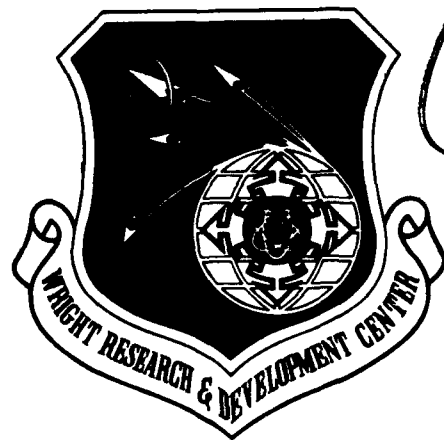


WRDC-TR-90-8007
Volume VIII
Part 24

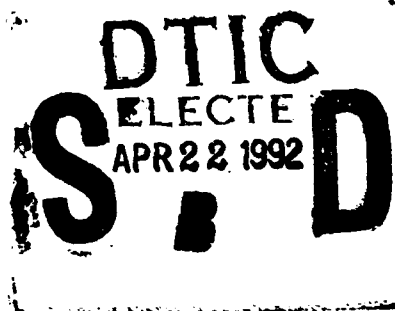
AD-A248 931



INTEGRATED INFORMATION SUPPORT SYSTEM (IISS)
Volume VIII - User Interface Subsystem
Part 24 - Report Writer Product Specification

S. Barker

Control Data Corporation
Integration Technology Services
2970 Presidential Drive
Fairborn, OH 45324-6209



September 1990

Final Report for Period 1 April 1987 - 31 December 1990

Approved for Public Release; Distribution is Unlimited

92-10290

MANUFACTURING TECHNOLOGY DIRECTORATE
WRIGHT RESEARCH AND DEVELOPMENT CENTER
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6533

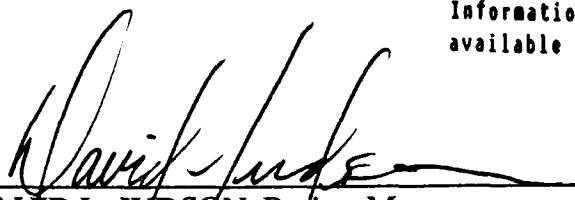
92 4 21 128

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever, regardless whether or not the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data. It should not, therefore, be construed or implied by any person, persons, or organization that the Government is licensing or conveying any rights or permission to manufacture, use, or market any patented invention that may in any way be related thereto.

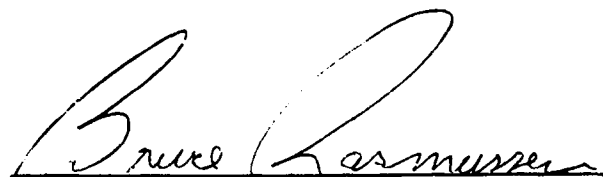
This technical report has been reviewed and is approved for publication.

This report is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations


DAVID L. JUDSON, Project Manager
WRDC/MTI
Wright-Patterson AFB, OH 45433-6533

25 July 91
DATE

FOR THE COMMANDER:


BRUCE A. RASMUSSEN, Chief
WRDC/MTI
Wright-Patterson AFB, OH 45433-6533

25 July 91
DATE

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify WRDC/MTI, Wright-Patterson Air Force Base, OH 45433-6533 to help us maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified		1b. RESTRICTIVE MARKINGS	
2a. SECURITY CLASSIFICATION AUTHORITY		3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for Public Release; Distribution is Unlimited.	
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE			
4. PERFORMING ORGANIZATION REPORT NUMBER(S) UTP620344402		5. MONITORING ORGANIZATION REPORT NUMBER(S) WRDC-TR-90-8007 Vol. VIII, Part 24	
6a. NAME OF PERFORMING ORGANIZATION Control Data Corporation; Integration Technology Services	6b. OFFICE SYMBOL (if applicable)	7a. NAME OF MONITORING ORGANIZATION WRDC/MTI	
6c. ADDRESS (City, State, and ZIP Code) 2970 Presidential Drive Fairborn, OH, 45324-6209		7b. ADDRESS (City, State, and ZIP Code) WPAFB, OH 45433-6533	
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Wright Research and Development Center, Air Force Systems Command, USAF	8b. OFFICE SYMBOL (if applicable) WRDC/MTI	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUM. F33600-87-C-0464	
8c. ADDRESS (City, State, and ZIP Code) Wright-Patterson AFB, Ohio 45433-6533		10. SOURCE OF FUNDING NOS.	
11. TITLE See block 19		PROGRAM ELEMENT NO. 78011F	PROJECT NO. 595600
		TASK NO. F95600	WORK UNIT NO. 20950607
12. PERSONAL AUTHOR(S) Structural Dynamics Research Corporation: Barker, S., Glandorf, F., et al.			
13a. TYPE OF REPORT Final Report	13b. TIME COVERED 4 / 1 / 87 - 12 / 31 / 90	14. DATE OF REPORT (Yr., Mo., Day) 1990 September 30	15. PAGE COUNT 450
16. SUPPLEMENTARY NOTES WRDC/MTI Project Priority 6203			
17. COSATI CODES		18. SUBJECT TERMS (Continue on reverse if necessary and identify block no.)	
FIELD	GROUP	SUB GR.	
1308	0905		
19. ABSTRACT (Continue on reverse if necessary and identify block number) This specification establishes the detailed design of the Report Writer computer program. BLOCK 11: INTEGRATED INFORMATION SUPPORT SYSTEM Vol VIII -User Interface Subsystem Part 24 - Report Writer Product Specification			
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED x SAME AS RPT. DTIC USERS		21. ABSTRACT SECURITY CLASSIFICATION Unclassified	
22a. NAME OF RESPONSIBLE INDIVIDUAL David L. Judson		22b. TELEPHONE NO. (Include Area Code) (513) 255-7371	22c. OFFICE SYMBOL WRDC/MTI

FOREWORD

This technical report covers work performed under Air Force Contract F33600-87-C-0464, DAPro Project. This contract is sponsored by the Manufacturing Technology Directorate, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio. It was administered under the technical direction of Mr. Bruce A. Rasmussen, Branch Chief, Integration Technology Division, Manufacturing Technology Directorate, through Mr. David L. Judson, Project Manager. The Prime Contractor was Integration Technology Services, Software Programs Division, of the Control Data Corporation, Dayton, Ohio, under the direction of Mr. W. A. Osborne. The DAPro Project Manager for Control Data Corporation was Mr. Jimmy P. Maxwell.

The DAPro project was created to continue the development, test, and demonstration of the Integrated Information Support System (IISS). The IISS technology work comprises enhancements to IISS software and the establishment and operation of IISS test bed hardware and communications for developers and users.

The following list names the Control Data Corporation subcontractors and their contributing activities:

SUBCONTRACTOR

ROLE

Control Data Corporation	Responsible for the overall Common Data Model design development and implementation, IISS integration and test, and technology transfer of IISS.
D. Appleton Company	Responsible for providing software information services for the Common Data Model and IDEF1X integration methodology.
ONTEK	Responsible for defining and testing a representative integrated system base in Artificial Intelligence techniques to establish fitness for use.
Simpact Corporation	Responsible for Communication development.
Structural Dynamics Research Corporation	Responsible for User Interfaces, Virtual Terminal Interface, and Network Transaction Manager design, development, implementation, and support.
Arizona State University	Responsible for test bed operations and support.

TABLE OF CONTENTS

		<u>Page</u>
SECTION 1.0	SCOPE	1-1
1.1	Identification	1-1
1.2	Functional Summary	1-1
SECTION 2.0	DOCUMENTS	2-1
2.1	Reference Documents	2-1
2.2	Terms and Abbreviations	2-2
SECTION 3.0	REQUIREMENTS	3-1
3.1	Structural Description	3-1
3.1.1	RWG	3-1
3.1.2	HRW	3-2
3.2	Functional Flow	3-2
3.3	Interfaces	3-4
3.3.1	Forms Language Compiler	3-4
3.3.2	CDM Data Dictionary	3-4
3.3.3	Generated Report Interfaces	3-4
3.4	Program Interrupts	3-4
3.5	Timing and Sequencing Description ..	3-4
3.6	Special Control Features	3-4
3.7	Storage Allocation	3-5
3.7.1	Data Base Definition	3-5
3.7.1.1	File Descriptions	3-5
3.7.1.2	Table Description	3-7
3.8	Object Code Creation	3-8
3.9	Adaptation Data	3-8
3.10	Detailed Design Description	3-8
3.10.1	Main Program List	3-8
3.10.2	Module List	3-10
3.10.3	External Routines List	3-20
3.10.4	Include File List	3-23
3.10.5	Where Include File Used List	3-25
3.10.6	Where External Routine Used List..	3-52
3.10.7	Main Program Parts List	3-69
3.10.8	Module Documentation	3-78
3.10.9	Include File Description	3-368
3.10.10	Hierarchy Chart	3-384
3.11	Program Listings Comments	3-437
SECTION 4.0	QUALITY ASSURANCE PROVISIONS	4-1
4.1	Introduction and Definitions	4-1
4.2	Computer Programming and Test Evaluation	4-1

LIST OF ILLUSTRATIONS

<u>Figure</u>	<u>Title</u>	<u>Page</u>
3-1	RWG Structure	3-2
3-2	Report Writer Environment Data Flow ..	3-3
3-3	Hierarchical Report Writer Data Flow..	3-4

Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	



SECTION 1

SCOPE

1.1 Identification

This specification establishes the detailed design of a computer program identified as the Report Writer hereinafter referred to as RW. RW is one configuration item of the Integrated Information Support System (IISS) User Interface (UI).

1.2 Functional Summary

The RW is used to report selected information stored in the database accessible through the Common Data Model (CDM).

The major functions of the RW are:

1. The placement and formatting of fixed textual information and database information, i.e., CDM data.
2. The summarization of simple statistical attributes of the reported information such as counts, sums, and averages.
3. The retrieval of data from the CDM.

SECTION 2

DOCUMENTS

2.1 Reference Documents

- [1] Structural Dynamics Research Corporation, Application Interface Product Specification, PS 620144700 , 1 November 1985.
- [2] Structural Dynamics Research Corporation, Forms Driven Form Editor Product Specification, PS 620144402 , 1 November 1985.
- [3] Structural Dynamics Research Corporation, Form Processor Product Specification, PS 620144200 , 1 November 1985.
- [4] Structural Dynamics Research Corporation, Forms Language Compiler Product Specification, PS 620144401 , 1 November 1985.
- [5] Structural Dynamics Research Corporation, Rapid Application Generator Product Specification, PS 620144502 , 1 November 1985.
- [6] Structural Dynamics Research Corporation, Text Editor Product Specification, PS 620144600 , 1 November 1985.
- [7] Structural Dynamics Research Corporation, User Interface Services Product Specification, PS 620144100 , 1 November 1985.
- [8] Structural Dynamics Research Corporation, Virtual Terminal Product Specification, PS 620144300 , 1 November 1985.
- [9] Structural Dynamics Research Corporation, Report Writer Development Specification, DS 620144501 , 1 November 1985.
- [10] Structural Dynamics Research Corporation, Report Writer Unit Test Plan, UTP620144501 , 1 November 1985.
- [11] Structural Dynamics Research Corporation, Report Writer User Manual, UM 620144501 , 1 November 1985.

2.2 Terms and Abbreviations

Application Generator: (AG), subset of the IISS User Interface that consists of software modules that generate IISS application code and associated form definitions based on a language input. The part of the AG that generates report programs is called the Report Writer. The part of the AG that generates interactive applications is called the Rapid Application Generator.

Application Interface: (AI), subset of the IISS User Interface that consists of the callable routines that are linked with applications that use the Form Processor or Virtual Terminal. The AI enables applications to be hosted on computers other than the host of the User Interface.

Application Process: (AP), a cohesive unit of software that can be initiated as a unit to perform some function or functions.

Attribute: field characteristic such as blinking, highlighted, black, etc. and various other combinations. Background attributes are defined for forms or windows only. Foreground attributes are defined for items. Attributes may be permanent, i.e., they remain the same unless changed by the application program, or they may be temporary, i.e., they remain in effect until the window is redisplayed.

Common Data Model: (CDM), IISS subsystem that describes common data application process formats, form definitions, etc. of the IISS and includes conceptual schema, external schemas, internal schemas, and schema transformation operators.

Computer Program Configuration Item: (CPCI), an aggregation of computer programs or any of their discrete portions which satisfies an end-use function.

Conceptual Schema: (CS), the standard definition used for all data in the CDM. It is based on IDEF1 information modelling.

Device Drivers: (DD), software modules written to handle I/O for a specific kind of terminal. The modules map terminal specific commands and data to a neutral format. Device Drivers are part of the UI Virtual Terminal.

Display List: is similar to the open list, except that it contains only those forms that have been added to the screen and are currently displayed on the screen.

External Schema: (ES), an application's view of the CDM's conceptual schema.

Field: two dimensional space on a terminal screen.

Form: structured view which may be imposed on windows or other forms. A form is composed of fields. These fields may be defined as forms, items, and windows.

Form Definition: (FD), forms definition language after compilation. It is read at runtime by the Form Processor.

Forms Definition Language: (FDL), the language in which electronic forms are defined.

Forms Driver Form Editor: (FD FE), subset of the FE which consists of a forms driven application used to create Form Definition files interactively.

Form Editor: (FE), subset of the IISS User Interface that is used to create definitions of forms. The FE consists of the Forms Driven Form Editor and the Forms Language Compiler.

Form Hierarchy: a graphic representation of the way in which forms, items and windows are related to their parent form.

Forms Language Compiler: (FLAN), subset of the FE that consists of a batch process that accepts a series of forms definition language statements and produces form definition files as output.

Form Processor: (FP), subset of the IISS User Interface that consists of a set of callable execution time routines available to an application program for form processing.

IISS Function Screen: the first screen that is displayed after logon. It allows the user to specify the function he wants to access and the device type and device name on which he is working.

Integrated Information Support System: (IISS), a test computing environment used to investigate, demonstrate and test the concepts of information management and information integration in the context of Aerospace Manufacturing. The IISS addresses the problems of integration of data resident on heterogeneous data bases supported by heterogeneous computers interconnected via a Local Area Network.

Item: non-decomposable area of a form in which hard-coded descriptive text may be placed and the only defined areas where user data may be input/output.

Message: descriptive text which may be returned in the standard message line on the terminal screen. They are used to warn of errors or provide other user information.

Message Line: a line on the terminal screen that is used to display messages.

Network Transaction Manager: (NTM), IISS subsystem that performs the coordination, communication and housekeeping functions required to integrate the Application Processes and System Services resident on the various hosts into a cohesive system.

Neutral Data Manipulation Language: (NDML), the command language by which the CDM is accessed for the purpose of extracting, deleting, adding, or modifying data.

Operating System: (OS), software supplied with a computer which allows it to supervise its own operations and manage access to hardware facilities such as memory and peripherals.

Page: instance of forms in windows that are created whenever a form is added to a window.

Paging and Scrolling: a method which allows a form to contain more data than can be displayed with provisions for viewing any portion of the data buffer.

Physical Device: a hardware terminal.

Presentation Schema: (PS), may be equivalent to a form. It is the view presented to the user of the application.

Qualified Name: the name of a form, item or window preceded by the hierarchy path so that it is uniquely identified.

Report Definition Language: (RDL), an extension of the Forms Definition Language that includes retrieval and calculation of database information and is used to define reports.

Report Writer: (RW), part of the Application Generator that generates source code for report programs based on a language input.

Report Writer Generator: (RWG), used to translate report definitions defined using the RDL into programs that access data bases via the CDM.

Subform: a form that is used within another form.

User Interface: (UI), IISS subsystem that controls the user's terminal and interfaces with the rest of the system. The UI consists of two major subsystems: the User Interface Development System (UIDS) and the User Interface Management System (UIMS).

User Interface Development System: (UIDS), collection of IISS User Interface subsystems that are used by applications programmers as they develop IISS applications. The UIDS includes the Form Editor and the Application Generator.

Window: dynamic area of a terminal screen on which predefined forms may be placed at run time.

SECTION 3
REQUIREMENTS

3.1 Structural Description

The Report Writer consists of the Report Writer Generator (RWG) and the Hierarchical Report Writer (HRW).

3.1.1 RWG

The RWG is used to translate report definitions defined using the Report Definition Language (RDL) into programs that access data bases via the CDM and report the extracted data in a formatted way. Conceptually, the RWG is a compiler that takes RDL as input and generates:

- o Binary form definition files that determine the layout of the report pages by parcing the RDL using the modules YTAB.C, FLANSP.C, and WRTFRM.C.
- o A data base query program that maps the CDM external schema to the presentation schema (forms defined by the FD files). The module NDMLGEN.C calls the COBOL module CDMESQY.PRC to get meta data about the report query from the CDM data dictionary to check for legal schema mappings. Illegal mappings are recorded in a warning file. This program also generates the appropriate NDML to do the query and must be precompiled using the NDML precompiler.
- o A control flow program based on the specified conditions. This is the main module of the generated report that uses the Application Interface to put data from the CDM into the report forms and arrange the printed output.

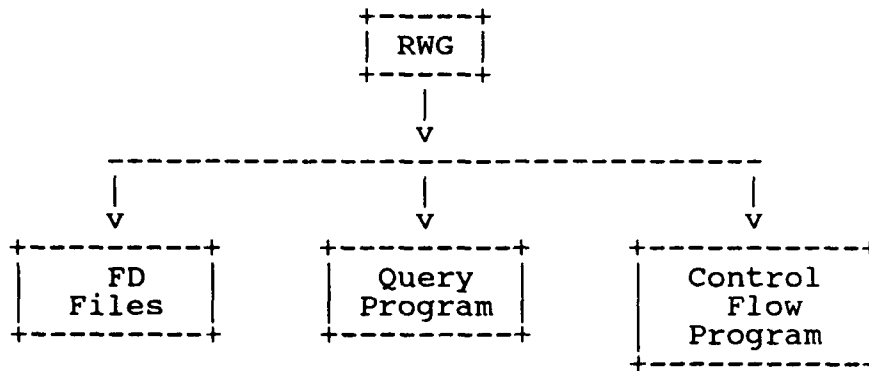


Figure 3-1. RWG Structure

3.1.2 HRW

The HRW is a post processor which takes a report generated by the Report Writer Generator and rearranges it into an appropriate tree structure. The data to be displayed can be either a true hierarchy where each box appears only once or a network where a box may appear more than once.

3.2 Functional Flow

Figure 3-2 is a data flow diagram of the Report Writer environment.

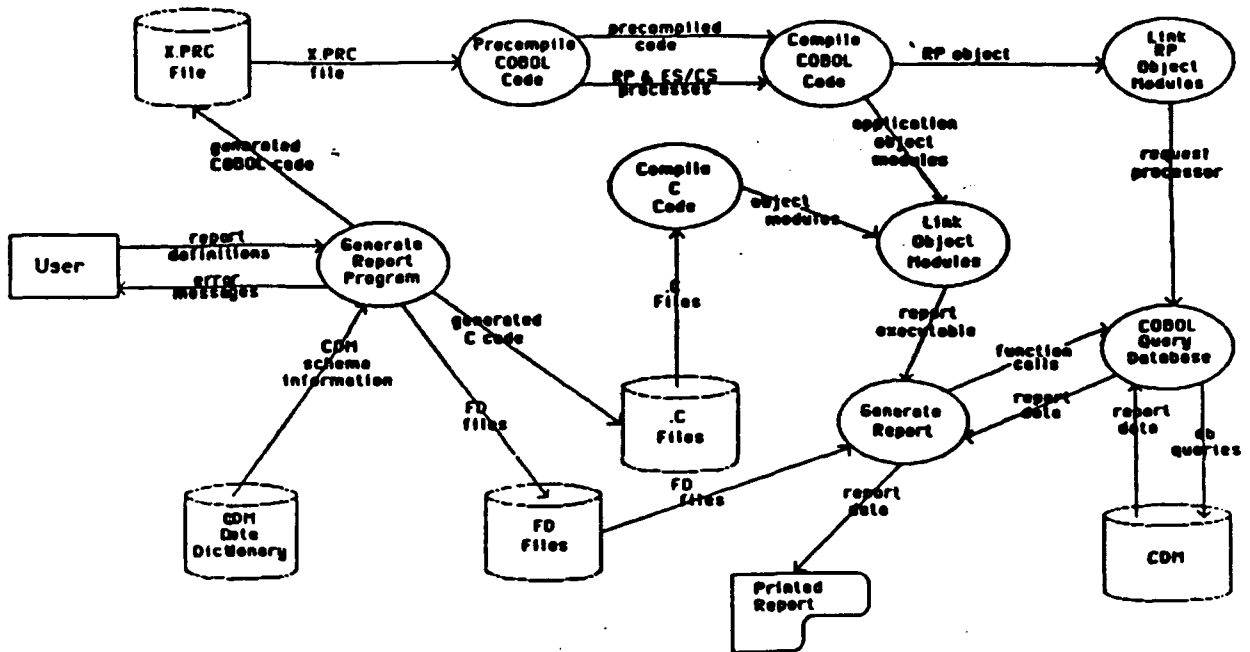


Figure 3-2. Report Writer Environment Data Flow

Figure 3-3 is a data flow diagram of the Hierarchical Report Writer.



Figure 3-3. Hierarchical Report Writer Data Flow

3.3 Interfaces

3.3.1 Forms Language Compiler

The RWG uses the modules YTAB.C, FLANSP.C, and WRTFRM.C of the Forms Language Compiler (FLAN) to generate binary form definition files from its RDL input file. FLAN also produces the internal data structures used to generate the control flow program.

3.3.2 CDM Data Dictionary

The data base query program extracts meta data about the report query from the CDM Data Dictionary to check for legal External Schema to Presentation Schema mappings.

3.3.3 Generated Report Interfaces

The generated report is like any other IISS application. It interfaces with the User Interface via the Application Interface and the CDM via the CDMP calls generated by the NDML precompiler. All applications that use the CDMP or the Application Interface of the UI also interface with the NTM.

3.4 Program Interrupts

This section does not apply to the detailed design of the Report Writer.

3.5 Timing and Sequencing Description

This section does not apply to the detailed design of the Report Writer.

3.6 Special Control Features

This section does not apply to the detailed design of the Report Writer.

3.7 Storage Allocation

The Report Writer Generator executable size is 492 blocks.

3.7.1 Data Base Definition

3.7.1.1 File Descriptions

1. FILE NAME: formname.FD - Form Definition file. The name of this file is dependent upon the form it describes.

PURPOSE: This file contains information about the structure and attributes of a form that is used at run time by the Form Processor.

DECLARATION:

```
typedef struct      /* version number record */
{
    char rectyp;      /* '1' */
    int  vernum;      /* current version number (2) */
    char linefeed;
} VERREC;

typedef struct      /* form record */
{
    char  form_name[10]; /* form name */
    char  background[10]; /* background name */
    short row;          /* starting row */
    short col;          /* starting col */
    short width;        /* width */
    short depth;        /* depth */
    short n_txtflds;     /* number of text fields */
    short n_datflds;     /* number of data fields */
    short s_txtbuf;      /* size of the text buffer */
    short s_defbuf;      /* size of the default buffer */
}
```

```
    char linefeed;
    } FRMREC;

typedef struct      /* text record */
{
    short row;      /* starting row */
    short col;      /* starting col */
    short len;      /* total length */
    char linefeed;
} TXTREC;

typedef struct      /* field record */
{
    char fld_name[10]; /* field name */
    char fld_type;     /* field type (F, I, W, A) */
    short row;         /* starting row */
    short col;         /* starting col */
    short width;       /* field width */
    short depth;       /* field depth */
    int min_value;     /* minimum value (if any) */
    int max_value;     /* maximum value (if any) */
    char help_line[80]; /* help text */
    char disp_att[10]; /* display attribute */
    short n_formats;   /* number of formats */
    char format[12][2]; /* format strings */
    short n_arydefs;   /* number of dimensions */
    struct /* dimension specification */
    {
        char dir;      /* repeat direction (H, V) */
        short cnt;     /* actual repeat count */
        short sp;      /* number of spaces between
                        repetitions */
        short dsp_size; /* display repeat count */
    } array_def[3];
    char linefeed;
} FLDREC;
```

2. FILE NAME: generated using the CDM file namer program with a TMP extension - the generated COBOL program processes the results of the NDML select and creates this Presentation Schema format file of the report data.

PURPOSE: This file is a temporary file that pertains to the current report query. It is input to the generated control flow program to produce the printed report and becomes obsolete after the report is generated. If the report

program terminates abnormally, this file may be examined to help determine the cause.

DECLARATION: The module GENDB.C generates a character type declaration based on the Presentation Schema sizes of the selected columns.

3. FILE NAME: *C.C - where * is the report name as given on the CREATE REPORT statement of the RDL file - generated C code.

PURPOSE: This is the control flow program generated by the RWG that uses the Application Interface to put data from the CDM into the report forms and arrange the printed output.

DECLARATION: Character (i.e., PIC X(80). in COBOL)

4. FILE NAME: *X.PRC where * is the report name as given on the CREATE REPORT statement of the RDL file - generated COBOL code that contains:

- External Schema COBOL record structures
- Presentation Schema COBOL record structures
- Machine Representation Conversion code

PURPOSE: This code contains the CDM query procedures to do the report query specified by the NDML SELECT and maps the External Schema to the Presentation Schema.

DECLARATION: Character (i.e., PIC X(80). in COBOL)

5. FILE NAME: *.WRN where * is the report name as given on the CREATE REPORT statement of the RDL file - generated error file listing any inconsistencies in the External to Presentation Schema mapping.

PURPOSE: This file should be examined by the developer to verify inconsistencies in form item sizes and external schema data.

DECLARATION: Character (i.e., PIC X(80). in COBOL)

3.7.1.2 Table Description

The database tables accessed by the RWG are under the control of the CDM and are predefined.

3.8 Object Code Creation

The RWG routines were compiled using a C compiler developed by Interactive Software under VAX/VMS. The generated C programs can be compiled using the same compiler. The generated COBOL program can be compiled using any ANSI COBOL compiler.

3.9 Adaptation Data

The C source modules for the RWG can be compiled using any UNIX version 7 compatible C compiler. The generated COBOL code must be precompiled using the NDML precompiler before being compiled by the COBOL compiler.

3.10 Detailed Design Description

3.10.1 Main Program List

The following is a list of all "Main Programs" which are modules that are not called by any other module being documented here. These modules are either program entry points or, if they are hooked into another set of programs via subroutine calls, they are the points the external programs can call and therefore enter through. To differentiate between the two types of entry points, look at the individual Module Documentation (section 3.10.8) and look at Module Type for each of the Main Program modules listed. Note whether the routine is a Program, Subroutine, or Function. If it is a Program, it is truly a main program entry point. If not, then it is merely called by other programs not being documented here.

REPORT WRITER Main Program List

Module Name -----	Purpose -----
GRP/MAIN	GENERATE APPLICATION/REPORT PROGRAM
HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER

3.10.2 Module List

The following is a list of all the modules being documented here along with their purpose. Each module has a unique name, no matter what language it was written in.

REPORT WRITER Module List

Module Name -----	Purpose -----
ACTRSV	ACTION RESOLVE
ADDCHK	ADD POSITION TO CHECK LIST
ARRANGE	ARRANGE CHART AND ASSIGNS PAGE NUMBERS
ASSIGN	ASSIGN FILE SECTION
BLDMOD	BUILD MODULE
BLDNODE	BUILD NODE
BLDSUB	BUILD SUBROUTINES
BSCODE	BUILD SUBROUTINE CODE
CALCSTAT	CALCULATE STATISTIC
CCONV	C CONVERSIONS
CDMESQY	PROGRAM NAME CDMESQY
CES	C ES
CESPS	C ES TO PS
CHKARY	CHECK ARRAY
CHKFLD	CHECK FIELD
CHKFRM	CHECK FORM
CHKGRP	CHECK FOR GROUP SEPERATORS OR END OF FILE
CHKSIZE	CHECK SIZE OF ITEMS DOING CONVERSIONS ON
CLOSEGAP	CLOSE GAP IN TREE
CLRNDP	CLEAR NODUPLICATE FIELDS
CLSFIL	CLOSE FILES

REPORT WRITER Module List

Module Name -----	Purpose -----
COBCONV	COBOL CONVERSIONS
COBES	COBOL ES RECORD
COBESPS	COBOL ES TO PS
COBPE	COBOL PE
COPYNODE	COPY A NODE IN TREE
CPE	C PE
CSTASH	CHARACTER STASH
CTLRSV	CONTROL RESOLVE
DASH	WRITE DASH '-'
DATAGEN	DATA DIVISION GENERATE
DBFREAD	GENERATE DATA BASE FREAD
DCLINDX	DECLARE INDEX VARIABLES
DELNODE	DELETE A SPECIFIED NODE IN TREE
DOINDEX	DO CHART INDEX
DRAWLEV	DRAW A LEVEL OF THE CHART
ENDGEN	END GERNERATE
ERROR	ISSUE ERROR MESSAGE
ESPSMAP	THE EXTERNAL SCHEMA AND PRESENTATION SCHEMA MAPPING
ESPSMAP/INDENT	INDENT
FATAL	ISSUE FATAL ERROR MESSAGE

REPORT WRITER Module List

Module Name -----	Purpose -----
FD	FD SECTION DECLARATIONS
FILELNK	FILE LINKAGE SECTION GENERATE
FLANCI	FLAN CALLABLE INTERFACE
FLDRSV	FIELD RESOLVE
FLDTYP	FIELD TYPE
FNDATT	FIND ATTRIBUTE
FNDFRM	FIND FORM
FRMPDAT	FORM PDATA
FRNTND	FRONT END FOR FORMS
GEN	GENERATE A LINE OF CODE
GENAA	GENERATE PROCEDURE "ADDACT" ADD AN ACTION
GENAAL	GENERATE PROCEDURE "ADDAL" ADD ACTION LIST
GENACT	GENERATE ACTIONS
GENAE	GENERATE ACTION EXIT
GENAH	GENERATE ACTION HELP
GENAI	GENERATE ACTION INSERT
GENAL	GENERATE ACTION LIST
GENAP	GENERATE ACTION PAGE
GENAQ	GENERATE ACTION QUERY (SELECT)
GENAR	GENERATE ACTION PRESENT
GENAS	GENERATE ACTION SET

REPORT WRITER Module List

Module Name -----	Purpose -----
GENAT	GENERATE ACTION SIGNAL
GENBEG	GENERATE BEGINNING OF APPLICATION OR REPORT
GENCHG	GENERATE CHANGE DECLARATIONS
GENDB	GENERATE DATA BASE RECORDS AND FILE DECLARATIONS
GENDOA	GENERATE PROCEDURE "DOACT" DO ACTION
GENDS	GENERATE DATA DATA STRUCTURES
GENFP	GENERATE FORM PATH
GENFS	GENERATE FORM DATA STRUCTURES
GENFSD	GENERATE FORM STRUCTURE DATA INITIALIZATION
GENINS	GENERATE INSERT DECLARATIONS
GENMAIN	GENERATE MAIN PROGRAM
GENNDP	GENERATE NODUPLICATE DECLARATIONS
GENPAG	GENERATE NEWPAG PROCEDURE
GETCOL	GET THE COLUMN NAME OF A TABLE.COLUMN OR COLUMN STRING
GETFILE	RETURN A FILE POINTER BASED ON INPUT FROM THE USER
GETFIT	GET SUBTREE THAT FITS ON PAGE
GETLOWLEF	GET LOWER LEFT CHILD NODE
GETLOWRIT	GET LOWER RIGHT CHILD NODE

REPORT WRITER Module List

<u>Module Name</u> -----	<u>Purpose</u> -----
GETPAR	GET PARENT NODE
GETPTH	GET PATH
GETSIZE	GET SUBTREE SIZE
GETTBL	GET A TABLE NAME
GETTOP	GET TOP OF TREE
GETUPLFT	GET UPPER LEFTMOST NODE
GFLDPT	GET FIELD POINTER
GRP/MAIN	GENERATE APPLICATION/REPORT PROGRAM
HASDATA	DETERMINE IF THERE ARE ANY SELECT STATEMENTS
HASITEM	THIS ROUTINE DETERMINES IF THERE IS AN ITEM WITHIN
HASLOWER	HAS A LOWER FORM WHICH READS THE SAME DATA RECORD?
HBALANC	HORIZONTAL TREE BALANCE
HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
INDENT	INDENT A LINE OF GENERATED CODE
INSERT	INSERT PROCEDURE
INSRSV	INSERT RESOLVE
INSWS	INSERT WORKING STORAGE SECTION
ISOPNE	DETERMINE IF THIS FIELD IS OPEN ENDED
MAKACT	MAKE ACTION LIST ELEMENT

REPORT WRITER Module List

Module Name -----	Purpose -----
MAKES	MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE
MAKES/CNUMPIC	C NUMBERS
MAKES/INDENT	INDENT
MAKES/NUMPIC	NUMBER PICTURE CLAUSE
MAKINS	MAKE INSERT
MAKINT	MAKE EXPRESSION INTO AN INTEGER
MAKPS	MAKES THE PRESENTATION SCHEMA RECORD STRUCTURE
MAKQR	MAKE QUALIFIED REFERENCE
MAKSTR	MAKE EXPRESSION INTO A STRING
MAKWH	MAKE WHERE
MAKWHEs	MAKE THE WHERE CLAUSE EXTERNAL SCHEMA VARIABLES
MAKWHEs/COBWHEs	COBOL WHERE ES
MAKWHEs/CWHEs	C WHERE ES
MAKWHEs/NUMPIC	NUMBER PICTURE CLAUSE
MAPDB	MAP DATABASE
MKINC	MAKE INCLUDE (ACTUALLY STRUCTURE TAGS ALA MAKINC)
MKPOS	MAKE POSITION NODE
MLPFRM	MAKE A LIST OF PRESENTED FORMS
MODPAGE	MODIFY PAGES

REPORT WRITER Module List

Module Name -----	Purpose -----
MOVCLD	MOVE CHILDREN
MOVECLD	MOVE CHILD'S POSITION
MYALLOC	MY MALLOC
NDMLGEN	NDML COBOL APPLICATION GENERATOR
NDMLLAB	GENERATE LABELS
NDMLLNK	LINKAGE SECTION
NEXTLEV	ADVANCE POINTERS TO NEXT LEVEL OF SUBTREE
NULBLK	BLANK FILL A STRING
OPNFIL	GENERATE OPEN FILE SECTION
PAGNODE	PAGE NODES
PAGTREE	PAGE TREE
PEMAP	THE PRESENTATION SCHEMA AND THE EXTERNAL SCHEMA AND MAPPING
PRNT	PRINT MODULE NAMES HIERARCHICALLY
PRNTREE	PRINT TREE
PROCEN	PROCEDURE DIVISION GENERATE
PSSTRC/COBSUB	COBOL SUBSTITUTE
PSSTRC/CSUB	C SUBSTITUTE
PSSTRC/INDENT	INDENT
PUTLIN	PRINT LEVEL OF TREE
READDB	READ DATA BASE

REPORT WRITER Module List

Module Name -----	Purpose -----
READTREE	READ DUMPTREE FILE
REPOS	REPOSITION MODULE EXPANSIONS
RSETNDP	RESET NODUPLICATE FIELDS TO VALUE OF NODUP%D
RSETSTAT	RESET STATISTIC
RWEXPD	REPORT WRITER EXPAND ARRAYS
RWOPN	REPORT WRITER OPEN FORMS
RWSP/FIXFRM	FIX UP A FORM
SAVEES	SAVE ES INFORMATION
SELECT	GENERATE SELECT CODE
SELGEN	SELECT GENERATE
SELLEN	COMPUTE LENGTH OF SELECT PS RECORD
SELMAP	MAP SELECTED DATA TO OUTPUT RECORD
SELOPN	SELECT OPEN
SELRSV	SELECT RESOLVE
SELWHR	SELECT WHERE
SELWS	SELECT WORKING STORAGE SECTION
SETNDP	SET NODUPLICATE FIELDS TO BLANK IF THEY ARE DUPLICATED
SORT	SORT MODULE NAMES
SPLICE	SPLICE TREE INTO ANOTHER TREE
SPLITNODE	SPLIT A NODE FOR PAGE BREAKS

REPORT WRITER Module List

Module Name -----	Purpose -----
STATRSV	STATISTIC RESOLVE
STDCODE	STANDARD COBOL CODE
STRIPLEV	DRAW STRIP CHART LEVEL
TRGRSV	TRIGGER RESOLVE
UQFOR	UNIVERSAL QUALIFIER FOR LOOP
UQPTH	UNIVERSAL QUALIFIER PATH
USING	GENERATE USING SECTION
VISITA	VISIT ARRAYS ON THIS FORM
WARNING	ISSUE WARNING MESSAGE
WINRSV	WINDOW RESOLVE
WRTEXP	WRITE EXPRESSION
WRTFRM	WRITE FORM
WRTFRM/DBFCLOS	DEFAULT BUFFER CLOSE
WRTFRM/FORMAT	INSERT FORMAT CODES
WRTFRM/TBFCLOS	TEXT BUFFER CLOSE
WRTFRM/WRTDBF	WRITE DEFAULT BUFFER
WRTFRM/WRTFLD	WRITE FIELD
WRTFRM/WRTTBF	WRITE TEXT BUFFER
WRTFRM/WRTTXT	WRITE TEXT
YYLEX	LEXICAL ANALYZER FOR FLAN
YYPARSE	FLAN PARSER

3.10.3 External Routines List

The following is a list of all routines or functions not documented here that are called by modules that are documented here. The first caller, in alphabetical order, is listed as well. See section 3.10.6 for a list of the modules that call each of these external routines.

