty
closewindow	remove_unlinked
closewindow	rev_pessimism
closewindow	showmemory
closewindow	source_use
closewindow	tda_whatif
closewindow	tech_base_ihoh
closewindow	tempupgrade
closewindow	te_tda_org
closewindow	total_engine
closewindow	tot_exp1
closewindow	tot_exp2
closewindow	tsamt
closewindow	upgrone
closewindow	user_defined
closewindow	user_totals
closewindow	view_capital
closewindow	view_capmaj
closewindow	view_center
closewindow	view_corecap

Called Routine	Calling Routine
closewindow	view_ctda
closewindow	view_foe_factors
closewindow	view_internal
closewindow	view_main_factors
closewindow	view_major
closewindow	view_off
closewindow	view_orgs
closewindow	view_personnel
closewindow	view_products
closewindow	view_rates
closewindow	view_revenue
closewindow	view_revtypes
closewindow	view_skills
closewindow	view_sources
closewindow	view_support
closewindow	view_tda_org
closewindow	warnem
closewindow	whatif
compare_wif	view_tda_org
compass	e_capital
compass	post_repl
compchg	e_pessimism
compcpb	bal_part2
compcpb	e_carry
compcpb	e_revenue
compcpb	e_rev_item
compcpb	recomp_foe
compcpb	source_use
compcpb	tech_base_ihoh
compcpb	tot_exp1
compcpb	tot_exp2
compose_line	user_defined
compwky	bal_part1
compwky	e_personnel
compwky	e_per_item
compwky	recomp_foe
copydata	assorted
copyone	copydata
corefilt	core_dollars
core_dollars	list_corecap
core_qc	list_corecap
core_tot	amccharts
ctreport	core_tot
ctr_toggle	view_center
c_o	user_defined
data_entry	bpm (main)

Called Routine	Calling Routine
dbedit	browse
dbedit	go_general
defineallfiles	bpm (main)
deleteone	delete_lab
deleteone	del_allbut
delete_lab	assorted
del_allbut	assorted
del_personnel	view_personnel
del_rec	del_personnel
del_rec	del_record
del_rec	del_revenue
del_rec	e_per_item
del_rec	put_cap
del_rec	put_factor
del_record	del_revenue
del_record	rev_pessimism
del_record	view_capital
del_record	view_capmaj
del_record	view_ctda
del_record	view_internal
del_record	view_major
del_record	view_off
del_record	view_orgs
del_record	view_products
del_record	view_revtypes
del_record	view_skills
del_record	view_sources
del_record	view_support
del_revenue	view_revenue
dircust	core_tot
dircust	list_ext1
dircust	list_int1
dircust	list_prod1
dircust	list_rev1
dircust	source_use
dircust	tot_carry
dircust	tot_exp1
dircust	tot_ext1
dircust	tot_int1
dircust	tot_netrev
dircust	tot_rev1
dircust	tot_rev2
dirorgcalc	opt_calc
disp_capital	anal_details
disp_contract	anal_details
disp_expense	anal_details

Called Routine	Calling Routine
disp_external	anal_details
disp_foe	view_totals
disp_foe_guts	ctr_toggle
disp_foe_guts	disp_foe
disp_foe_guts	dotoggle
disp_foe_guts	recalc_all
disp_foe_guts	recalc_lab
disp_foe_guts	view_center
disp_internal	anal_details
disp_rates	view_rates
disp_wkyrs	anal_details
docheck	assorted
doindex	chkidx
doit	disp_foe
doit	dopessimism
doit	opt_personnel
doit	rev_pessimism
doit	tda_whatif
doit	view_capital
doit	view_capmaj
doit	view_center
doit	view_corecap
doit	view_ctda
doit	view_foe_factors
doit	view_internal
doit	view_main_factors
doit	view_major
doit	view_off
doit	view_orgs
doit	view_personnel
doit	view_products
doit	view_rates
doit	view_revenue
doit	view_revtypes
doit	view_skills
doit	view_sources
doit	view_support
doit	view_tda_org
doload	assorted
dopack	bpm (main)
dopessimism	rev_pessimism
doproj_org_tda	view_tda_org
doreindex	assorted
doskip	doit
doskip	whatsnext
dosvers	bpm (main)

Called Routine	Calling Routine
dotoggle	disp_foe
doviewtext	assorted
doviewtext	list_capital
doviewtext	list_center
doviewtext	list_corecap
doviewtext	list_factors
doviewtext	list_personnel
doviewtext	list_pessimism
doviewtext	list_rates
doviewtext	list_revenue
doviewtext	list_tda_org
doviewtext	list_tda_wif
doviewtext	list_totals
doviewtext	opt_list
dtos	user_defined
edit_baseyear	sys_edit
edit_foe_factors	view_foe_factors
edit_main_factors	view_main_factors
edit_personnel	view_personnel
edit_revenue	view_revenue
escape	anal_details
escape	assorted
escape	baseline
escape	data_entry
escape	sys_edit
escape	whatif
est_factor	e_factor
eval_toggle	tda_whatif
expenses	list_center
e_capital	view_capital
e_capmaj	view_capmaj
e_carry	edit_revenue
e_corecap	view_corecap
e_cpb_ga	alloc_cost
e_cpb_wky	alloc_cost
e_ctda	view_ctda
e_factor	edit_foe_factors
e_factor	edit_main_factors
e_internal	view_internal
e_major	view_major
e_off	view_off
e_opt_pers	opt_calc
e_opt_pers	opt_personnel
e_org	view_orgs
e_personnel	edit_personnel
e_personnel	view_personnel

Called Routine	Calling Routine
e_per_item	edit_personnel
e_pessimism	rev_pessimism
e_product	view_products
e_revenue	edit_revenue
e_revenue	view_revenue
e_revtype	view_revtypes
e_rev_item	edit_revenue
e_rev_item	e_carry
e_skill	view_skills
e_source	view_sources
e_support	view_support
e_tda_org	view_tda_org
e_tda_tgt	tda_whatif
e_tda_what	tda_whatif
files	bpm (main)
fillnf	list_cap1
fillnf	list_core1
fillnf	list_ctda
fillnf	list_ext1
fillnf	list_fac1
fillnf	list_int1
fillnf	list_off
fillnf	list_orgs
fillnf	list_pers1
fillnf	list_pess_rev
fillnf	list_prod1
fillnf	list_rev1
fillnf	list_revtypes
fillnf	list_skills
fillnf	list_sources
fillnf	list_wif1
fillnf	tot_cap1
fillnf	tot_carry
fillnf	tot_ext1
fillnf	tot_int1
fillnf	tot_netrev
fillnf	tot_pers1
fillnf	tot_pess_rev
fillnf	tot_rev1
fillnf	tot_rev2
fillpmu	disp_foe
fillpmu	dopessimism
fillpmu	opt_personnel
fillpmu	rev_pessimism
fillpmu	tda_whatif
fillpmu	view_capital

Called Routine	Calling Routine
fillpmu	view_capmaj
fillpmu	view_center
fillpmu	view_corecap
fillpmu	view_ctda
fillpmu	view_foe_factors
fillpmu	view_internal
fillpmu	view_main_factors
fillpmu	view_major
fillpmu	view_off
fillpmu	view_orgs
fillpmu	view_personnel
fillpmu	view_products
fillpmu	view_rates
fillpmu	view_revenue
fillpmu	view_revtypes
fillpmu	view_skills
fillpmu	view_sources
fillpmu	view_support
fillpmu	view_tda_org
fillpu	alloc_cost
fillpu	amccharts
fillpu	anal_details
fillpu	assorted
fillpu	balances
fillpu	baseline
fillpu	data_entry
fillpu	edit_foe_factors
fillpu	edit_main_factors
fillpu	edit_personnel
fillpu	edit_revenue
fillpu	expenses
fillpu	list_capital
fillpu	list_center
fillpu	list_corecap
fillpu	list_factors
fillpu	list_personnel
fillpu	list_pessimism
fillpu	list_rates
fillpu	list_revenue
fillpu	list_tda_org
fillpu	list_tda_wif
fillpu	list_totals
fillpu	opt_fact
fillpu	opt_list
fillpu	opt_view
fillpu	sys_edit

Called Routine	Calling Routine
fillpu	ts_amounts
fillpu	whatif
fillpvp	est_factor
fillpvp	e_tda_tgt
fillpvp	move_personnel
fillpvp	move_revenue
fillpvp	source_use
findkey	e_personnel
findkey	e_revenue
findkey	load_orgs
findkey	move_personnel
findkey	move_revenue
findkey	openfile
findkey	reloaddata
gasubsidy	opt_calc
genload	doload
genmsgwindow	showdemomsg
genmsgwindow	view_rates
genmsgwindow	whatif
getdatetime	chkidx
getfileid	openfile
getfiltcond	user_defined
getfiltcond	user_totals
getthese	bpm (main)
getthese	copydata
getthese	core_tot
getthese	doload
getthese	edit_baseyear
getthese	est_factor
getthese	e_tda_tgt
getthese	limit_overtime
getthese	move_personnel
getthese	move_revenue
getthese	setdest
getthese	source_use
getthese	split_chart
get_array	aaddf
get_array	amcc4
get_array	anonzero
get_array	avgload
get_array	bal_part2
get_array	compose_line
get_array	edit_revenue
get_array	e_capital
get_array	e_capmaj
get_array	e_internal

Called Routine	Calling Routine
get_array	e_major
get_array	e_pessimism
get_array	e_revenue
get_array	e_support
get_array	get_factor
get_array	get_totals
get_array	get_wkyrs
get_array	intl_diag
get_array	listcapstr
get_array	net_carry
get_array	net_rev
get_array	pess_adj
get_array	pess_adj_exp
get_array	pess_scan
get_array	post_repl
get_array	recalc_tda
get_array	recomp_foe
get_array	source_use
get_array	split_chart
get_array	tech_base_ihoh
get_array	tot_exp1
get_array	tot_exp2
get_array	tsamt
get_cap	core_tot
get_cap	e_corecap
get_cost_tot	amcc_ihoh
get_cost_tot	disp_external
get_cost_tot	disp_foe_guts
get_cost_tot	disp_rates
get_cost_tot	dopessimism
get_cost_tot	l_ext
get_cost_tot	l_main
get_cost_tot	recalc_tda
get_cost_tot	reloaddata
get_cost_tot	retrieve_costs
get_factor	load_foe_factors
get_factor	load_main_factors
get_pic	browse
get_set	dobrowse
get_set	getthese
get_set	help
get_totals	amcc_ihoh
get_totals	core_tot
get_totals	disp_capital
get_totals	disp_contract
get_totals	disp_expense

Called Routine	Calling Routine
get_totals	disp_external
get_totals	disp_foe_guts
get_totals	disp_internal
get_totals	disp_rates
get_totals	disp_wkyrs
get_totals	dopessimism
get_totals	get_cost_tot
get_totals	l_cap
get_totals	l_exp
get_totals	l_ext
get_totals	l_int
get_totals	l_main
get_totals	l_supt
get_totals	l_wkyrs
get_totals	org_nets
get_totals	recomp_lab
get_totals	rem_ts_cost
get_totals	retrieve_costs
get_wkyrs	amcc3
get_wkyrs	bal_part1
get_wkyrs	doload
get_wkyrs	edit_personnel
get_wkyrs	e_personnel
get_wkyrs	list_avg_fact
get_wkyrs	recalc_tda
get_wkyrs	recomp_foe
go_corecap	view_corecap
go_general	isinfile
go_general	opt_personnel
go_general	rev_pessimism
go_general	tda_whatif
go_general	view_ctda
go_general	view_off
go_general	view_orgs
go_general	view_revtypes
go_general	view_skills
go_general	view_sources
go_spec	view_capital
go_spec	view_capmaj
go_spec	view_internal
go_spec	view_major
go_spec	view_personnel
go_spec	view_products
go_spec	view_revenue
go_spec	view_support
go_spec	view_tda_org