REPORT WRITER External Routines List

Module Name -----	First User -----
ABS	RWEXPD
ADDFRM	FRNTND
ATOF	YYLEX
atoi	YYLEX
BLEN	CHKFLD
CALLOC	GRP/MAIN
COPFLD	WINRSV
DELFLD	FLANCI
ERRPRO	CDMESQY
ESCPY	COBCONV
FCLOSE	WRTFRM
FGETS	DRAWLEV
FOPEN	WRTFRM
FPRINTF	PSSTRC/COBSUB
FPUTS	DOINDEX
FREE	CHKFLD
FSEEK	STRIPLEV
FTELL	READTREE
FWRITE	WRTFRM/WRTFLD
GDATA	HRW/MAIN
GETC	READTREE
INITAL	FRNTND
INITFP	FRNTND
INSMAP	PROCGEN
ISALNUM	YYLEX
ISALPHA	YYLEX
ISDIGIT	YYLEX
ISSPACE	YYLEX
MAKFLD	YYPARSE
MALLOC	WINRSV
MAP	PROCGEN
MAX	GETSIZE
MEMCMP	HRW/MAIN
MEMCPY	STRIPLEV
MEMSET	DRAWLEV
MIN	GETSIZE
OISCR	FRNTND
OUTSCR	HRW/MAIN
PMSGLC	GRP/MAIN
PMSGLS	BLDMOD
PRINTF	PRNT

REPORT WRITER External Routines List

Module Name -----	First User -----
PSESMAP	PROCGEN
PTHPTR	UQPTH
PUTATT	HRW/MAIN
PUTC	PUTLIN
PUTCUR	HRW/MAIN
SPRINTF	GETFILE
STRASN	CHKARY
STRCAT	YYPARSE
STRCHR	PUTLIN
STRCMP	RWSP/FIXFRM
STRCPY	GETPTH
STRLEN	READTREE
STRNCMP	SAVEES
STRNCPY	WRTFRM/WRTFLD
STRSPN	GENAS
STRUPC	SORT
SYSMSG	WRTFRM
TERMFP	GRP/MAIN
TOUPPER	YYLEX
TRMNAT	HRW/MAIN
TRMNDML	GRP/MAIN
UNGETC	YYLEX
YYERROR	YYPARSE

3.10.4 Include File List

The following is a list of all include files called in by modules being documented here. Each include file has a unique name regardless of the language being used. The purpose of each include file is listed as well. A more complete description of each include file is given in section 3.10.9. The purpose listed is the one that is in the source code of the include file.

A purpose of "**** PURPOSE NOT FOUND BY STRIPPER ****" indicates that a purpose statement was not written into the include file itself. The most common reason for this is that the include file comes from system libraries that were not developed by the project, such as 'C' libraries that are provided with the 'C' compiler.

See section 3.10.6 for a set of lists which show all the modules which call in each of these include files.

REPORT WRITER Include File List

File Name -----	Purpose -----
CHART	CHART INCLUDE FILE
CTLCHR	CONTROL CHARACTERS
CTYPE	**** PURPOSE NOT FOUND BY STRIPPER ****
ERRPRO	PROCESS ERROR INCLUDE FILE
FFFV2	FORM FILE FORMAT - VERSION 2
FLAN.Y"	**** PURPOSE NOT FOUND BY STRIPPER ****
FPCODE	FORM PROCESSOR RETURN CODES
FPD	FORM PROCESSOR DATA
FPDINI	FPD INITIALIZATION
FPPARM	FORM PROCESSOR PARAMETERS
HRWFRM	HRW FORM DEFINITION
MATH	**** PURPOSE NOT FOUND BY STRIPPER ****
NTM	NTM INTERFACE INCLUDE FILE
RW	REPORT WRITER DEFINITIONS
SRVRET	AS THE RETURN GIVEN A TABLE-FULL ERROR
STDIO	**** PURPOSE NOT FOUND BY STRIPPER ****
STDTP	STANDARD TYPE DEFINITIONS

3.10.5 Where Include File Used List

The following lists each include file from 3.10.4 and all the modules documented in this specification which include them. The purpose of each module is listed as well.

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
CHART		
	ARRANGE	ARRANGE CHART AND ASSIGNS PAGE NUMBERS
	BLDMOD	BUILD MODULE
	BLDNODE	BUILD NODE
	CLOSEGAP	CLOSE GAP IN TREE
	COPYNODE	COPY A NODE IN TREE
	DELNODE	DELETE A SPECIFIED NODE IN TREE
	DOINDEX	DO CHART INDEX
	DRAWLEV	DRAW A LEVEL OF THE CHART
	GETFIT	GET SUBTREE THAT FITS ON PAGE
	GETLOWLEF	GET LOWER LEFT CHILD NODE
	GETLOWRIT	GET LOWER RIGHT CHILD NODE
	GETPAR	GET PARENT NODE
	GETSIZE	GET SUBTREE SIZE
	GETTOP	GET TOP OF TREE
	GETUPLFT	GET UPPER LEFTMOST NODE
	HBALANC	HORIZONTAL TREE BALANCE
	HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
	MODPAGE	MODIFY PAGES
	MOVCLD	MOVE CHILDREN
	MOVECLD	MOVE CHILD'S POSITION
	NEXTLEV	ADVANCE POINTERS TO NEXT LEVEL OF SUBTREE
	PAGNODE	PAGE NODES
	PAGTREE	PAGE TREE
	PRNT	PRINT MODULE NAMES HIERARCHICALLY
	PRNTREE	PRINT TREE
	PUTLIN	PRINT LEVEL OF TREE
	READTREE	READ DUMPTREE FILE
	REPOS	REPOSITION MODULE EXPANSIONS
	SORT	SORT MODULE NAMES
	SPLICE	SPLICE TREE INTO ANOTHER TREE
	SPLITNODE	SPLIT A NODE FOR PAGE BREAKS
	STRIPLEV	DRAW STRIP CHART LEVEL
CTLCHR		
	ASSIGN	ASSIGN FILE SECTION
	CLSFIL	CLOSE FILES

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	DATAGEN	DATA DIVISION GENERATE
	ENDGEN	END GERNERATE
	FD	FD SECTION DECLARATIONS
	FILELNK	FILE LINKAGE SECTION GENERATE
	INDENT	INDENT A LINE OF GENERATED CODE
	INSERT	INSERT PROCEDURE
	INWS	INSERT WORKING STORAGE SECTION
	NDMLGEN	NDML COBOL APPLICATION GENERATOR
	NDMLLAB	GENERATE LABELS
	NDMLLNK	LINKAGE SECTION
	NULBLK	BLANK FILL A STRING
	OPNFIL	GENERATE OPEN FILE SECTION
	PROCGEN	PROCEDURE DIVISION GENERATE
	SAVEES	SAVE ES INFORMATION
	SELECT	GENERATE SELECT CODE
	SELGEN	SELECT GENERATE
	SELLEN	COMPUTE LENGTH OF SELECT PS RECORD
	SELMAP	MAP SELECTED DATA TO OUTPUT RECORD
	SELWS	SELECT WORKING STORAGE SECTION
	STDCODE	STANDARD COBOL CODE
	USING	GENERATE USING SECTION

CTYPE

MAKACT	MAKE ACTION LIST ELEMENT
YYLEX	LEXICAL ANALYZER FOR FLAN
YYPARSE	FLAN PARSER

ERRPRO

CDMESQY	PROGRAM NAME	CDMESQY
---------	--------------	---------

FFFV2

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
--------------------------	-------------------------	----------------------------

	WRTFRM	WRITE FORM
	WRTFRM/DB	DEFAULT BUFFER CLOSE
	WRTFRM/FO	INSERT FORMAT CODES
	WRTFRM/TB	TEXT BUFFER CLOSE
	WRTFRM/WR	WRITE DEFAULT BUFFER
	WRTFRM/WR	WRITE FIELD
	WRTFRM/WR	WRITE TEXT BUFFER
	WRTFRM/WR	WRITE TEXT

FLAN.Y"

	MAKACT	MAKE ACTION LIST ELEMENT
	YLEX	LEXICAL ANALYZER FOR FLAN
	YYPARSE	FLAN PARSER

FPCODE

	ACTRSV	ACTION RESOLVE
	ADDCHK	ADD POSITION TO CHECK LIST
	ASSIGN	ASSIGN FILE SECTION
	CALCSTAT	CALCULATE STATISTIC
	CHKARY	CHECK ARRAY
	CHKFLD	CHECK FIELD
	CHKFRM	CHECK FORM
	CLSFIL	CLOSE FILES
	CSTASH	CHARACTER STASH
	CTLRSV	CONTROL RESOLVE
	DATAGEN	DATA DIVISION GENERATE
	ENDGEN	END GERNERATE
	FD	FD SECTION DECLARATIONS
	FILELNK	FILE LINKAGE SECTION GENERATE
	FLANCI	FLAN CALLABLE INTERFACE
	FLDRSV	FIELD RESOLVE
	FLDTYP	FIELD TYPE
	FNDATT	FIND ATTRIBUTE
	FNDFRM	FIND FORM

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	GETCOL	GET THE COLUMN NAME OF A TABLE.COLUMN OR COLUMN STRING
	GETPTH	GET PATH
	GETTBL	GET A TABLE NAME
	GFLDPT	GET FIELD POINTER
	HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
	INDENT	INDENT A LINE OF GENERATED CODE
	INSERT	INSERT PROCEDURE
	INSRSV	INSERT RESOLVE
	INWS	INSERT WORKING STORAGE SECTION
	MAKINS	MAKE INSERT
	MAKINT	MAKE EXPRESSION INTO AN INTEGER
	MAKPS	MAKES THE PRESENTATION SCHEMA RECORD STRUCTURE
	MAKSTR	MAKE EXPRESSION INTO A STRING
	MAKWH	MAKE WHERE
	MAKWHE	MAKE THE WHERE CLAUSE EXTERNAL SCHEMA VARIABLES
	MAKWHE/C	COBOL WHERE ES
	MAKWHE/C	C WHERE ES
	MAKWHE/N	NUMBER PICTURE CLAUSE
	MKPOS	MAKE POSITION NODE
	MLPFRM	MAKE A LIST OF PRESENTED FORMS
	MYALLO	MY MALLOC
	NDMLGEN	NDML COBOL APPLICATION GENERATOR
	NDMLLAB	GENERATE LABELS
	NDMLLNK	LINKAGE SECTION
	NULBLK	BLANK FILL A STRING
	OPNFIL	GENERATE OPEN FILE SECTION
	PROCGEN	PROCEDURE DIVISION GENERATE
	PSSTRC/CO	COBOL SUBSTITUTE
	PSSTRC/CS	C SUBSTITUTE
	PSSTRC/IN	INDENT
	RSETSTAT	RESET STATISTIC
	RWEXPD	REPORT WRITER EXPAND ARRAYS
	RWOPN	REPORT WRITER OPEN FORMS
	RWSP/FIXF	FIX UP A FORM
	SAVEES	SAVE ES INFORMATION
	SELECT	GENERATE SELECT CODE
	SELGEN	SELECT GENERATE

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	SELLEN	COMPUTE LENGTH OF SELECT PS RECORD
	SELMAP	MAP SELECTED DATA TO OUTPUT RECORD
	SELRSV	SELECT RESOLVE
	SELWS	SELECT WORKING STORAGE SECTION
	STATRSV	STATISTIC RESOLVE
	STDCODE	STANDARD COBOL CODE
	TRGRSV	TRIGGER RESOLVE
	UQPTH	UNIVERSAL QUALIFIER PATH
	USING	GENERATE USING SECTION
	WINRSV	WINDOW RESOLVE
	WRTEXP	WRITE EXPRESSION
	WRTFRM	WRITE FORM
	WRTFRM/DB	DEFAULT BUFFER CLOSE
	WRTFRM/FO	INSERT FORMAT CODES
	WRTFRM/TB	TEXT BUFFER CLOSE
	WRTFRM/WR	WRITE DEFAULT BUFFER
	WRTFRM/WR	WRITE FIELD
	WRTFRM/WR	WRITE TEXT BUFFER
	WRTFRM/WR	WRITE TEXT

FPD

ACTRSV	ACTION RESOLVE
ADDCHK	ADD POSITION TO CHECK LIST
ASSIGN	ASSIGN FILE SECTION
BLDSUB	BUILD SUBROUTINES
BSCODE	BUILD SUBROUTINE CODE
CALCSTAT	CALCULATE STATISTIC
CCONV	C CONVERSIONS
CES	C ES
CESPS	C ES TO PS
CHKARY	CHECK ARRAY
CHKFLD	CHECK FIELD
CHKFRM	CHECK FORM
CHKGRP	CHECK FOR GROUP SEPERATORS OR END OF FILE
CHKSIZE	CHECK SIZE OF ITEMS DOING CONVERSIONS ON
CLRNDP	CLEAR NODUPLICATE FIELDS
CLSFIL	CLOSE FILES

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	COBCONV	COBOL CONVERSIONS
	COBES	COBOL ES RECORD
	COBESPS	COBOL ES TO PS
	COBPE	COBOL PE
	CPE	C PE
	CSTASH	CHARACTER STASH
	CTLRSV	CONTROL RESOLVE
	DASH	WRITE DASH '-'
	DATAGEN	DATA DIVISION GENERATE
	DBFREAD	GENERATE DATA BASE FREAD
	DCLINDX	DECLARE INDEX VARIABLES
	ENDGEN	END GERNERATE
	ESPSMAP	THE EXTERNAL SCHEMA AND PRESENTATION SCHEMA MAPPING
	ESPSMAP/I	INDENT
	FD	FD SECTION DECLARATIONS
	FILELNK	FILE LINKAGE SECTION GENERATE
	FLANCI	FLAN CALLABLE INTERFACE
	FLDRSV	FIELD RESOLVE
	FLDTYP	FIELD TYPE
	FNDATT	FIND ATTRIBUTE
	FNDFRM	FIND FORM
	FRMPDAT	FORM PDATA
	GEN	GENERATE A LINE OF CODE
	GENAA	GENERATE PROCEDURE "ADDACT" ADD AN ACTION
	GENAAL	GENERATE PROCEDURE "ADDAL" ADD ACTION LIST
	GENACT	GENERATE ACTIONS
	GENAE	GENERATE ACTION EXIT
	GENAH	GENERATE ACTION HELP
	GENAI	GENERATE ACTION INSERT
	GENAL	GENERATE ACTION LIST
	GENAP	GENERATE ACTION PAGE
	GENAQ	GENERATE ACTION QUERY (SELECT)
	GENAR	GENERATE ACTION PRESENT
	GENAS	GENERATE ACTION SET
	GENAT	GENERATE ACTION SIGNAL
	GENBEG	GENERATE BEGINNING OF APPLICATION OR REPORT
	GENCHG	GENERATE CHANGE DECLARATIONS

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	GENDB	GENERATE DATA BASE RECORDS AND FILE DECLARATIONS
	GENDOA	GENERATE PROCEDURE "DOACT" DO ACTION
	GENDS	GENERATE DATA DATA STRUCTURES
	GENFP	GENERATE FORM PATH
	GENFS	GENERATE FORM DATA STRUCTURES
	GENFSD	GENERATE FORM STRUCTURE DATA INITIALIZATION
	GENINS	GENERATE INSERT DECLARATIONS
	GENMAIN	GENERATE MAIN PROGRAM
	GENNDP	GENERATE NODUPLICATE DECLARATIONS
	GENPAG	GENERATE NEWPAG PROCEDURE
	GETCOL	GET THE COLUMN NAME OF A TABLE.COLUMN OR COLUMN STRING
	GETFILE	RETURN A FILE POINTER BASED ON INPUT FROM THE USER
	GETPTH	GET PATH
	GETTBL	GET A TABLE NAME
	GFLDPT	GET FIELD POINTER
	GRP/MAIN	GENERATE APPLICATION/REPORT PROGRAM
	HASDATA	DETERMINE IF THERE ARE ANY SELECT STATEMENTS
	HASITEM	THIS ROUTINE DETERMINES IF THERE IS AN ITEM WITHIN
	HASLOWER	HAS A LOWER FORM WHICH READS THE SAME DATA RECORD?
	INDENT	INDENT A LINE OF GENERATED CODE
	INSERT	INSERT PROCEDURE
	INSRSV	INSERT RESOLVE
	INSWS	INSERT WORKING STORAGE SECTION
	ISOPNE	DETERMINE IF THIS FIELD IS OPEN ENDED
	MAKACT	MAKE ACTION LIST ELEMENT
	MAKES	MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE
	MAKES/CNU	C NUMBERS
	MAKES/IND	INDENT
	MAKES/NUM	NUMBER PICTURE CLAUSE
	MAKINS	MAKE INSERT
	MAKINT	MAKE EXPRESSION INTO AN INTEGER
	MAKPS	MAKES THE PRESENTATION SCHEMA RECORD STRUCTURE

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	MAKQR	MAKE QUALIFIED REFERENCE
	MAKSTR	MAKE EXPRESSION INTO A STRING
	MAKWH	MAKE WHERE
	MAKWHE	MAKE THE WHERE CLAUSE EXTERNAL SCHEMA VARIABLES
	MAKWHE/C	COBOL WHERE ES
	MAKWHE/C	C WHERE ES
	MAKWHE/N	NUMBER PICTURE CLAUSE
	MAPDB	MAP DATABASE
	MKINC	MAKE INCLUDE (ACTUALLY STRUCTURE TAGS ALA MAKINC)
	MKPOS	MAKE POSITION NODE
	MLPFRM	MAKE A LIST OF PRESENTED FORMS
	MYALLO	MY MALLOC
	NDMLGEN	NDML COBOL APPLICATION GENERATOR
	NDMLLAB	GENERATE LABELS
	NDMLLNK	LINKAGE SECTION
	NULBLK	BLANK FILL A STRING
	OPNFIL	GENERATE OPEN FILE SECTION
	PEMAP	THE PRESENTATION SCHEMA AND THE EXTERNAL SCHEMA AND MAPPING
	PROCGEN	PROCEDURE DIVISION GENERATE
	PSSTRC/CO	COBOL SUBSTITUTE
	PSSTRC/CS	C SUBSTITUTE
	PSSTRC/IN	INDENT
	READDB	READ DATA BASE
	RSETNDP	RESET NODUPLICATE FIELDS TO VALUE OF NODUP%D
	RSETSTAT	RESET STATISTIC
	RWEXPD	REPORT WRITER EXPAND ARRAYS
	RWOPN	REPORT WRITER OPEN FORMS
	RWSP/FIXF	FIX UP A FORM
	SAVEES	SAVE ES INFORMATION
	SELECT	GENERATE SELECT CODE
	SELGEN	SELECT GENERATE
	SELLEN	COMPUTE LENGTH OF SELECT PS RECORD
	SELMAP	MAP SELECTED DATA TO OUTPUT RECORD
	SELOPN	SELECT OPEN
	SELRSV	SELECT RESOLVE
	SELWHR	SELECT WHERE

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	SELWS	SELECT WORKING STORAGE SECTION
	SETNDP	SET NODUPLICATE FIELDS TO BLANK IF THEY ARE DUPLICATED
	STATRSV	STATISTIC RESOLVE
	STDCODE	STANDARD COBOL CODE
	TRGRSV	TRIGGER RESOLVE
	UQFOR	UNIVERSAL QUALIFIER FOR LOOP
	UQPTH	UNIVERSAL QUALIFIER PATH
	USING	GENERATE USING SECTION
	VISITA	VISIT ARRAYS ON THIS FORM
	WINRSV	WINDOW RESOLVE
	WRTEXP	WRITE EXPRESSION
	WRTFRM	WRITE FORM
	WRTFRM/DB	DEFAULT BUFFER CLOSE
	WRTFRM/FO	INSERT FORMAT CODES
	WRTFRM/TB	TEXT BUFFER CLOSE
	WRTFRM/WR	WRITE DEFAULT BUFFER
	WRTFRM/WR	WRITE FIELD
	WRTFRM/WR	WRITE TEXT BUFFER
	WRTFRM/WR	WRITE TEXT
	YYLEX	LEXICAL ANALYZER FOR FLAN
	YYPARSE	FLAN PARSER
FPDINI	BLDSUB	BUILD SUBROUTINES
	BSCODE	BUILD SUBROUTINE CODE
	CHKGRP	CHECK FOR GROUP SEPERATORS OR END OF FILE
	CLRNDP	CLEAR NODUPLICATE FIELDS
	DBFREAD	GENERATE DATA BASE FREAD
	GEN	GENERATE A LINE OF CODE
	GENPAG	GENERATE NEWPAG PROCEDURE
	GETFILE	RETURN A FILE POINTER BASED ON INPUT FROM THE USER
	GRP/MAIN	GENERATE APPLICATION/REPORT PROGRAM
	HASDATA	DETERMINE IF THERE ARE ANY SELECT STATEMENTS

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	HASITEM	THIS ROUTINE DETERMINES IF THERE IS AN ITEM WITHIN
	HASLOWER	HAS A LOWER FORM WHICH READS THE SAME DATA RECORD?
	ISOPNE	DETERMINE IF THIS FIELD IS OPEN ENDED
	MAKQR	MAKE QUALIFIED REFERENCE
	MAPDB	MAP DATABASE
	READDB	READ DATA BASE
	RSETNDP	RESET NODUPLICATE FIELDS TO VALUE OF NODUP%D
	SETNDP	SET NODUPLICATE FIELDS TO BLANK IF THEY ARE DUPLICATED
	VISITA	VISIT ARRAYS ON THIS FORM

FPPARM

ASSIGN	ASSIGN FILE SECTION
BLDSUB	BUILD SUBROUTINES
BSCODE	BUILD SUBROUTINE CODE
CHKGRP	CHECK FOR GROUP SEPERATORS OR END OF FILE
CLRNDP	CLEAR NODUPLICATE FIELDS
CLSFIL	CLOSE FILES
DATAGEN	DATA DIVISION GENERATE
DBFREAD	GENERATE DATA BASE FREAD
ENDGEN	END GERNERATE
FD	FD SECTION DECLARATIONS
FILELNK	FILE LINKAGE SECTION GENERATE
FRNTND	FRONT END FOR FORMS
GEN	GENERATE A LINE OF CODE
GENPAG	GENERATE NEWPAG PROCEDURE
GETFILE	RETURN A FILE POINTER BASED ON INPUT FROM THE USER
GRP/MAIN	GENERATE APPLICATION/REPORT PROGRAM
HASDATA	DETERMINE IF THERE ARE ANY SELECT STATEMENTS
HASITEM	THIS ROUTINE DETERMINES IF THERE IS AN ITEM WITHIN

REPORT WRITER Where-include-file-used List

Inclu- File -----	Module Name -----	Module Purpose -----
	HASLOWER	HAS A LOWER FORM WHICH READS THE SAME DATA RECORD?
	HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
	INDENT	INDENT A LINE OF GENERATED CODE
	INSERT	INSERT PROCEDURE
	INSWS	INSERT WORKING STORAGE SECTION
	ISOPNE	DETERMINE IF THIS FIELD IS OPEN ENDED
	MAKACT	MAKE ACTION LIST ELEMENT
	MAKINS	MAKE INSERT
	MAKPS	MAKES THE PRESENTATION SCHEMA RECORD STRUCTURE
	MAKQR	MAKE QUALIFIED REFERENCE
	MAKWH	MAKE WHERE
	MAKWHE	MAKE THE WHERE CLAUSE EXTERNAL SCHEMA VARIABLES
	MAKWHE/C	COBOL WHERE ES
	MAKWHE/C	C WHERE ES
	MAKWHE/N	NUMBER PICTURE CLAUSE
	MAPDB	MAP DATABASE
	NDMLGEN	NDML COBOL APPLICATION GENERATOR
	NDMLLAB	GENERATE LABELS
	NDMLLNK	LINKAGE SECTION
	NULBLK	BLANK FILL A STRING
	OPNFIL	GENERATE OPEN FILE SECTION
	PROCGEN	PROCEDURE DIVISION GENERATE
	PSSTRC/CO	COBOL SUBSTITUTE
	PSSTRC/CS	C SUBSTITUTE
	PSSTRC/IN	INDENT
	READDB	READ DATA BASE
	RSETNDP	RESET NODUPLICATE FIELDS TO VALUE OF NODUP%D
	SAVEES	SAVE ES INFORMATION
	SELECT	GENERATE SELECT CODE
	SELGEN	SELECT GENERATE
	SELLEN	COMPUTE LENGTH OF SELECT PS RECORD
	SELMAP	MAP SELECTED DATA TO OUTPUT RECORD
	SELWS	SELECT WORKING STORAGE SECTION
	SETNDP	SET NODUPLICATE FIELDS TO BLANK IF THEY ARE DUPLICATED
	STDCODE	STANDARD COBOL CODE

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	USING VISITA YYLEX YYPARSE	GENERATE USING SECTION VISIT ARRAYS ON THIS FORM LEXICAL ANALYZER FOR FLAN FLAN PARSER
HRWFRM	HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
MATH	MAKACT YYLEX YYPARSE	MAKE ACTION LIST ELEMENT LEXICAL ANALYZER FOR FLAN FLAN PARSER
NTM	ASSIGN BLDSUB BSCODE CHKGRP CLRNDP CLSFIL DATAGEN DBFREAD ENDGEN FD FILELNK FRNTND GEN GENPAG GETFILE GRP/MAIN	ASSIGN FILE SECTION BUILD SUBROUTINES BUILD SUBROUTINE CODE CHECK FOR GROUP SEPERATORS OR END OF FILE CLEAR NODUPLICATE FIELDS CLOSE FILES DATA DIVISION GENERATE GENERATE DATA BASE FREAD END GERNERATE FD SECTION DECLARATIONS FILE LINKAGE SECTION GENERATE FRONT END FOR FORMS GENERATE A LINE OF CODE GENERATE NEWPAG PROCEDURE RETURN A FILE POINTER BASED ON INPUT FROM THE USER GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	HASDATA	DETERMINE IF THERE ARE ANY SELECT STATEMENTS
	HASITEM	THIS ROUTINE DETERMINES IF THERE IS AN ITEM WITHIN
	HASLOWER	HAS A LOWER FORM WHICH READS THE SAME DATA RECORD?
	INDENT	INDENT A LINE OF GENERATED CODE
	INSERT	INSERT PROCEDURE
	INSWS	INSERT WORKING STORAGE SECTION
	ISOPNE	DETERMINE IF THIS FIELD IS OPEN ENDED
	MAKQR	MAKE QUALIFIED REFERENCE
	MAPDB	MAP DATABASE
	NDMLGEN	NDML COBOL APPLICATION GENERATOR
	NDMLLAB	GENERATE LABELS
	NDMLLNK	LINKAGE SECTION
	NULBLK	BLANK FILL A STRING
	OPNFIL	GENERATE OPEN FILE SECTION
	PROCEN	PROCEDURE DIVISION GENERATE
	READDB	READ DATA BASE
	RSETNDP	RESET NODUPLICATE FIELDS TO VALUE OF NODUP%D
	SAVEES	SAVE ES INFORMATION
	SELECT	GENERATE SELECT CODE
	SELGEN	SELECT GENERATE
	SELLEN	COMPUTE LENGTH OF SELECT PS RECORD
	SELMAP	MAP SELECTED DATA TO OUTPUT RECORD
	SELWS	SELECT WORKING STORAGE SECTION
	SETNDP	SET NODUPLICATE FIELDS TO BLANK IF THEY ARE DUPLICATED
	STDCODE	STANDARD COBOL CODE
	USING	GENERATE USING SECTION
	VISITA	VISIT ARRAYS ON THIS FORM
RW	ACTRSV	ACTION RESOLVE
	ADDCHK	ADD POSITION TO CHECK LIST
	ASSIGN	ASSIGN FILE SECTION

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	BLDSUB	BUILD SUBROUTINES
	BSCODE	BUILD SUBROUTINE CODE
	CALCSTAT	CALCULATE STATISTIC
	CCONV	C CONVERSIONS
	CES	C ES
	CESPS	C ES TO PS
	CHKARY	CHECK ARRAY
	CHKFLD	CHECK FIELD
	CHKFRM	CHECK FORM
	CHKGRP	CHECK FOR GROUP SEPERATORS OR END OF FILE
	CHKSIZE	CHECK SIZE OF ITEMS DOING CONVERSIONS ON
	CLRNDP	CLEAR NODUPLICATE FIELDS
	CLSFIL	CLOSE FILES
	COBCONV	COBOL CONVERSIONS
	COBES	COBOL ES RECORD
	COBESPS	COBOL ES TO PS
	COBPE	COBOL PE
	CPE	C PE
	CSTASH	CHARACTER STASH
	CTLRSV	CONTROL RESOLVE
	DASH	WRITE DASH '-'
	DATAGEN	DATA DIVISION GENERATE
	DBFREAD	GENERATE DATA BASE FREAD
	DCLINDX	DECLARE INDEX VARIABLES
	ENDGEN	END GERNERATE
	ESPSMAP	THE EXTERNAL SCHEMA AND PRESENTATION SCHEMA MAPPING
	ESPSMAP/I	INDENT
	FD	FD SECTION DECLARATIONS
	FILELNK	FILE LINKAGE SECTION GENERATE
	FLANCI	FLAN CALLABLE INTERFACE
	FLDRSV	FIELD RESOLVE
	FLDTYP	FIELD TYPE
	FNDATT	FIND ATTRIBUTE
	FNDFRM	FIND FORM
	FRMPDAT	FORM PDATA
	GEN	GENERATE A LINE OF CODE
	GENAA	GENERATE PROCEDURE "ADDACT" ADD AN ACTION
	GENAAL	GENERATE PROCEDURE "ADDAL" ADD ACTION LIST
	GENACT	GENERATE ACTIONS

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	GENAE	GENERATE ACTION EXIT
	GENAH	GENERATE ACTION HELP
	GENAI	GENERATE ACTION INSERT
	GENAL	GENERATE ACTION LIST
	GENAP	GENERATE ACTION PAGE
	GENAQ	GENERATE ACTION QUERY (SELECT)
	GENAR	GENERATE ACTION PRESENT
	GENAS	GENERATE ACTION SET
	GENAT	GENERATE ACTION SIGNAL
	GENBEG	GENERATE BEGINNING OF APPLICATION OR REPORT
	GENCHG	GENERATE CHANGE DECLARATIONS
	GENDB	GENERATE DATA BASE RECORDS AND FILE DECLARATIONS
	GENDOA	GENERATE PROCEDURE "DOACT" DO ACTION
	GENDS	GENERATE DATA DATA STRUCTURES
	GENFP	GENERATE FORM PATH
	GENFS	GENERATE FORM DATA STRUCTURES
	GENFSD	GENERATE FORM STRUCTURE DATA INITIALIZATION
	GENINS	GENERATE INSERT DECLARATIONS
	GENMAIN	GENERATE MAIN PROGRAM
	GENNDP	GENERATE NODUPLICATE DECLARATIONS
	GENPAG	GENERATE NEWPAG PROCEDURE
	GETCOL	GET THE COLUMN NAME OF A TABLE.COLUMN OR COLUMN STRING
	GETFILE	RETURN A FILE POINTER BASED ON INPUT FROM THE USER
	GETPTH	GET PATH
	GETTBL	GET A TABLE NAME
	GFLDPT	GET FIELD POINTER
	GRP/MAIN	GENERATE APPLICATION/REPORT PROGRAM
	HASDATA	DETERMINE IF THERE ARE ANY SELECT STATEMENTS
	HASITEM	THIS ROUTINE DETERMINES IF THERE IS AN ITEM WITHIN
	HASLOWER	HAS A LOWER FORM WHICH READS THE SAME DATA RECORD?
	INDENT	INDENT A LINE OF GENERATED CODE
	INSERT	INSERT PROCEDURE

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	INSRSV	INSERT RESOLVE
	INSWS	INSERT WORKING STORAGE SECTION
	ISOPNE	DETERMINE IF THIS FIELD IS OPEN ENDED
	MAKACT	MAKE ACTION LIST ELEMENT
	MAKES	MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE
	MAKES/CNU	C NUMBERS
	MAKES/IND	INDENT
	MAKES/NUM	NUMBER PICTURE CLAUSE
	MAKINS	MAKE INSERT
	MAKINT	MAKE EXPRESSION INTO AN INTEGER
	MAKPS	MAKES THE PRESENTATION SCHEMA RECORD STRUCTURE
	MAKQR	MAKE QUALIFIED REFERENCE
	MAKSTR	MAKE EXPRESSION INTO A STRING
	MAKWH	MAKE WHERE
	MAKWHEs	MAKE THE WHERE CLAUSE EXTERNAL SCHEMA VARIABLES
	MAKWHEs/C	COBOL WHERE ES
	MAKWHEs/C	C WHERE ES
	MAKWHEs/N	NUMBER PICTURE CLAUSE
	MAPDB	MAP DATABASE
	MKINC	MAKE INCLUDE (ACTUALLY STRUCTURE TAGS ALA MAKINC)
	MKPOS	MAKE POSITION NODE
	MLPFRM	MAKE A LIST OF PRESENTED FORMS
	MYALLOC	MY MALLOC
	NDMLGEN	NDML COBOL APPLICATION GENERATOR
	NDMLLAB	GENERATE LABELS
	NDMLLNK	LINKAGE SECTION
	NULBLK	BLANK FILL A STRING
	OPNFIL	GENERATE OPEN FILE SECTION
	PEMAP	THE PRESENTATION SCHEMA AND THE EXTERNAL SCHEMA AND MAPPING
	PROCGEN	PROCEDURE DIVISION GENERATE
	PSSTRC/CO	COBOL SUBSTITUTE
	PSSTRC/CS	C SUBSTITUTE
	PSSTRC/IN	INDENT
	READDB	READ DATA BASE
	RSETNDP	RESET NODUPLICATE FIELDS TO VALUE OF NODUP%D

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	RSETSTAT	RESET STATISTIC
	RWEXPD	REPORT WRITER EXPAND ARRAYS
	RWOPN	REPORT WRITER OPEN FORMS
	RWSP/FIXF	FIX UP A FORM
	SAVEES	SAVE ES INFORMATION
	SELECT	GENERATE SELECT CODE
	SELGEN	SELECT GENERATE
	SELLEN	COMPUTE LENGTH OF SELECT PS RECORD
	SELMAP	MAP SELECTED DATA TO OUTPUT RECORD
	SELOPN	SELECT OPEN
	SELRSV	SELECT RESOLVE
	SELWHR	SELECT WHERE
	SELWS	SELECT WORKING STORAGE SECTION
	SETNDP	SET NODUPLICATE FIELDS TO BLANK IF THEY ARE DUPLICATED
	STATRSV	STATISTIC RESOLVE
	STDCODE	STANDARD COBOL CODE
	TRGRSV	TRIGGER RESOLVE
	UQFOR	UNIVERSAL QUALIFIER FOR LOOP
	UQPTH	UNIVERSAL QUALIFIER PATH
	USING	GENERATE USING SECTION
	VISITA	VISIT ARRAYS ON THIS FORM
	WINRSV	WINDOW RESOLVE
	WRTEXP	WRITE EXPRESSION
	YLEX	LEXICAL ANALYZER FOR FLAN
	YYPARSE	FLAN PARSER
SRVRET	CDMESQY	PROGRAM NAME CDMESQY
STDIO	ADDCHK	ADD POSITION TO CHECK LIST
	ARRANGE	ARRANGE CHART AND ASSIGNS PAGE NUMBERS
	ASSIGN	ASSIGN FILE SECTION

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	BLDSUB	BUILD SUBROUTINES
	BSCODE	BUILD SUBROUTINE CODE
	CCONV	C CONVERSIONS
	CES	C ES
	CESPS	C ES TO PS
	CHKARY	CHECK ARRAY
	CHKFLD	CHECK FIELD
	CHKFRM	CHECK FORM
	CHKGRP	CHECK FOR GROUP SEPERATORS OR END OF FILE
	CHKSIZE	CHECK SIZE OF ITEMS DOING CONVERSIONS ON
	CLRNDP	CLEAR NODUPLICATE FIELDS
	CLSFIL	CLOSE FILES
	COBCONV	COBOL CONVERSIONS
	COBES	COBOL ES RECORD
	COBESPS	COBOL ES TO PS
	COBPE	COBOL PE
	CPE	C PE
	CSTASH	CHARACTER STASH
	DASH	WRITE DASH '-'
	DATAGEN	DATA DIVISION GENERATE
	DBFREAD	GENERATE DATA BASE FREAD
	DOINDEX	DO CHART INDEX
	DRAWLEV	DRAW A LEVEL OF THE CHART
	ENDGEN	END GERNERATE
	ESPSMAP	THE EXTERNAL SCHEMA AND PRESENTATION SCHEMA MAPPING
	ESPSMAP/I	INDENT
	FD	FD SECTION DECLARATIONS
	FILELNK	FILE LINKAGE SECTION GENERATE
	FLANCI	FLAN CALLABLE INTERFACE
	FLDTYP	FIELD TYPE
	FNDATT	FIND ATTRIBUTE
	GEN	GENERATE A LINE OF CODE
	GENBEG	GENERATE BEGINNING OF APPLICATION OR REPORT
	GENCHG	GENERATE CHANGE DECLARATIONS
	GENDB	GENERATE DATA BASE RECORDS AND FILE DECLARATIONS
	GENDS	GENERATE DATA DATA STRUCTURES
	GENFP	GENERATE FORM PATH