Called Routine	Calling Routine
handles	bpm (main)
help	a_handler
if	doskip
incr_baseyear	sys_edit
indexkey	browse
indexkey	chkidx
indexkey	del_rec
indexkey	doskip
indexkey	findkey
indexkey	go_spec
indexkey	isinfile
indexord	core_dollars
indexord	core_tot
indexord	list_engine
indexord	total_engine
initscreen	scrollfile
intl_diag	list_center
intl_diag	list_totals
isinfile	e_capital
isinfile	e_internal
isinfile	e_personnel
isinfile	e_pessimism
isinfile	e_revenue
isinfile	list_pers1
isinfile	list_skills
isinfile	move_personnel
isinfile	move_revenue
isinfile	tot_pers1
ispath	copydata
ispath	doload
larray	allocate_costs
larray	amcc2
larray	amcc3
larray	amcc5
larray	amcc_ihoh
larray	balance1
larray	balance2
larray	balance3
larray	bal_part1
larray	bal_part2
larray	intl_diag
larray	list_avg_fact
larray	list_pess1
larray	l_cap
larray	l_exp
larray	l_ext

Called Routine	Calling Routine
larray	l_int
larray	l_main
larray	l_supt
larray	l_wkyrs
larray	org_nets
larray	split_chart
larray	tech_base_ihoh
larray	total_engine
larray	tot_exp1
larray	tot_exp2
larray	tsamt
larray	user_defined
larray	wr_arr
lastrec	browse
lastrec	xbrowse
leftjust	showeditmsg
limit_overtime	sys_edit
list_avg_fact	list_factors
list_cap1	list_capital
list_capital	view_capital
list_center	view_center
list_core1	list_corecap
list_corecap	view_corecap
list_ctda	view_ctda
list_engine	user_defined
list_ext1	list_revenue
list_fac1	list_factors
list_factors	view_foe_factors
list_factors	view_main_factors
list_int1	list_revenue
list_off	view_off
list_orgs	view_orgs
list_pers1	list_personnel
list_personnel	view_personnel
list_pess1	list_pessimism
list_pessimism	dopessimism
list_pess_rev	list_pessimism
list_prod1	list_revenue
list_rates	view_rates
list_rev1	list_revenue
list_revenue	view_revenue
list_revtypes	view_revtypes
list_skills	view_skills
list_sources	view_sources
list_tda1	list_tda_wif
list_tda2	list_tda_wif

Called Routine	Calling Routine
list_tda3	list_tda_wif
list_tda4	list_tda_org
list_tda5	list_tda_wif
list_tda_org	view_tda_org
list_tda_wif	tda_whatif
list_tot1	list_center
list_tot1	list_totals
list_totals	disp_foe
list_wif1	list_tda_org
list_wif1	list_tda_wif
list_wif1	opt_list
loadpsf	user_defined
loadpsf	user_totals
loadrevs	reloaddata
load_foe_factors	bal_part1
load_foe_factors	bal_part2
load_foe_factors	list_avg_fact
load_foe_factors	recalc_tda
load_foe_factors	recomp_foe
load_foe_factors	source_use
load_foe_factors	tech_base_ihoh
load_foe_factors	tot_exp1
load_foe_factors	tot_exp2
load_foe_factors	view_foe_factors
load_foe_factors	view_personnel
load_foe_factors	view_revenue
load_foe_factors	view_totals
load_main_factors	bpm (main)
load_orgs	bpm (main)
ltrim	fillfld
ltrim	get_pic
l_cap	list_tot1
l_exp	list_tot1
l_ext	list_tot1
l_int	intl_diag
l_int	list_tot1
l_main	list_tot1
l_menu	doit
l_supt	list_tot1
l_wkyrs	list_tot1
minmax	e_capital
minmax	e_capmaj
minmax	e_factor
minmax	e_internal
minmax	e_major
minmax	e_per_item

Called Routine	Calling Routine
minmax	e_rev_item
minmax	e_support
minmax	post_repl
minmax	post_totals
minmax	put_totals
minmax	recomp_foe
minmax	view_capmaj
minmax	view_internal
minmax	view_major
minmax	view_support
move_personnel	view_personnel
move_revenue	view_revenue
net_rev	net_revenue
net_rev	recalc_tda
nonzero	allocate_costs
nonzero	anonzero
nonzero	e_per_item
nonzero	put_factor
nonzero	put_totals
nonzero	total_engine
norec	rev_pessimism
norec	tda_whatif
norec	view_capital
norec	view_capmaj
norec	view_ctda
norec	view_internal
norec	view_major
norec	view_off
norec	view_orgs
norec	view_personnel
norec	view_products
norec	view_revenue
norec	view_revtypes
norec	view_skills
norec	view_sources
norec	view_support
norec	view_tda_org
okapn	corefilt
okapn	e_pessimism
okapn	e_revenue
okapn	e_revtype
okapn	list_core1
okapn	list_ext1
okapn	list_int1
okapn	list_pess_rev
okapn	list_prod1