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	GENFS	GENERATE FORM DATA STRUCTURES
	GENFSD	GENERATE FORM STRUCTURE DATA INITIALIZATION
	GENINS	GENERATE INSERT DECLARATIONS
	GENMAIN	GENERATE MAIN PROGRAM
	GENNDP	GENERATE NODUPLICATE DECLARATIONS
	GENPAG	GENERATE NEWPAG PROCEDURE
	GETFILE	RETURN A FILE POINTER BASED ON INPUT FROM THE USER
	GFLDPT	GET FIELD POINTER
	GRP/MAIN	GENERATE APPLICATION/REPORT PROGRAM
	HASDATA	DETERMINE IF THERE ARE ANY SELECT STATEMENTS
	HASITEM	THIS ROUTINE DETERMINES IF THERE IS AN ITEM WITHIN
	HASLOWER	HAS A LOWER FORM WHICH READS THE SAME DATA RECORD?
	HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
	INDENT	INDENT A LINE OF GENERATED CODE
	INSERT	INSERT PROCEDURE
	INSWS	INSERT WORKING STORAGE SECTION
	ISOPNE	DETERMINE IF THIS FIELD IS OPEN ENDED
	MAKACT	MAKE ACTION LIST ELEMENT
	MAKES	MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE
	MAKES/CNU	C NUMBERS
	MAKES/IND	INDENT
	MAKES/NUM	NUMBER PICTURE CLAUSE
	MAKINS	MAKE INSERT
	MAKINT	MAKE EXPRESSION INTO AN INTEGER
	MAKPS	MAKES THE PRESENTATION SCHEMA RECORD STRUCTURE
	MAKQR	MAKE QUALIFIED REFERENCE
	MAKSTR	MAKE EXPRESSION INTO A STRING
	MAKWH	MAKE WHERE
	MAKWHE	MAKE THE WHERE CLAUSE EXTERNAL SCHEMA VARIABLES
	MAKWHE/C	COBOL WHERE ES
	MAKWHE/C	C WHERE ES
	MAKWHE/N	NUMBER PICTURE CLAUSE
	MAPDB	MAP DATABASE

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	MKINC	MAKE INCLUDE (ACTUALLY STRUCTURE TAGS ALA MAKINC)
	MKPOS	MAKE POSITION NODE
	MYALLOC	MY MALLOC
	NDMLGEN	NDML COBOL APPLICATION GENERATOR
	NDMLLAB	GENERATE LABELS
	NDMLLNK	LINKAGE SECTION
	NULBLK	BLANK FILL A STRING
	OPNFIL	GENERATE OPEN FILE SECTION
	PEMAP	THE PRESENTATION SCHEMA AND THE EXTERNAL SCHEMA AND MAPPING
	PRNTREE	PRINT TREE
	PROCGEN	PROCEDURE DIVISION GENERATE
	PSSTRC/CO	COBOL SUBSTITUTE
	PSSTRC/CS	C SUBSTITUTE
	PSSTRC/IN	INDENT
	PUTLIN	PRINT LEVEL OF TREE
	READDB	READ DATA BASE
	READTREE	READ DUMPTREE FILE
	RSETNDP	RESET NODUPLICATE FIELDS TO VALUE OF NODUP%D
	SAVEES	SAVE ES INFORMATION
	SELECT	GENERATE SELECT CODE
	SELGEN	SELECT GENERATE
	SELLEN	COMPUTE LENGTH OF SELECT PS RECORD
	SELMAP	MAP SELECTED DATA TO OUTPUT RECORD
	SELWS	SELECT WORKING STORAGE SECTION
	SETNDP	SET NODUPLICATE FIELDS TO BLANK IF THEY ARE DUPLICATED
	STDCODE	STANDARD COBOL CODE
	STRIPLEV	DRAW STRIP CHART LEVEL
	USING	GENERATE USING SECTION
	VISITA	VISIT ARRAYS ON THIS FORM
	WRTEXP	WRITE EXPRESSION
	WRTFRM	WRITE FORM
	WRTFRM/DB	DEFAULT BUFFER CLOSE
	WRTFRM/FO	INSERT FORMAT CODES
	WRTFRM/TB	TEXT BUFFER CLOSE
	WRTFRM/WR	WRITE DEFAULT BUFFER
	WRTFRM/WR	WRITE FIELD

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	WRTFRM/WR	WRITE TEXT BUFFER
	WRTFRM/WR	WRITE TEXT
	YYLEX	LEXICAL ANALYZER FOR FLAN
	YYPARSE	FLAN PARSER

STDTPP

ACTRSV	ACTION RESOLVE
ADDCHK	ADD POSITION TO CHECK LIST
ARRANGE	ARRANGE CHART AND ASSIGNS PAGE NUMBERS
ASSIGN	ASSIGN FILE SECTION
BLDMOD	BUILD MODULE
BLDNODE	BUILD NODE
BLDSUB	BUILD SUBROUTINES
BSCODE	BUILD SUBROUTINE CODE
CALCSTAT	CALCULATE STATISTIC
CCONV	C CONVERSIONS
CES	C ES
CESPS	C ES TO PS
CHKARY	CHECK ARRAY
CHKFLD	CHECK FIELD
CHKFRM	CHECK FORM
CHKGRP	CHECK FOR GROUP SEPERATORS OR END OF FILE
CHKSIZE	CHECK SIZE OF ITEMS DOING CONVERSIONS ON
CLOSEGAP	CLOSE GAP IN TREE
CLRNDP	CLEAR NODUPLICATE FIELDS
CLSFIL	CLOSE FILES
COBCONV	COBOL CONVERSIONS
COBES	COBOL ES RECORD
COBESPS	COBOL ES TO PS
COBPE	COBOL PE
COPYNODE	COPY A NODE IN TREE
CPE	C PE
CSTASH	CHARACTER STASH
CTLRSV	CONTROL RESOLVE
DASH	WRITE DASH '-'
DATAGEN	DATA DIVISION GENERATE
DBFREAD	GENERATE DATA BASE FREAD

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	DCLINDX	DECLARE INDEX VARIABLES
	DELNODE	DELETE A SPECIFIED NODE IN TREE
	DOINDEX	DO CHART INDEX
	DRAWLEV	DRAW A LEVEL OF THE CHART
	ENDGEN	END GERNERATE
	ERROR	ISSUE ERROR MESSAGE
	ESPSMAP	THE EXTERNAL SCHEMA AND PRESENTATION SCHEMA MAPPING
	ESPSMAP/I	INDENT
	FATAL	ISSUE FATAL ERROR MESSAGE
	FD	FD SECTION DECLARATIONS
	FILELNK	FILE LINKAGE SECTION GENERATE
	FLANCI	FLAN CALLABLE INTERFACE
	FLDRSV	FIELD RESOLVE
	FLDTYP	FIELD TYPE
	FNDATT	FIND ATTRIBUTE
	FNDFRM	FIND FORM
	FRMPDAT	FORM PDATA
	FRNTND	FRONT END FOR FORMS
	GEN	GENERATE A LINE OF CODE
	GENAA	GENERATE PROCEDURE "ADDACT" ADD AN ACTION
	GENAAL	GENERATE PROCEDURE "ADDAL" ADD ACTION LIST
	GENACT	GENERATE ACTIONS
	GENAE	GENERATE ACTION EXIT
	GENAH	GENERATE ACTION HELP
	GENAI	GENERATE ACTION INSERT
	GENAL	GENERATE ACTION LIST
	GENAP	GENERATE ACTION PAGE
	GENAQ	GENERATE ACTION QUERY (SELECT)
	GENAR	GENERATE ACTION PRESENT
	GENAS	GENERATE ACTION SET
	GENAT	GENERATE ACTION SIGNAL
	GENBEG	GENERATE BEGINNING OF APPLICATION OR REPORT
	GENCHG	GENERATE CHANGE DECLARATIONS
	GENDB	GENERATE DATA BASE RECORDS AND FILE DECLARATIONS
	GENDOA	GENERATE PROCEDURE "DOACT" DO ACTION
	GENDS	GENERATE DATA DATA STRUCTURES
	GENFP	GENERATE FORM PATH

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	GENFS	GENERATE FORM DATA STRUCTURES
	GENFSD	GENERATE FORM STRUCTURE DATA INITIALIZATION
	GENINS	GENERATE INSERT DECLARATIONS
	GENMAIN	GENERATE MAIN PROGRAM
	GENNDP	GENERATE NODUPLICATE DECLARATIONS
	GENPAG	GENERATE NEWPAG PROCEDURE
	GETCOL	GET THE COLUMN NAME OF A TABLE.COLUMN OR COLUMN STRING
	GETFILE	RETURN A FILE POINTER BASED ON INPUT FROM THE USER
	GETFIT	GET SUBTREE THAT FITS ON PAGE
	GETLOWLEF	GET LOWER LEFT CHILD NODE
	GETLOWRIT	GET LOWER RIGHT CHILD NODE
	GETPAR	GET PARENT NODE
	GETPTH	GET PATH
	GETSIZE	GET SUBTREE SIZE
	GETTBL	GET A TABLE NAME
	GETTOP	GET TOP OF TREE
	GETUPLFT	GET UPPER LEFTMOST NODE
	GFLDPT	GET FIELD POINTER
	GRP/MAIN	GENERATE APPLICATION/REPORT PROGRAM
	HASDATA	DETERMINE IF THERE ARE ANY SELECT STATEMENTS
	HASITEM	THIS ROUTINE DETERMINES IF THERE IS AN ITEM WITHIN
	HASLOWER	HAS A LOWER FORM WHICH READS THE SAME DATA RECORD?
	HBALANC	HORIZONTAL TREE BALANCE
	HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
	INDENT	INDENT A LINE OF GENERATED CODE
	INSERT	INSERT PROCEDURE
	INSRSV	INSERT RESOLVE
	INSWS	INSERT WORKING STORAGE SECTION
	ISOPNE	DETERMINE IF THIS FIELD IS OPEN ENDED
	MAKACT	MAKE ACTION LIST ELEMENT
	MAKES	MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE
	MAKES/CNU	C NUMBERS
	MAKES/IND	INDENT
	MAKES/NUM	NUMBER PICTURE CLAUSE

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	MAKINS	MAKE INSERT
	MAKINT	MAKE EXPRESSION INTO AN INTEGER
	MAKPS	MAKES THE PRESENTATION SCHEMA RECORD STRUCTURE
	MAKQR	MAKE QUALIFIED REFERENCE
	MAKSTR	MAKE EXPRESSION INTO A STRING
	MAKWH	MAKE WHERE
	MAKWHE	MAKE THE WHERE CLAUSE EXTERNAL SCHEMA VARIABLES
	MAKWHE/C	COBOL WHERE ES
	MAKWHE/C	C WHERE ES
	MAKWHE/N	NUMBER PICTURE CLAUSE
	MAPDB	MAP DATABASE
	MKINC	MAKE INCLUDE (ACTUALLY STRUCTURE TAGS ALA MAKINC)
	MKPOS	MAKE POSITION NODE
	MLPFRM	MAKE A LIST OF PRESENTED FORMS
	MODPAGE	MODIFY PAGES
	MOVCLD	MOVE CHILDREN
	MOVECLD	MOVE CHILD'S POSITION
	MYALLOC	MY MALLOC
	NDMLGEN	NDML COBOL APPLICATION GENERATOR
	NDMLLAB	GENERATE LABELS
	NDMLLNK	LINKAGE SECTION
	NEXTLEV	ADVANCE POINTERS TO NEXT LEVEL OF SUBTREE
	NULBLK	BLANK FILL A STRING
	OPNFIL	GENERATE OPEN FILE SECTION
	PAGNODE	PAGE NODES
	PAGTREE	PAGE TREE
	PEMAP	THE PRESENTATION SCHEMA AND THE EXTERNAL SCHEMA AND MAPPING
	PRNT	PRINT MODULE NAMES HIERARCHICALLY
	PRNTREE	PRINT TREE
	PROCEN	PROCEDURE DIVISION GENERATE
	PSSTRC/CO	COBOL SUBSTITUTE
	PSSTRC/CS	C SUBSTITUTE
	PSSTRC/IN	INDENT
	PUTLIN	PRINT LEVEL OF TREE
	READDB	READ DATA BASE
	READTREE	READ DUMPTREE FILE

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	REPOS	REPOSITION MODULE EXPANSIONS
	RSETNDP	RESET NODUPLICATE FIELDS TO VALUE OF NODUP%D
	RSETSTAT	RESET STATISTIC
	RWEXPD	REPORT WRITER EXPAND ARRAYS
	RWOPN	REPORT WRITER OPEN FORMS
	RWSP/FIXF	FIX UP A FORM
	SAVEES	SAVE ES INFORMATION
	SELECT	GENERATE SELECT CODE
	SELGEN	SELECT GENERATE
	SELLEN	COMPUTE LENGTH OF SELECT PS RECORD
	SELMAP	MAP SELECTED DATA TO OUTPUT RECORD
	SELOPN	SELECT OPEN
	SELRSV	SELECT RESOLVE
	SELWHR	SELECT WHERE
	SELWS	SELECT WORKING STORAGE SECTION
	SETNDP	SET NODUPLICATE FIELDS TO BLANK IF THEY ARE DUPLICATED
	SORT	SORT MODULE NAMES
	SPLICE	SPLICE TREE INTO ANOTHER TREE
	SPLITNODE	SPLIT A NODE FOR PAGE BREAKS
	STATRSV	STATISTIC RESOLVE
	STDCODE	STANDARD COBOL CODE
	STRIPLEV	DRAW STRIP CHART LEVEL
	TRGRSV	TRIGGER RESOLVE
	UQFOR	UNIVERSAL QUALIFIER FOR LOOP
	UQPTH	UNIVERSAL QUALIFIER PATH
	USING	GENERATE USING SECTION
	VISITA	VISIT ARRAYS ON THIS FORM
	WARNING	ISSUE WARNING MESSAGE
	WINRSV	WINDOW RESOLVE
	WRTEXP	WRITE EXPRESSION
	WRTFRM	WRITE FORM
	WRTFRM/DB	DEFAULT BUFFER CLOSE
	WRTFRM/FO	INSERT FORMAT CODES
	WRTFRM/TB	TEXT BUFFER CLOSE
	WRTFRM/WR	WRITE DEFAULT BUFFER
	WRTFRM/WR	WRITE FIELD
	WRTFRM/WR	WRITE TEXT BUFFER
	WRTFRM/WR	WRITE TEXT

REPORT WRITER Where-include-file-used List

Include File -----	Module Name -----	Module Purpose -----
	YYLEX	LEXICAL ANALYZER FOR FLAN
	YYPARSE	FLAN PARSER

3.10.6 Where External Routine Used List

The following lists each external function or routine listed in 3.10.3 and all the documented modules which call it. The purpose of each module is listed as well.

REPORT WRITER Where-external-routine-used List

System Module -----	Module Name -----	Module Purpose -----
ABS	CHKARY CHKFRM CHKFRM RWEXPD	CHECK ARRAY CHECK FORM CHECK FORM REPORT WRITER EXPAND ARRAYS
ADDFRM	FRNTND HRW/MAIN	FRONT END FOR FORMS MAIN MODULE FOR HIERARCHICAL REPORT WRITER
ATOF	YYLEX	LEXICAL ANALYZER FOR FLAN
ATOI	CCONV CES COBCONV COBES HRW/MAIN SAVEES YYLEX	C CONVERSIONS C ES COBOL CONVERSIONS COBOL ES RECORD MAIN MODULE FOR HIERARCHICAL REPORT WRITER SAVE ES INFORMATION LEXICAL ANALYZER FOR FLAN
BLN	CHKFLD CHKSIZE MKINC PSSTRC/COBCOBOL PSSTRC/CSUC SELLEN	CHECK FIELD CHECK SIZE OF ITEMS DOING CONVERSIONS ON MAKE INCLUDE (ACTUALLY STRUCTURE TAGS ALA MAKINC) PSSTRC/COBCOBOL SUBSTITUTE PSSTRC/CSUC SUBSTITUTE COMPUTE LENGTH OF SELECT PS RECORD

REPORT WRITER Where-external-routine-used List

System Module -----	Module Name -----	Module Purpose -----
CALLOC	GRP/MAIN	GENERATE APPLICATION/REPORT PROGRAM
COPFLD	RWEXPD RWSP/FIXFR WINRSV	REPORT WRITER EXPAND ARRAYS UP A FORM WINDOW RESOLVE
DELFLD	FLANCI	FLAN CALLABLE INTERFACE
ERRPRO	CDMESQY	PROGRAM NAME CDMESQY
ESCPY	CCONV CES COBCONV COBES GETTBL HRW/MAIN SAVEES	C CONVERSIONS C ES COBOL CONVERSIONS COBOL ES RECORD GET A TABLE NAME MAIN MODULE FOR HIERARCHICAL REPORT WRITER SAVE ES INFORMATION
FCLOSE	HRW/MAIN NDMLGEN WRTFRM	MAIN MODULE FOR HIERARCHICAL REPORT WRITER NDML COBOL APPLICATION GENERATOR WRITE FORM

REPORT WRITER Where-external-routine-used List

System Module -----	Module Name -----	Module Purpose -----
FGETS	DRAWLEV	DRAW A LEVEL OF THE CHART
	READTREE	READ DUMPTREE FILE
	STRIPLEV	DRAW STRIP CHART LEVEL
FOPEN	GETFILE	RETURN A FILE POINTER BASED ON INPUT FROM THE USER
	GRP/MAIN	GENERATE APPLICATION/REPORT PROGRAM
	HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
	NDMLGEN	NDML COBOL APPLICATION GENERATOR
	WRTFRM	WRITE FORM
FPRINTF	ASSIGN	ASSIGN FILE SECTION
	CONV	C CONVERSIONS
	CES	C ES
	CHKSIZE	CHECK SIZE OF ITEMS DOING CONVERSIONS ON
	CLSFIL	CLOSE FILES
	COBCONV	COBOL CONVERSIONS
	COBES	COBOL ES RECORD
	COBESPS	COBOL ES TO PS
	COBPE	COBOL PE
	DATAGEN	DATA DIVISION GENERATE
	ENDGEN	END GERNERATE
	FD	FD SECTION DECLARATIONS
	FILELNK	FILE LINKAGE SECTION GENERATE
	GEN	GENERATE A LINE OF CODE
	INSERT	INSERT PROCEDURE
	INSWS	INSERT WORKING STORAGE SECTION
	MAKES/CNUMC	NUMBERS
	MAKES/NUMPNUMBER	PICTURE CLAUSE
	MAKINS	MAKE INSERT

REPORT WRITER Where-external-routine-used List

System Module -----	Module Name -----	Module Purpose -----
	MAKWH	MAKE WHERE
	MAKWHE	MAKE THE WHERE CLAUSE EXTERNAL SCHEMA VARIABLES
	MAKWHE/COCOBOL	WHERE ES
	MAKWHE/NUNUMBER	PICTURE CLAUSE
	NDMLLAB	GENERATE LABELS
	OPNFIL	GENERATE OPEN FILE SECTION
	PRNTREE	PRINT TREE
	PROCEN	PROCEDURE DIVISION GENERATE
	PSSTRC/COBCOBOL	SUBSTITUTE
	PSSTRC/CSUC	SUBSTITUTE
	SELGEN	SELECT GENERATE
	SELWS	SELECT WORKING STORAGE SECTION
	STDCODE	STANDARD COBOL CODE
	USING	GENERATE USING SECTION
 FPUTS		
	DOINDEX	DO CHART INDEX
	PRNTREE	PRINT TREE
 FREE		
	CHKFLD	CHECK FIELD
	CHKFRM	CHECK FORM
	DELNODE	DELETE A SPECIFIED NODE IN TREE
	DOINDEX	DO CHART INDEX
	DRAWLEV	DRAW A LEVEL OF THE CHART
	STRIPLEV	DRAW STRIP CHART LEVEL
	WINRSV	WINDOW RESOLVE
	WRTEXP	WRITE EXPRESSION
	YYPARSE	FLAN PARSER
 FSEEK		
	DRAWLEV	DRAW A LEVEL OF THE CHART
	STRIPLEV	DRAW STRIP CHART LEVEL

REPORT WRITER Where-external-routine-used List

System Module -----	Module Name -----	Module Purpose -----
FTELL	READTREE	READ DUMPTREE FILE
FWRITE	WRTFRM	WRITE FORM
	WRTFRM/DBFDEFAULT	BUFFER CLOSE
	WRTFRM/TBFTEXT	BUFFER CLOSE
	WRTFRM/WRTWRITE	DEFAULT BUFFER
	WRTFRM/WRTWRITE	FIELD
	WRTFRM/WRTWRITE	TEXT BUFFER
	WRTFRM/WRTWRITE	TEXT
GDATA	FRNTND	FRONT END FOR FORMS
	HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
GETC	DRAWLEV	DRAW A LEVEL OF THE CHART
	READTREE	READ DUMPTREE FILE
	STRIPLEV	DRAW STRIP CHART LEVEL
	YLEX	LEXICAL ANALYZER FOR FLAN
INITAL	FRNTND	FRONT END FOR FORMS
	HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
INITFP		

REPORT WRITER Where-external-routine-used List

System Module -----	Module Name -----	Module Purpose -----
	FRNTND	FRONT END FOR FORMS
	HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
INSMAP		
	PROCEN	PROCEDURE DIVISION GENERATE
ISALNUM		
	YYLEX	LEXICAL ANALYZER FOR FLAN
ISALPHA		
	YYLEX	LEXICAL ANALYZER FOR FLAN
ISDIGIT		
	YYLEX	LEXICAL ANALYZER FOR FLAN
ISSPACE		
	YYLEX	LEXICAL ANALYZER FOR FLAN
MAKFLD		
	YYPARSE	FLAN PARSER
MALLOC		
	BLDMOD	BUILD MODULE
	BLDNODE	BUILD NODE
	DOINDEX	DO CHART INDEX
	DRAWLEV	DRAW A LEVEL OF THE CHART

REPORT WRITER Where-external-routine-used List

System Module -----	Module Name -----	Module Purpose -----
	MLPFRM	MAKE A LIST OF PRESENTED FORMS
	MYALLOC	MY MALLOC
	SORT	SORT MODULE NAMES
	STRIPLEV	DRAW STRIP CHART LEVEL
	UQPTH	UNIVERSAL QUALIFIER PATH
	WINRSV	WINDOW RESOLVE
MAP	PROCEN	PROCEDURE DIVISION GENERATE
MAX	CHKFLD	CHECK FIELD
	CHKFRM	CHECK FORM
	DCLINDX	DECLARE INDEX VARIABLES
	GETSIZE	GET SUBTREE SIZE
	HBALANC	HORIZONTAL TREE BALANCE
MEMCMP	FRNTND	FRONT END FOR FORMS
	HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
MEMCPY	CHKFLD	CHECK FIELD
	DOINDEX	DO CHART INDEX
	DRAWLEV	DRAW A LEVEL OF THE CHART
	STRIPLEV	DRAW STRIP CHART LEVEL
	WRTEXP	WRITE EXPRESSION
	WRTFRM/WRT	WRITE FIELD
	YYPARSE	FLAN PARSER

REPORT WRITER Where-external-routine-used List

System Module -----	Module Name -----	Module Purpose -----
MEMSET	CHKFLD	CHECK FIELD
	DOINDEX	DO CHART INDEX
	DRAWLEV	DRAW A LEVEL OF THE CHART
	STRIPLEV	DRAW STRIP CHART LEVEL
MIN	GETSIZE	GET SUBTREE SIZE
	STRIPLEV	DRAW STRIP CHART LEVEL
OISCR	FRNTND	FRONT END FOR FORMS
	GRP/MAIN	GENERATE APPLICATION/REPORT PROGRAM
	HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
OUTSCR	HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
PMSGLC	FRNTND	FRONT END FOR FORMS
	GRP/MAIN	GENERATE APPLICATION/REPORT PROGRAM
PMSGLS	BLDMOD	BUILD MODULE
	ERROR	ISSUE ERROR MESSAGE
	FATAL	ISSUE FATAL ERROR MESSAGE
	HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
	WARNING	ISSUE WARNING MESSAGE

REPORT WRITER Where-external-routine-used List

System Module -----	Module Name -----	Module Purpose -----
PRINTF	PRNT YYPARSE	PRINT MODULE NAMES HIERARCHICALLY FLAN PARSER
PSESMAP	PROCGEN	PROCEDURE DIVISION GENERATE
PTHPTR	GETPTH UQPTH	GET PATH UNIVERSAL QUALIFIER PATH
PUTATT	HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
PUTC	DOINDEX ESPSMAP/ININDENT INDENT MAKES/INDEINDENT PRNTREE PSSTRC/INDINDENT PUTLIN	DO CHART INDEX INDENT A LINE OF GENERATED CODE PRINT TREE PRINT LEVEL OF TREE
PUTCUR	HRW/MAIN	MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Where-external-routine-used List

System Module -----	Module Name -----	Module Purpose -----
SPRINTF		
	BLDMOD	BUILD MODULE
	BSCODE	BUILD SUBROUTINE CODE
	CALCSTAT	CALCULATE STATISTIC
	CHKGRP	CHECK FOR GROUP SEPERATORS OR END OF FILE
	CLRNDP	CLEAR NODUPLICATE FIELDS
	DBFREAD	GENERATE DATA BASE FREAD
	DCLINDX	DECLARE INDEX VARIABLES
	DOINDEX	DO CHART INDEX
	DRAWLEV	DRAW A LEVEL OF THE CHART
	ERROR	ISSUE ERROR MESSAGE
	FATAL	ISSUE FATAL ERROR MESSAGE
	FRMPDAT	FORM PDATA
	FRNTND	FRONT END FOR FORMS
	GENAAL	GENERATE PROCEDURE "ADDAL" ADD ACTION LIST
	GENAH	GENERATE ACTION HELP
	GENAI	GENERATE ACTION INSERT
	GENAP	GENERATE ACTION PAGE
	GENAQ	GENERATE ACTION QUERY (SELECT)
	GENAR	GENERATE ACTION PRESENT
	GENAS	GENERATE ACTION SET
	GENAT	GENERATE ACTION SIGNAL
	GENBEG	GENERATE BEGINNING OF APPLICATION OR REPORT
	GENCHG	GENERATE CHANGE DECLARATIONS
	GENDB	GENERATE DATA BASE RECORDS AND FILE DECLARATIONS
	GENDOA	GENERATE PROCEDURE "DOACT" DO ACTION
	GENDS	GENERATE DATA DATA STRUCTURES
	GENFP	GENERATE FORM PATH
	GENFS	GENERATE FORM DATA STRUCTURES
	GENFSD	GENERATE FORM STRUCTURE DATA INITIALIZATION
	GENINS	GENERATE INSERT DECLARATIONS
	GENMAIN	GENERATE MAIN PROGRAM
	GENNDP	GENERATE NODUPLICATE DECLARATIONS

REPORT WRITER Where-external-routine-used List

System Module -----	Module Name -----	Module Purpose -----
	GETFILE	RETURN A FILE POINTER BASED ON INPUT FROM THE USER
	MAKQR	MAKE QUALIFIED REFERENCE
	MAPDB	MAP DATABASE
	MKINC	MAKE INCLUDE (ACTUALLY STRUCTURE TAGS ALA MAKINC)
	NDMLGEN	NDML COBOL APPLICATION GENERATOR
	PRNTREE	PRINT TREE
	RSETNDP	RESET NODUPLICATE FIELDS TO VALUE OF NODUP%D
	RSETSTAT	RESET STATISTIC
	SELGEN	SELECT GENERATE
	SELOPN	SELECT OPEN
	SELWHR	SELECT WHERE
	SETNDP	SET NODUPLICATE FIELDS TO BLANK IF THEY ARE DUPLICATED
	STDCODE	STANDARD COBOL CODE
	UQFOR	UNIVERSAL QUALIFIER FOR LOOP
	VISITA	VISIT ARRAYS ON THIS FORM
	WARNING	ISSUE WARNING MESSAGE
	WRTEXP	WRITE EXPRESSION
	WRTFRM	WRITE FORM
	YYPARSE	FLAN PARSER
STRASN	CHKARY	CHECK ARRAY
	CHKFRM	CHECK FORM
	RWEXPD	REPORT WRITER EXPAND ARRAYS
	WRTFRM	WRITE FORM
STRCAT	DCLINDX	DECLARE INDEX VARIABLES
	GRP/MAIN	GENERATE APPLICATION/REPORT PROGRAM
	MAKES/CNUMC	NUMBERS
	MAKQR	MAKE QUALIFIED REFERENCE
	YYPARSE	FLAN PARSER

REPORT WRITER Where-external-routine-used List

System Module -----	Module Name -----	Module Purpose -----
STRCHR		
	DASH	WRITE DASH '-'
	FRNTND	FRONT END FOR FORMS
	GENDOA	GENERATE PROCEDURE "DOACT" DO ACTION
	GETCOL	GET THE COLUMN NAME OF A TABLE.COLUMN OR COLUMN STRING
	GETPTH	GET PATH
	GETTBL	GET A TABLE NAME
	GRP/MAIN	GENERATE APPLICATION/REPORT PROGRAM
	MAPDB	MAP DATABASE
	NULBLK	BLANK FILL A STRING
	PUTLIN	PRINT LEVEL OF TREE
	UQPTH	UNIVERSAL QUALIFIER PATH
	YYPARSE	FLAN PARSER
STRCMP		
	BLDMOD	BUILD MODULE
	DOINDEX	DO CHART INDEX
	FNDATT	FIND ATTRIBUTE
	FNDFRM	FIND FORM
	GENAR	GENERATE ACTION PRESENT
	GETTBL	GET A TABLE NAME
	GFLDPT	GET FIELD POINTER
	RWSP/FIXFR	FIX UP A FORM
	SELWS	SELECT WORKING STORAGE SECTION
	SORT	SORT MODULE NAMES
	YYLEX	LEXICAL ANALYZER FOR FLAN
	YYPARSE	FLAN PARSER
STRCPY		
	BLDMOD	BUILD MODULE
	CSTASH	CHARACTER STASH
	DCLINDX	DECLARE INDEX VARIABLES

REPORT WRITER Where-external-routine-used List

System Module -----	Module Name -----	Module Purpose -----
	GENAS	GENERATE ACTION SET
	GETCOL	GET THE COLUMN NAME OF A TABLE.COLUMN OR COLUMN STRING
	GETPTH	GET PATH
	GETTBL	GET A TABLE NAME
	GRP/MAIN	GENERATE APPLICATION/REPORT PROGRAM
	INSWS	INSERT WORKING STORAGE SECTION
	MAKES	MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE
	MAKQR	MAKE QUALIFIED REFERENCE
	NULBLK	BLANK FILL A STRING
	SELGEN	SELECT GENERATE
	SELWS	SELECT WORKING STORAGE SECTION
	SORT	SORT MODULE NAMES
	UQPTH	UNIVERSAL QUALIFIER PATH
	WRTFRM	WRITE FORM
	WRTFRM/WRT	WRITE FIELD
	YYPARSE	FLAN PARSER

STRLEN

CHKFLD	CHECK FIELD
CHKFRM	CHECK FORM
CSTASH	CHARACTER STASH
DCLINDX	DECLARE INDEX VARIABLES
DOINDEX	DO CHART INDEX
DRAWLEV	DRAW A LEVEL OF THE CHART
ERROR	ISSUE ERROR MESSAGE
FATAL	ISSUE FATAL ERROR MESSAGE
GENAS	GENERATE ACTION SET
GENFSD	GENERATE FORM STRUCTURE DATA INITIALIZATION
MAKES	MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE
MAPDB	MAP DATABASE
MKINC	MAKE INCLUDE (ACTUALLY STRUCTURE TAGS ALA MAKINC)
PRNTREE	PRINT TREE
PUTLIN	PRINT LEVEL OF TREE
READTREE	READ DUMPTREE FILE

REPORT WRITER Where-external-routine-used List

System Module -----	Module Name -----	Module Purpose -----
	RSETNDP	RESET NODUPLICATE FIELDS TO VALUE OF NODUP%D
	SAVEES	SAVE ES INFORMATION
	SETNDP	SET NODUPLICATE FIELDS TO BLANK IF THEY ARE DUPLICATED
	STRIPLEV	DRAW STRIP CHART LEVEL
	VISITA	VISIT ARRAYS ON THIS FORM
	WARNING	ISSUE WARNING MESSAGE
	WRTEXP	WRITE EXPRESSION
	WRTFRM	WRITE FORM
	WRTFRM/WRT	WRITE TEXT
	YYPARSE	FLAN PARSER
STRNCMP	CCONV	C CONVERSIONS
	SAVEES	SAVE ES INFORMATION
STRNCPY	MAKES	MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE
	NDMLGEN	NDML COBOL APPLICATION GENERATOR
	WRTFRM/WRT	WRITE FIELD
	YYPARSE	FLAN PARSER
STRSPN	GENAS	GENERATE ACTION SET
STRUPC	GETPTH	GET PATH
	SORT	SORT MODULE NAMES
	STDCODE	STANDARD COBOL CODE
	YYPARSE	FLAN PARSER

REPORT WRITER Where-external-routine-used List

System Module -----	Module Name -----	Module Purpose -----
SYSMSG	CHKFLD NDMLGEN WRTFRM	CHECK FIELD NDML COBOL APPLICATION GENERATOR WRITE FORM
TERMPF	GRP/MAIN HRW/MAIN	GENERATE APPLICATION/REPORT PROGRAM MAIN MODULE FOR HIERARCHICAL REPORT WRITER
TOUPPER	YYLEX	LEXICAL ANALYZER FOR FLAN
TRMNAT	FRNTND HRW/MAIN	FRONT END FOR FORMS MAIN MODULE FOR HIERARCHICAL REPORT WRITER
TRMNDML	GRP/MAIN	GENERATE APPLICATION/REPORT PROGRAM
UNGETC	READTREE YYLEX	READ DUMPTREE FILE LEXICAL ANALYZER FOR FLAN

REPORT WRITER Where-external-routine-used List

System Module -----	Module Name -----	Module Purpose -----
YYERROR	YYPARSE	FLAN PARSER

3.10.7 Main Program Parts List

The following lists each Main Program listed in 3.10.1 and all the modules which are called either by that module itself or by any of the documented modules which it calls. It is possible for a non-main module to be listed more than once if it is called by multiple modules. The called modules, in this case known as program parts, are marked as to whether they are documented here. If so, the phrase "well-defined module" appears by the module name, if not it is an "external routine". The Purpose of the Main Program module is listed as well.