Called Routine	Calling Routine
okapn	list_rev1
okapn	list_revtypes
okapn	tot_carry
okapn	tot_ext1
okapn	tot_int1
okapn	tot_netrev
okapn	tot_pess_rev
okapn	tot_rev1
okapn	tot_rev2
okass	e_capital
okass	e_capmaj
okass	list_cap1
okass	list_ext1
okass	tot_cap1
okass	tot_ext1
okcat	e_pessimism
okcat	e_revenue
okcat	e_revtype
okcat	list_ext1
okcat	list_int1
okcat	list_pess_rev
okcat	list_prod1
okcat	list_rev1
okcat	list_revtypes
okcat	tot_carry
okcat	tot_ext1
okcat	tot_int1
okcat	tot_netrev
okcat	tot_pess_rev
okcat	tot_rev1
okcat	tot_rev2
okdc	list_ext1
okdc	list_int1
okdc	list_prod1
okdc	list_rev1
okdc	tot_carry
okdc	tot_ext1
okdc	tot_int1
okdc	tot_netrev
okdc	tot_rev1
okdc	tot_rev2
okext	list_ext1
okext	tot_ext1
okfact	list_fac1
okfoe	corefilt
okfoe	e_pessimism

Called Routine	Calling Routine
okfoe	list_cap1
okfoe	list_core1
okfoe	list_ctda
okfoe	list_ext1
okfoe	list_fac1
okfoe	list_int1
okfoe	list_orgs
okfoe	list_pers1
okfoe	list_pess_rev
okfoe	list_prod1
okfoe	list_rev1
okfoe	list_wif1
okfoe	tot_cap1
okfoe	tot_carry
okfoe	tot_ext1
okfoe	tot_int1
okfoe	tot_netrev
okfoe	tot_pers1
okfoe	tot_pess_rev
okfoe	tot_rev1
okfoe	tot_rev2
okgrp	e_major
okgrp	e_pessimism
okgrp	e_revenue
okgrp	e_source
okgrp	list_ext1
okgrp	list_int1
okgrp	list_pess_rev
okgrp	list_prod1
okgrp	list_rev1
okgrp	list_sources
okgrp	tot_carry
okgrp	tot_ext1
okgrp	tot_int1
okgrp	tot_netrev
okgrp	tot_pess_rev
okgrp	tot_rev1
okgrp	tot_rev2
okintlab	e_internal
oklab	corefilt
oklab	e_off
oklab	e_pessimism
oklab	list_cap1
oklab	list_core1
oklab	list_ctda
oklab	list_ext1

Called Routine	Calling Routine
oklab	list_fac1
oklab	list_int1
oklab	list_off
oklab	list_orgs
oklab	list_pers1
oklab	list_pess_rev
oklab	list_prod1
oklab	list_rev1
oklab	list_wif1
oklab	move_personnel
oklab	move_revenue
oklab	tot_cap1
oklab	tot_carry
oklab	tot_ext1
oklab	tot_int1
oklab	tot_netrev
oklab	tot_pers1
oklab	tot_pess_rev
oklab	tot_rev1
oklab	tot_rev2
okontda	e_org
okoper	getfiltcond
okorg	e_org
okorg	list_orgs
okpers	e_personnel
okpers	list_pers1
okpers	tot_pers1
okpos	e_personnel
okpos	list_pers1
okpos	tot_pers1
okps	e_personnel
okps	list_pers1
okps	tot_pers1
okskl	e_skill
okskl	list_pers1
okskl	list_skills
okskl	tot_pers1
oksrc	e_major
oksrc	e_pessimism
oksrc	e_revenue
oksrc	e_support
oksrc	list_ext1
oksrc	list_int1
oksrc	list_pess_rev
oksrc	list_prod1
oksrc	list_rev1

Called Routine	Calling Routine
oksrc	list_sources
oksrc	tot_carry
oksrc	tot_ext1
oksrc	tot_int1
oksrc	tot_netrev
oksrc	tot_pess_rev
oksrc	tot_rev1
oksrc	tot_rev2
okwky	list_pers1
okwky	tot_pers1
okyn	list_orgs
oneorgcalc	opt_calc
one_measure	list_tda2
openfile	amcc1
openfile	amcc1a
openfile	amcc1b
openfile	amcc1c
openfile	amcc1d
openfile	amcc1e
openfile	amcc2
openfile	amcc3
openfile	amcc4
openfile	amcc5
openfile	avgload
openfile	balance1
openfile	balance2
openfile	balance3
openfile	checkone
openfile	copyone
openfile	core_dollars
openfile	core_qc
openfile	core_tot
openfile	ctreport
openfile	deleteone
openfile	del_revenue
openfile	disp_foe
openfile	docheck
openfile	doload
openfile	dopack
openfile	dopessimism
openfile	doreindex
openfile	edit_baseyear
openfile	e_revenue
openfile	genload
openfile	get_factor
openfile	get_totals

Called Routine	Calling Routine
openfile	help
openfile	incr_baseyear
openfile	intl_diag
openfile	isinfile
openfile	list_avg_fact
openfile	list_engine
openfile	list_ext1
openfile	list_int1
openfile	list_prod1
openfile	list_tda1
openfile	list_tda3
openfile	list_tda5
openfile	list_wif1
openfile	load_foe_factors
openfile	load_main_factors
openfile	load_orgs
openfile	move_revenue
openfile	oksrc
openfile	opt_personnel
openfile	org_nets
openfile	post_totals
openfile	project_tda
openfile	put_factor
openfile	recalc_all
openfile	recalc_tda
openfile	recomp_center
openfile	recomp_lab
openfile	reloaddata
openfile	removeone
openfile	remove_empty
openfile	remove_unlinked
openfile	rev_pessimism
openfile	source_use
openfile	tda_whatif
openfile	tech_base_ihoh
openfile	total_engine
openfile	tot_cap1
openfile	tot_carry
openfile	tot_exp1
openfile	tot_exp2
openfile	tot_ext1
openfile	tot_int1
openfile	tot_netrev
openfile	tot_pers1
openfile	tot_rev1
openfile	tot_rev2