REPORT WRITER Main Program Parts List

Main Pgm Name -----	Module Name -----	Module Type -----
GRP/MAIN	Purpose-->	GENERATE APPLICATION/REPORT PROGRAM
	ABS	External routine
	ACTRSV	Well-defined module
	ADDCHK	Well-defined module
	ADDFRM	External routine
	ASSIGN	Well-defined module
	ATOF	External routine
	ATOI	External routine
	BLDSUB	Well-defined module
	BLN	External routine
	BSCODE	Well-defined module
	CALCSTAT	Well-defined module
	CALLOC	External routine
	CCONV	Well-defined module
	CDMESQY	Well-defined module
	CES	Well-defined module
	CESPS	Well-defined module
	CHKARY	Well-defined module
	CHKFLD	Well-defined module
	CHKFRM	Well-defined module
	CHKGRP	Well-defined module
	CHKSIZE	Well-defined module
	CLRNDP	Well-defined module
	CLSFIL	Well-defined module
	COBCONV	Well-defined module
	COBES	Well-defined module
	COBESPS	Well-defined module
	COBPE	Well-defined module
	COPFLD	External routine
	CPE	Well-defined module
	CSTASH	Well-defined module
	CTLRSV	Well-defined module
	DASH	Well-defined module
	DATAGEN	Well-defined module
	DBFREAD	Well-defined module
	DCLINDX	Well-defined module
	DELFLD	External routine
	ENDGEN	Well-defined module
	ERROR	Well-defined module

REPORT WRITER Main Program Parts List

Main Pgm Name -----	Module Name -----	Module Type -----
	ERRPRO	External routine
	ESCPY	External routine
	ESPSMAP	Well-defined module
	ESPSMAP/INDENT	Well-defined module
	FATAI	Well-defined module
	FCLOSE	External routine
	FD	Well-defined module
	FILELNK	Well-defined module
	FLANCI	Well-defined module
	FLDRSV	Well-defined module
	FLDTYP	Well-defined module
	FNDATT	Well-defined module
	FNDFRM	Well-defined module
	FOPEN	External routine
	FPRINTF	External routine
	FREE	External routine
	FRMPDAT	Well-defined module
	FRNTND	Well-defined module
	FWRITE	External routine
	GDATA	External routine
	GEN	Well-defined module
	GENAA	Well-defined module
	GENAAL	Well-defined module
	GENACT	Well-defined module
	GENAE	Well-defined module
	GENAH	Well-defined module
	GENAI	Well-defined module
	GENAL	Well-defined module
	GENAP	Well-defined module
	GENAQ	Well-defined module
	GENAR	Well-defined module
	GENAS	Well-defined module
	GENAT	Well-defined module
	GENBEG	Well-defined module
	GENCHG	Well-defined module
	GENDB	Well-defined module
	GENDOA	Well-defined module
	GENDS	Well-defined module
	GENFP	Well-defined module
	GENFS	Well-defined module

REPORT WRITER Main Program Parts List

Main Pgm Name -----	Module Name -----	Module Type -----
	GENFSD	Well-defined module
	GENINS	Well-defined module
	GENMAIN	Well-defined module
	GENNDP	Well-defined module
	GENPAG	Well-defined module
	GETC	External routine
	GETCOL	Well-defined module
	GETFILE	Well-defined module
	GETPTH	Well-defined module
	GETTBL	Well-defined module
	GFLDPT	Well-defined module
	HASDATA	Well-defined module
	HASITEM	Well-defined module
	HASLOWER	Well-defined module
	INDENT	Well-defined module
	INITAL	External routine
	INITFP	External routine
	INSERT	Well-defined module
	INSMAP	External routine
	INSRSV	Well-defined module
	INSWS	Well-defined module
	ISALNUM	External routine
	ISALPHA	External routine
	ISDIGIT	External routine
	ISOPNE	Well-defined module
	ISSPACE	External routine
	MAKACT	Well-defined module
	MAKES	Well-defined module
	MAKES/CNUMPIC	Well-defined module
	MAKES/INDENT	Well-defined module
	MAKES/NUMPIC	Well-defined module
	MAKFLD	External routine
	MAKINS	Well-defined module
	MAKINT	Well-defined module
	MAKPS	Well-defined module
	MAKQR	Well-defined module
	MAKSTR	Well-defined module
	MAKWH	Well-defined module
	MAKWHEs	Well-defined module
	MAKWHEs/COBWHEs	Well-defined module

REPORT WRITER Main Program Parts List

Main Pgm Name -----	Module Name -----	Module Type -----
	MAKWHE/CWHES	Well-defined module
	MAKWHE/NUMPIC	Well-defined module
	MALLOC	External routine
	MAP	External routine
	MAPDB	Well-defined module
	MAX	External routine
	MEMCMP	External routine
	MEMCPY	External routine
	MEMSET	External routine
	MKINC	Well-defined module
	MKPOS	Well-defined module
	MLPFRM	Well-defined module
	MYALLOC	Well-defined module
	NDMLGEN	Well-defined module
	NDMLLAB	Well-defined module
	NDMLLNK	Well-defined module
	NULBLK	Well-defined module
	OISCR	External routine
	OPNFIL	Well-defined module
	PEMAP	Well-defined module
	PMSGLC	External routine
	PMSGLS	External routine
	PRINTF	External routine
	PROCGEN	Well-defined module
	PSEMAP	External routine
	PSSTRC/COBSUB	Well-defined module
	PSSTRC/CSUB	Well-defined module
	PSSTRC/INDENT	Well-defined module
	PTHPTR	External routine
	PUTC	External routine
	READDB	Well-defined module
	RSETNDP	Well-defined module
	RSETSTAT	Well-defined module
	RWEXPD	Well-defined module
	RWOPN	Well-defined module
	RWSP/FIXFRM	Well-defined module
	SAVEES	Well-defined module
	SELECT	Well-defined module
	SELGEN	Well-defined module
	SELLEN	Well-defined module

REPORT WRITER Main Program Parts List

Main Pgm Name -----	Module Name -----	Module Type -----
	SELMAP	Well-defined module
	SELOPN	Well-defined module
	SELRSV	Well-defined module
	SELWHR	Well-defined module
	SELWS	Well-defined module
	SETNDP	Well-defined module
	SPRINTF	External routine
	STATRSV	Well-defined module
	STDCODE	Well-defined module
	STRASN	External routine
	STRCAT	External routine
	STRCHR	External routine
	STRCMP	External routine
	STRCPY	External routine
	STRLEN	External routine
	STRNCMP	External routine
	STRNCPY	External routine
	STRSPN	External routine
	STRUPC	External routine
	SYSMSG	External routine
	TERMFP	External routine
	TOUPPER	External routine
	TRGRSV	Well-defined module
	TRMNAT	External routine
	TRMNDML	External routine
	UNGETC	External routine
	UQFOR	Well-defined module
	UQPTH	Well-defined module
	USING	Well-defined module
	VISITA	Well-defined module
	WARNING	Well-defined module
	WINRSV	Well-defined module
	WRTEXP	Well-defined module
	WRTFRM	Well-defined module
	WRTFRM/DBFCLOS	Well-defined module
	WRTFRM/FORMAT	Well-defined module
	WRTFRM/TBFCLOS	Well-defined module
	WRTFRM/WRTDBF	Well-defined module
	WRTFRM/WRTFLD	Well-defined module
	WRTFRM/WRTTBF	Well-defined module

REPORT WRITER Main Program Parts List

Main Pgm Name -----	Module Name -----	Module Type -----
	WRTFRM/WRTTXT	Well-defined module
	YYERROR	External routine
	YYLEX	Well-defined module
	YYPARSE	Well-defined module

REPORT WRITER Main Program Parts List

Main Pgm Name -----	Module Name -----	Module Type -----
HRW/MAIN	Purpose-->	MAIN MODULE FOR HIERARCHICAL REPORT WRITER
	ADDFRM	External routine
	ARRANGE	Well-defined module
	ATOI	External routine
	BLDMOD	Well-defined module
	BLDNODE	Well-defined module
	CLOSEGAP	Well-defined module
	COPYNODE	Well-defined module
	DELNODE	Well-defined module
	DOINDEX	Well-defined module
	DRAWLEV	Well-defined module
	ESCPY	External routine
	FCLOSE	External routine
	FGETS	External routine
	FOPEN	External routine
	FPRINTF	External routine
	FPUTS	External routine
	FREE	External routine
	FSEEK	External routine
	FTELL	External routine
	GDATA	External routine
	GETC	External routine
	GETFIT	Well-defined module
	GETLOWLEF	Well-defined module
	GETLOWRIT	Well-defined module
	GETPAR	Well-defined module
	GETSIZE	Well-defined module
	GETTOP	Well-defined module
	GETUPLFT	Well-defined module
	HBALANC	Well-defined module
	INITAL	External routine
	INITFP	External routine
	MALLOC	External routine
	MAX	External routine
	MEMCMP	External routine
	MEMCPY	External routine
	MEMSET	External routine
	MIN	External routine
	MODPAGE	Well-defined module

REPORT WRITER Main Program Parts List

Main Pgm Name -----	Module Name -----	Module Type -----
	MOVCLD	Well-defined module
	MOVECLD	Well-defined module
	NEXTLEV	Well-defined module
	OISCR	External routine
	OUTSCR	External routine
	PAGNODE	Well-defined module
	PAGTREE	Well-defined module
	PMSGLS	External routine
	PRNTREE	Well-defined module
	PUTATT	External routine
	PUTC	External routine
	PUTCUR	External routine
	PUTLIN	Well-defined module
	READTREE	Well-defined module
	REPOS	Well-defined module
	SORT	Well-defined module
	SPLICE	Well-defined module
	SPLITNODE	Well-defined module
	SPRINTF	External routine
	STRCHR	External routine
	STRCMP	External routine
	STRCPY	External routine
	STRIPLEV	Well-defined module
	STRLEN	External routine
	STRUPC	External routine
	TERMFP	External routine
	TRMNAT	External routine
	UNGETC	External routine

3.10.8 Module Documentation

The following documentation describes information which is specific to each individual module being documented in this specification as listed in section 3.10.2. It provides a compact way of getting information that would be otherwise buried within each module's source code.

The specific items in this module documentation have the following meanings:

NAME:	Name of program Module.
PURPOSE:	Purpose of Module as detailed in the source code.
LANGUAGE:	Programming language source code is written in. The choices are: VAX-11 FORTRAN C (I/S-1 Workbench 'C') VAX-11 COBOL
MODULE TYPE:	Whether a Program, Subroutine, or Function.
SOURCE FILE:	Name of Source File from file specification.
SOURCE FILE TYPE:	Source File Extension from file specification.
HOST:	Whether this is a host-dependent routine (VAX or IBM) or blank if host-independent.
SUBSYSTEM:	IISS sub-system this file resides in.
SUBDIRECTORY:	Sub-directory of that subsystem in which this file resides.
DOCUMENTATION GROUP:	Name of documentation group of which this source file is a member.
DESCRIPTION:	A description of the module as obtained from the source code.
ARGUMENTS:	The arguments with which this routine is called if it is a Subroutine or a Function.
INCLUDE FILES:	A list of all the files that are included into this module as well as their purposes.

ROUTINES CALLED: Subroutines or Functions, either documented or external, called by this module, if any.

CALLED DIRECTLY BY: The documented routines which call this module, if any.

USED IN MAIN PROGRAM(S): The documented Main Programs which contain this module in their parts list according to the list in section 3.10.7.

The Module Documentation is arranged alphabetically according to Module Name.

REPORT WRITER Module Documentation

NAME: ACTRSV
PURPOSE: ACTION RESOLVE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

ACTRSV(ACTPTR, TRGPTR)
 ACTLST *ACTPTR;
 TRGLST *TRGPTR;

INPUTS:

ACTPTR - ACTION LIST FROM WHICH TO LOOK FOR PATHS.
TRGPTR - TRIGGER ASSOCIATED WITH THIS ACTION.

DESCRIPTION

RESOLVES ALL QUALIFIED NAMES INTO FIELD POINTERS FOR ALL
 NAMES
WHICH ARE ROOTED IN ACTLST (ACTION LIST).

ARGUMENTS:

ACTPTR = ACTLST *
TRGPTR = TRGLST *

INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

GETPTH - GET PATH
ERROR - ISSUE ERROR MESSAGE
SELRSV - SELECT RESOLVE
INSRSV - INSERT RESOLVE
UQPTH - UNIVERSAL QUALIFIER PATH

CALLED DIRECTLY BY:

TRGRSV - TRIGGER RESOLVE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: ADDCHK
PURPOSE: ADD POSITION TO CHECK LIST
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS

VOID ADDCHK(POSPTR)
POS *POSPTR;

DESCRIPTION

ADDS THE SPECIFIED POSITION TO THE OVERLAP CHECK LIST

ARGUMENTS:

POSPTR = POS *

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

FLD TYP - FIELD TYPE
ERROR - ISSUE ERROR MESSAGE

CALLED DIRECTLY BY:

CHKFRM - CHECK FORM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: ARRANGE
PURPOSE: ARRANGE CHART AND ASSIGNS PAGE NUMBERS
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: ARRANGE
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION: -----

SYNOPSIS
ARRANGE()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION
THIS ROUTINE ASSIGNS PAGE NUMBERS.

INCLUDE FILES: -----

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
CHART - CHART INCLUDE FILE

ROUTINES CALLED: -----

GETLOWLEF - GET LOWER LEFT CHILD NODE
GETTOP - GET TOP OF TREE

CALLED DIRECTLY BY: -----

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

USED IN MAIN PROGRAM(S): -----

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: ASSIGN
PURPOSE: ASSIGN FILE SECTION
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

SPTR = SELECT *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

FPRINTF
INDENT - INDENT A LINE OF GENERATED CODE

CALLED DIRECTLY BY:

DATAGEN - DATA DIVISION GENERATE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: BLDMOD
PURPOSE: BUILD MODULE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: MODULE * ()
SOURCE FILE: BLDMOD
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
BLDMOD()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

RETURNS A POINTER TO THE NAMED MODULE. THE MODULE IS
ALLOCATED IF IT
DOES NOT ALREADY EXIST

ARGUMENTS:

MODULE_NAME = CHAR []
FILEPOS = LONG
WIDTH = INT
DEPTH = INT
TOP_POS = INT
BOT_POS = INT
L_MARGIN = INT

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

STRCMP

SPRINTF
PMSGLS
MALLOC
STRCPY

CALLED DIRECTLY BY:

COPYNODE - COPY A NODE IN TREE
GETPAR - GET PARENT NODE
READTREE - READ DUMPTREE FILE

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: BLDNODE
PURPOSE: BUILD NODE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: NODE * ()
SOURCE FILE: BLDNODE
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
BLDNODE()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

THIS ROUTINE BUILDS A LINKED LIST FOR TREE RELATIONSHIPS,
IT
SETS UP PARENT - CHILD RELATIONSHIPS AS WELL AS LEFT -
RIGHT.

ARGUMENTS:

PARENT_PTR = NODE *
LEFT_PTR = NODE *
RIGHT_PTR = NODE *
MODULE_PTR = MODULE *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

MALLOC

CALLED DIRECTLY BY:

COPYNODE - COPY A NODE IN TREE
GETPAR - GET PARENT NODE
MOVCLD - MOVE CHILDREN
READTREE - READ DUMPTREE FILE
SPLITNODE - SPLIT A NODE FOR PAGE BREAKS

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: BLDSUB
PURPOSE: BUILD SUBROUTINES
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

BLDSUB (DP)
FIELD *DP;

INPUTS/OUTPUTS:
NONE

INPUTS:
(DP) - FIELD POINTER

OUTPUTS:
NONE

DESCRIPTION

THIS ROUTINE TRAVERSES THE FORMS HIERARCHY LOOKING FOR
FORMS
WHICH HAVE A SELECT WHICH TARGETS TO ITEMS ON THE FORM OR
ONE
OF ITS SUBFORMS. WHEN IT FINDS ONE IT CALLS BSCODE WHICH
GENERATES A FORM PROCEDURE.

ARGUMENTS:

DP = FIELD *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPDINI - FPD INITIALIZATION
FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

HASDATA - DETERMINE IF THERE ARE ANY SELECT STATEMENTS
BSCODE - BUILD SUBROUTINE CODE
BLDSUB - BUILD SUBROUTINES

CALLED DIRECTLY BY:

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM
BLDSUB - BUILD SUBROUTINES

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: BSCODE
PURPOSE: BUILD SUBROUTINE CODE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

BSCODE(DP)
FIELD *DP;

INPUTS/OUTPUTS:
NONE

INPUTS:
(DP) - FIELD POINTER

OUTPUTS:
NONE

DESCRIPTION

THE FIELD POINTER WHICH IS PASSED TO THIS ROUTINE IS
POINTING
TO A FIELD WHOSE CONTENTS ARE A FORM. THIS ROUTINE
GENERATES THE
CODE FOR A SUBROUTINE THAT CORRESPONDS TO THE NAME OF THAT
FORM.
THIS PROCEDURE IMPLEMENTS THE "INSTANTIATION RULES". THE
FORM
PROCEDURES ARE OF THE FORM:

```
FORMNAME(FORMPTR, FORMPATH)
  STRUCT FRM%D *FORMPTR;          POINTER TO DATA
                                  STRUCTURE OF FORM.
  CHAR *FORMPATH;                PATH IN FORM PROCESSOR
                                  TO FORM.
  {
    <DECLARE SOME VARIABLES>

                                  "VISIT ALL ITEMS ON
FORM".
                                  COPY DATA VALUES TO
ITEMS ON FORM.
```

```
MEMCPY(FORMPTR->FIELD, DBR%D.FIELD, SIZE);  
. . .  
                                "VISIT ALL ARRAYS ON  
                                FORM".  
FOR (I = 0; !DONE; I++)  
{  
    <CHECK FOR GROUP SEPERATOR OR END OF FILE OF  
    DATA RECORDS WHICH TARGET TO THESE SUBFORMS.>  
  
    <CHECK FOR OVERFLOW ON THIS ARRAY.>  
  
    <CALL THE SUBFORM'S PROCEDURE.>  
}  
. . .  
  
    <READ NEXT DATA RECORD AND CHECK FOR CHANGE  
    CONDITIONS.>  
  
    RETURN <TRUE IF ANY CONDITIONS TRIPPED OR READ END  
    OF FILE.>  
}
```

ARGUMENTS:

DP = FIELD *

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPDINI - FPD INITIALIZATION
FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

SPRINTF
GEN - GENERATE A LINE OF CODE
READDB - READ DATA BASE
RSETNDP - RESET NODUPLICATE FIELDS TO VALUE OF NODUP%D
MAPDB - MAP DATABASE
VISITA - VISIT ARRAYS ON THIS FORM
HASITEM - THIS ROUTINE DETERMINES IF THERE IS AN ITEM
 WITHIN
SETNDP - SET NODUPLICATE FIELDS TO BLANK IF THEY ARE
 DUPLICATED

CALLED DIRECTLY BY:

BLDSUB - BUILD SUBROUTINES

PS 620344501
30 September 1990

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: CALCSTAT
PURPOSE: CALCULATE STATISTIC
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

FP = FIELD *
DP = FIELD *

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

CALCSTAT - CALCULATE STATISTIC
MAKQR - MAKE QUALIFIED REFERENCE
SPRINTF
GEN - GENERATE A LINE OF CODE

CALLED DIRECTLY BY:

FRMPDAT - FORM PDATA
CALCSTAT - CALCULATE STATISTIC

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: CCONV
PURPOSE: C CONVERSIONS
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: MAKES
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

ES = ESDTYPE *
TBLSTR = CHAR *
SELNO = INT

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

ATOI
ESCPY
STRNCMP
FPRINTF
MAKES/INDENT -- INDENT

CALLED DIRECTLY BY:

MAKES - MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: CDMESQY
PURPOSE: PROGRAM NAME CDMESQY
LANGUAGE: VAX-11 COBOL
MODULE TYPE: SUBROUTINE
SOURCE FILE: CDMESQY
SOURCE FILE TYPE: .PRC
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SELECTS ALL THE DATA ITEMS FOR A GIVEN VIEW AND RETRIEVES
EACH ES DATA ITEM'S MACHINE TYPE, SIZE, AND NUMBER OF DECIMAL
DIGITS.
THIS INFORMATION IS RETURNED TO THE CALLING PROGRAM IN AN
ARRAY STRUCTURE.
THIS ROUTINE WILL CHANGE WHEN DOMAINS AND DATA TYPES ARE
COMPLETELY DEFINED FOR THE CDM.

ARGUMENTS:

VIEW = DSPLY [X(30)]

INCLUDE FILES:

SRVRET - AS THE RETURN GIVEN A TABLE-FULL ERROR
ERRPRO - PROCESS ERROR INCLUDE FILE

ROUTINES CALLED:

ERRPRO

CALLED DIRECTLY BY:

MAKES - MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: CES
PURPOSE: C ES
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: MAKES
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

ES = ESDTYPE *
SELNO = INT
TBLNUM = INT
REC_CNT = INT

INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

FPRINTF
ATOI
ESCPY
MAKES/INDENT - INDENT

CALLED DIRECTLY BY:

MAKES - MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: CESPS
PURPOSE: C ES TO PS
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: ESPSMAP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

SELPTR = SELECT *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

CALLED DIRECTLY BY:

ESPSMAP - THE EXTERNAL SCHEMA AND PRESENTATION SCHEMA
MAPPING

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: CHKARY
PURPOSE: CHECK ARRAY
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS

VOID CHKARY(ARYPTR)
FIELD *ARYPTR;

DESCRIPTION

GENERATES POSITIONS FOR EACH ELEMENT OF AN ARRAY FOR
OVERLAP CHECKING

ARGUMENTS:

ARYPTR = FIELD *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

MYALLOC - MY MALLOC
ABS
STRASN

CALLED DIRECTLY BY:

CHKFRM - CHECK FORM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: CHKFLD
PURPOSE: CHECK FIELD
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS

VOID CHKFLD()

DESCRIPTION

CHECKS THE CURRENT FIELD FOR COMPLETENESS AND CONSISTENCY

INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

FNDATT - FIND ATTRIBUTE
ERROR - ISSUE ERROR MESSAGE
MEMSET
MAX
FREE
WRTEXP - WRITE EXPRESSION
BLEN
MEMCPY
SYSMSG
MYALLOC - MY MALLOC
STRLEN

CALLED DIRECTLY BY:

YYPARSE - FLAN PARSER

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: CHKFRM
PURPOSE: CHECK FORM
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS

VOID CHKFRM()

DESCRIPTION

CHECKS THE CURRENT FORM FOR COMPLETENESS AND CONSISTENCY

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

WARNING - ISSUE WARNING MESSAGE
ADDCHK - ADD POSITION TO CHECK LIST
CHKARY - CHECK ARRAY
ABS
STRLEN
FREE
FLDTYP - FIELD TYPE
ERROR - ISSUE ERROR MESSAGE
GFLDPT - GET FIELD POINTER
ABS
MAX
STRASN
FNDATT - FIND ATTRIBUTE

CALLED DIRECTLY BY:

YYPARSE - FLAN PARSER

PS 620344501
30 September 1990

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: CHKGRP
PURPOSE: CHECK FOR GROUP SEPERATORS OR END OF FILE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

CHKGRP(FP)
FIELD *FP;

INPUTS/OUTPUTS:
NONE

INPUTS:
FP - FIELD POINTER

OUTPUTS:
NONE

DESCRIPTION

CHECKS IF THE DATA RECORD WHICH TARGETS TO THE FORM (FP)
HAS A
GROUP SEPERATOR OR END OF FILE. IF SO IT CLEARS THE
NODUP%D FIELDS
AND READS ANOTHER RECORD.

ARGUMENTS:

FP = FIELD *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPDINI - FPD INITIALIZATION
FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

HASLOWER - HAS A LOWER FORM WHICH READS THE SAME DATA
RECORD?
SPRINTF
GEN - GENERATE A LINE OF CODE
CLRNDP - CLEAR NODUPLICATE FIELDS
DBFREAD - GENERATE DATA BASE FREAD

CALLED DIRECTLY BY:

VISITA - VISIT ARRAYS ON THIS FORM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: CHKSIZE
PURPOSE: CHECK SIZE OF ITEMS DOING CONVERSIONS ON
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: CHKSIZE
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

```
CHKSIZE(DPTR, FPTR, DIR)
CDMDTYPE *DPTR;
FIELD *FPTR;
CHAR DIR;
```

DESCRIPTION

CHECK THE SIZE OF THE CDM DATA TYPE TO THE PRESENTATION
ITEM SIZE
ONLY PUT OUT A WARNING MESSAGE IF TRUNCATION WILL OCCUR
ON CONVERSION

ARGUMENTS:

```
DPTR = CDMDTYPE *
FPTR = FIELD *
DIR = CHAR
```

INCLUDE FILES:

```
STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
```

ROUTINES CALLED:

```
BLN
FPRINTF
```

CALLED DIRECTLY BY:

```
COBESPS - COBOL ES TO PS
COBPE - COBOL PE
```

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: CLOSEGAP
PURPOSE: CLOSE GAP IN TREE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: CLSGAP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
CLOSEGAP()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

THIS ROUTINE CUTS A SECTION OUT OF THE TREE. THE SECTION FROM FIRST_PTR TO LAST_PTR AND ALL OF THE RELATED CHILDREN ARE UNLINKED AND THE RESULTING HOLE IS SPLICED.

ARGUMENTS:

FIRST_PTR = NODE *
LAST_PTR = NODE *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

NEXTLEV - ADVANCE POINTERS TO NEXT LEVEL OF SUBTREE

CALLED DIRECTLY BY:

COPYNODE - COPY A NODE IN TREE

DELNODE - DELETE A SPECIFIED NODE IN TREE
MOVCLD - MOVE CHILDREN
REPOS - REPOSITION MODULE EXPANSIONS
SPLITNODE - SPLIT A NODE FOR PAGE BREAKS

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: CLRNDP
PURPOSE: CLEAR NODUPLICATE FIELDS
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

CLRNDP(SP)
SELECT *SP;

INPUTS/OUTPUTS:
NONE

INPUTS:
SP - SELECT POINTER

OUTPUTS:
NONE

DESCRIPTION

CLEARs ALL THE NODUP%D FIELDS WHICH THIS SELECT AND ALL
ITS CHILDREN
TARGET TO.

ARGUMENTS:

SP = SELECT *

INCLUDE FILES:

STDYTP	- STANDARD TYPE DEFINITIONS
STDIO	- **** PURPOSE NOT FOUND BY STRIPPER ****
FPD	- FORM PROCESSOR DATA
FPDINI	- FPD INITIALIZATION
FPPARM	- FORM PROCESSOR PARAMETERS
RW	- REPORT WRITER DEFINITIONS
NTM	- NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

SPRINTF
GEN - GENERATE A LINE OF CODE
CLRNDP - CLEAR NODUPLICATE FIELDS

CALLED DIRECTLY BY:

GENAQ - GENERATE ACTION QUERY (SELECT)
CHKGRP - CHECK FOR GROUP SEPERATORS OR END OF FILE
CLRNDP - CLEAR NODUPLICATE FIELDS

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: CLSFIL
PURPOSE: CLOSE FILES
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

SPTR = SELECT *

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

CLSFIL - CLOSE FILES
FPRINTF
INDENT - INDENT A LINE OF GENERATED CODE

CALLED DIRECTLY BY:

CLSFIL - CLOSE FILES
PROCEN - PROCEDURE DIVISION GENERATE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: COBCONV
PURPOSE: COBOL CONVERSIONS
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: MAKES
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

ES = ESDTYPE *
TBLSTR = CHAR *
SELNO = INT
REC_CNT = INT

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

MAKES/NUMPIC - NUMBER PICTURE CLAUSE
MAKES/CNUMPIC - C NUMBERS
ATOI
DASH - WRITE DASH '-'
ESCPY
FPRINTF
MAKES/INDENT - INDENT

CALLED DIRECTLY BY:

MAKES - MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: COBES
PURPOSE: COBOL ES RECORD
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: MAKES
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

ES = ESDTYPE *
SELNO = INT
TBLNUM = INT
REC_CNT = INT

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

MAKES/NUMPIC - NUMBER PICTURE CLAUSE
FPRINTF
MAKES/INDENT - INDENT
ATOI
DASH - WRITE DASH '-'
ESCPY

CALLED DIRECTLY BY:

MAKES - MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: COBESPS
PURPOSE: COBOL ES TO PS
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: ESPSMAP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

SELPTR = SELECT *

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

DASH - WRITE DASH '~'
GETTBL - GET A TABLE NAME
GETCOL - GET THE COLUMN NAME OF A TABLE.COLUMN OR COLUMN
STRING
CHKSIZE - CHECK SIZE OF ITEMS DOING CONVERSIONS ON
FPRINTF
ESPSMAP/INDENT - INDENT

CALLED DIRECTLY BY:

ESPSMAP - THE EXTERNAL SCHEMA AND PRESENTATION SCHEMA
MAPPING

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: COBPE
PURPOSE: COBOL PE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: PEMAP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

STR1 = CHAR *
STR2 = CHAR *
FPTR = FIELD *
DPTR = CMDMDTYPE *

INCLUDE FILES:

STDTYPE - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

FPRINTF
CHKSIZE - CHECK SIZE OF ITEMS DOING CONVERSIONS ON
INDENT - INDENT A LINE OF GENERATED CODE

CALLED DIRECTLY BY:

PEMAP - THE PRESENTATION SCHEMA AND THE EXTERNAL SCHEMA
AND MAPPING

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: COPYNODE
PURPOSE: COPY A NODE IN TREE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: COPNODE
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
COPYNODE()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

ARGUMENTS:

PAGE_PTR = NODE *
NODE_PTR = NODE *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

BLDMOD - BUILD MODULE
BLDNODE - BUILD NODE
CLOSEGAP - CLOSE GAP IN TREE

CALLED DIRECTLY BY:

PAGNODE - PAGE NODES

PS 620344501
30 September 1990

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: CPE
PURPOSE: C PE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: PEMAP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

STR1 = CHAR *
STR2 = CHAR *
FPTR = FIELD *
DPTR = CMDMDTYPE *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

CALLED DIRECTLY BY:

PEMAP - THE PRESENTATION SCHEMA AND THE EXTERNAL SCHEMA
AND MAPPING

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: CSTASH
PURPOSE: CHARACTER STASH
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

----- SYNOPSIS

CHAR *CSTASH(S)
CHAR *S;

DESCRIPTION

SAVES THE SPECIFIED CHARACTER STRING AND RETURNS A
POINTER TO IT

ARGUMENTS:

S = CHAR *

INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

STRCPY
STRLEN
MYALLOC - MY MALLOC

CALLED DIRECTLY BY:

YYLEX - LEXICAL ANALYZER FOR FLAN
YYPARSE - FLAN PARSER

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: CTLRSV
PURPOSE: CONTROL RESOLVE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

CTLRSV(CTLPTR)
CTLLST *CTLPTR;

INPUTS:

CTLPTR - CONTROL LIST FROM WHICH TO LOOK FOR PATHS.

DESCRIPTION

RESOLVES ALL QUALIFIED NAMES INTO FIELD POINTERS FOR ALL
NAMES
WHICH ARE ROOTED IN CTLLST'S (CONTROL LISTS).

ARGUMENTS:

CTLPTR = CTLLST *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

GETPTH - GET PATH
ERROR - ISSUE ERROR MESSAGE

CALLED DIRECTLY BY:

FLDRSV - FIELD RESOLVE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: DASH
PURPOSE: WRITE DASH '-'
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: MAKES
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

STR = CHAR []

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

STRCHR

CALLED DIRECTLY BY:

COBESPS - COBOL ES TO PS
COBES - COBOL ES RECORD
COBCONV - COBOL CONVERSIONS
MAKES - MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE
MAKWHEs/COBWHEs - COBOL WHERE ES
SELGEN - SELECT GENERATE
SELWS - SELECT WORKING STORAGE SECTION
INSWS - INSERT WORKING STORAGE SECTION
INSERT - INSERT PROCEDURE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: DATAGEN
PURPOSE: DATA DIVISION GENERATE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

LANG = INT
APNAME = CHAR *
TYPE = CHAR

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

NDMLLNK - LINKAGE SECTION
FILELNK - FILE LINKAGE SECTION GENERATE
INDENT - INDENT A LINE OF GENERATED CODE
FPRINTF
SELWS - SELECT WORKING STORAGE SECTION
INSWS - INSERT WORKING STORAGE SECTION
FD - FD SECTION DECLARATIONS
ASSIGN - ASSIGN FILE SECTION

CALLED DIRECTLY BY:

STDCODE - STANDARD COBOL CODE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: DBFREAD
PURPOSE: GENERATE DATA BASE FREAD
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

```
DBFREAD(SP, GENCHG)
  SELECT *SP;
  BOOL GENCHG;
```

INPUTS/OUTPUTS:
NONE

INPUTS:
SP - SELECT POINTER INDICATES DATA RECORD TO READ.
GENCHG - IF TRUE THEN ALSO GENERATE THE CHECK CHANGE
 CONDITION CODE.

OUTPUTS:
NONE

DESCRIPTION

GENERATES THE FREAD TO READ THE DATA RECORD ASSOCIATED
 WITH A SELECT.
SETS THE DBCODE TO INDICATE STATUS (TRUE INDICATES AN EOF
 OR GROUP
SEPERATOR WAS READ). IF GENCHG IS TRUE THEN ALSO GENERATE
 THE CODE
TO CHECK CHANGE CONDITIONS.

ARGUMENTS:

SP = SELECT *
GENCHG = BOOL

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPDINI - FPD INITIALIZATION

FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

SPRINTF
GEN - GENERATE A LINE OF CODE

CALLED DIRECTLY BY:

SELOPN - SELECT OPEN
READDB - READ DATA BASE
CHKGRP - CHECK FOR GROUP SEPERATORS OR END OF FILE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: DCLINDX
PURPOSE: DECLARE INDEX VARIABLES
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

DCLINDX(TP)
TRGLST *TP;

INPUTS:

TP - CONDITION WHICH REQUIRES INDEX VARIABLES.

DESCRIPTION

THIS PROCEDURE DECLARES THE INDEX VARIABLES USED IN
CONDITIONS AND
ACTIONS WHICH MAKE USE OF UNIVERSAL QUALIFICATION. THESE
DECLARATIONS
FOR CONDITIONS AND ACTIONS RESPECTIVELY ARE:

INT TINDX%D,...;
INT AINDX%D,...;

ARGUMENTS:

TP = TRGLST *

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

MAX
STRCPY
SPRINTF
STRLEN
STRCAT
GEN - GENERATE A LINE OF CODE

CALLED DIRECTLY BY:

GENDOA - GENERATE PROCEDURE "DOACT" DO ACTION

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: DELNODE
PURPOSE: DELETE A SPECIFIED NODE IN TREE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: DELNODE
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
DELNODE()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

DELETES THE SPECIFIED NODE, FIXES ITS NEIGHBORS'
POINTERS, AND DELETES
ITS MODULE IF IT WAS THE LAST REFERENCE TO IT.

ARGUMENTS:

NODE_PTR = NODE *

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

DELNODE - DELETE A SPECIFIED NODE IN TREE
CLOSEGAP - CLOSE GAP IN TREE
FREE

CALLED DIRECTLY BY:

DELNODE - DELETE A SPECIFIED NODE IN TREE
HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER
MOVCLD - MOVE CHILDREN
REPOS - REPOSITION MODULE EXPANSIONS

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: DOINDEX
PURPOSE: DO CHART INDEX
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: DOINDEX
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS

```
VOID DOINDEX(OUTCHART, PAGE_WIDTH, PAGE_DEPTH)
FILE OUTCHART;
INT PAGE_WIDTH;
INT PAGE_DEPTH;
```

INPUTS:

```
OUTCHART - OUTPUT FILE
PAGE_WIDTH - OUTPUT PAGE WIDTH
PAGE_DEPTH - OUTPUT PAGE DEPTH
```

DESCRIPTION

PRINTS THE INDEX FOR THE CHART. IF A MODULE HAS AN
EXPANSION, THE
PAGE NUMBER WHERE THE EXPANSION APPEARS FOLLOWS THE
MODULE NAME.

ARGUMENTS:

```
OUTCHART = FILE *
PAGE_WIDTH = INT
PAGE_DEPTH = INT
```

INCLUDE FILES:

```
STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
CHART - CHART INCLUDE FILE
```

ROUTINES CALLED:

```
SORT - SORT MODULE NAMES
MALLOC
STRCMP
PUTC
FREE
```

FPUTS
MEMSET
STRLEN
MEMCPY
SPRINTF
GETTOP - GET TOP OF TREE

CALLED DIRECTLY BY:

PRNTREE - PRINT TREE

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: DRAWLEV
PURPOSE: DRAW A LEVEL OF THE CHART
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: DRAWLEV
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
DRAWLEV()

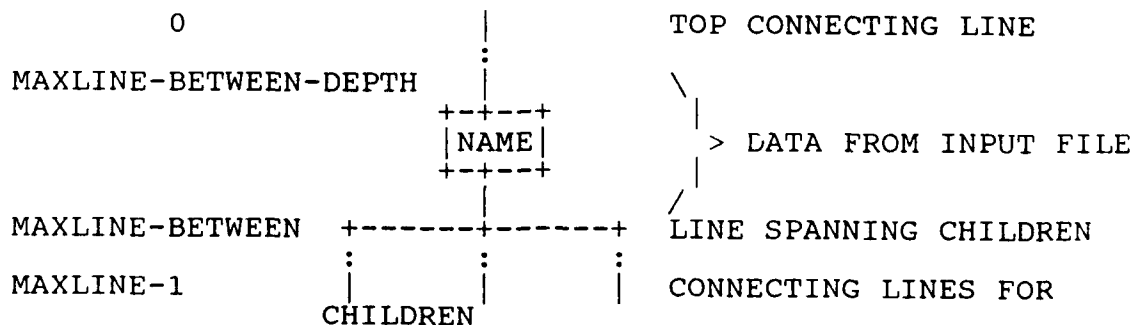
INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

THIS ROUTINE CONTROLS THE DRAWING OF A LEVEL OF THE CHART.
A LEVEL CONSISTS OF MAXLINE LINES:



ARGUMENTS:

TEMPFILE = FILE *
OUTCHART = FILE *
START_PTR = NODE *
CHARSET = INT
PAGE_WIDTH = INT

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

MALLOC
FSEEK
GETC
MEMCPY
FGETS
STRLEN
PUTLIN - PRINT LEVEL OF TREE
FREE
MEMSET
SPRINTF
GETTOP - GET TOP OF TREE

CALLED DIRECTLY BY:

PRNTREE - PRINT TREE

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: ENDGEN
PURPOSE: END GERNERATE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

GENERATE THE ENDING CODE WHICH CLOSES THE FILES AND DOES THE
NDML ERROR PROCESSING

ARGUMENTS:

LANG = INT

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

FPRINTF
INDENT - INDENT A LINE OF GENERATED CODE

CALLED DIRECTLY BY:

STDCODE - STANDARD COBOL CODE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: ERROR
PURPOSE: ISSUE ERROR MESSAGE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: FLUIERR
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

----- SYNOPSIS

```
VOID ERROR(S, A, B, C, D, E, F)
    CHAR *S, *A, *B, *C, *D, *E, *F;
```

DESCRIPTION

PRINTS AN ERROR MESSAGE ON STDERR AND INCREMENTS THE
NUMBER OF ERRORS

ARGUMENTS:

S = CHAR *
A = CHAR *
B = CHAR *
C = CHAR *
D = CHAR *
E = CHAR *
F = CHAR *

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS

ROUTINES CALLED:

PMSGLS
STRLEN
SPRINTF

CALLED DIRECTLY BY:

MKINC - MAKE INCLUDE (ACTUALLY STRUCTURE TAGS ALA
MAKINC)
GETFILE - RETURN A FILE POINTER BASED ON INPUT FROM THE
USER
MAKES - MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE
INSRSV - INSERT RESOLVE

SELRSV	- SELECT RESOLVE
CTLRSV	-- CONTROL RESOLVE
STATRSV	- STATISTIC RESOLVE
TRGRSV	- TRIGGER RESOLVE
ACTRSV	- ACTION RESOLVE
MLPFRM	- MAKE A LIST OF PRESENTED FORMS
CHKFLD	- CHECK FIELD
CHKFRM	- CHECK FORM
ADDCHK	- ADD POSITION TO CHECK LIST
YYLEX	- LEXICAL ANALYZER FOR FLAN
YYPARSE	- FLAN PARSER

USED IN MAIN PROGRAM(S) :

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: ESPSMAP
PURPOSE: THE EXTERNAL SCHEMA AND PRESENTATION
SCHEMA MAPPING
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: ESPSMAP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

```
ESPSMAP (LANG, SELPTR)  
INT LANG;  
SELECT *SELPTR;
```

DESCRIPTION

GENERATES THE CODE TO TRANSFORM AN EXTERNAL SCHEMA DATA
ITEM INTO
A PRESENTATION SCHEMA FORM ITEMS AND VICE VERSA.

ARGUMENTS:

```
-----  
LANG =          INT  
SELPTR =        SELECT *
```

INCLUDE FILES:

```
-----  
STDTyp - STANDARD TYPE DEFINITIONS  
STDIO  - **** PURPOSE NOT FOUND BY STRIPPER ****  
FPD    - FORM PROCESSOR DATA  
RW     - REPORT WRITER DEFINITIONS
```

ROUTINES CALLED:

```
-----  
CESPS - C ES TO PS  
COBESPS - COBOL ES TO PS
```

CALLED DIRECTLY BY:

```
-----  
SELMAP - MAP SELECTED DATA TO OUTPUT RECORD
```

PS 620344501
30 September 1990

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: ESPSMAP/INDENT
PURPOSE: INDENT
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: ESPSMAP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

M = INT
T = INT

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

PUTC

CALLED DIRECTLY BY:

COBESPS - COBOL ES TO PS

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: FATAL
PURPOSE: ISSUE FATAL ERROR MESSAGE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: FLUIERR
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS

```
VOID FATAL(S, A, B, C, D, E, F)
    CHAR *S, *A, *B, *C, *D, *E, *F;
```

DESCRIPTION

PRINTS A FATAL MESSAGE ON STDERR AND EXITS

ARGUMENTS:

```
S = CHAR *
A = CHAR *
B = CHAR *
C = CHAR *
D = CHAR *
E = CHAR *
F = CHAR *
```

INCLUDE FILES:

```
STDYTP - STANDARD TYPE DEFINITIONS
```

ROUTINES CALLED:

```
SPRINTF
STRLEN
PMSGLS
```

CALLED DIRECTLY BY:

```
MYALLOC - MY MALLOC
YYLEX - LEXICAL ANALYZER FOR FLAN
YYPARSE - FLAN PARSER
```

PS 620344501
30 September 1990

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: FD
PURPOSE: FD SECTION DECLARATIONS
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

SPTR = SELECT *

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

SELLEN - COMPUTE LENGTH OF SELECT PS RECORD
FPRINTF
INDENT - INDENT A LINE OF GENERATED CODE

CALLED DIRECTLY BY:

DATAGEN - DATA DIVISION GENERATE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: FILELNK
PURPOSE: FILE LINKAGE SECTION GENERATE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

SPTR = SELECT *

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

INDENT - INDENT A LINE OF GENERATED CODE
FPRINTF

CALLED DIRECTLY BY:

DATAGEN - DATA DIVISION GENERATE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: FLANCI
PURPOSE: FLAN CALLABLE INTERFACE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS

CHAR *FLANCI (FPTR)
FILE *FPTR;

INPUTS:

FPTR - FILE TO BE COMPILED

DESCRIPTION

COMPILES THE SPECIFIED FILE INTO THE LOCAL OPEN LIST.

ARGUMENTS:

FPTR = FILE *

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

YYPARSE - FLAN PARSER
DELFLD

CALLED DIRECTLY BY:

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: FLDRSV
PURPOSE: FIELD RESOLVE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS
FLDRSV(DP)
FIELD *DP;

INPUTS:
DP - FIELD FROM WHICH TO BEGIN SEARCH.

DESCRIPTION

RESOLVES ALL QUALIFIED NAMES INTO FIELD POINTERS FOR ALL
NAMES
WHICH ARE ROOTED IN FIELDS.

ARGUMENTS:

DP = FIELD *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

FLDRSV - FIELD RESOLVE
CTLRSV - CONTROL RESOLVE
STATRSV - STATISTIC RESOLVE

CALLED DIRECTLY BY:

RWOPN - REPORT WRITER OPEN FORMS
FLDRSV - FIELD RESOLVE

PS 620344501
30 September 1990

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: FLDTYP
PURPOSE: FIELD TYPE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: FLANSF
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

----- SYNOPSIS

CHAR *FLDTYP(C)
CHAR C;

DESCRIPTION

RETURNS A STRING OF THE SPECIFIED FIELD TYPE

ARGUMENTS:

C = CHAR

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

CALLED DIRECTLY BY:

CHKFRM - CHECK FORM
ADDCHK - ADD POSITION TO CHECK LIST

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: FNDATT
PURPOSE: FIND ATTRIBUTE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: ATTMAP * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

----- SYNOPSIS

ATTMAP *FNDATT(S)
CHAR *S;

DESCRIPTION

RETURNS A POINTER TO THE SPECIFIED ATTRIBUTE IN THE
ATTRIBUTE MAP

ARGUMENTS:

S = CHAR *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

STRCMP

CALLED DIRECTLY BY:

RWEXPD - REPORT WRITER EXPAND ARRAYS
RWSP/FIXFR - FIX UP A FORM
CHKFLD - CHECK FIELD
CHKFRM - CHECK FORM
YYPARSE - FLAN PARSER

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: FNDFRM
PURPOSE: FIND FORM
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: FIELD * ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

FIELD *FNDFRM(STR)
CHAR STR[];

INPUTS:

STR - NAME OF FORM TO FIND

DESCRIPTION

FINDS THE NAMED FORM ON THE OPNLST AND RETURNS A POINTER
TO IT.
RETURNS A NULL IF THE FORM CAN NOT BE FOUND.

ARGUMENTS:

STR = CHAR []

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

STRCMP

CALLED DIRECTLY BY:

GENAR - GENERATE ACTION PRESENT
MLPFRM - MAKE A LIST OF PRESENTED FORMS
WINRSV - WINDOW RESOLVE

PS 620344501
30 September 1990

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: FRMPDAT
PURPOSE: FORM PDATA
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

FRMPDAT(FDP)
FIELD *FDP;

INPUTS:

FDP - POINTER TO A FORM.

DESCRIPTION

GENERATES A PDATA FOR THE FORM POINTED TO BY FDP IF THERE
ARE ANY
ITEMS ON IT.

ARGUMENTS:

FDP = FIELD *

INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

HASITEM - THIS ROUTINE DETERMINES IF THERE IS AN ITEM
WITHIN
CALCSTAT - CALCULATE STATISTIC
SPRINTF
GEN - GENERATE A LINE OF CODE
RSETSTAT - RESET STATISTIC

CALLED DIRECTLY BY:

GENAP - GENERATE ACTION PAGE
GENAR - GENERATE ACTION PRESENT

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: FRNTND
PURPOSE: FRONT END FOR FORMS
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: RWFRNT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW

DESCRIPTION:

SYNOPSIS

CHAR *FRNTND()

INPUTS/OUTPUTS:
NONE

INPUTS:
NONE

OUTPUTS:
NONE

DESCRIPTION

THIS FUNCTION PRESENTS A TOP LEVEL FORM REQUESTING A
FILE NAME FROM
THE USER. IT RETURNS THAT FILE NAME TO GRP. THE NAME OF
THE FORM IS
"APFRONT.FDL" FOR THE APPLICATION GENERATOR AND
"RWFRONT.FDL" FOR THE
REPORT WRITER AND "FLFRONT.FDL" FOR FLAN. IT IS
HARDCODED INTO THE
ROUTINE. THERE IS ONE COPY OF THIS ROUTINE FOR THE AP
AND ONE FOR
THE RW AND ONE FOR FLAN.

ARGUMENTS:

FILNAM = CHAR [41]

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
FPPARM - FORM PROCESSOR PARAMETERS
NTM - NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

STRCHR
INITAL
MEMCMP
TRMNAT
PMSGLC
INITFP
ADDFRM
GDATA
OISCR
SPRINTF

CALLED DIRECTLY BY:

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GEN
PURPOSE: GENERATE A LINE OF CODE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

GEN (STRING)
CHAR STRING[];

INPUTS/OUTPUTS:
NONE

INPUTS:
STRING - THIS IS THE LINE OF CODE TO GENERATE

OUTPUTS:
NONE

DESCRIPTION

THIS ROUTINE WILL MOVE A LINE OF CODE TO THE OUTPUT FILE,
IT ALSO TAKES CARE OF BALANCING RIGHT AND LEFT BRACKETS
AS WELL AS ALIGNING # TYPE STATEMENTS.

ARGUMENTS:

STRING = CHAR []

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPDINI - FPD INITIALIZATION
FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

FPRINTF

CALLED DIRECTLY BY:

GENMAIN	- GENERATE MAIN PROGRAM
GENAAL	- GENERATE PROCEDURE "ADDAL" ADD ACTION LIST
GENAA	- GENERATE PROCEDURE "ADDACT" ADD AN ACTION
GENDOA	- GENERATE PROCEDURE "DOACT" DO ACTION
DCLINDEX	- DECLARE INDEX VARIABLES
GENAL	- GENERATE ACTION LIST
UQFOR	- UNIVERSAL QUALIFIER FOR LOOP
GENAP	- GENERATE ACTION PAGE
GENAR	- GENERATE ACTION PRESENT
GENAQ	- GENERATE ACTION QUERY (SELECT)
GENAS	- GENERATE ACTION SET
GENAE	- GENERATE ACTION EXIT
GENAH	- GENERATE ACTION HELP
GENAT	- GENERATE ACTION SIGNAL
GENAI	- GENERATE ACTION INSERT
SELWHR	- SELECT WHERE
SELOPN	- SELECT OPEN
FRMPDAT	- FORM PDATA
GENBEG	- GENERATE BEGINNING OF APPLICATION OR REPORT
MKINC	- MAKE INCLUDE (ACTUALLY STRUCTURE TAGS ALA MAKINC)
GENDB	- GENERATE DATA BASE RECORDS AND FILE DECLARATIONS
GENFS	- GENERATE FORM DATA STRUCTURES
GENDS	- GENERATE DATA DATA STRUCTURES
GENFSD	- GENERATE FORM STRUCTURE DATA INITIALIZATION
GENFP	- GENERATE FORM PATH
GENNDP	- GENERATE NODUPLICATE DECLARATIONS
GENCHG	- GENERATE CHANGE DECLARATIONS
GENINS	- GENERATE INSERT DECLARATIONS
BSCODE	- BUILD SUBROUTINE CODE
MAPDB	- MAP DATABASE
VISITA	- VISIT ARRAYS ON THIS FORM
CHKGRP	- CHECK FOR GROUP SEPERATORS OR END OF FILE
CLRNDP	- CLEAR NODUPLICATE FIELDS
GENPAG	- GENERATE NEWPAG PROCEDURE
DBFREAD	- GENERATE DATA BASE FREAD
SETNDP	- SET NODUPLICATE FIELDS TO BLANK IF THEY ARE DUPLICATED
RSETNDP	- RESET NODUPLICATE FIELDS TO VALUE OF NODUP%D
CALCSTAT	- CALCULATE STATISTIC
RSETSTAT	- RESET STATISTIC

USED IN MAIN PROGRAM(S):

GRP/MAIN	- GENERATE APPLICATION/REPORT PROGRAM
----------	---------------------------------------

REPORT WRITER Module Documentation

NAME: GENAA
PURPOSE: GENERATE PROCEDURE "ADDACT" ADD AN ACTION
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS
GENAA()

DESCRIPTION

THIS ROUTINE GENERATES A PROGRAM THAT WILL ADD AN ACTION TO THE ACTION LIST AT RUN TIME. THE PROGRAM THAT IS GENERATED BY THIS ROUTINE IS FIXED AND IS NOT CHANGED FOR ANY REPORT, IT IS ALWAYS THE SAME PROGRAM.

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

GEN - GENERATE A LINE OF CODE

CALLED DIRECTLY BY:

GENACT - GENERATE ACTIONS

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENAAL
PURPOSE: GENERATE PROCEDURE "ADDAL" ADD ACTION LIST
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS
GENAAL()

DESCRIPTION

THIS ROUTINE TRAVERSES THE TRIGGER DATA STRUCTURE FOR EACH
TRIGGER IT GENERATES A CALL TO ADD EACH ACTION.
PRIORITIES FOR ACTIONS ARE DETERMINED BY THIS ROUTINE.

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

GEN - GENERATE A LINE OF CODE
SPRINTF

CALLED DIRECTLY BY:

GENACT - GENERATE ACTIONS

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENACT
PURPOSE: GENERATE ACTIONS
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS
GENACT()

DESCRIPTION

THIS ROUTINE GENERATES THE CODE TO IMPLEMENT TRIGGERS AND ACTIONS. IT GENERATES CODE TO ADD A LIST OF ACTIONS TO ADD AN ACTION AND GENERATES THE CODE NECESSARY TO PERFORM AN ACTION.

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

GENAAL - GENERATE PROCEDURE "ADDAL" ADD ACTION LIST
GENAA - GENERATE PROCEDURE "ADDACT" ADD AN ACTION
GENDOA - GENERATE PROCEDURE "DOACT" DO ACTION

CALLED DIRECTLY BY:

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENAE
PURPOSE: GENERATE ACTION EXIT
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

GENAE(TP, AP)
TRGLST *TP;
ACTLST *AP;

INPUTS:

TP - CONDITION ASSOCIATED WITH THIS ACTION.
AP - THIS ACTION.

DESCRIPTION

GENERATES THE EXIT ACTION

ARGUMENTS:

TP = TRGLST *
AP = ACTLST *

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

GEN - GENERATE A LINE OF CODE

CALLED DIRECTLY BY:

GENAL - GENERATE ACTION LIST

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENAH
PURPOSE: GENERATE ACTION HELP
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

GENAH(TP, AP)
TRGLST *TP;
ACTLST *AP;

INPUTS:

TP - CONDITION ASSOCIATED WITH THIS ACTION.
AP - THIS ACTION.

DESCRIPTION

GENERATES THE HELP ACTION

ARGUMENTS:

TP = TRGLST *
AP = ACTLST *

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

SPRINTF
GEN - GENERATE A LINE OF CODE

CALLED DIRECTLY BY:

GENAL - GENERATE ACTION LIST

PS 620344501
30 September 1990

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENAI
PURPOSE: GENERATE ACTION INSERT
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

GENAI(TP, AP)
TRGLST *TP;
ACTLST *AP;

INPUTS:

TP - CONDITION ASSOCIATED WITH THIS ACTION.
AP - THIS ACTION.

DESCRIPTION

GENERATES THE INSERT ACTION

ARGUMENTS:

TP = TRGLST *
AP = ACTLST *

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

MAKQR - MAKE QUALIFIED REFERENCE
SPRINTF
GEN - GENERATE A LINE OF CODE

CALLED DIRECTLY BY:

GENAL - GENERATE ACTION LIST

PS 620344501
30 September 1990

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENAL
PURPOSE: GENERATE ACTION LIST
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

GENAL(TP, AP)
TRGLST *TP;
ACTLST *AP;

INPUTS:

TP - CONDITION TO WHICH THIS ACTION BELONGS.
AP - ACTION TO GENERATE CODE FOR.

DESCRIPTION

CALL THE PROCEDURE WHICH GENERATES THE CODE TO IMPLEMENT AN ACTION. ALSO CALLS PROCEDURE TO GENERATE FOR LOOPS FOR UNIVERSAL QUALIFICATION.

ARGUMENTS:

TP = TRGLST *
AP = ACTLST *

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

UQFOR - UNIVERSAL QUALIFIER FOR LOOP
GEN - GENERATE A LINE OF CODE
CFNAP - GENERATE ACTION PAGE
GENAR - GENERATE ACTION PRESENT
GENAQ - GENERATE ACTION QUERY (SELECT)
GENAS - GENERATE ACTION SET
GENAE - GENERATE ACTION EXIT

GENAH	- GENERATE ACTION HELP
GENAT	- GENERATE ACTION SIGNAL
GENAI	- GENERATE ACTION INSERT

CALLED DIRECTLY BY:

GENDOA - GENERATE PROCEDURE "DOACT" DO ACTION

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENAP
PURPOSE: GENERATE ACTION PAGE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

GENAP(TP, AP)
TRGLST *TP;
ACTLST *AP;

INPUTS:

TP - CONDITION ASSOCIATED WITH THIS ACTION.
AP - THIS ACTION.

DESCRIPTION

GENERATES THE PAGE ACTION

ARGUMENTS:

TP = TRGLST *
AP = ACTLST *

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

GEN - GENERATE A LINE OF CODE
FRMPDAT - FORM PDATA
SPRINTF

CALLED DIRECTLY BY:

GENAL - GENERATE ACTION LIST

PS 620344501
30 September 1990

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENAQ
PURPOSE: GENERATE ACTION QUERY (SELECT)
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

GENAQ(TP, AP)
TRGLST *TP;
ACTLST *AP;

INPUTS:

TP - CONDITION ASSOCIATED WITH THIS ACTION.
AP - THIS ACTION.

DESCRIPTION

GENERATES THE SELECT ACTION

ARGUMENTS:

TP = TRGLST *
AP = ACTLST *

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

SELWHR - SELECT WHERE
SPRINTF
GEN - GENERATE A LINE OF CODE
CLRNDP - CLEAR NODUPLICATE FIELDS
SELOPN - SELECT OPEN

CALLED DIRECTLY BY:

GENAL - GENERATE ACTION LIST

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENAR
PURPOSE: GENERATE ACTION PRESENT
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

GENAR(TP, AP)
TRGLST *TP;
ACTLST *AP;

INPUTS:

TP - CONDITION ASSOCIATED WITH THIS ACTION.
AP - THIS ACTION.

DESCRIPTION

GENERATES THE PRESENT ACTION

ARGUMENTS:

TP = TRGLST *
AP = ACTLST *

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

FNDFRM - FIND FORM
GEN - GENERATE A LINE OF CODE
ISOPNE - DETERMINE IF THIS FIELD IS OPEN ENDED
HASDATA - DETERMINE IF THERE ARE ANY SELECT STATEMENTS
HASITEM - THIS ROUTINE DETERMINES IF THERE IS AN ITEM
WITHIN
STRCMP
FRMPDAT - FORM PDATA
SPRINTF

CALLED DIRECTLY BY:

GENAL - GENERATE ACTION LIST

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENAS
PURPOSE: GENERATE ACTION SET
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

```
GENAS(TP, AP)
  TRGLST *TP;
  ACTLST *AP;
```

INPUTS:

TP - CONDITION ASSOCIATED WITH THIS ACTION.
AP - THIS ACTION.

DESCRIPTION

GENERATES THE SET ACTION

ARGUMENTS:

```
TP = TRGLST *
AP = ACTLST *
```

INCLUDE FILES:

```
STDTP - STANDARD TYPE DEFINITIONS
FPD   - FORM PROCESSOR DATA
RW    - REPORT WRITER DEFINITIONS
```

ROUTINES CALLED:

```
ISOPNE - DETERMINE IF THIS FIELD IS OPEN ENDED
MAKQR  - MAKE QUALIFIED REFERENCE
STRCPY
SPRINTF
GEN     - GENERATE A LINE OF CODE
STRSPN
STRLEN
```

CALLED DIRECTLY BY:

GENAL - GENERATE ACTION LIST

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENAT
PURPOSE: GENERATE ACTION SIGNAL
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

GENAT(TP, AP)
TRGLST *TP;
ACTLST *AP;

INPUTS:

TP - CONDITION ASSOCIATED WITH THIS ACTION.
AP - THIS ACTION.

DESCRIPTION

GENERATES THE SIGNAL ACTION

ARGUMENTS:

TP = TRGLST *
AP = ACTLST *

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

WARNING - ISSUE WARNING MESSAGE
SPRINTF
GEN - GENERATE A LINE OF CODE

CALLED DIRECTLY BY:

GENAL - GENERATE ACTION LIST

PS 620344501
30 September 1990

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENBEG
PURPOSE: GENERATE BEGINNING OF APPLICATION OR
REPORT
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENMN2
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

GENBEG(NAME)
CHAR NAME[];

INPUTS:

NAME - NAME OF THE APPLICATION OR REPORT

OUTPUTS:

NONE

DESCRIPTION

THIS ROUTINE GENERATES THE PROLOG FOR AN APPLICATION OR A
REPORT.

IT CONSISTS OF THE #INCLUDE'S, THE ACTION STRUCTURE AND
POINTERS,
AND DECLARATIONS FOR SEVERAL OTHER FIXED SIZE VARIABLES.

ARGUMENTS:

NAME = CHAR []

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

GEN - GENERATE A LINE OF CODE
PRINTF

CALLED DIRECTLY BY:

GENMAIN - GENERATE MAIN PROGRAM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENCHG
PURPOSE: GENERATE CHANGE DECLARATIONS
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENMN2
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS
GENCHG()

INPUTS/OUTPUTS:
NONE

INPUTS:
NONE

OUTPUTS:
NONE

DESCRIPTION

THIS ROUTINE GENERATES THE DECLARATION TO HOLD THE LAST
VALUE OF
AN ITEM WHICH HAS A CHANGE CONDITION ON IT. THE FORM OF THE
DECLARATION IS:

CHAR CHG%D[SIZE]; %D - NUMBER OF FIELD, SIZE OF
FIELD.

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

SPRINTF
GEN - GENERATE A LINE OF CODE

CALLED DIRECTLY BY:

GENMAIN - GENERATE MAIN PROGRAM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENDB
PURPOSE: GENERATE DATA BASE RECORDS AND FILE
DECLARATIONS
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENMN2
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

GENDB(COUNT)
INT COUNT;

INPUTS:

COUNT - THE NUMBER OF SELECTS IN THIS LIST

OUTPUTS:

NONE

DESCRIPTION

GENERATES DECLARATIONS FOR SELECT FILES: FILE POINTERS,
NAMES AND
STATUS CODES.

ARGUMENTS:

COUNT = INT

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

SPRINTF
GEN - GENERATE A LINE OF CODE

CALLED DIRECTLY BY:

GENMAIN - GENERATE MAIN PROGRAM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENDOA
PURPOSE: GENERATE PROCEDURE "DOACT" DO ACTION
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS
GENDOA()

DESCRIPTION

THIS ROUTINE GENERATES THE CODE NECESSARY TO PERFORM AN
ACTION AT RUN TIME. IT GENERATES CODE FOR EACH ACTION FOR
EACH
TRIGGER.

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

GEN - GENERATE A LINE OF CODE
DCLINDX - DECLARE INDEX VARIABLES
GENAL - GENERATE ACTION LIST
STRCHR
SPRINTF

CALLED DIRECTLY BY:

GENACT - GENERATE ACTIONS

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENDS
PURPOSE: GENERATE DATA DATA STRUCTURES
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENMN2
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS
GENDS()

INPUTS/OUTPUTS:
NONE

INPUTS:
NONE

OUTPUTS:
NONE

DESCRIPTION

THIS ROUTINE GENERATES A DATA STRUCTURE FOR EACH SELECT STATEMENT. THESE ARE OF THE FORM:

```
STRUCT
{
    CHAR DBNAME[20];           FIELDS TO GET DATA.
    CHAR DBID[4];
    CHAR HOSTID[3];
    CHAR DBMSNAME[30];
    CHAR CR;                   CARRIAGE RETURN PAD.
    } DBR%D;                   %D - NUMBER OF SELECT (0 IS
                                FIRST).
```

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

GEN - GENERATE A LINE OF CODE
SPRINTF

CALLED DIRECTLY BY:

GENMAIN - GENERATE MAIN PROGRAM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENFP
PURPOSE: GENERATE FORM PATH
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENMN2
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS
GENFP()

INPUTS/OUTPUTS:
NONE

INPUTS:
NONE

OUTPUTS:
NONE

DESCRIPTION

THIS ROUTINE GENERATES THE DECLARATION TO HOLD A PATH NAME FOR OPEN ENDED FORMS AND THE TOP FORM(S). THE FORM OF THE DECLARATIONS IS:

CHAR PATH%D[120] = "FORMNAME"; %D IS THE NUMBER OF THE
FORM AND FORMNAME IS THE NAME OF THE
FORM.

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

SPRINTF
GEN - GENERATE A LINE OF CODE

CALLED DIRECTLY BY:

GENMAIN - GENERATE MAIN PROGRAM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENFS
PURPOSE: GENERATE FORM DATA STRUCTURES
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENMN2
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

GENFS(DP)
FIELD *DP;

INPUTS/OUTPUTS:
NONE

INPUTS:
(DP) - FIELD POINTER

OUTPUTS:
NONE

DESCRIPTION

THIS ROUTINE USES THE STRUCTURE TAGS TO ALLOCATE SPACE FOR FORM DATA FOR CURRENT AND PREVIOUS IT GENERATES THIS CODE FOR OPEN ENDED FORMS AND FOR OPEN ENDED ARRAYS. THESE ARE DECLARED AS FOLLOWS:

STRUCT FRM%D FRM%DC, FRM%DP; %D - NUMBER OF THE FORM.

ARGUMENTS:

DP = FIELD *

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

ISOPNE - DETERMINE IF THIS FIELD IS OPEN ENDED
HASITEM - THIS ROUTINE DETERMINES IF THERE IS AN ITEM

WITHIN

SPRINTF
GEN - GENERATE A LINE OF CODE
GENFS - GENERATE FORM DATA STRUCTURES

CALLED DIRECTLY BY:

GENMAIN - GENERATE MAIN PROGRAM
GENFS - GENERATE FORM DATA STRUCTURES

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENFSD
PURPOSE: GENERATE FORM STRUCTURE DATA
INITIALIZATION
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENMN2
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

GENFSD(DP)
FIELD *DP;

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

THIS ROUTINE INITIALIZES THE FORM CURRENT AND PREVIOUS
BUFFERS TO BLANK FOR BOTH OPEN ENDED FORMS AND OPEN
ENDED ITEMS. THESE ARE OF THE FORM:

MEMSET(&FRM%DC, ' ', SIZEOF FRM%DC); %D IS THE NUMBER OF
THE FORM.

ARGUMENTS:

DP = FIELD *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

ISOPNE - DETERMINE IF THIS FIELD IS OPEN ENDED

HASITEM - THIS ROUTINE DETERMINES IF THERE IS AN ITEM
 WITHIN
SPRINTF
STRLEN
GEN - GENERATE A LINE OF CODE
GENFSD - GENERATE FORM STRUCTURE DATA INITJALIZATION
MAKQR - MAKE QUALIFIED REFERENCE

CALLED DIRECTLY BY:

GENMAIN - GENERATE MAIN PROGRAM
GENFSD - GENERATE FORM STRUCTURE DATA INITIALIZATION

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENINS
PURPOSE: GENERATE INSERT DECLARATIONS
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENMN2
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS GENINS()

INPUTS/OUTPUTS:
NONE

INPUTS:
NONE

OUTPUTS:
NONE

DESCRIPTION

THIS ROUTINE GENERATES THE DECLARATIONS FOR THE NDML
INSERT ACTION.
THE FORM OF THIS DECLARATION IS:

```
STRUCT
{
  STRUCT
  {
    CHAR DBID[4];
    CHAR HOSTID[3];
  } INSERT%D;          %D - NUMBER OF INSERT (0
                        IS FIRST).
  STRUCT
  {
    CHAR DBID[4];          FIELDS OF FORM TO BE
    INSERTED.
    CHAR HOSTID[3];
  } INSERT1;            ONE PRESENTATION SCHEME
    RECORD.
  CHAR DUMMY;           DUMMY FIELD IF THERE ARE
    NO INSERTS.
} INSERTPS;            NAME OF INSERT STRUCTURE .
```

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

GEN - GENERATE A LINE OF CODE
SPRINTF

CALLED DIRECTLY BY:

GENMAIN - GENERATE MAIN PROGRAM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENMAIN
PURPOSE: GENERATE MAIN PROGRAM
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENMAIN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW

DESCRIPTION: -----

SYNOPSIS GENMAIN()

INPUTS/OUTPUTS:
NONE

INPUTS:
NONE

OUTPUTS:
NONE

DESCRIPTION

THIS ROUTINE GENERATES THE FOLLOWING:

1. INCLUDE STATEMENTS
2. MAKINC IS USED TO GENERATE FORM DEFINITION
FUNCTION TAGS
3. GENCS IS CALLED TO INITIALIZE CONDITION FLAGS
4. GENFS IS USED TO GENERATE CURRENT AND PREVIOUS
BUFFERS FOR FORMS
5. GENFP IS USED TO GENERATE PATH DECLARATIONS
PER FORM
6. GENDS IS USED TO GENERATE DATA STRUCTURES FOR
EACH SELECT STATEMENT
7. TYPEDEF AND ACTION LIST POINTERS ARE GENERATED
8. THE FILE POINTERS, A DATABASE CODE AND FILE NAME
DATA STRUCTURES ARE GENERATED FOR EACH SELECT
STATEMENT
9. THE GLOBAL VARIABLE I IS DECLARED MAIN AND
DECLARATIONS FOR VARIABLES USED IN MAIN ARE
DECLARED
10. CURRENT FORM BUFFERS ARE INITIALIZED TO BLANKS
11. THE CALL TO INITFP
14. THE STARTUP CONDITION'S ACTIONS ARE ADDED TO THE
ACTION LIST.

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

GENBEG - GENERATE BEGINNING OF APPLICATION OR REPORT
MKINC - MAKE INCLUDE (ACTUALLY STRUCTURE TAGS ALA
 MAKINC)
GEN - GENERATE A LINE OF CODE
GENFSD - GENERATE FORM STRUCTURE DATA INITIALIZATION
SPRINTF
GENFS - GENERATE FORM DATA STRUCTURES
GENFP - GENERATE FORM PATH
GENNDP - GENERATE NODUPLICATE DECLARATIONS
GENCHG - GENERATE CHANGE DECLARATIONS
GENDS - GENERATE DATA DATA STRUCTURES
GENDB - GENERATE DATA BASE RECORDS AND FILE DECLARATIONS
GENINS - GENERATE INSERT DECLARATIONS

CALLED DIRECTLY BY:

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENNDP
PURPOSE: GENERATE NODUPLICATE DECLARATIONS
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENMN2
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS
GENNDP()

INPUTS/OUTPUTS:
NONE

INPUTS:
NONE

OUTPUTS:
NONE

DESCRIPTION

GENERATES DECLARATIONS FOR THE NODUP OPTION ON ITEMS. THE
DECLARATIONS
ARE OF THE FORM:

CHAR NODUP%D[SIZE]; %D - IS THE NUMBER OF THE FIELD,
 SIZE OF FIELD.

INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

SPRINTF
GEN - GENERATE A LINE OF CODE

CALLED DIRECTLY BY:

GENMAIN - GENERATE MAIN PROGRAM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GENPAG
PURPOSE: GENERATE NEWPAG PROCEDURE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS
GENPAG()

INPUTS/OUTPUTS:
NONE

INPUTS:
NONE

OUTPUTS:
NONE

DESCRIPTION

GENERATES THE PROCEDURE NEWPAG WHICH INCREMENTS THE FIELD
' . PAGENO; '
AND THEN DOES AN OUTSCR.

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPDINI - FPD INITIALIZATION
FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

GEN - GENERATE A LINE OF CODE

CALLED DIRECTLY BY:

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

PS 620344501
30 September 1990

USED IN MAIN PROGRAM(S) :

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GETCOL
PURPOSE: GET THE COLUMN NAME OF A TABLE.COLUMN OR
COLUMN STRING
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

OUTSTR = CHAR []
COLNAM = CHAR []

INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

STRCHR
STRCPY

CALLED DIRECTLY BY:

COBESPS - COBOL ES TO PS
MAKWHEs/COBWHEs - COBOL WHERE ES
SELGEN - SELECT GENERATE
SELWS - SELECT WORKING STORAGE SECTION
INSERT - INSERT PROCEDURE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GETFILE
PURPOSE: RETURN A FILE POINTER BASED ON INPUT FROM
THE USER
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: FILE * ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

```
FILE *GETFILE(NAMPTR)
CHAR *NAMPTR;
```

INPUTS/OUTPUTS:
NONE

INPUTS:
NAMPTR - STRING WITH NAME OF FILE.

OUTPUTS:
FILE POINTER IS RETURNED THROUGH THE FUNCTION REFERENCE

DESCRIPTION

GETFILE OPENS THE FILE NAMED BY THE INPUT PARAMETER. IF
THE USER
DOES NOT SPECIFY THE .FDL SUFFIX IT IS
AUTOMATICALLY APPENDED. THE FILE IS THEN OPENED.

ARGUMENTS:

NAMPTR = CHAR *

INCLUDE FILES:

```
STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPDINI - FPD INITIALIZATION
FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
```

ROUTINES CALLED:

ERROR - ISSUE ERROR MESSAGE
FOPEN
SPRINTF

CALLED DIRECTLY BY:

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GETFIT
PURPOSE: GET SUBTREE THAT FITS ON PAGE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: NODE * ()
SOURCE FILE: GETFIT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
GETFIT()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

THIS ROUTINE RETURNS A POINTER TO THE LARGEST SUBTREE (UP
TO THE
ENTIRE TREE) THAT WILL FIT ON A PAGE.

ARGUMENTS:

NODE_PTR = NODE *
PAGE_WIDTH = INT
PAGE_DEPTH = INT

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

GETSIZE - GET SUBTREE SIZE
GETFIT - GET SUBTREE THAT FITS ON PAGE

CALLED DIRECTLY BY:

GETFIT - GET SUBTREE THAT FITS ON PAGE
PAGTREE - PAGE TREE

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: GETLOWLEF
PURPOSE: GET LOWER LEFT CHILD NODE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: NODE * ()
SOURCE FILE: GETLWLF
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS

GETLOWLEF()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

GIVEN A NODE, GET THE FIRST NODE ON THE NEXT LOWER LEVEL
THAT IS A CHILD OF THIS NODE OR A CHILD OF SOME NODE RIGHT
OF THIS NODE

ARGUMENTS:

START_PTR = NODE *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

CALLED DIRECTLY BY:

ARRANGE - ARRANGE CHART AND ASSIGNS PAGE NUMBERS
HBALANC - HORIZONTAL TREE BALANCE
MOVECLD - MOVE CHILD'S POSITION
PAGNODE - PAGE NODES
PRNTREE - PRINT TREE
READTREE - READ DUMPTREE FILE
SPLICE - SPLICE TREE INTO ANOTHER TREE

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: GETLOWRIT
PURPOSE: GET LOWER RIGHT CHILD NODE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: NODE * ()
SOURCE FILE: GETLWRT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
GETLOWRIT()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

GIVEN A NODE, GET THE RIGHT-MOST NODE ON THE NEXT LOWER
LEVEL
THAT IS A CHILD OF THIS NODE OR A CHILD OF SOME NODE TO
THE LEFT
OF THIS NODE

ARGUMENTS:

START_PTR = NODE *

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

CALLED DIRECTLY BY:

READTREE - READ DUMPTREE FILE
SPLICE - SPLICE TREE INTO ANOTHER TREE

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: GETPAR
PURPOSE: GET PARENT NODE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: NODE * ()
SOURCE FILE: GETPAR
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
GETPAR()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

THIS ROUTINE RETURNS THE SPECIFIED PARENT NODE WHICH IS
CREATED IF
REQUIRED.

ARGUMENTS:

PARENT_NAME = CHAR []
FILEPOS = LONG
WIDTH = INT
DEPTH = INT
TOP_POS = INT
BOT_POS = INT
L_MARGIN = INT

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

BLDMOD - BUILD MODULE
BLDNODE - BUILD NODE

CALLED DIRECTLY BY:

READTREE - READ DUMPTREE FILE

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: GETPTH
PURPOSE: GET PATH
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

```
CHAR *GETPTH(PATH, DPP, FLDLST)
CHAR PATH[];
FIELD **DPP, *FLDLST;
```

INPUTS:

PATH - PATH TO BE RESOLVED INTO A POINTER.
FLDLST - FIELD HIERARCHY TO SEARCH FOR A PATH.

OUTPUTS:

DPP - POINTER TO POINTER TO FIELD INDICATED BY PATH.

DESCRIPTION

RESOLVES A QUALIFIED NAME INTO A FIELD POINTER. REPEATEDLY
CALLS
PTHPTR WITH FORMS IN THE TOPLST (SEE MLPFRM()).

ARGUMENTS:

```
PATH = CHAR []
DPP = FIELD **
FLDLST = FIELD *
```

INCLUDE FILES:

```
STDYTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
```

ROUTINES CALLED:

```
PTHPTR
STRCPY
STRUPC
STRCHR
```

CALLED DIRECTLY BY:

INSRSV - INSERT RESOLVE
SELRSV - SELECT RESOLVE
CTLRSV - CONTROL RESOLVE
STATRSV - STATISTIC RESOLVE
TRGRSV - TRIGGER RESOLVE
ACTRSV - ACTION RESOLVE
WINRSV - WINDOW RESOLVE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GETSIZE
PURPOSE: GET SUBTREE SIZE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: GETSIZE
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
HEADER()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

THIS ROUTINE RETURNS THE WIDTH AND DEPTH OF A SUB-TREE

ARGUMENTS:

FIRST_PTR = NODE *
WIDTH = INT *
DEPTH = INT *

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

NEXTLEV - ADVANCE POINTERS TO NEXT LEVEL OF SUBTREE
MIN
MAX

CALLED DIRECTLY BY:

GETFIT - GET SUBTREE THAT FITS ON PAGE
PAGTREE - PAGE TREE
PRNTREE - PRINT TREE

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: GETTBL
PURPOSE: GET A TABLE NAME
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

OUTSTR = CHAR []
TNUM = INT *
COLNAM = CHAR []
SELPTR = SELECT *

INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

STRCHR
ESCPY
NULBLK - BLANK FILL A STRING
STRCMP
STRCPY

CALLED DIRECTLY BY:

COBESPS - COBOL ES TO PS
MAKWHES/COBWHES - COBOL WHERE ES
SELGEN - SELECT GENERATE
SELWS - SELECT WORKING STORAGE SECTION

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GETTOP
PURPOSE: GET TOP OF TREE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: NODE * ()
SOURCE FILE: GETTOP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
GETTOP()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

ARGUMENTS:

START_PTR = NODE *

INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

CALLED DIRECTLY BY:

ARRANGE - ARRANGE CHART AND ASSIGNS PAGE NUMBERS
DOINDEX - DO CHART INDEX
DRAWLEV - DRAW A LEVEL OF THE CHART

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: GETUPLFT
PURPOSE: GET UPPER LEFTMOST NODE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: NODE * ()
SOURCE FILE: GETUPLF
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
GETUPLFT()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION
GIVEN A NODE, GET THE NODE ON THE NEXT HIGHER LEVEL
FARTHEST TO THE LEFT

ARGUMENTS:

START_PTR = NODE *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

CALLED DIRECTLY BY:

HBALANC - HORIZONTAL TREE BALANCE
PAGNODE - PAGE NODES

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: GFLDPT
PURPOSE: GET FIELD POINTER
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: FIELD * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS

FIELD *GFLDPT(FLDPTR, S)
FIELD *FLDPTR;
CHAR *S;

DESCRIPTION

RETURN A POINTER TO THE NAMED FIELD ON THE SPECIFIED FORM.