Called Routine	Calling Routine
openfile	unlinked
openfile	view_capital
openfile	view_capmaj
openfile	view_center
openfile	view_corecap
openfile	view_ctda
openfile	view_foe_factors
openfile	view_internal
openfile	view_main_factors
openfile	view_major
openfile	view_off
openfile	view_orgs
openfile	view_personnel
openfile	view_products
openfile	view_rates
openfile	view_revenue
openfile	view_revtypes
openfile	view_skills
openfile	view_sources
openfile	view_support
openfile	view_tda_org
openfile	whatif
openwindow	alloc_one
openwindow	balance1
openwindow	balance2
openwindow	balance3
openwindow	bpm (main)
openwindow	choose
openwindow	choosemenu
openwindow	compare_wif
openwindow	copydata
openwindow	corefilt
openwindow	core_dollars
openwindow	core_qc
openwindow	core_tot
openwindow	delete_lab
openwindow	del_allbut
openwindow	disp_capital
openwindow	disp_contract
openwindow	disp_expense
openwindow	disp_external
openwindow	disp_foe
openwindow	disp_internal
openwindow	disp_wkyrs
openwindow	dobrowse
openwindow	docheck

Called Routine	Calling Routine
openwindow	doindex
openwindow	dolicense
openwindow	doload
openwindow	dopack
openwindow	dopessimism
openwindow	doproj_org_tda
openwindow	doreindex
openwindow	edit_baseyear
openwindow	e_cpb_ga
openwindow	e_cpb_wky
openwindow	genmsgwindow
openwindow	getfiltcond
openwindow	getthese
openwindow	go_general
openwindow	help
openwindow	incr_baseyear
openwindow	list_avg_fact
openwindow	list_engine
openwindow	list_tot1
openwindow	optselect
openwindow	opt_fact1
openwindow	opt_fact2
openwindow	opt_personnel
openwindow	opt_save
openwindow	org_nets
openwindow	pess_scan
openwindow	project_tda
openwindow	recalc_tda
openwindow	recalc_year
openwindow	recomp_center
openwindow	recomp_lab
openwindow	remove_empty
openwindow	remove_unlinked
openwindow	rev_pessimism
openwindow	showmemory
openwindow	source_use
openwindow	tda_whatif
openwindow	tech_base_ihoh
openwindow	tempupgrade
openwindow	te_tda_org
openwindow	total_engine
openwindow	tot_exp1
openwindow	tot_exp2
openwindow	tsamt
openwindow	upgrone
openwindow	user_defined

Called Routine	Calling Routine
openwindow	user_totals
openwindow	view_capital
openwindow	view_capmaj
openwindow	view_center
openwindow	view_corecap
openwindow	view_ctda
openwindow	view_foe_factors
openwindow	view_internal
openwindow	view_main_factors
openwindow	view_major
openwindow	view_off
openwindow	view_orgs
openwindow	view_personnel
openwindow	view_products
openwindow	view_rates
openwindow	view_revenue
openwindow	view_revtypes
openwindow	view_skills
openwindow	view_sources
openwindow	view_support
openwindow	view_tda_org
openwindow	warnem
openwindow	whatif
optselect	accept_rates
optselect	addnewfile
optselect	alloc_one
optselect	alloc_wky
optselect	bpm (main)
optselect	copydata
optselect	delete_lab
optselect	del_allbut
optselect	del_personnel
optselect	del_record
optselect	del_revenue
optselect	docheck
optselect	doload
optselect	dopack
optselect	dopessimism
optselect	doproj_org_tda
optselect	doreindex
optselect	doviewtext
optselect	eval_toggle
optselect	e_factor
optselect	e_revenue
optselect	incr_baseyear
optselect	list_avg_fact

Called Routine	Calling Routine
optselect	opt_save
optselect	pessfilt
optselect	pess_scan
optselect	post_repl
optselect	project_tda
optselect	recalc_all
optselect	recalc_lab
optselect	recalc_tda
optselect	remove_empty
optselect	remove_unlinked
optselect	setdest
optselect	source_use
optselect	te_tda_org
optselect	tgt_toggle
optselect	toggle_list
optselect	user_defined
optselect	user_totals
optselect	whatif
optselect	xmemo
opt_calc	e_opt_pers
opt_calc	opt_fact
opt_disp	e_opt_pers
opt_fact	opt_personnel
opt_fact1	opt_fact
opt_fact2	opt_fact
opt_list	opt_personnel
opt_list1	opt_list
opt_list2	opt_list
opt_list3	opt_list
opt_personnel	whatif
opt_save	opt_personnel
opt_view	opt_personnel
opt_view1	opt_view
opt_view2	opt_view
opt_view3	opt_view
orblank	e_capital
orblank	e_internal
orblank	e_personnel
orblank	e_revenue
org_measure	list_tda4
org_nets	list_center
pad	amcc2
pad	amcc5
pad	bpm (main)
pad	choosemenu
pad	copydata

Called Routine	Calling Routine
pad	core_dollars
pad	core_tot
pad	ctreport
pad	ctr_toggle
pad	disp_capital
pad	disp_contract
pad	disp_expense
pad	disp_external
pad	disp_foe_guts
pad	disp_internal
pad	disp_wkyrs
pad	doload
pad	doviewtext
pad	getthese
pad	help
pad	intl_diag
pad	ispath
pad	list_center
pad	list_pess1
pad	list_tot1
pad	loadrevs
pad	load_foe_factors
pad	l_main
pad	l_menu
pad	oksrc
pad	one_measure
pad	opt_disp
pad	org_measure
pad	org_nets
pad	post_totals
pad	put_factor
pad	rem_ts_cost
pad	rerecalc_tda
pad	retrieve_costs
pad	show_arr
pad	split_chart
pad	total_engine
pad	tot_exp1
pad	tot_exp2
pad	user_defined
pad	user_totals
pad	view_corecap
pad	writescreen
pcttoggle	opt_personnel
pessfilt	rev_pessimism
pess_adj	pess_scan

Called Routine	Calling Routine
pess_adj_exp	pess_scan
pess_scan	dopessimism
pess_show	alloc_cost
pess_show	dopessimism
popq	corefilt
popq	e_pessimism
popq	user_defined
popq	user_totals
post_repl	view_capital
post_totals	recomp_center
post_totals	recomp_lab
post_totals	rem_ts_cost
project_tda	tda_whatif
proj_org_tda	doproj_org_tda
proj_org_tda	project_tda
put_array	amcc4
put_array	avgload
put_array	e_capital
put_array	e_capmaj
put_array	e_internal
put_array	e_major
put_array	e_per_item
put_array	e_pessimism
put_array	e_rev_item
put_array	e_support
put_array	pess_adj
put_array	pess_adj_exp
put_array	pess_scan
put_array	post_repl
put_array	post_totals
put_array	put_factor
put_array	put_totals
put_array	view_capmaj
put_array	view_internal
put_array	view_major
put_array	view_support
put_cap	e_corecap
put_factor	accept_rates
put_factor	e_factor
put_factor	recomp_foe
put_totals	post_totals
put_totals	recomp_foe
put_totals	recomp_lab
readbkwd	readlinebkwd
readfwd	initscreen
readfwd	readlinefwd

Called Routine	Calling Routine
readjust	source_use
readlinebkwd	readscrbkwd
readlinebkwd	scrollfile
readlinefwd	readscrfwd
readlinefwd	scrollfile
readscrbkwd	scrollfile
readscrfwd	scrollfile
recalc_all	view_center
recalc_lab	disp_foe
recalc_tda	opt_personnel
recalc_tda	tda_whatif
recalc_year	eval_toggle
recalc_year	e_tda_what
recalc_year	opt_save
recalc_year	project_tda
recalc_year	recalc_tda
recalc_yr	proj_org_tda
recalc_yr	recalc_year
recalc_yr	show_overall
recalc_yr	te_tda_org
recalc_yr	view_tda_org
recomp_center	recalc_all
recomp_foe	recomp_lab
recomp_lab	recalc_all
recomp_lab	recalc_lab
reloaddata	e_opt_pers
reloaddata	opt_fact
removeone	remove_empty
remove_empty	sys_edit
remove_unlinked	assorted
rem_old_names	user_defined
rem_old_names	user_totals
rem_ts_cost	recomp_center
rerecalc_tda	recalc_tda
restscreen	closewindow
restscreen	doit
restscreen	scrollfile
retrieve_costs	source_use
retrieve_costs	tech_base_ihoh
retrieve_costs	tot_exp1
retrieve_costs	tot_exp2
rev_pessimism	whatif
savepsf	user_defined
savepsf	user_totals
scrollbkwd	scrollfile
scrollfile	doviewtext

Called Routine	Calling Routine
scrollfwd	scrollfile
sel_foe	view_capital
sel_foe	view_corecap
sel_foe	view_foe_factors
sel_foe	view_personnel
sel_foe	view_revenue
sel_foe	view_totals
sel_lab_foe	copydata
sel_lab_foe	delete_lab
sel_lab_foe	del_allbut
sel_lab_foe	doload
sel_lab_foe	list_avg_fact
sel_lab_foe	pessfilt
setdest	amcc1
setdest	amcc1a
setdest	amcc1b
setdest	amcc1c
setdest	amcc1d
setdest	amcc1e
setdest	amcc2
setdest	amcc3
setdest	amcc4
setdest	amcc5
setdest	amcc_ihoh
setdest	balance1
setdest	balance2
setdest	balance3
setdest	core_dollars
setdest	core_qc
setdest	core_tot
setdest	docheck
setdest	intl_diag
setdest	list_avg_fact
setdest	list_engine
setdest	list_pess1
setdest	list_tda1
setdest	list_tda2
setdest	list_tda3
setdest	list_tda4
setdest	list_tda5
setdest	list_tot1
setdest	opt_list1
setdest	opt_list2
setdest	opt_list3
setdest	org_nets
setdest	source_use

Called Routine	Calling Routine
setdest	tech_base_ihoh
setdest	total_engine
setdest	tot_exp1
setdest	tot_exp2
setdest	tsamt
setdest	win_print
set_salary	e_personnel
shadowwin	openwindow
showdemomsg	bpm (main)
showeditmsg	alloc_one
showeditmsg	alloc_wky
showeditmsg	amccharts
showeditmsg	anal_details
showeditmsg	balances
showeditmsg	baseline
showeditmsg	bpm (main)
showeditmsg	choose
showeditmsg	choosemenu
showeditmsg	compare_wif
showeditmsg	corefilt
showeditmsg	ctr_toggle
showeditmsg	data_entry
showeditmsg	disp_capital
showeditmsg	disp_contract
showeditmsg	disp_expense
showeditmsg	disp_external
showeditmsg	disp_foe
showeditmsg	disp_internal
showeditmsg	disp_wkyrs
showeditmsg	dobrowse
showeditmsg	dopessimism
showeditmsg	doviewtext
showeditmsg	edit_personnel
showeditmsg	edit_revenue
showeditmsg	eval_toggle
showeditmsg	expenses
showeditmsg	e_capital
showeditmsg	e_capmaj
showeditmsg	e_corecap
showeditmsg	e_cpb_ga
showeditmsg	e_cpb_wky
showeditmsg	e_ctda
showeditmsg	e_factor
showeditmsg	e_internal
showeditmsg	e_major
showeditmsg	e_off