ARGUMENTS:

FLDPTR = FIELD *
S = CHAR *

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

STRCMP

CALLED DIRECTLY BY:

CHKFRM - CHECK FORM
YYPARSE - FLAN PARSER

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: GRP/MAIN
PURPOSE: GENERATE APPLICATION/REPORT PROGRAM
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

MAIN() ! THE EXECUTABLE IS NAMED "GRP" OR "GAP".

INPUTS/OUTPUTS:
NONE

INPUTS:
NONE.

OUTPUTS:
NONE

DESCRIPTION

THIS IS THE MAIN ROUTINE FOR THE APPLICATION/REPORT
GENERATING PROGRAM.
IT PROMPTS THE USER FOR HIS .FDL DEFINITION FILE, CALLS
FLAN TO
PARSE THE APPLICATION OR REPORT DEFINITION, WRITES OUT THE
FD FILES,
GENERATES THE SPECIFIC DATA STRUCTURES, AND ESTABLISHES THE
HIERARCHICAL RELATIONSHIP BETWEEN THE SELECT STATEMENTS
AND THE
FORM HIERARCHY. IT THEN GENERATES THE C CODE IN THE
FOLLOWING STEPS:

1. GENERATES THE MAIN PROGRAM
2. GENERATES THE CODE FOR EACH SUB-ROUTINE WHERE THESE
SUB-ROUTINES
CORRESPOND TO FORMS IN THE HIERARCHY
3. GENERATES THE CODE TO PROCESS ON CONDITIONS AND ACTIONS
4. GENERATES THE COBOL CODE TO PROCESS THE SELECT
STATEMENTS

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****

FPD	- FORM PROCESSOR DATA
FPDINI	- FPD INITIALIZATION
FPPARM	- FORM PROCESSOR PARAMETERS
RW	- REPORT WRITER DEFINITIONS
NTM	- NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

FRNTND	- FRONT END FOR FORMS
GETFILE	- RETURN A FILE POINTER BASED ON INPUT FROM THE USER
CALLOC	
FLANCI	- FLAN CALLABLE INTERFACE
FOPEN	
STRCAT	
STRCPY	
WRTFRM	- WRITE FORM
RWOPN	- REPORT WRITER OPEN FORMS
GENMAIN	- GENERATE MAIN PROGRAM
BLDSUB	- BUILD SUBROUTINES
GENACT	- GENERATE ACTIONS
GENPAG	- GENERATE NEWPAG PROCEDURE
NDMLGEN	- NDML COBOL APPLICATION GENERATOR
PMSGLC	
OISCR	
TERMFP	
TRMNDML	
STRCHR	

REPORT WRITER Module Documentation

NAME: HASDATA
PURPOSE: DETERMINE IF THERE ARE ANY SELECT STATEMENTS
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

THAT TARGET TO THE SCOPE OF THIS FORM.

SYNOPSIS

HASDATA(DP)
FIELD *DP;

INPUTS/OUTPUTS:
NONE

INPUTS:
(DP) - FIELD POINTER

OUTPUTS:
HASDATA RETURNS A TRUE OR A FALSE VALUE DEPENDING ON
WHETHER
ANY DATA WERE FOUND.

DESCRIPTION

THIS ROUTINE TRAVERSES THE FORM PROCESSOR DATA HIERARCHY TO
DETERMINE IF ANY SELECT STATEMENT TARGETS TO AN ITEM
WITHIN THE
SCOPE INDICATED BY THE FIELD POINTER WHICH IS PASSED IN AS
AN
INPUT PARAMETER. THE SCOPE IS DETERMINED BY NOT PROCESSING
PAST WINDOWS.

ARGUMENTS:

DP = FIELD *

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPDINI - FPD INITIALIZATION

FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

HASDATA - DETERMINE IF THERE ARE ANY SELECT STATEMENTS

CALLED DIRECTLY BY:

GENAR - GENERATE ACTION PRESENT
BLDSUB - BUILD SUBROUTINES
HASDATA - DETERMINE IF THERE ARE ANY SELECT STATEMENTS
VISITA - VISIT ARRAYS ON THIS FORM
SETNDP - SET NODUPLICATE FIELDS TO BLANK IF THEY ARE
DUPLICATED
RSETNDP - RESET NODUPLICATE FIELDS TO VALUE OF NODUP%D

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: HASITEM
PURPOSE: THIS ROUTINE DETERMINES IF THERE IS AN
ITEM WITHIN
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

THE SCOPE OF REFERENCE.

SYNOPSIS

HASITEM(DP)
FIELD *DP;

INPUTS/OUTPUTS:
NONE

INPUTS:
(DP) - FIELD POINTER

OUTPUTS:
RETURNS TRUE IF AN ITEM IS WITHIN THE SCOPE OF REFERENCE.

DESCRIPTION

THIS ROUTINE TRAVERSES THE FORMS HIERARCHY LOOKING FOR
ITEMS.

THE SCOPE OF REFERENCE IS DETERMINED BY NOT TRAVERSING PAST
OPEN ENDED ARRAYS OR WINDOWS. THE ROUTINE STOPS WHEN AN
ITEM
IS FOUND.

ARGUMENTS:

DP = FIELD *

INCLUDE FILES:

STDTP	- STANDARD TYPE DEFINITIONS
STDIO	- **** PURPOSE NOT FOUND BY STRIPPER ****
FPD	- FORM PROCESSOR DATA
FPDINI	- FPD INITIALIZATION
FPPARM	- FORM PROCESSOR PARAMETERS
RW	- REPORT WRITER DEFINITIONS
NTM	- NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

HASITEM - THIS ROUTINE DETERMINES IF THERE IS AN ITEM
 WITHIN

CALLED DIRECTLY BY:

GENAR - GENERATE ACTION PRESENT
FRMPDAT - FORM PDATA
MKINC - MAKE INCLUDE (ACTUALLY STRUCTURE TAGS ALA
 MAKINC)
GENFS - GENERATE FORM DATA STRUCTURES
GENFSD - GENERATE FORM STRUCTURE DATA INITIALIZATION
BSCODE - BUILD SUBROUTINE CODE
VISITA - VISIT ARRAYS ON THIS FORM
HASITEM - THIS ROUTINE DETERMINES IF THERE IS AN ITEM
 WITHIN

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: HASLOWER
PURPOSE: HAS A LOWER FORM WHICH READS THE SAME
DATA RECORD?
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

HASLOWER(FP, VP)
FIELD *FP;
VARLST *VP;

INPUTS/OUTPUTS:
NONE

INPUTS:
FP - FIELD POINTER
VP - VARIABLE LIST FROM A SELECT

OUTPUTS:
NONE

DESCRIPTION

CHECKS THE FORM FP TO SEE IF ANY OF THE VARIABLES IN THE
SELECT LIST
TARGET TO A FORM WHICH IS LOWER IN THE HIERARCHY THAN FP.
USED BY
CHKGRP AND READDB.

ARGUMENTS:

FP = FIELD *
VP = VARLST *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPDINI - FPD INITIALIZATION
FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE

CALLED DIRECTLY BY:

READDB - READ DATA BASE
CHKGRP - CHECK FOR GROUP SEPERATORS OR END OF FILE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: HBALANC
PURPOSE: HORIZONTAL TREE BALANCE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: HBALANC
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
HBALANC()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

THIS ROUTINE BALANCES THE TREE. STARTING AT THE BOTTOM
LEFT OF THE
TREE AND MOVING FIRST RIGHT AND THEN UPWARD, EACH NODE IS
POSITIONED
TO THE RIGHT OF ITS NEIGHBOR. IF THE CENTER OF ITS
CHILDREN IS FURTHER
RIGHT, THE NODE IS MOVED RIGHT, OTHERWISE THE CHILDREN
ARE MOVED RIGHT.

ARGUMENTS:

FIRST_PTR = NODE *

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

GETLOWLEF - GET LOWER LEFT CHILD NODE

MOVECLD - MOVE CHILD'S POSITION
MAX
GETUPLFT - GET UPPER LEFTMOST NODE

CALLED DIRECTLY BY:

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: HRW/MAIN
PURPOSE: MAIN MODULE FOR HIERARCHICAL REPORT WRITER
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: HRW
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
MAIN()

DESCRIPTION
MAIN PROGRAM FOR HIERARCHICAL REPORT WRITER.

INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPCODE - FORM PROCESSOR RETURN CODES
FPPARM - FORM PROCESSOR PARAMETERS
CHART - CHART INCLUDE FILE
HRWFRM - HRW FORM DEFINITION

ROUTINES CALLED:

INITAL
INITFP
ADDFRM
GDATA
ESCPY
FOPEN
ATOI
MEMCMP
PMSGLS
PUTATT
PUTCUR
FCLOSE
DELNODE - DELETE A SPECIFIED NODE IN TREE
TERMFP
TRMNAT
READTREE - READ DUMPTREE FILE
REPOS - REPOSITION MODULE EXPANSIONS
MODPAGE - MODIFY PAGES
PAGTREE - PAGE TREE

ARRANGE	- ARRANGE CHART AND ASSIGNS PAGE NUMBERS
HBALANC	- HORIZONTAL TREE BALANCE
PRNTREE	- PRINT TREE
OUTSCR	
OISCR	

REPORT WRITER Module Documentation

NAME: INDENT
PURPOSE: INDENT A LINE OF GENERATED CODE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

M = INT

INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

PUTC

CALLED DIRECTLY BY:

MAKWHE/COBWHE - COBOL WHERE ES
MAKWHE - MAKE THE WHERE CLAUSE EXTERNAL SCHEMA VARIABLES
SELGEN - SELECT GENERATE
ASSIGN - ASSIGN FILE SECTION
FD - FD SECTION DECLARATIONS
CLSFIL - CLOSE FILES
ENDGEN - END GERNERATE
PROCEN - PROCEDURE DIVISION GENERATE
DATAGEN - DATA DIVISION GENERATE
FILELNK - FILE LINKAGE SECTION GENERATE
OPNFIL - GENERATE OPEN FILE SECTION
USING - GENERATE USING SECTION
SELWS - SELECT WORKING STORAGE SECTION
INSWS - INSERT WORKING STORAGE SECTION

INSERT	- INSERT PROCEDURE
NDMLLAB	- GENERATE LABELS
COBPE	- COBOL PE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: INSERT
PURPOSE: INSERT PROCEDURE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

LANG = INT

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

INDENT - INDENT A LINE OF GENERATED CODE
FPRINTF
GETCOL - GET THE COLUMN NAME OF A TABLE.COLUMN OR COLUMN
STRING
DASH - WRITE DASH '-'

CALLED DIRECTLY BY:

STDCODE - STANDARD COBOL CODE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: INSRSV
PURPOSE: INSERT RESOLVE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

```
VOID INSRSV(INSPTR, TRGPTR, ACTPTR)
    INSERT *INSPTR;
    TRGLST *TRGPTR;
    ACTLST *ACTPTR;
```

INPUTS:

INSPTR - INSERT FROM WHICH TO LOOK FOR PATH.
TRGPTR - CONDITION THIS INSERT IS ASSOCIATED WITH.
ACTPTR - ACTION THIS INSERT IS ASSOCIATED WITH.

DESCRIPTION

RESOLVES ALL QUALIFIED NAMES INTO FIELD POINTERS FOR ALL
NAMES
WHICH ARE ROOTED IN INSERT (SELECT, VALUE LIST).

ARGUMENTS:

INSPTR =	INSERT *
TRGPTR =	TRGLST *
ACTPTR =	ACTLST *

INCLUDE FILES:

STDTP	- STANDARD TYPE DEFINITIONS
FPD	- FORM PROCESSOR DATA
FPCODE	- FORM PROCESSOR RETURN CODES
RW	- REPORT WRITER DEFINITIONS

ROUTINES CALLED:

UQPTH	- UNIVERSAL QUALIFIER PATH
ERROR	- ISSUE ERROR MESSAGE
GETPTH	- GET PATH

CALLED DIRECTLY BY:

ACTRSV - ACTION RESOLVE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: INSW
PURPOSE: INSERT WORKING STORAGE SECTION
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

LANG = INT

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

NULBLK - BLANK FILL A STRING
STRCPY
DASH - WRITE DASH '-'
INDENT - INDENT A LINE OF GENERATED CODE
FPRINTF
MAKES - MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE
SAVEES - SAVE ES INFORMATION

CALLED DIRECTLY BY:

DATAGEN - DATA DIVISION GENERATE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: ISOPNE
PURPOSE: DETERMINE IF THIS FIELD IS OPEN ENDED
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

ISOPNE (DP)
FIELD *DP;

INPUTS/OUTPUTS:
NONE

INPUTS:
(DP) - FIELD POINTER

OUTPUTS:
THIS ROUTINE RETURNS TRUE IF THIS FIELD
IS OPEN ENDED.

DESCRIPTION

THIS LOOKS UP THE FORMS HIERARCHY TREE TO DETERMINE IF ITS
AN FORM OF AN OPEN ENDED ARRAY.

ARGUMENTS:

DP = FIELD *

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPDINI - FPD INITIALIZATION
FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE

CALLED DIRECTLY BY:

GENAR - GENERATE ACTION PRESENT
GENAS - GENERATE ACTION SET

GENFS	- GENERATE FORM DATA STRUCTURES
GENFSD	- GENERATE FORM STRUCTURE DATA INITIALIZATION
MAKQR	- MAKE QUALIFIED REFERENCE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MAKACT
PURPOSE: MAKE ACTION LIST ELEMENT
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: YTAB
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

----- SYNOPSIS

VOID MAKACT(TYPE)
CHAR TYPE;

DESCRIPTION

MAKES AN ACTLST NODE, PUTS IN VALUES AND ADDS IT TO THE
LIST

ARGUMENTS:

TYPE = CHAR

INCLUDE FILES:

FLAN.Y" - **** PURPOSE NOT FOUND BY STRIPPER ****
STDYYP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
MATH - **** PURPOSE NOT FOUND BY STRIPPER ****

ROUTINES CALLED:

MYALLOC - MY MALLOC

CALLED DIRECTLY BY:

YYPARSE - FLAN PARSER

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MAKES
PURPOSE: MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: MAKES
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

```
MAKES (LANG, SPTR, REC_CNT_PTR)
INT LANG;
TBLIST *TPTR;
INT SELNO;
INT *REC_CNT_PTR;
```

DESCRIPTION

WRITES A RECORD STRUCTURE ON A FILE IN THE CURRENT
DIRECTORY FOR THE GIVEN TABLE OR VIEWNAME.
ALSO CREATES A EDIT CONVERSION RECORD STRUCTURE FOR THE
EACH
EXTERNAL SCHEMA DATA ITEM

ARGUMENTS:

```
LANG = INT
TBLNAM = CHAR *
TBLNUM = INT
SELNO = INT
REC_CNT_PTR = INT *
```

INCLUDE FILES:

```
STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
```

ROUTINES CALLED:

```
CDMESQY - PROGRAM NAME CDMESQY
ERROR - ISSUE ERROR MESSAGE
NULBLK - BLANK FILL A STRING
DASH - WRITE DASH '-'
CES - C ES
```

COBES	- COBOL ES RECORD
CCONV	- C CONVERSIONS
COBCONV	- COBOL CONVERSIONS
STRCPY	
STRNCPY	
STRLEN	

CALLED DIRECTLY BY:

SELWS	- SELECT WORKING STORAGE SECTION
INSWS	- INSERT WORKING STORAGE SECTION

USED IN MAIN PROGRAM(S):

GRP/MAIN	- GENERATE APPLICATION/REPORT PROGRAM
----------	---------------------------------------

REPORT WRITER Module Documentation

NAME: MAKES/CNUMPIC
PURPOSE: C NUMBERS
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: MAKES
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

M = INT
T = INT

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

FPRINTF
STRCAT

CALLED DIRECTLY BY:

COBCONV - COBOL CONVERSIONS

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MAKES/INDENT
PURPOSE: INDENT
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: MAKES
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

M = INT
T = INT

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

PUTC

CALLED DIRECTLY BY:

CES - C ES
COBES - COBOL ES RECORD
CCONV - C CONVERSIONS
COBCONV - COBOL CONVERSIONS

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MAKES/NUMPIC
PURPOSE: NUMBER PICTURE CLAUSE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: MAKES
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

M = INT
T = INT

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

FPRINTF

CALLED DIRECTLY BY:

COBES - COBOL ES RECORD
COBCONV - COBOL CONVERSIONS

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MAKINS
PURPOSE: MAKE INSERT
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: PSSTRC
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

LANG = INT
IPTR = INSERT *

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

PSSTRC/INDENT - INDENT
FPRINTF
PSSTRC/CSUB - C SUBSTITUTE
PSSTRC/COBSUB - COBOL SUBSTITUTE

CALLED DIRECTLY BY:

NDMLLNK - LINKAGE SECTION

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MAKINT
PURPOSE: MAKE EXPRESSION INTO AN INTEGER
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: ENODE * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

----- SYNOPSIS

ENODE *MAKINT(EP)
ENODE *EP;

DESCRIPTION

CONVERT THE SPECIFIED EXPRESSION TO INTEGER AND RETURN
POINTER TO NEW
EXPRESSION.

ARGUMENTS:

EP = ENODE *

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

MYALLOC - MY MALLOC

CALLED DIRECTLY BY:

YYPARSE - FLAN PARSER

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MAKPS
PURPOSE: MAKES THE PRESENTATION SCHEMA RECORD
STRUCTURE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: PSSTRC
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

```
MAKPS(LANG, SPTR)
INT LANG;
SELECT *SPTR;
```

DESCRIPTION

WRITES A RECORD STRUCTURE ON A FILE IN THE CURRENT
DIRECTORY FOR THE GIVEN SELECT.
THE RECORD STRUCTURE INCLUDES ALL THE FIELDS ON THE FORM
THAT THE
SELECT IS SELECTING INTO.

ARGUMENTS:

```
LANG = INT
SPTR = SELECT *
```

INCLUDE FILES:

```
STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
```

ROUTINES CALLED:

```
PSSTRC/CSUB - C SUBSTITUTE
PSSTRC/COBSUB - COBOL SUBSTITUTE
```

CALLED DIRECTLY BY:

```
SELWS - SELECT WORKING STORAGE SECTION
```


PS 620344501
30 September 1990

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MAKQR
PURPOSE: MAKE QUALIFIED REFERENCE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

```
CHAR *MAKQR(DP, SUFFIX, S1, TFLDP, AFLDP)
    FIELD *DP;
    CHAR SUFFIX;
    CHAR S1[];
    FLDLST *TFLDP, *AFLDP;
```

INPUTS/OUTPUTS:
NONE

INPUTS:

DP - FIELD POINTER TO AN ITEM.
SUFFIX - THIS IS A CHARACTER VALUE OF EITHER C OR P
 TO REPRESENT CURRENT OR PREVIOUS.
TFLDP - LIST OF FIELDS TO GENERATE A "TINDX%D" INDEX
 REFERENCE.
AFLDP - LIST OF FIELDS TO GENERATE A "AINDX%D" INDEX
 REFERENCE.

OUTPUTS:

S1 - THIS IS THE QUALIFIED REFERENCE CHARACTER STRING

DESCRIPTION

THIS ROUTINE STARTS AT THE ITEM POINTER LOOKING UP THE FORM
PROCESSOR HIERARCHY TO GENERATE A FULLY QUALIFIED
 REFERENCE WHICH
CORRESPONDS TO THOSE CURRENT AND PREVIOUS DATA STRUCTURES
 GENERATED
BY MAKINC.

ARGUMENTS:

DP = FIELD *
SUFFIX = CHAR
S1 = CHAR []
TFLDP = FLDLST *
AFLDP = FLDLST *

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPDINI - FPD INITIALIZATION
FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

STRCAT
STRCPY
SPRINTF
ISOPNE - DETERMINE IF THIS FIELD IS OPEN ENDED

CALLED DIRECTLY BY:

GENAS - GENERATE ACTION SET
GENAI - GENERATE ACTION INSERT
SELWHR - SELECT WHERE
GENFSD - GENERATE FORM STRUCTURE DATA INITIALIZATION
MAPDB - MAP DATABASE
VISITA - VISIT ARRAYS ON THIS FORM
SETNDP - SET NODUPLICATE FIELDS TO BLANK IF THEY ARE
DUPLICATED
RSETNDP - RESET NODUPLICATE FIELDS TO VALUE OF NODUP%D
CALCSTAT - CALCULATE STATISTIC
RSETSTAT - RESET STATISTIC

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MAKSTR
PURPOSE: MAKE EXPRESSION INTO A STRING
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: ENODE * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

----- SYNOPSIS

ENODE *MAKSTR(EP)
ENODE *EP;

DESCRIPTION

CONVERT THE SPECIFIED EXPRESSION TO STRING AND RETURN
POINTER TO NEW
EXPRESSION.

ARGUMENTS:

EP = ENODE *

INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

MYALLOC - MY MALLOC

CALLED DIRECTLY BY:

YYPARSE - FLAN PARSER

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MAKWH
PURPOSE: MAKE WHERE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: PSSTRC
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

LANG = INT
SPTR = SELECT *

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

PSSTRC/INDENT - INDENT
FPRINTF
PSSTRC/CSUB - C SUBSTITUTE
PSSTRC/COBSUB - COBOL SUBSTITUTE

CALLED DIRECTLY BY:

NDMLLNK - LINKAGE SECTION

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MAKWHES
PURPOSE: MAKE THE WHERE CLAUSE EXTERNAL SCHEMA
VARIABLES
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: MAKWHES
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

```
MAKWHES(LANG, SPTR)
INT LANG;
SELECT *SPTR;
```

DESCRIPTION

WRITES A WHERE CLAUSE EXTERNAL SCHEMA RECORD STRUCTURE
FOR ALL
EXTERNAL SCHEMA COLUMNS THAT MAP TO PRESENTATION ITEMS IN
THE
WHERE CLAUSE OF THE SELECT. IT IS ALLOWABLE FOR ONE ES
ITEM TO
MAP TO MORE THAN ONE PS ITEM

ARGUMENTS:

```
LANG = INT
SPTR = SELECT *
```

INCLUDE FILES:

```
STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
```

ROUTINES CALLED:

```
INDENT - INDENT A LINE OF GENERATED CODE
FPRINTF
MAKWHES/CWHES - C WHERE ES
MAKWHES/COBWHES - COBOL WHERE ES
```

CALLED DIRECTLY BY:

SELWS - SELECT WORKING STORAGE SECTION

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MAKWHES/COBWHES
PURPOSE: COBOL WHERE ES
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: MAKWHES
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

ESWH_PTR = PREDOPER *
COLWH_PTR = PREDOPER *
SPTR = SELECT *
LOOPCNT = INT

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

MAKWHES/NUMPIC - NUMBER PICTURE CLAUSE
FPRINTF
INDENT - INDENT A LINE OF GENERATED CODE
DASH - WRITE DASH '-'
GETTBL - GET A TABLE NAME
GETCOL - GET THE COLUMN NAME OF A TABLE.COLUMN OR COLUMN
STRING

CALLED DIRECTLY BY:

MAKWHES - MAKE THE WHERE CLAUSE EXTERNAL SCHEMA VARIABLES

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MAKWHES/CWHES
PURPOSE: C WHERE ES
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: MAKWHES
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

ESWH_PTR = PREDOPER *
COLWH_PTR = PREDOPER *
SPTR = SELECT *
LOOPCNT = INT

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

CALLED DIRECTLY BY:

MAKWHES - MAKE THE WHERE CLAUSE EXTERNAL SCHEMA VARIABLES

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MAKWHES/NUMPIC
PURPOSE: NUMBER PICTURE CLAUSE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: MAKWHES
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

M = INT
T = INT

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

FPRINTF

CALLED DIRECTLY BY:

MAKWHES/COBWHES - COBOL WHERE ES

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MAPDB
PURPOSE: MAP DATABASE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

INPUTS/OUTPUTS:
NONE

INPUTS:
(DP) - FIELD POINTER

OUTPUTS:
NONE

DESCRIPTION

TRAVERSES ALL SELECTS LOOKING FOR ONES THAT TARGET TO THE
SCOPE
OF THE FORM INDICATED BY THE INPUT PARAMETER. IT GENERATES
STATEMENTS OF THE FORM:

MEMCPY(FRMPTR->FIELD, DBR%D.FIELD, SIZE); %D - NUMBER OF
SELECT.

ARGUMENTS:

FP = FIELD *

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPDINI - FPD INITIALIZATION
FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

STRCHR
MAKQR - MAKE QUALIFIED REFERENCE
SPRINTF
STRLEN
GEN - GENERATE A LINE OF CODE

CALLED DIRECTLY BY:

BSCODE - BUILD SUBROUTINE CODE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MKINC
PURPOSE: MAKE INCLUDE (ACTUALLY STRUCTURE TAGS ALA MAKINC)
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENMN2
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

```
    MKINC(FP)
        FIELD *FP;
```

INPUTS/OUTPUTS:
NONE

INPUTS:
FP - FORM POINTER

OUTPUTS:
NONE

DESCRIPTION

GENERATES THE STRUCTURE TAGS FOR ALL THE FORMS USED IN AN APPLICATION OR REPORT. THIS UPPER LEVEL PROCEDURE TRAVERSES ALL FORMS WHICH ARE PRESENTED IN WINDOWS. THE DATA STRUCTURES ARE OF THE FORM:

```
#IFDEF FRM7                                MAKE SURE THE FORM IS
                                           DECLARED ONCE ONLY.
    STRUCT FRM7                            STRUCTURE TAG.
    {
        CHAR DBID[4];                      DATA FIELDS (ITEMS) ON
        THE FORM.
        CHAR DBNAME[20];
        CHAR HOSTID[3];
        CHAR DBMSNAME[30];
    } ; (* INSRT *)
#define FRM7                                DEFINE A SYMBOL.
#endif
#ifdef FRM3
    STRUCT FRM3
    {
        CHAR PDATE[10];
        STRUCT FRM7 FRM7[10];  (* INSRT *)  A SUBFORM OF
```

```
                FORM 3.  
            } ; (* SEL45 *)  
#DEFINE FRM3  
#ENDIF
```

ARGUMENTS:

FP = FIELD *

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

MKINC - MAKE INCLUDE (ACTUALLY STRUCTURE TAGS ALA
 MAKINC)
HASITEM - THIS ROUTINE DETERMINES IF THERE IS AN ITEM
 WITHIN
SPRINTF
STRLEN
BLEN
GEN - GENERATE A LINE OF CODE
ERROR - ISSUE ERROR MESSAGE

CALLED DIRECTLY BY:

GENMAIN - GENERATE MAIN PROGRAM
MKINC - MAKE INCLUDE (ACTUALLY STRUCTURE TAGS ALA
 MAKINC)

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MKPOS
PURPOSE: MAKE POSITION NODE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: POS * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS

```
POS *MKPOS(HPOS, HMIN, HLOC, HREF, VPOS, VMIN, VLOC, VREF)
  INT HPOS, HMIN, HLOC;
  CHAR *HREF;
  INT VPOS, VMIN, VLOC;
  CHAR *VREF;
```

DESCRIPTION

CREATES THE SPECIFIED POSITION NODE AND ADDS IT TO THE LIST. HPOS AND VPOS ARE THE REFERENCE POINTS ON THE CURRENT FIELD, HMIN AND VMIN ARE THE LOCATION RELATIVE TO THE REFERENCE FIELD, HLOC AND VLOC ARE THE REFERENCE POINTS ON THE REFERENCE FIELD, AND HREF AND VREF ARE THE REFERENCE FIELDS.

ARGUMENTS:

```
HPOS = INT
HMIN = INT
HLOC = INT
HREF = CHAR *
VPOS = INT
VMIN = INT
VLOC = INT
VREF = CHAR *
```

INCLUDE FILES:

```
STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES
```

ROUTINES CALLED:

MYALLOC - MY MALLOC

CALLED DIRECTLY BY:

YYPARSE - FLAN PARSER

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MLPFRM
PURPOSE: MAKE A LIST OF PRESENTED FORMS
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS
MLPFRM()

DESCRIPTION

MAKES TWO LISTS OF PRESENTED FORMS. ONE LIST POINTED TO BY
PRSFrm,
CONTAINS ALL PRESENTED FORMS. THE SECOND LIST POINTED TO
BY TOPFRM,
CONTAINS ALL FORMS PRESENTED IN THE WINDOW SCREEN.

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

FNDFRM - FIND FORM
ERROR - ISSUE ERROR MESSAGE
MALLOC

CALLED DIRECTLY BY:

RwOPN - REPORT WRITER OPEN FORMS

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: MODPAGE
PURPOSE: MODIFY PAGES
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: MODPAGE
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
MODPAGE()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

THIS ROUTINE MOVES ANY NODE WHICH IS REFERENCED MORE THAN
ONCE TO
ITS OWN PAGE.

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

SPLITNODE - SPLIT A NODE FOR PAGE BREAKS

CALLED DIRECTLY BY:

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: MOVCLD
PURPOSE: MOVE CHILDREN
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: MOVCLD
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
MOVCLD()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

MOVE ALL THE CHILDREN FROM FROM_NODE TO TO_NODE,
FOLLOWING ANY EXISTING
TO_NODE CHILDREN

ARGUMENTS:

FROM_NODE = NODE *
TO_NODE = NODE *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

BLDNODE - BUILD NODE
CLOSEGAP - CLOSE GAP IN TREE
SPLICE - SPLICE TREE INTO ANOTHER TREE
DELNODE - DELETE A SPECIFIED NODE IN TREE

CALLED DIRECTLY BY:

REPOS - REPOSITION MODULE EXPANSIONS

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: MOVECLD
PURPOSE: MOVE CHILD'S POSITION
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: MOVECLD
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
MOVECLD()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

INCREMENT THIS CHILD'S POSITION AND EVERYTHING TO THE
RIGHT ALONG THIS LEVEL (INCLUDING THEIR CHILDREN)
ON DOWN AND THIS CHILD'S CHILDREN ON DOWN)

ARGUMENTS:

FIRST_PTR = NODE *
OFFSET = INT

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

GETLOWLEF - GET LOWER LEFT CHILD NODE

CALLED DIRECTLY BY:

HBALANC - HORIZONTAL TREE BALANCE
PAGTREE - PAGE TREE

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: MYALLOC
PURPOSE: MY MALLOC
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

----- SYNOPSIS

CHAR *MYALLOC(SIZE)
UNSIGNED SIZE;

DESCRIPTION

ALLOCATE THE SPECIFIED MEMORY IF POSSIBLE, ELSE ISSUE
FATAL ERROR

ARGUMENTS:

SIZE = UNSIGNED

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

FATAL - ISSUE FATAL ERROR MESSAGE
MALLOC

CALLED DIRECTLY BY:

CHKFLD - CHECK FIELD
CHKARY - CHECK ARRAY
CSTASH - CHARACTER STASH
WRTEXP - WRITE EXPRESSION
MKPOS - MAKE POSITION NODE
MAKINT - MAKE EXPRESSION INTO AN INTEGER
MAKSTR - MAKE EXPRESSION INTO A STRING
MAKACT - MAKE ACTION LIST ELEMENT
YYPARSE - FLAN PARSER

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: NDMLGEN
PURPOSE: NDML COBOL APPLICATION GENERATOR
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS
NDMLGEN()

DESCRIPTION
CALLS THE APPROPRIATE ROUTINES TO GENERATE THE
PRESENTATION
SCHEMA RECORD STRUCTURE, THE EXTERNAL SCHEMA RECORD
STRUCTURE
AND THE CONVERSION CODE TO GO FROM ONE CDM DATA TYPE TO
ANOTHER.
AND THE NDML COMMANDS SPECIFIED.

ARGUMENTS:

LANG = INT
APNAME = CHAR []

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

UTRNCPY
SPRINTF
FOPEN
SYSMSG
STD CODE - STANDARD COBOL CODE
FCLOSE

CALLED DIRECTLY BY:

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: NDMLLAB
PURPOSE: GENERATE LABELS
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

TYPE = CHAR

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

INDENT - INDENT A LINE OF GENERATED CODE
FPRINTF

CALLED DIRECTLY BY:

PROCGEN - PROCEDURE DIVISION GENERATE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: NDMLLNK
PURPOSE: LINKAGE SECTION
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

LANG = INT
TYPE = CHAR

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

MAKWH - MAKE WHERE
MAKINS - MAKE INSERT

CALLED DIRECTLY BY:

DATAGEN - DATA DIVISION GENERATE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: NEXTLEV
PURPOSE: ADVANCE POINTERS TO NEXT LEVEL OF SUBTREE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NEXTLEV
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
NEXTLEV()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION
ADVANCES L_PTR AND R_PTR TO THE NEXT LEVEL OF A SUBTREE

ARGUMENTS:

L_PTR = NODE **
R_PTR = NODE **

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

CALLED DIRECTLY BY:

CLOSEGAP - CLOSE GAP IN TREE
GETSIZE - GET SUBTREE SIZE
REPOS - REPOSITION MODULE EXPANSIONS

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: NULBLK
PURPOSE: BLANK FILL A STRING
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

TMPSTR = CHAR []
INSTR = CHAR []

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

STRCHR
STRCPY

CALLED DIRECTLY BY:

MAKES - MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE
SELGEN - SELECT GENERATE
SAVEES - SAVE ES INFORMATION
SELWS - SELECT WORKING STORAGE SECTION
INSWS - INSERT WORKING STORAGE SECTION
GETTBL - GET A TABLE NAME

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: OPNFIL
PURPOSE: GENERATE OPEN FILE SECTION
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

SPTR = SELECT *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

INDENT - INDENT A LINE OF GENERATED CODE
FPRINTF
OPNFIL - GENERATE OPEN FILE SECTION

CALLED DIRECTLY BY:

SELGEN - SELECT GENERATE
PROCEN - PROCEDURE DIVISION GENERATE
OPNFIL - GENERATE OPEN FILE SECTION

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: PAGNODE
PURPOSE: PAGE NODES
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: PAGNODE
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
PAGNODE()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

THIS ROUTINE DIVIDES ANY NODES WHICH ARE TOO BIG TO FIT
ON A SINGLE
PAGE.

ARGUMENTS:

FIRST_PTR = NODE *
PAGE_WIDTH = INT

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

GETLOWLEF - GET LOWER LEFT CHILD NODE
GETUPLFT - GET UPPER LEFTMOST NODE
COPYNODE - COPY A NODE IN TREE

CALLED DIRECTLY BY:

PAGTREE - PAGE TREE

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: PAGTREE
PURPOSE: PAGE TREE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: PAGTREE
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
PAGTREE()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION
THIS ROUTINE DIVIDES THE TREE INTO PAGES.

ARGUMENTS:

PAGE_WIDTH = INT
PAGE_DEPTH = INT

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

PAGNODE - PAGE NODES
GETFIT - GET SUBTREE THAT FITS ON PAGE
SPLITNODE - SPLIT A NODE FOR PAGE BREAKS
GETSIZE - GET SUBTREE SIZE
MOVECLD - MOVE CHILD'S POSITION

CALLED DIRECTLY BY:

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: PEMAP
PURPOSE: THE PRESENTATION SCHEMA AND THE EXTERNAL
SCHEMA AND MAPPING
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: PEMAP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

```
PEMAP(LANG, STR1, STR2, FPTR, DPTR)
INT LANG;
CHAR *STR1;
CHAR *STR2;
FIELD *FPTR;
STRUCT DTYPE *DPTR;
```

DESCRIPTION

GENERATES THE CODE TO TRANSFORM AN PRESENTATION SCHEMA
DATA ITEM INTO
A EXTERNAL SCHEMA ITEM. THIS IS DONE ON A PER ITEM BASIS
AND THE
SOURCE AND DESTINATION STRINGS OF CODE (STR1, STR2) ARE
PASSED IN SO
THE RESULTING CODE MAY USE THE CORRECT VARIABLES.

ARGUMENTS:

```
LANG = INT
STR1 = CHAR *
STR2 = CHAR *
FPTR = FIELD
DPTR = CDMDTYPE *
```

INCLUDE FILES:

```
STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
```

ROUTINES CALLED:

CPE - C PE
COBPE - COBOL PE

CALLED DIRECTLY BY:

SELGEN - SELECT GENERATE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: PRNT
PURPOSE: PRINT MODULE NAMES HIERARCHICALLY
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: PRNT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
PRNT()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

THIS IS A ROUTINE TO PRINT MODULE NAMES IN A HIERARCHICAL
ORDER.
USEFUL FOR DEBUGGING PURPOSES.

ARGUMENTS:

FIRST_PTR = NODE *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

PRINTF
PRNT - PRINT MODULE NAMES HIERARCHICALLY

CALLED DIRECTLY BY:

PRNT - PRINT MODULE NAMES HIERARCHICALLY

REPORT WRITER Module Documentation

NAME: PRNTREE
PURPOSE: PRINT TREE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: PRNTREE
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS

PRNTREE(TOP_MODULE_PTR, TOP_NODE_PTR, OUTCHART, CHARSET,
STRIP,
PAGE_WIDTH, PAGE_DEPTH)

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

THIS ROUTINE PRINTS THE TREE.