Called Routine	Calling Routine
showeditmsg	e_opt_pers
showeditmsg	e_org
showeditmsg	e_personnel
showeditmsg	e_per_item
showeditmsg	e_pessimism
showeditmsg	e_product
showeditmsg	e_revenue
showeditmsg	e_revtype
showeditmsg	e_rev_item
showeditmsg	e_skill
showeditmsg	e_source
showeditmsg	e_support
showeditmsg	e_tda_org
showeditmsg	e_tda_what
showeditmsg	go_corecap
showeditmsg	go_general
showeditmsg	go_spec
showeditmsg	isinfile
showeditmsg	list_capital
showeditmsg	list_center
showeditmsg	list_corecap
showeditmsg	list_factors
showeditmsg	list_personnel
showeditmsg	list_pessimism
showeditmsg	list_rates
showeditmsg	list_revenue
showeditmsg	list_tda_org
showeditmsg	list_tda_wif
showeditmsg	list_totals
showeditmsg	okapn
showeditmsg	okass
showeditmsg	okcat
showeditmsg	okdc
showeditmsg	okext
showeditmsg	okfact
showeditmsg	okfoe
showeditmsg	okgrp
showeditmsg	okintlab
showeditmsg	oklab
showeditmsg	okoper
showeditmsg	okorg
showeditmsg	okpers
showeditmsg	okpos
showeditmsg	okps
showeditmsg	okskl
showeditmsg	oksrc

Called Routine	Calling Routine
showeditmsg	okwky
showeditmsg	okyn
showeditmsg	opt_fact
showeditmsg	opt_fact1
showeditmsg	opt_fact2
showeditmsg	opt_list
showeditmsg	opt_personnel
showeditmsg	opt_view
showeditmsg	rev_pessimism
showeditmsg	sys_edit
showeditmsg	tda_whatif
showeditmsg	te_tda_org
showeditmsg	tgt_toggle
showeditmsg	ts_amounts
showeditmsg	view_capital
showeditmsg	view_capmaj
showeditmsg	view_center
showeditmsg	view_corecap
showeditmsg	view_ctda
showeditmsg	view_foe_factors
showeditmsg	view_internal
showeditmsg	view_main_factors
showeditmsg	view_major
showeditmsg	view_off
showeditmsg	view_orgs
showeditmsg	view_personnel
showeditmsg	view_products
showeditmsg	view_revenue
showeditmsg	view_revtypes
showeditmsg	view_skills
showeditmsg	view_sources
showeditmsg	view_support
showeditmsg	view_tda_org
showeditmsg	whatif
showrecmsg	eval_toggle
showrecmsg	tda_whatif
showrecmsg	te_tda_org
showrecmsg	tgt_toggle
showrecmsg	view_tda_org
show_arr	user_defined
show_arr	user_totals
show_line_length	user_defined
show_overall	e_tda_org
show_overall	te_tda_org
show_overall	tgt_toggle
source_use	expenses

Called Routine	Calling Routine
split_chart	amcc1
split_chart	amcc1a
split_chart	amcc1b
split_chart	amcc1c
split_chart	amcc1d
split_chart	amcc1e
split_chart	amcc4
sys_edit	bpm (main)
tda_eval_comp	e_tda_what
tda_eval_comp	recalc_yr
tda_tot	e_tda_org
tda_tot_comp	proj_org_tda
tda_tot_comp	recalc_year
tda_tot_comp	recalc_yr
tda_tot_comp	tda_tot
tda_tot_comp	te_tda_org
tda_tot_comp	te_tot
tda_tot_comp	tgt_toggle
tda_whatif	whatif
tech_base_ihoh	expenses
tempupgrade	upgrade
te_tda_org	view_tda_org
te_tot	te_tda_org
tgt_toggle	view_tda_org
time	list_engine
time	total_engine
tlbr	alloc_one
tlbr	corefilt
tlbr	disp_capital
tlbr	disp_contract
tlbr	disp_expense
tlbr	disp_external
tlbr	disp_foe
tlbr	disp_internal
tlbr	disp_wkyrs
tlbr	dopessimism
tlbr	e_cpb_ga
tlbr	e_cpb_wky
tlbr	go_general
tlbr	opt_fact1
tlbr	opt_fact2
tlbr	opt_personnel
tlbr	rev_pessimism
tlbr	tda_whatif
tlbr	te_tda_org
tlbr	user_defined

Called Routine	Calling Routine
tlbr	user_totals
tlbr	view_capital
tlbr	view_capmaj
tlbr	view_center
tlbr	view_corecap
tlbr	view_ctda
tlbr	view_foe_factors
tlbr	view_internal
tlbr	view_main_factors
tlbr	view_major
tlbr	view_off
tlbr	view_orgs
tlbr	view_personnel
tlbr	view_products
tlbr	view_rates
tlbr	view_revenue
tlbr	view_revtypes
tlbr	view_skills
tlbr	view_sources
tlbr	view_support
tlbr	view_tda_org
toggle_list	assorted
total_engine	user_totals
tot_cap1	list_capital
tot_cap1	list_center
tot_cap1	list_totals
tot_carry	list_revenue
tot_ctda	e_ctda
tot_exp1	expenses
tot_exp2	expenses
tot_ext1	list_center
tot_ext1	list_revenue
tot_ext1	list_totals
tot_int1	list_center
tot_int1	list_revenue
tot_int1	list_totals
tot_netrev	list_revenue
tot_pers1	list_center
tot_pers1	list_personnel
tot_pers1	list_totals
tot_pess_rev	list_pessimism
tot_rev1	list_center
tot_rev1	list_revenue
tot_rev1	list_totals
tot_rev2	list_center
tot_rev2	list_revenue