ARGUMENTS:

TEMPFILE = FILE *
OUTCHART = FILE *
CHARSET = INT
STRIP = BOOL
PAGE_WIDTH = INT
PAGE_DEPTH = INT

INCLUDE FILES:

STDYIP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

FPRINTF

GETSIZE - GET SUBTREE SIZE
PUTC
GETLOWLEF - GET LOWER LEFT CHILD NODE
STRIPLEV - DRAW STRIP CHART LEVEL
DRAWLEV - DRAW A LEVEL OF THE CHART
DOINDEX - DO CHART INDEX
FPUTS
SPRINTF
STRLEN

CALLED DIRECTLY BY:

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: PROCGEN
PURPOSE: PROCEDURE DIVISION GENERATE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

LANG = INT
TYPE = CHAR

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

MAP
USING - GENERATE USING SECTION
OPNFIL - GENERATE OPEN FILE SECTION
NDMLLAB - GENERATE LABELS
CLSFIL - CLOSE FILES
INSMAP
SELMAP - MAP SELECTED DATA TO OUTPUT RECORD
INDENT - INDENT A LINE OF GENERATED CODE
FPRINTF
PSESMAP

CALLED DIRECTLY BY:

STDCODE - STANDARD COBOL CODE

PS 620344501
30 September 1990

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: PSSTRC/COBSUB
PURPOSE: COBOL SUBSTITUTE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: PSSTRC
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

DP = FIELD *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

BLN
FPRINTF
PSSTRC/INDENT - INDENT

CALLED DIRECTLY BY:

MAKPS - MAKES THE PRESENTATION SCHEMA RECORD STRUCTURE
MAKWH - MAKE WHERE
MAKINS - MAKE INSERT

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: PSSTRC/CSUB
PURPOSE: C SUBSTITUTE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: PSSTRC
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

DP = FIELD *

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

BLN
FPRINTF
PSSTRC/INDENT - INDENT

CALLED DIRECTLY BY:

MAKPS - MAKES THE PRESENTATION SCHEMA RECORD STRUCTURE
MAKWH - MAKE WHERE
MAKINS - MAKE INSERT

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: PSSTRC/INDENT
PURPOSE: INDENT
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: PSSTRC
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

M = INT

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

PUTC

CALLED DIRECTLY BY:

PSSTRC/CSU - C SUBSTITUTE
PSSTRC/COBSUB - COBOL SUBSTITUTE
MAKWH - MAKE WHERE
MAKINS - MAKE INSERT

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: PUTLIN
PURPOSE: PRINT LEVEL OF TREE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: PUTLIN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
PUTLIN()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

PRINT A FORMATTED LEVEL OF THE TREE

ARGUMENTS:

OUTCHART = FILE *
MAXLINE = INT
LINE = CHAR **
CHARSET = INT

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

STRLEN
STRCHR
PUTC

CALLED DIRECTLY BY:

DRAWLEV - DRAW A LEVEL OF THE CHART
STRIPLEV - DRAW STRIP CHART LEVEL

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: READDB
PURPOSE: READ DATA BASE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

READDB(FP)
FIELD *FP;

INPUTS/OUTPUTS:
NONE

INPUTS:
FP - FIELD POINTER

OUTPUTS:
NONE

DESCRIPTION

TRAVERSES THE LIST OF SELECTS LOOKING FOR ONES THAT TARGET
TO ITEMS
ON THE FORM INDICATED BY THE INPUT PRARMETER. WHEN ONE IS
FOUND IT
CALLS DBFREAD TO READ A DATA RECORD AND CHECK FOR CHANGE
CONDITIONS.

ARGUMENTS:

FP = FIELD *

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPDINI - FPD INITIALIZATION
FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

HASLOWER - HAS A LOWER FORM WHICH READS THE SAME DATA
 RECORD?
DBFREAD - GENERATE DATA BASE FREAD

CALLED DIRECTLY BY:

BSCODE - BUILD SUBROUTINE CODE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: READTREE
PURPOSE: READ DUMPTREE FILE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: RDTREE
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
 READTREE()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

 THIS ROUTINE READS THE FILE CREATED BY DUMPTREE AND
 REBUILDS THE TREE.

ARGUMENTS:

 TEMPFILE = FILE *
 INVERT = BOOL

INCLUDE FILES:

 STDYTP - STANDARD TYPE DEFINITIONS
 STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
 CHART - CHART INCLUDE FILE

ROUTINES CALLED:

 BLDNODE - BUILD NODE
 FTELL
 GETC
 UNGETC
 FGETS
 STRLEN

GETPAR - GET PARENT NODE
GETLOWRIT - GET LOWER RIGHT CHILD NODE
GETLOWLEF - GET LOWER LEFT CHILD NODE
BLDMOD - BUILD MODULE

CALLED DIRECTLY BY:

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: REPOS
PURPOSE: REPOSITION MODULE EXPANSIONS
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: REPOS
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
REPOS()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION
REPOSITION MODULE EXPANSIONS TO THE FIRST REFERENCE TO
THE MODULE
EXPANSION.

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

CLOSEGAP - CLOSE GAP IN TREE
SPLICE - SPLICE TREE INTO ANOTHER TREE
NEXTLEV - ADVANCE POINTERS TO NEXT LEVEL OF SUBTREE
MOVCLD - MOVE CHILDREN
DELNODE - DELETE A SPECIFIED NODE IN TREE

CALLED DIRECTLY BY:

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: RSETNDP
PURPOSE: RESET NODUPPLICATE FIELDS TO VALUE OF
NODUP%D
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

RSETNDP(FP, DP)
FIELD *FP, *DP;

INPUTS/OUTPUTS:
NONE

INPUTS:
FP - PARENT FORM OF DP (HELP IN RECURSION).
DP - FIELD THAT MIGHT HAVE NODUP OPTION.

OUTPUTS:
NONE

DESCRIPTION

TRAVERSES THE FORM HIERARCHY LOOKING FOR ITEMS UNDER FP
WHICH HAVE
THE NODUP OPTION. WHEN IT FINDS ONE IT GENERATES CODE TO
COPY THE
NODUP%D VALUE TO THE FORM FIELD.

ARGUMENTS:

FP = FIELD *
DP = FIELD *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPDINI - FPD INITIALIZATION
FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

RSETNDP - RESET NODUPLICATE FIELDS TO VALUE OF NODUP%D
MAKQR - MAKE QUALIFIED REFERENCE
SPRINTF
STRLEN
GEN - GENERATE A LINE OF CODE
HASDATA - DETERMINE IF THERE ARE ANY SELECT STATEMENTS

CALLED DIRECTLY BY:

BSCODE - BUILD SUBROUTINE CODE
RSETNDP - RESET NODUPLICATE FIELDS TO VALUE OF NODUP%D

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: RSETSTAT
PURPOSE: RESET STATISTIC
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

FP = FIELD *
DP = FIELD *

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

RSETSTAT - RESET STATISTIC
MAKQR - MAKE QUALIFIED REFERENCE
SPRINTF
GEN - GENERATE A LINE OF CODE

CALLED DIRECTLY BY:

FRMPDAT - FORM PDATA
RSETSTAT - RESET STATISTIC

USED IN MAIN PROGRAM(S):

GRP/MAIN -- GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: RWEXPD
PURPOSE: REPORT WRITER EXPAND ARRAYS
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

```
CHAR *RWEXPD(FDP, USELST)
    FIELD *FDP;
    FIELD **USELST;
```

INPUTS:

```
    FIELD *FDP;  ** THE FORM YOU WISH EXPANDED **
    FIELD **USELST; ** WHERE TO LOOK FOR EXPANDING
                   SUBFORMS
```

DESCRIPTION

THIS GUY IS RESPONSIBLE FOR EXPANDING AN ARRAY WHICH WAS PARTIALLY CONSTRUCTED BY FLAN. IT TAKES A POINTER TO THE FORM TO BE EXPANDED AND A POINTER TO THE POINTER TO THE LIST FROM WHICH SUBFORMS MAY BE TAKEN. IF A SUBFORM IS NOT FOUND THE FIELD'S DISPLAY ATTRIBUTE IS SET TO INPUT. THE CASE WHERE BOTH A FIELD AND THE SUBFORM HAVE PROMPTS IS RESOLVED BY CREATING A SPECIAL FIELD TO HOLD THE FIELD'S PROMPTS. USELST MUST BE A POINTER TO A POINTER BECAUSE DELFLD IS USED AND THAT'S WHAT IT NEEDS.

ARGUMENTS:

FDP = FIELD *
USELST = FIELD **

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

COPFLD
ABS
FNDATT - FIND ATTRIBUTE
STRASN
RWSP/FIXFRM - FIX UP A FORM

CALLED DIRECTLY BY:

RWSP/FIXFR - FIX UP A FORM
RWOPN - REPORT WRITER OPEN FORMS

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: RWOPN
PURPOSE: REPORT WRITER OPEN FORMS
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS
VOID RWOPN()

DESCRIPTION

CREATES AN "OPEN LIST" OF FORMS. FROM THE STRUCTURES
CREATED BY FLAN
SUBFORMS ARE COPIED IN PLACE AND ARRAYS ARE EXPANDED TO
THEIR FULL
SIZE.

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

RWEXPD - REPORT WRITER EXPAND ARRAYS
MLPFRM - MAKE A LIST OF PRESENTED FORMS
WINRSV - WINDOW RESOLVE
FLDRSV - FIELD RESOLVE
TRGRSV - TRIGGER RESOLVE

CALLED DIRECTLY BY:

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: RWSP/FIXFRM
PURPOSE: FIX UP A FORM
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

```
CHAR *FIXFRM(DP, USELST)
    FIELD *DP;
    FIELD **USELST;
```

INPUTS:

DP - DUMMY FORM FIELD TO BE FIXED UP.
USELST - WHERE TO LOOK FOR THE SUBFORM TO COPY.

DESCRIPTION

FIXES A SUBFORM BY LOCATING IT AND ATTACHING IT IN PLACE
AND
EXPANDING IT IF REQUIRED.

ARGUMENTS:

DP = FIELD *
USELST = FIELD **

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

RWEXPD - REPORT WRITER EXPAND ARRAYS
COPFLD
FNDATT - FIND ATTRIBUTE
STRCMP

CALLED DIRECTLY BY:

RWEXPD - REPORT WRITER EXPAND ARRAYS

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: SAVEES
PURPOSE: SAVE ES INFORMATION
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

ITMNAM = CHAR [CDMCOLNAMLEN +1]
DPTR = CDMdtype *
REC_CNT = INT

INCLUDE FILES:

STDtyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

NULBLK - BLANK FILL A STRING
STRNCMP
STRLEN
ESCPY
ATOI

CALLED DIRECTLY BY:

SELWS - SELECT WORKING STORAGE SECTION
INSWS - INSERT WORKING STORAGE SECTION

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: SELECT
PURPOSE: GENERATE SELECT CODE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

LANG = INT

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

SELGEN - SELECT GENERATE

CALLED DIRECTLY BY:

STDCODE - STANDARD COBOL CODE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: SELGEN
PURPOSE: SELECT GENERATE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

LANG = INT
PPTR = SELECT *
SPTR = SELECT *
TOPSEL = SELECT *

INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

SELGEN - SELECT GENERATE
NULBLK - BLANK FILL A STRING
STRCPY
PEMAP - THE PRESENTATION SCHEMA AND THE EXTERNAL SCHEMA
AND MAPPING
SPRINTF
DASH - WRITE DASH '-'
GETTBL - GET A TABLE NAME
GETCOL - GET THE COLUMN NAME OF A TABLE.COLUMN OR COLUMN
STRING
OPNFIL - GENERATE OPEN FILE SECTION
FPRINTF
INDENT - INDENT A LINE OF GENERATED CODE

CALLED DIRECTLY BY:

SELGEN	- SELECT GENERATE
SELECT	- GENERATE SELECT CODE

USED IN MAIN PROGRAM(S):

GRP/MAIN	- GENERATE APPLICATION/REPORT PROGRAM
----------	---------------------------------------

REPORT WRITER Module Documentation

NAME: SELLEN
PURPOSE: COMPUTE LENGTH OF SELECT PS RECORD
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

SPTR = SELECT *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

BLN

CALLED DIRECTLY BY:

FD - FD SECTION DECLARATIONS

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: SELMAP
PURPOSE: MAP SELECTED DATA TO OUTPUT RECCRD
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

LANG = INT
SPTR = SELECT *

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

ESPSMAP - THE EXTERNAL SCHEMA AND PRESENTATION SCHEMA
MAPPING
SELMAP - MAP SELECTED DATA TO OUTPUT RECORD

CALLED DIRECTLY BY:

PROC GEN - PROCEDURE DIVISION GENERATE
SELMAP - MAP SELECTED DATA TO OUTPUT RECORD

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: SELOPN
PURPOSE: SELECT OPEN
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS
 SELOPN(SP)
 SELECT *SP;

INPUTS:
SP - POINTER TO SELECT TO HAVE ITS DATA FILE OPENED.

DESCRIPTION

GENERATES CODE TO OPEN THE DATA FILE ASSOCIATED WITH THIS
 SELECT
ACTION.

ARGUMENTS:

SP = SELECT *

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

SPRINTF
GEN - GENERATE A LINE OF CODE
DBFREAD - GENERATE DATA BASE FREAD
SELOPN - SELECT OPEN

CALLED DIRECTLY BY:

GENAQ - GENERATE ACTION QUERY (SELECT)
SELOPN - SELECT OPEN

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: SELRSV
PURPOSE: SELECT RESOLVE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

```
VOID SELRSV(SELPTR, TRGPTR, ACTPTR)
    SELECT *SELPTR;
    TRGPTR *TRGPTR;
    ACTPTR *ACTPTR;
```

INPUTS:

SELPTR - SELECT FROM WHICH TO LOOK FOR PATHS.
TRGPTR - CONDITION THIS SELECT IS ASSOCIATED WITH.
ACTPTR - ACTION THIS SELECT IS ASSOCIATED WITH.

DESCRIPTION

RESOLVES ALL QUALIFIED NAMES INTO FIELD POINTERS FOR ALL
NAMES
WHICH ARE ROOTED IN SELECT (SELECT, VARIABLE LIST, WHERE
LIST).

ARGUMENTS:

SELPTR = SELECT *
TRGPTR = TRGLST *
ACTPTR = ACTLST *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

SELRSV - SELECT RESOLVE
UQPTH - UNIVERSAL QUALIFIER PATH
ERROR - ISSUE ERROR MESSAGE
GETPTH - GET PATH

CALLED DIRECTLY BY:

SELRSV - SELECT RESOLVE
ACTRSV - ACTION RESOLVE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: SELWHR
PURPOSE: SELECT WHERE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

```
SELWHR(SP, TP, AP)
  SELECT *SP;
  TRGLST *TP;
  ACTLST *AP;
```

INPUTS:

SP - POINTER TO SELECT ACTION (NEEDED SINCE SELECTS CAN BE
NESTED).
TP - CONDITION ASSOCIATED WITH THIS ACTION.
AP - THIS ACTION.

DESCRIPTION

GENERATES CODE TO COPY DATA FROM A FORM STRUCTURE TO THE
WHERE STRUCTURE
FOR THOSE SELECTS WHICH HAVE A QUALIFIED NAME IN THE WHERE
CLAUSE.

ARGUMENTS:

SP = SELECT *
TP = TRGLST *
AP = ACTLST *

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

SPRINTF
MAKQR - MAKE QUALIFIED REFERENCE
GEN - GENERATE A LINE OF CODE
SELWHR - SELECT WHERE

CALLED DIRECTLY BY:

GENAQ - GENERATE ACTION QUERY (SELECT)
SELWHR - SELECT WHERE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: SELWS
PURPOSE: SELECT WORKING STORAGE SECTION
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

SPTR = SELECT *
LANG = INT

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

MAKPS - MAKES THE PRESENTATION SCHEMA RECORD STRUCTURE
NULBLK - BLANK FILL A STRING
STRCPY
DASH - WRITE DASH '-'
INDENT - INDENT A LINE OF GENERATED CODE
FPRINTF
MAKWHEs - MAKE THE WHERE CLAUSE EXTERNAL SCHEMA VARIABLES
SELWS - SELECT WORKING STORAGE SECTION
MAKES - MAKES THE EXTERNAL SCHEMA RECORD STRUCTURE
GETCOL - GET THE COLUMN NAME OF A TABLE.COLUMN OR COLUMN
STRING
GETTBL - GET A TABLE NAME
STRCMP
SAVEES - SAVE ES INFORMATION

CALLED DIRECTLY BY:

DATAGEN - DATA DIVISION GENERATE
SELWS - SELECT WORKING STORAGE SECTION

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: SETNDP
PURPOSE: SET NODUPPLICATE FIELDS TO BLANK IF THEY
ARE DUPLICATED
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

```
SETNDP(FP, DP)
    FIELD *FP, *DP;
```

INPUTS/OUTPUTS:
NONE

INPUTS:
FP - PARENT FORM OF DP (HELP IN RECURSION).
DP - FIELD THAT MIGHT HAVE NODUP OPTION.

OUTPUTS:
NONE

DESCRIPTION

TRAVERSES THE FORM HIERARCHY LOOKING FOR ITEMS UNDER FP
WHICH HAVE
THE NODUP OPTION. WHEN IT FINDS ONE IT GENERATES CODE TO
CHECK FOR
DUPLICATE VALUES AND BLANKS THE FORM FORM IF THERE ARE
DUPLICATE
VALUES.

ARGUMENTS:

```
FP = FIELD *
DP = FIELD *
```

INCLUDE FILES:

```
STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPDINI - FPD INITIALIZATION
```

FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

SETNDP - SET NODUPLICATE FIELDS TO BLANK IF THEY ARE
DUPLICATED
MAQOR - MAKE QUALIFIED REFERENCE
SPRINTF
STRLEN
GEN - GENERATE A LINE OF CODE
HASDATA - DETERMINE IF THERE ARE ANY SELECT STATEMENTS

CALLED DIRECTLY BY:

BSCODE - BUILD SUBROUTINE CODE
SETNDP - SET NODUPLICATE FIELDS TO BLANK IF THEY ARE
DUPLICATED

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: SORT
PURPOSE: SORT MODULE NAMES
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: SORT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
SORT()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

ARGUMENTS:

MAX_LEN = INT

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

MALLOC
STRCPY
STRUPC
STRCMP

CALLED DIRECTLY BY:

DOINDEX - DO CHART INDEX

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: SPLICE
PURPOSE: SPLICE TREE INTO ANOTHER TREE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: SPLICE
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
SPLICE()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

ARGUMENTS:

FIRST_NODE_PTR = NODE *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

GETLOWLEF - GET LOWER LEFT CHILD NODE
GETLOWRIT - GET LOWER RIGHT CHILD NODE

CALLED DIRECTLY BY:

MOVCLD - MOVE CHILDREN
REPOS - REPOSITION MODULE EXPANSIONS

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: SPLITNODE
PURPOSE: SPLIT A NODE FOR PAGE BREAKS
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: SPLNODE
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
SPLITNODE()

INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

WHEN A PAGE BREAK OCCURS A DUPLICATE NODE IS CREATED IN
ORDER TO
BEGIN A NEW PAGE. THE CHILDREN OF THE OLD NODE BECOME THE
CHILDREN
OF THE NEW NODE

ARGUMENTS:

FIRST_PTR = NODE *
NODE_PTR = NODE *

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

BLDNODE - BUILD NODE
CLOSEGAP - CLOSE GAP IN TREE

CALLED DIRECTLY BY:

MODPAGE - MODIFY PAGES
PAGTREE - PAGE TREE

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: STATRSV
PURPOSE: STATISTIC RESOLVE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS
STATRSV(STATPTR)
STATLST *STATPTR;

INPUTS:
STATPTR - STATISTIC LIST FROM WHICH TO LOOK FOR PATHS.

DESCRIPTION

RESOLVES ALL QUALIFIED NAMES INTO FIELD POINTERS FOR ALL
NAMES
WHICH ARE ROOTED IN STATLST (STATISTIC LIST).

ARGUMENTS:

STATPTR = STATLST *

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

GETPTH - GET PATH
ERROR - ISSUE ERROR MESSAGE

CALLED DIRECTLY BY:

FLDRSV - FIELD RESOLVE

PS 620344501
30 September 1990

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: STD CODE
PURPOSE: STANDARD COBOL CODE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

GENERATE THE NECESSARY DIVISIONS FOR COBOL AND THE DATA
STRUCTURES NECESSARY FOR NTM PROCESSING.
FOR A COBOL PROGRAM TO DO JUST NDML AND WRITE DATA TO FILES
MUST CONSTRUCT FILE SECTION CORRECTLY.
FOR A C PROGRAM WOULD BE DECLARING ALL NTM VARIABLES AS
EXTERNAL TO THE C GENERATED PROCEDURE.

ARGUMENTS:

LANG = INT
APNAME = CHAR *
TYPE = CHAR

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STD IO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

ENDGEN - END GENERATE
SELECT - GENERATE SELECT CODE
SPRINTF
PROCEN - PROCEDURE DIVISION GENERATE
FPRINTF
DATAGEN - DATA DIVISION GENERATE
STRUPC
INSERT - INSERT PROCEDURE

CALLED DIRECTLY BY:

NDMLGEN - NDML COBOL APPLICATION GENERATOR

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: STRIPLEV
PURPOSE: DRAW STRIP CHART LEVEL
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: STRPLEV
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: HRW
DOCUMENTATION GROUP: HRW

DESCRIPTION:

SYNOPSIS
STRIPLEV()

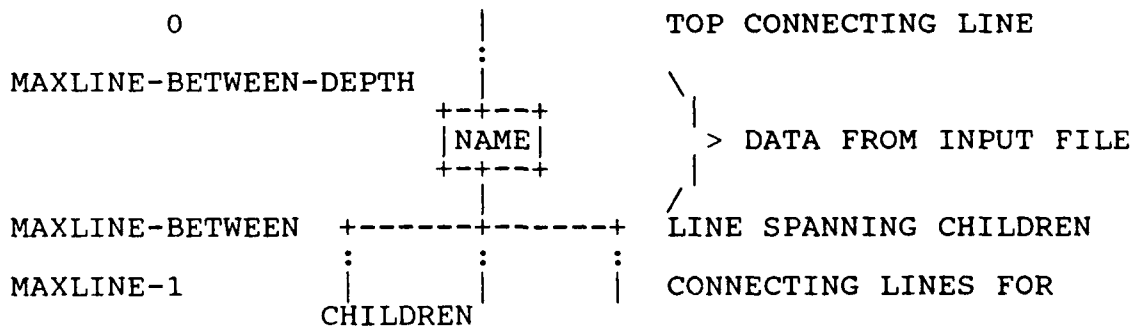
INPUTS/OUTPUTS:

INPUTS:

OUTPUTS:

DESCRIPTION

THIS ROUTINE DRAWS A LEVEL OF A STRIPPED CHART.
A LEVEL CONSISTS OF MAXLINE LINES:



ARGUMENTS:

TEMPFILE = FILE *
OUTCHART = FILE *
START_PTR = NODE *
CHARSET = INT
START_POS = INT
PAGE_WIDTH = INT

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
CHART - CHART INCLUDE FILE

ROUTINES CALLED:

MALLOC
MIN
PUTLIN - PRINT LEVEL OF TREE
FREE
FSEEK
GETC
MEMCPY
FGETS
STRLEN
MEMSET

CALLED DIRECTLY BY:

PRNTREE - PRINT TREE

USED IN MAIN PROGRAM(S):

HRW/MAIN - MAIN MODULE FOR HIERARCHICAL REPORT WRITER

REPORT WRITER Module Documentation

NAME: TRGRSV
PURPOSE: TRIGGER RESOLVE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

TRGRSV(TRGPTR)
TRGLST *TRGPTR;

INPUTS:

TRGPTR - CONDITION LIST FROM WHICH TO LOOK FOR PATHS.

DESCRIPTION

RESOLVES ALL QUALIFIED NAMES INTO FIELD POINTERS FOR ALL
NAMES
WHICH ARE ROOTED IN TRGLST (CONDITION LIST).

ARGUMENTS:

TRGPTR = TRGLST *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

GETPTH - GET PATH
ERROR - ISSUE ERROR MESSAGE
ACTRSV - ACTION RESOLVE
UQPTH - UNIVERSAL QUALIFIER PATH

CALLED DIRECTLY BY:

RWOPN - REPORT WRITER OPEN FORMS

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: UQFOR
PURPOSE: UNIVERSAL QUALIFIER FOR LOOP
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GENACT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

UQFOR(FLDP, TYPE)
FLDLST *FLDP;
CHAR TYPE;

INPUTS:

FLDP - POINTER TO LIST OF FIELDS WHICH REQUIRE UNIVERSAL
QUALIFICATION.
TYPE - 'T' FOR CONDITIONAL INDEX (TINDX%D) 'A' FOR ACTION
(AINDX%D).

DESCRIPTION

GENERATES THE FOR LOOP FOR UNIVERSAL QUALIFICATION.

ARGUMENTS:

FLDP = FLDLST *
TYPE = CHAR

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

SPRINTF
GEN - GENERATE A LINE OF CODE

CALLED DIRECTLY BY:

GENAL - GENERATE ACTION LIST

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: UQPTH
PURPOSE: UNIVERSAL QUALIFIER PATH
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS

```
CHAR *UQPTH(PATH, DP, TFLDPP, AFLDPP)
    CHAR  PATH[];
    FIELD *DP;
    FLDLST **TFLDPP, **AFLDPP;
```

INPUTS/OUTPUTS:

TFLDPP - POINTER TO POINTER OF CONDITION INDEX FIELDS.
AFLDPP - POINTER TO POINTER OF ACTION INDEX FIELDS.

INPUTS:

PATH - PATH WITH UNIVERSAL QUALIFIERS IN IT.
DP - FIRST INSTANCE OF PATH.

DESCRIPTION

MAKES A LIST OF FIELDS WHICH REQUIRE UNIVERSAL
QUALIFICATION FOR
A CONDITION AND ACTION.

ARGUMENTS:

```
PATH = CHAR []
DP = FIELD *
TFLDPP = FLDLST **
AFLDPP = FLDLST **
```

INCLUDE FILES:

```
STDYTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
```

ROUTINES CALLED:

MALLOC

PTHPTR
STRCHR
STRCPY

CALLED DIRECTLY BY:

INSRSV - INSERT RESOLVE
SELRSV - SELECT RESOLVE
TRGRSV - TRIGGER RESOLVE
ACTRSV - ACTION RESOLVE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: USING
PURPOSE: GENERATE USING SECTION
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: NDMLGEN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

ARGUMENTS:

SPTR = SELECT *

INCLUDE FILES:

STD TYP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE
CTLCHR - CONTROL CHARACTERS

ROUTINES CALLED:

INDENT - INDENT A LINE OF GENERATED CODE
FPRINTF

CALLED DIRECTLY BY:

PROCGEN - PROCEDURE DIVISION GENERATE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: VISITA
PURPOSE: VISIT ARRAYS ON THIS FORM
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GRP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

----- SYNOPSIS

VISITA(DP)
FIELD *DP;

INPUTS/OUTPUTS:
NONE

INPUTS:
(DP) - FIELD POINTER

OUTPUTS:
NONE

DESCRIPTION

VISIT AN ARRAY BY GENERATING A LOOP TO GO THRU THE
ELEMENTS IN THE ARRAY.
CHECK FOR GROUP SEPERATORS/END OF FILE, OVERFLOW
CONDITIONS AND CALL THE
PROCEDURE WHICH IMPLEMENTS THE SUBFORM.

ARGUMENTS:

DP = FIELD *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPDINI - FPD INITIALIZATION
FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
NTM - NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

VISITA - VISIT ARRAYS ON THIS FORM
HASDATA - DETERMINE IF THERE ARE ANY SELECT STATEMENTS
GEN - GENERATE A LINE OF CODE
MAKQR - MAKE QUALIFIED REFERENCE
HASITEM - THIS ROUTINE DETERMINES IF THERE IS AN ITEM
 WITHIN
CHKGRP - CHECK FOR GROUP SEPERATORS OR END OF FILE
SPRINTF
STRLEN

CALLED DIRECTLY BY:

BSCODE - BUILD SUBROUTINE CODE
VISITA - VISIT ARRAYS ON THIS FORM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: WARNING
PURPOSE: ISSUE WARNING MESSAGE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: FLUIERR
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS

VOID WARNING(S, A, B, C, D, E, F)
CHAR *S, *A, *B, *C, *D, *E, *F;

DESCRIPTION

PRINTS A WARNING MESSAGE ON STDERR

ARGUMENTS:

S = CHAR *
A = CHAR *
B = CHAR *
C = CHAR *
D = CHAR *
E = CHAR *
F = CHAR *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS

ROUTINES CALLED:

PMSGLS
STRLEN
SPRINTF

CALLED DIRECTLY BY:

GENAT - GENERATE ACTION SIGNAL
CHKFRM - CHECK FORM
YYLEX - LEXICAL ANALYZER FOR FLAN
YYPARSE - FLAN PARSER

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: WINRSV
PURPOSE: WINDOW RESOLVE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: RWSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: RW
DOCUMENTATION GROUP: RW/AP

DESCRIPTION:

SYNOPSIS
WINRSV()

DESCRIPTION

ALL FORMS WHICH ARE PRESENTED IN WINDOWS ARE ADDED TO
THOSE WINDOWS
SO QUALIFIED NAMES MAY BE RESOLVED INTO POINTERS.

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
RW - REPORT WRITER DEFINITIONS

ROUTINES CALLED:

MALLOC
GETPTH - GET PATH
FNDFRM - FIND FORM
COPFLD
FREE

CALLED DIRECTLY BY:

RWOPN - REPORT WRITER OPEN FORMS

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: WRTEXP
PURPOSE: WRITE EXPRESSION
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

----- SYNOPSIS

CHAR *WRTEXP(EP)
ENODE *EP;

INPUTS:

EP - EXPRESSION TO WRITE

OUTPUTS:

RETURNS A POINTER TO THE WRITTEN EXPRESSION OR NULL
FOR ERRORS

DESCRIPTION

RETURNS A POINTER TO THE CHARACTER STRING REPRESENTING
THE GIVEN
EXPRESSION, OR NULL IF AN ERROR IS DETECTED.

ARGUMENTS:

EP = ENODE *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

FREE
WRTEXP - WRITE EXPRESSION
MEMCPY
MYALLOC - MY MALLOC
STRLEN
SPRINTF

CALLED DIRECTLY BY:

CHKFLD	- CHECK FIELD
WRTEXP	- WRITE EXPRESSION

USED IN MAIN PROGRAM(S):

GRP/MAIN	- GENERATE APPLICATION/REPORT PROGRAM
----------	---------------------------------------

REPORT WRITER Module Documentation

NAME: WRTFRM
PURPOSE: WRITE FORM
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: WRTFRM
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FP
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS

```
CHAR *WRTFRM(FP)
    FIELD *FP;
```

INPUTS:

FP - POINTER TO FORM TO WRITE OUT

DESCRIPTION

WRITES THE SPECIFIED FORM INTO A .FD FILE.

ARGUMENTS:

OPNPTR = FIELD *

INCLUDE FILES:

```
STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
FFFV2 - FORM FILE FORMAT - VERSION 2
```

ROUTINES CALLED:

```
SPRINTF
FOPEN
SYSMSG
FWRITE
FCLOSE
WRTFRM/WRTTXT - WRITE TEXT
WRTFRM/WRTFLD - WRITE FIELD
WRTFRM/WRTTBF - WRITE TEXT BUFFER
WRTFRM/TBFCLOS - TEXT BUFFER CLOSE
WRTFRM/WRTDBF - WRITE DEFAULT BUFFER
WRTFRM/DBFCLOS - DEFAULT BUFFER CLOSE
```


STRASN
STRCPY
STRLEN

CALLED DIRECTLY BY:

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: WRTFRM/DBFCLOS
PURPOSE: DEFAULT BUFFER CLOSE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: WRTFRM
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FP
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

----- SYNOPSIS

```
DBFCLOS(FPTR, I, LINE)
FILE *FPTR;
INT I;
CHAR LINE[81];
```

DESCRIPTION

WRITES THE LAST LINE OF THE DEFAULT LINE BUFFER.

ARGUMENTS:

FPTR = FILE *
I = INT
LINE = CHAR [81]

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
FFFV2 - FORM FILE FORMAT - VERSION 2

ROUTINES CALLED:

FWRITE

CALLED DIRECTLY BY:

WRTFRM - WRITE FORM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: WRTFRM/FORMAT
PURPOSE: INSERT FORMAT CODES
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: WRTFRM
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FP
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

----- SYNOPSIS

```
FORMAT(FLDREC, FMT1, FMT2)
  FLDREC  *FLDREC;
  CHAR    FMT1, FMT2;
```

DESCRIPTION

INSERTS THE SPECIFIED FORMAT INTO THE SPECIFIED FIELD
RECORD.

ARGUMENTS:

FLDREC = FLDREC *
FMT1 = CHAR
FMT2 = CHAR

INCLUDE FILES:

STDTyp - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
FFV2 - FORM FILE FORMAT - VERSION 2

CALLED DIRECTLY BY:

WRTFRM/WRTFLD - WRITE FIELD

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: WRTFRM/TBFCLOS
PURPOSE: TEXT BUFFER CLOSE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: WRTFRM
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FP
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS

```
TBFCLOS(FPTR, I, LINE)
    FILE *FPTR;
    INT I;
    CHAR LINE[];
```

DESCRIPTION

WRITES THE LAST LINE OF THE TEXT LINE BUFFER.

ARGUMENTS:

```
FPTR = FILE *
I = INT
LINE = CHAR []
```

INCLUDE FILES:

```
STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
FFV2 - FORM FILE FORMAT - VERSION 2
```

ROUTINES CALLED:

FWRITE

CALLED DIRECTLY BY:

WRTFRM - WRITE FORM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: WRTFRM/WRTDBF
PURPOSE: WRITE DEFAULT BUFFER
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: WRTFRM
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FP
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS

```
INT WRTDBF(FPTR, FLDPTR, I, LINE)
FILE *FPTR;
FIELD *FLDPTR;
INT I;
CHAR LINE[81];
```

DESCRIPTION

COPIES THE SPECIFIED FIELD DEFAULT VALUE INTO THE DEFAULT
VALUE LINE
BUFFER STARTING AT THE SPECIFIED POSITION AND WRITING THE
LINE BUFFER
WHEN FULL. RETURNS THE NEXT POSITION TO USE.

ARGUMENTS:

```
FPTR = FILE *
FLDPTR = FIELD *
I = INT
LINE = CHAR [81]
```

INCLUDE FILES:

```
STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
FFV2 - FORM FILE FORMAT - VERSION 2
```

ROUTINES CALLED:

FWRITE

CALLED DIRECTLY BY:

WRTFRM - WRITE FORM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: WRTFRM/WRTFLD
PURPOSE: WRITE FIELD
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: WRTFRM
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FP
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

----- SYNOPSIS

WRTFLD(FPTR, FLDPTR)
FILE *FPTR;
FIELD *FLDPTR;

DESCRIPTION

WRITES THE FIELD RECORD FOR THE SPECIFIED FIELD STRUCTURE.

ARGUMENTS:

FPTR = FILE *
FLDPTR = FIELD *

INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
FFFV2 - FORM FILE FORMAT - VERSION 2

ROUTINES CALLED:

FWRITE
STRCPY
WRTFRM/FORMAT - INSERT FORMAT CODES
STRNCPY
MEMCPY

CALLED DIRECTLY BY:

WRTFRM - WRITE FORM

PS 620344501
30 September 1990

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: WRTFRM/WRTTBF
PURPOSE: WRITE TEXT BUFFER
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: WRTFRM
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FP
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS

```
INT WRTTBF(FPTR, TXTPTR, I, LINE)
    FILE *FPTR;
    TEXT *TXTPTR;
    CHAR LINE[81];
    INT I;
```

DESCRIPTION

COPIES THE SPECIFIED TEXT INTO THE TEXT LINE BUFFER
STARTING AT THE
SPECIFIED POSITION AND WRITING THE LINE BUFFER WHEN FULL.
RETURNS THE
NEXT POSITION TO USE.

ARGUMENTS:

```
FPTR = FILE *
TXTPTR = TEXT *
I = INT
LINE = CHAR [81]
```

INCLUDE FILES:

```
STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
FFV2 - FORM FILE FORMAT - VERSION 2
```

ROUTINES CALLED:

FWRITE

CALLED DIRECTLY BY:

WRTFRM - WRITE FORM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: WRTFRM/WRTTXT
PURPOSE: WRITE TEXT
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: WRTFRM
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FP
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

----- SYNOPSIS

WRTTXT(FPTR, TXTPTR)
FILE *FPTR;
TEXT *TXTPTR;

DESCRIPTION

WRITES THE TEXT RECORD FOR THE SPECIFIED TEXT STRUCTURE.

ARGUMENTS:

FPTR = FILE *
TXTPTR = TEXT *

INCLUDE FILES:

STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
FFV2 - FORM FILE FORMAT - VERSION 2

ROUTINES CALLED:

FWRITE
STRLEN

CALLED DIRECTLY BY:

WRTFRM - WRITE FORM

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: YYLEX
PURPOSE: LEXICAL ANALYZER FOR FLAN
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: YTAB
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS
INT LEX()

OUTPUTS:
SETS YYLVAL TO THE TOKEN VALUE (IF ANY)
RETURN THE TOKEN TYPE

DESCRIPTION
RECOGNIZES TOKENS (KEYWORDS, IDENTIFIERS, NUMBERS, ETC.),
SETS YYLVAL,
AND RETURNS THE APPROPRIATE TOKEN TYPE.

INCLUDE FILES:

FLAN.Y" - **** PURPOSE NOT FOUND BY STRIPPER ****
STDYYP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
MATH - **** PURPOSE NOT FOUND BY STRIPPER ****

ROUTINES CALLED:

GETC
ERROR - ISSUE ERROR MESSAGE
ISALNUM
ISDIGIT
FATAL - ISSUE FATAL ERROR MESSAGE
UNGETC
WARNING - ISSUE WARNING MESSAGE
STRCMP
CSTASH - CHARACTER STASH
ATOF
ISALPHA

TOUPPER
ATOI
ISSPACE

CALLED DIRECTLY BY:

YYPARSE - FLAN PARSER

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

REPORT WRITER Module Documentation

NAME: YYPARSE
PURPOSE: FLAN PARSER
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: YTAB
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

DESCRIPTION
DEFINITION LANGUAGE GRAMMAR.