Called Routine	Calling Routine
tot_rev2	list_totals
tsamt	ts_amounts
ts_amounts	list_center
unlinked	remove_unlinked
upgrade	bpm (main)
upgrone	upgrade
user_defined	list_cap1
user_defined	list_core1
user_defined	list_ctda
user_defined	list_ext1
user_defined	list_fac1
user_defined	list_int1
user_defined	list_off
user_defined	list_orgs
user_defined	list_pers1
user_defined	list_pess_rev
user_defined	list_prod1
user_defined	list_rev1
user_defined	list_revtypes
user_defined	list_skills
user_defined	list_sources
user_defined	list_wif1
user_totals	tot_cap1
user_totals	tot_carry
user_totals	tot_ext1
user_totals	tot_int1
user_totals	tot_netrev
user_totals	tot_pers1
user_totals	tot_pess_rev
user_totals	tot_rev1
user_totals	tot_rev2
view_capital	data_entry
view_capmaj	edit_revenue
view_center	baseline
view_corecap	data_entry
view_ctda	sys_edit
view_foe_factors	data_entry
view_foe_factors	disp_foe
view_internal	edit_revenue
view_main_factors	sys_edit
view_major	edit_revenue
view_off	sys_edit
view_orgs	sys_edit
view_personnel	data_entry
view_products	edit_revenue
view_rates	baseline

Called Routine	Calling Routine
view_revenue	data_entry
view_revtypes	sys_edit
view_skills	sys_edit
view_sources	sys_edit
view_support	edit_revenue
view_tda_org	tda_whatif
view_totals	baseline
warnem	accept_rates
warnem	avgload
warnem	bpm (main)
warnem	checklevel
warnem	copydata
warnem	doload
warnem	doviewtext
warnem	edit_foe_factors
warnem	edit_personnel
warnem	edit_revenue
warnem	est_factor
warnem	e_corecap
warnem	e_cpb_ga
warnem	findkey
warnem	genload
warnem	go_general
warnem	go_spec
warnem	incr_baseyear
warnem	isinfile
warnem	ispath
warnem	nofunction
warnem	norec
warnem	okontda
warnem	oksrc
warnem	opt_calc
warnem	proj_org_tda
warnem	setdest
warnem	toggle_list
warnem	upgrade
warnem	user_defined
warnem	user_totals
warnem	view_corecap
warnem	view_foe_factors
whatif	bpm (main)
whatsnext	del_personnel
whatsnext	del_record
whatsnext	del_revenue
whatsnext	move_personnel
whatsnext	move_revenue

Called Routine	Calling Routine
wiforg	compare_wif
wiforg	defineallfiles
wiforg	list_tda1
wiforg	list_tda3
wiforg	list_tda5
wiforg	org_measure
wiforg	project_tda
wiforg	proj_org_tda
wiforg	te_tda_org
wiforg	view_tda_org
win_print	list_capital
win_print	list_center
win_print	list_corecap
win_print	list_factors
win_print	list_personnel
win_print	list_pessimism
win_print	list_rates
win_print	list_revenue
win_print	list_tda_org
win_print	list_tda_wif
win_print	list_totals
win_print	opt_list
wkyrs_adjust	tda_tot_comp
wkyrs_adjust	te_tda_org
writescreen	scrollfile
wr_arr	addupfactors
wr_arr	alloc_one
wr_arr	disp_capital
wr_arr	disp_contract
wr_arr	disp_expense
wr_arr	disp_external
wr_arr	disp_foe_guts
wr_arr	disp_internal
wr_arr	disp_rates
wr_arr	disp_wkyrs
wr_arr	dopessimism
wr_arr	e_personnel
wr_arr	e_revenue
wr_arr	pess_show
wr_arr	view_foe_factors
wr_arr	view_main_factors
wr_duls	disp_capital
wr_duls	disp_contract
wr_duls	disp_expense
wr_duls	disp_external
wr_duls	disp_foe_guts

Called Routine	Calling Routine
wr_duls	disp_internal
wr_duls	disp_rates
wr_duls	disp_wkyrs
wr_duls	e_capital
wr_duls	e_personnel
wr_duls	e_revenue
wr_uls	alloc_one
wr_uls	disp_capital
wr_uls	disp_contract
wr_uls	disp_expense
wr_uls	disp_external
wr_uls	disp_foe_guts
wr_uls	disp_internal
wr_uls	disp_rates
wr_uls	disp_wkyrs
wr_uls	dopessimism
wr_uls	e_capital
wr_uls	e_capmaj
wr_uls	e_cpb_ga
wr_uls	e_cpb_wky
wr_uls	e_internal
wr_uls	e_major
wr_uls	e_personnel
wr_uls	e_pessimism
wr_uls	e_revenue
wr_uls	e_support
wr_uls	pess_show
wr_uls	view_foe_factors
wr_uls	view_main_factors
wr_years	alloc_one
wr_years	disp_capital
wr_years	disp_contract
wr_years	disp_expense
wr_years	disp_external
wr_years	disp_foe_guts
wr_years	disp_internal
wr_years	disp_rates
wr_years	disp_wkyrs
wr_years	dopessimism
wr_years	e_capital
wr_years	e_capmaj
wr_years	e_cpb_ga
wr_years	e_cpb_wky
wr_years	e_internal
wr_years	e_major
wr_years	e_personnel

Called Routine	Calling Routine
wr_years	e_pessimism
wr_years	e_revenue
wr_years	e_support
wr_years	view_foe_factors
wr_years	view_main_factors
years	amcc2
years	amcc3
years	amcc5
years	amcc_ihoh
years	bal_part1
years	bal_part2
years	intl_diag
years	list_avg_fact
years	list_pess1
years	l_cap
years	l_exp
years	l_ext
years	l_int
years	l_main
years	l_supt
years	l_wkyrs
years	org_nets
years	split_chart
years	tech_base_ihoh
years	total_engine
years	tot_exp1
years	tot_exp2
years	tsamt
years	user_defined
years	wr_years
zaprecord	add_rec