INCLUDE FILES:

FLAN.Y" - **** PURPOSE NOT FOUND BY STRIPPER ****
STDYTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPPARM - FORM PROCESSOR PARAMETERS
RW - REPORT WRITER DEFINITIONS
MATH - **** PURPOSE NOT FOUND BY STRIPPER ****

ROUTINES CALLED:

PRINTF
STRUPC
STRNCPY
FREE
STRCAT
MYALLOC - MY MALLOC
MEMCPY
MAKACT - MAKE ACTION LIST ELEMENT
MAKINT - MAKE EXPRESSION INTO AN INTEGER
STRCMP
STRLEN
WARNING - ISSUE WARNING MESSAGE
SPRINTF
MKPOS - MAKE POSITION NODE
FATAL - ISSUE FATAL ERROR MESSAGE
STRCPY
CHKFLD - CHECK FIELD
CHKFRM - CHECK FORM
STRCHR
ERROR - ISSUE ERROR MESSAGE

MAKSTR - MAKE EXPRESSION INTO A STRING
CSTASH - CHARACTER STASH
GFLDPT - GET FIELD POINTER
MAKFLD
FNDATT - FIND ATTRIBUTE
YYERROR
YYLEX - LEXICAL ANALYZER FOR FLAN

CALLED DIRECTLY BY:

FLANCI - FLAN CALLABLE INTERFACE

USED IN MAIN PROGRAM(S):

GRP/MAIN - GENERATE APPLICATION/REPORT PROGRAM

3.10.9 Include File Descriptions

The following list contains a purpose and description of each include file listed in 3.10.4 as specified in the source code. The language it is written in is also given.

REPORT WRITER Include File Description

FILE NAME: CHART
PURPOSE: CHART INCLUDE FILE
LANGUAGE: C

DESCRIPTION:

DESCRIPTION
GLOBAL DECLARATIONS FOR CHART.

REPORT WRITER Include File Description

FILE NAME: CTLCHR
PURPOSE: CONTROL CHARACTERS
LANGUAGE: C

DESCRIPTION:

DESCRIPTION
DEFINITIONS OF ALL CONTROL CHARACTERS TO AVOID CHARACTER
SET
DEPENDENCIES.

REPORT WRITER Include File Description

FILE NAME: ERRPRO
PURPOSE: PROCESS ERROR INCLUDE FILE
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

REPORT WRITER Include File Description

FILE NAME: FFFV2
PURPOSE: FORM FILE FORMAT - VERSION 2
LANGUAGE: C

DESCRIPTION:

DESCRIPTION
RECORD LAYOUTS FOR THE BINARY FORM DEFINITION FILE

REPORT WRITER Include File Description

FILE NAME: FLAN
PURPOSE: FLAN INTERNAL STRUCTURES
LANGUAGE: C

DESCRIPTION:

DESCRIPTION
AUXILIARY DATA STRUCTURES USED BY FLAN.

REPORT WRITER Include File Description

FILE NAME: FPCODE
PURPOSE: FORM PROCESSOR RETURN CODES
LANGUAGE: C

DESCRIPTION:

REPORT WRITER Include File Description

FILE NAME: FPD
PURPOSE: FORM PROCESSOR DATA
LANGUAGE: C

DESCRIPTION:

DESCRIPTION
DATA DEFINITIONS FOR ALL FORM PROCESSOR (INCLUDING
MONITER) DATA.

REPORT WRITER Include File Description

FILE NAME: FPDINI
PURPOSE: FPD INITIALIZATION
LANGUAGE: C

DESCRIPTION:

DESCRIPTION
INITIALIZED VERSION OF UID FOR INCLUSION IN MAIN PROGRAM.

REPORT WRITER Include File Description

FILE NAME: FPPARM
PURPOSE: FORM PROCESSOR PARAMETERS
LANGUAGE: C

DESCRIPTION:

DESCRIPTION: THESE DATA DEFINITIONS ARE USED
IN THE FORM PROCESSOR ROUTINES.

REPORT WRITER Include File Description

FILE NAME: HRWFRM
PURPOSE: HRW FORM DEFINITION
LANGUAGE: C

DESCRIPTION:

REPORT WRITER Include File Description

FILE NAME: NTM
PURPOSE: NTM INTERFACE INCLUDE FILE
LANGUAGE: C

DESCRIPTION:

DESCRIPTION
INCLUDE FILE FOR NTM INTERFACE

REPORT WRITER Include File Description

FILE NAME: RW
PURPOSE: REPORT WRITER DEFINITIONS
LANGUAGE: C

DESCRIPTION:

DESCRIPTION

REPORT WRITER Include File Description

FILE NAME: SRVRET
PURPOSE: AS THE RETURN GIVEN A TABLE-FULL ERROR
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

MODIFIED 11/2/83 TO INCLUDE RET-CODE-5	*
MODIFIED 1/9/84 TO INCREASE ALL ERROR CODES TO PIC X(5)	*
AND TO ELIMINATE ALPHA'S	*
MODIFIED 1/26/84 TO ADD RET-CODE FOR GETUSR-NOT-SUCC	*
SRV-SUCCESSFUL ADDED FOR GENERIC RETURN	*
MODIFIED 2/7/84 TO ADD ERROR CODES FOR ENTRY-NOT-FOUND	*
MODIFIED 2/8/84 TO ADD WTHST-NOT-SUCCESSFUL	*
MODIFIED 2/20/84 TO ADD TSTMOD NEW CODES.	*
MODIFIED 20 AUG 84 INITIALIZE ALL LOCAL VARIABLES TO SPACES OR 0.	
MODIFIED 5/21/85 TO ADD RCL AND FILGEN RETURN CODES	

REPORT WRITER Include File Description

FILE NAME: STD TYP
PURPOSE: STANDARD TYPE DEFINITIONS
LANGUAGE: C

DESCRIPTION:

DESCRIPTION

THIS FILE ENSURES THAT THE FOLLOWING STANDARD TYPES ARE
AVAILABLE:

FLOAT	- SINGLE PRECISION FLOAT
DOUBLE	- DOUBLE PRECISION FLOAT
LONG	- 32 BIT (OR LARGER) SIGNED INTEGER
LBITS	- 32 BITS (OR MORE) FOR BIT MANIPULATION
INT	- NATURAL SIZE SIGNED INTEGER
UNSIGNED	- NATURAL SIZE UNSIGNED INTEGER
BOOL	- NATURAL SIZE LOGICAL (ZERO / NON-ZERO ONLY)
SHORT	- 16 BIT (OR LARGER) SIGNED INTEGER
USHORT	- 16 BIT (OR LARGER) UNSIGNED INTEGER
BITS	- 16 BITS (OR MORE) FOR BIT MANIPULATION
CHAR	- SINGLE MACHINE CHARACTER (REAL CHARACTERS ALWAYS POSITIVE)
TINY	- 8 BIT (OR LARGER) SIGNED INTEGER
UTINY	- 8 BIT (OR LARGER) UNSIGNED INTEGER
TBITS	- 8 BITS (OR MORE) FOR BIT MANIPULATION
TBOOL	- 8 BIT (OR LARGER) LOGICAL (ZERO / NON-ZERO ONLY)
METACHAR	- 16 BIT (OR LARGER) AUGMENTED CHARACTER (SIGNED)
VOID	- FUNCTION THAT RETURNS NO VALUE
FORTTRAN	- STORAGE CLASS FOR FOREIGN (NON-C) ROUTINES OR C ROUTINES WHICH ARE CALLABLE FROM FOREIGN ROUTINES

SINCE NOT ALL COMPILERS SUPPORT USHORT, TINY, AND UTINY,
THE FUNCTIONS

USHORT(), TINY(), AND UTINY() SHOULD BE USED WHENEVER
REFERENCING THEM.

IN ADDITION, THE FOLLOWING UTILITY MACROS ARE DEFINED:

LURSHIFT(N, B)	- UNSIGNED LONG RIGHT SHIFT
MAX(A, B)	- MAXIMUM OF A AND B
MIN(A, B)	- MINIMUM OF A AND B

REPORT WRITER Include File Description

ABS(A)	- ABSOLUTE VALUE OF A
STRASN(A, B)	- TRANSPORTABLE A = B FOR STRUCTURES
NULL	- NULL POINTER VALUE (0)
TRUE	- 1
FALSE	- 0
SUCCESS	- EXIT(SUCCESS) INDICATES SUCCESSFUL COMPLETION
FAILURE	- EXIT(FAILURE) INDICATES ERRORS

THE FOLLOWING SYMBOLS SHOULD BE DEFINED BASED ON THE
COMPILER BEING USED:

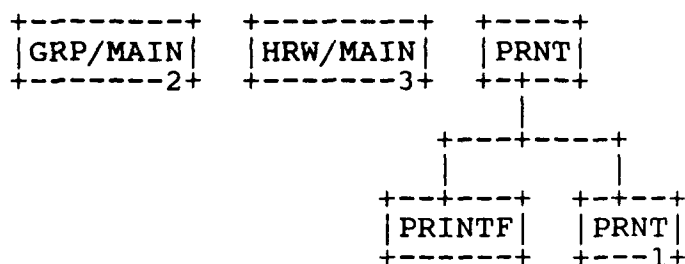
USHORT	- COMPILER SUPPORTS UNSIGNED SHORT
TINY	- COMPILER TREATS CHAR AS SIGNED
UTINY	- CHAR IS SIGNED AND COMPILER SUPPORTS UNSIGNED CHAR
VOID	- COMPILER SUPPORTS VOID
FORTTRAN	- COMPILER SUPPORTS FORTRAN
STRASN	- DEFINE APPROPRIATE MACRO
SUCCESS	- DEFINE APPROPRIATE VALUE IF NOT 0
FAILURE	- DEFINE APPROPRIATE VALUE IF NOT 1

3.10.10 Hierarchy Chart

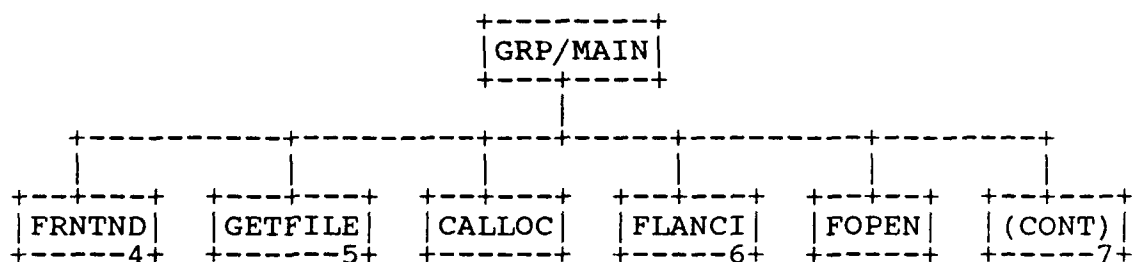
The following hierarchy charts show the relationships between all of the modules mentioned in the preceding section. A module may call a subroutine several times within its code, but the call will only be shown once as a single relationship on this hierarchy chart. All modules shown at the top of the first page are considered Main Programs as described in section 3.10.1 above.

There is an internal paging scheme as marked by the numbers in the upper right corner of each page. An index after the last page of the chart shows where a routine and its calls are first defined. If a routine has no page reference, it either makes no calls or is an external routine. A continuation box on the end of a tree limb shows where that the tree continues on the page numbered mentioned. A number in a box with a routine name points to the page where the routine is further defined within the hierarchy tree. If there is no number in a box, the routine either makes no calls or is an external routine.

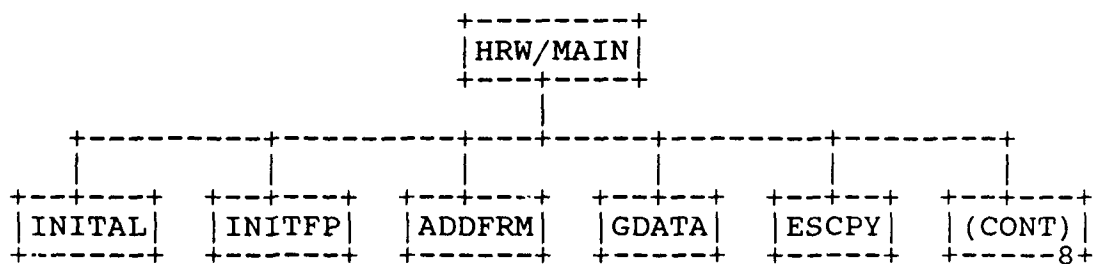
1



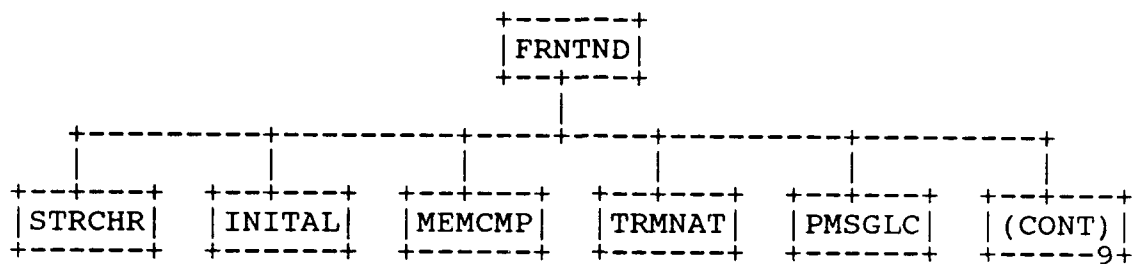
2



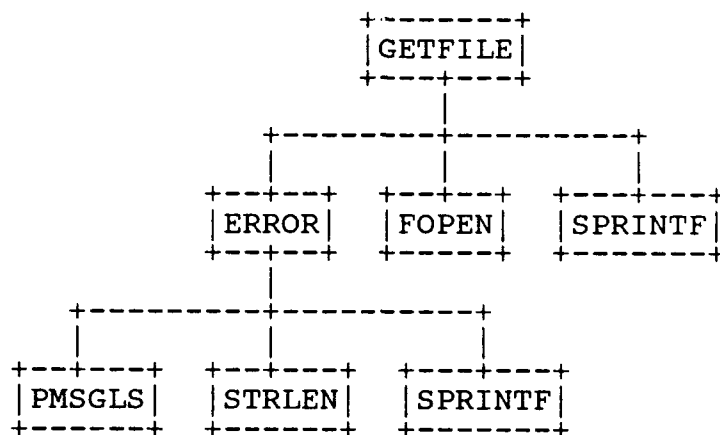
3



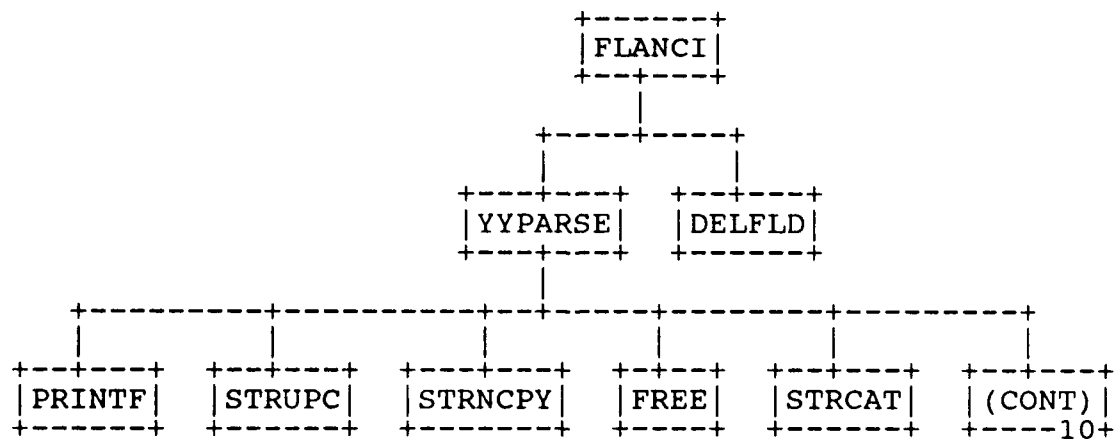
4



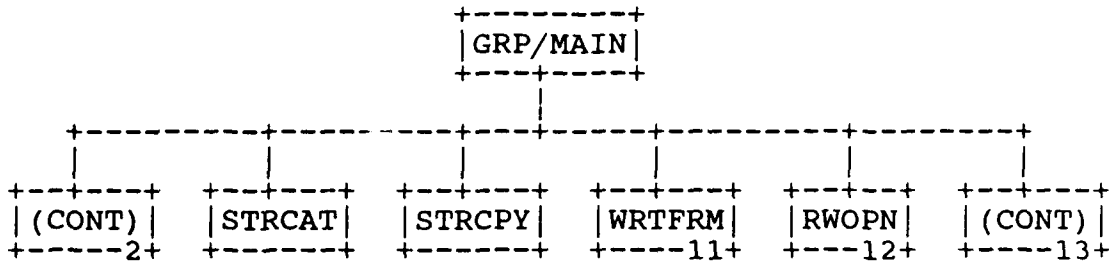
5



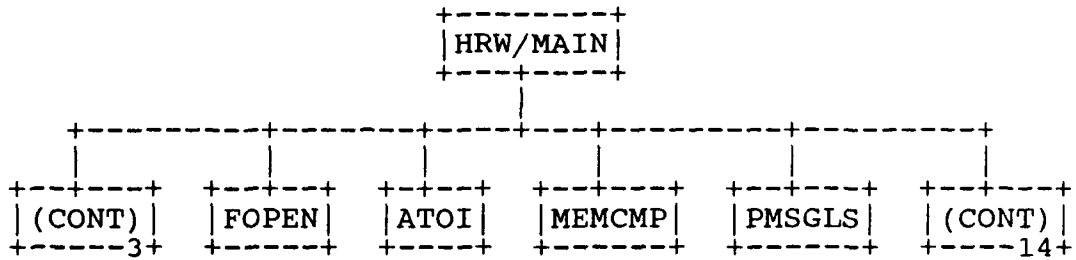
6



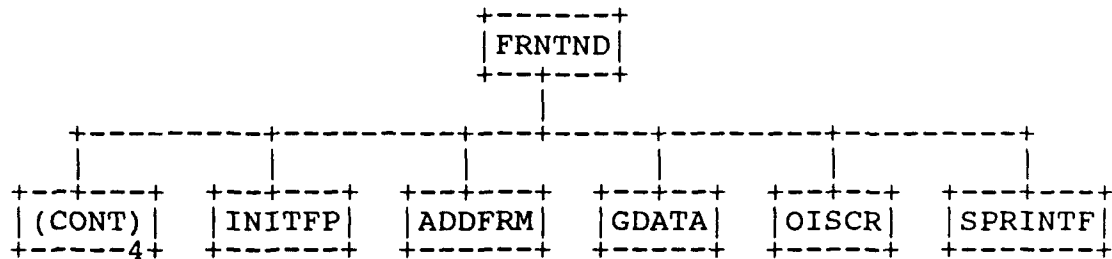
7



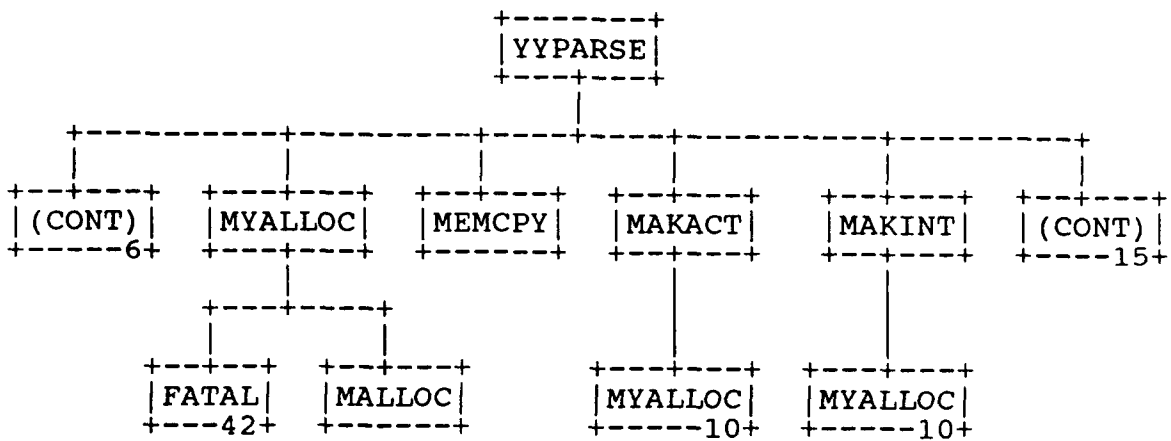
8



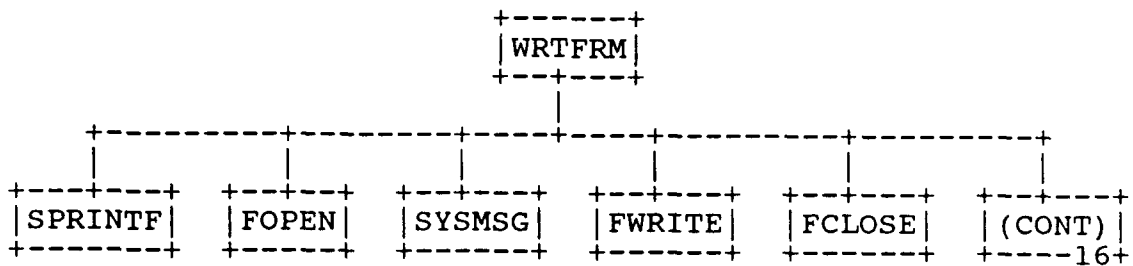
9



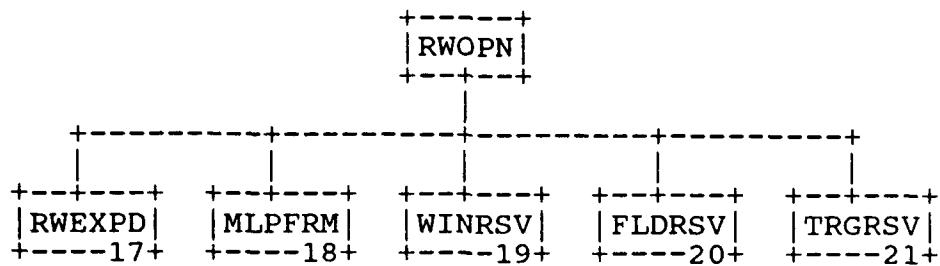
10



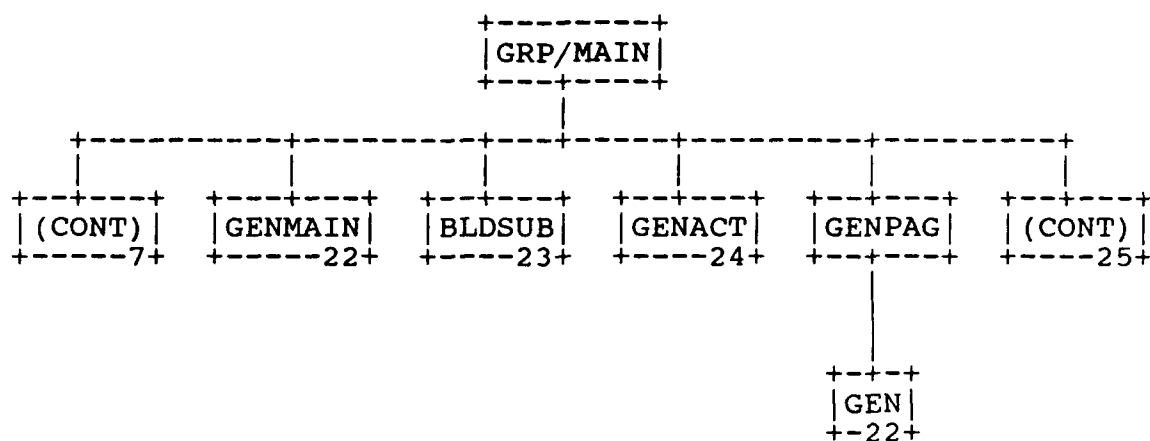
11



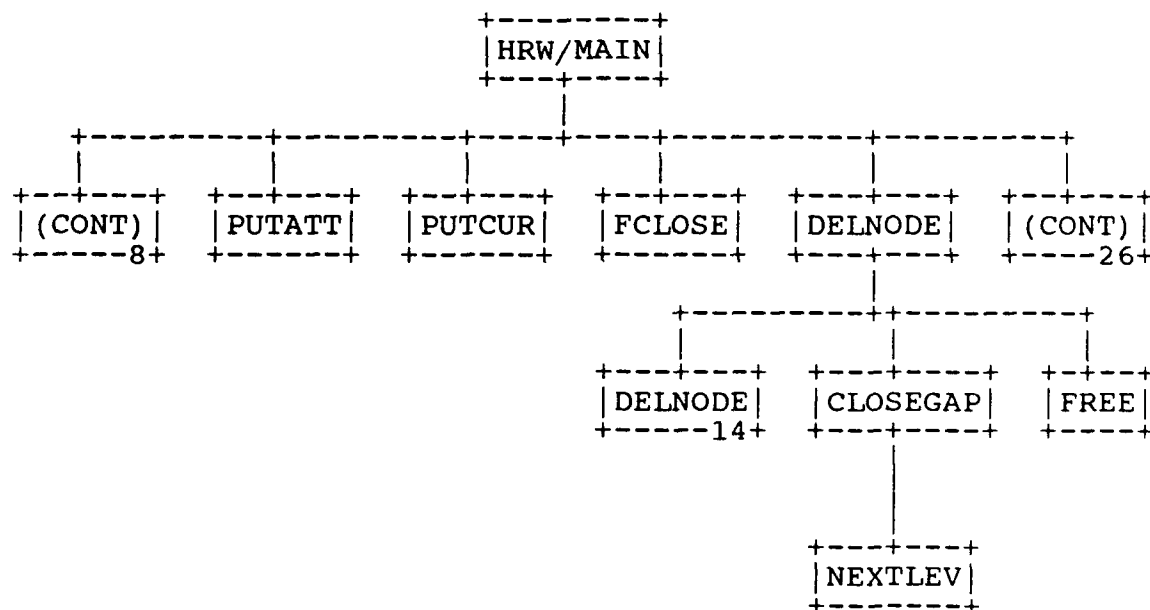
12



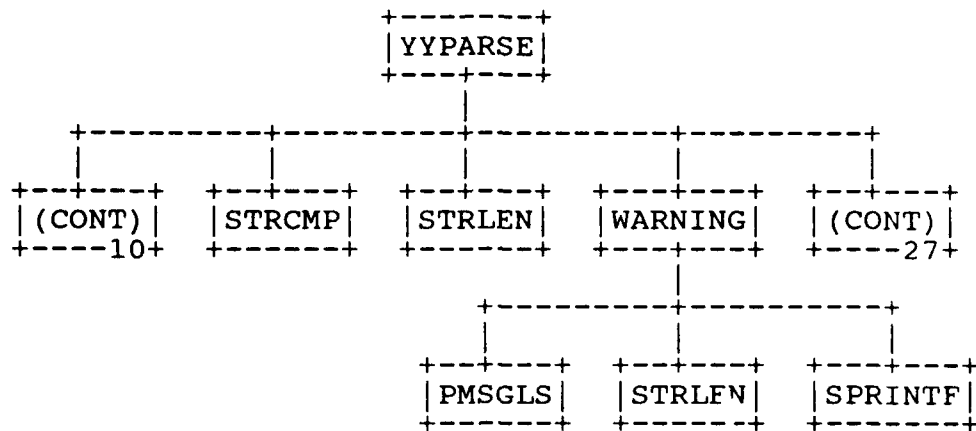
13



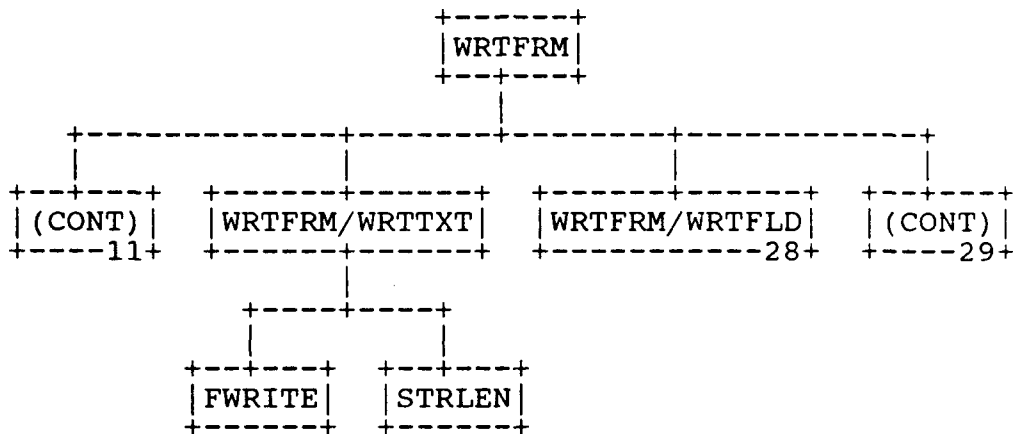
14



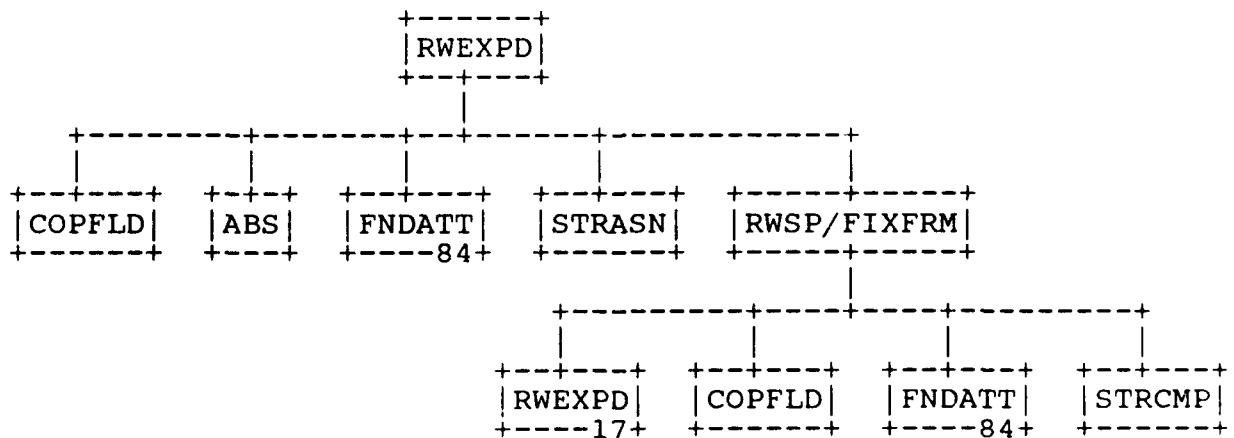
15



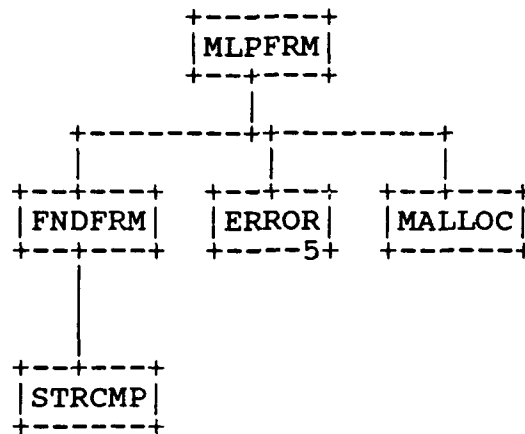
16



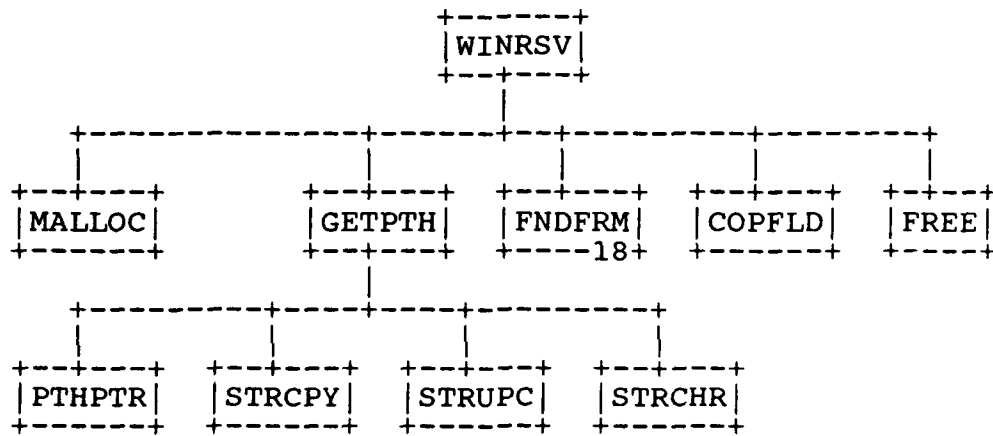
17



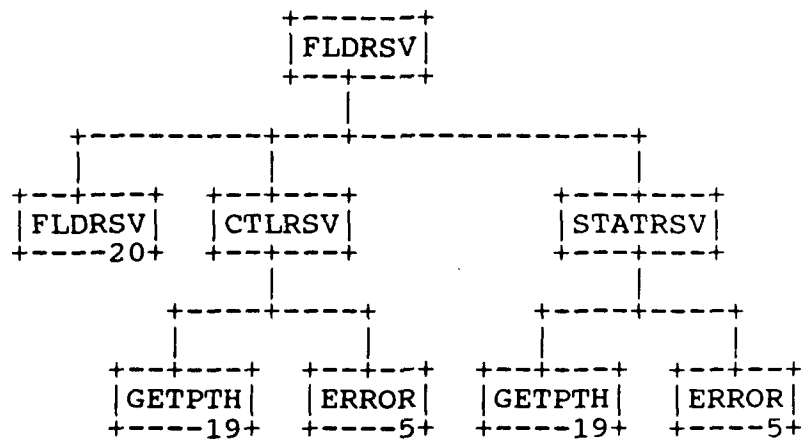
18



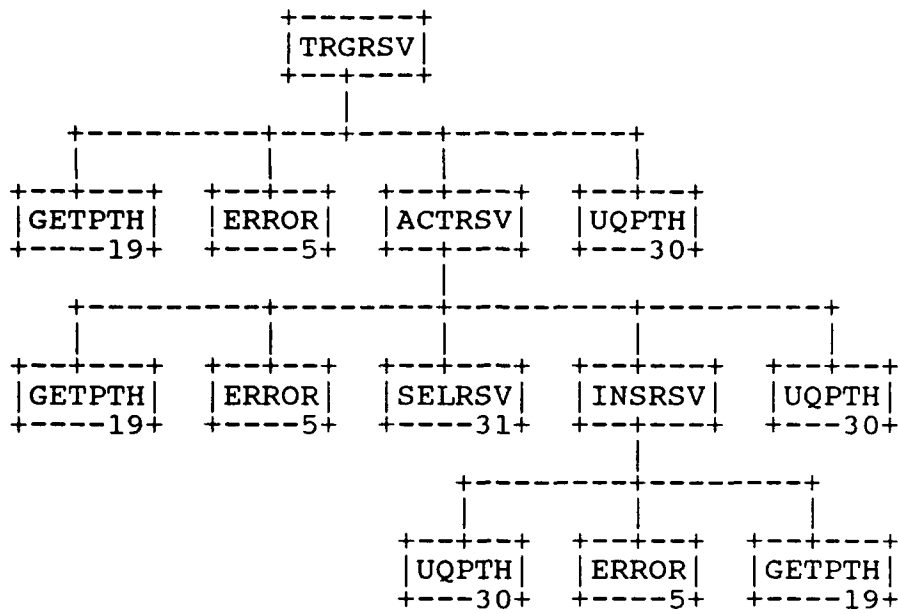
19



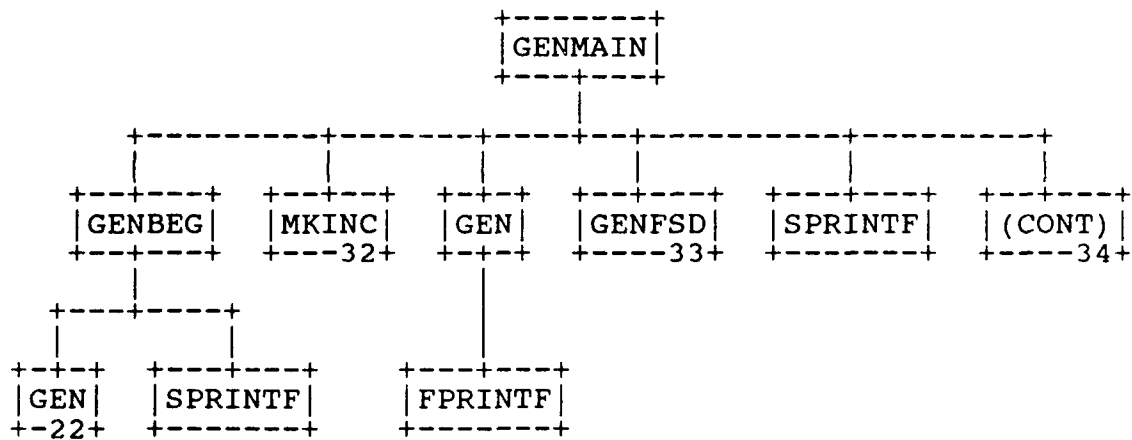
20



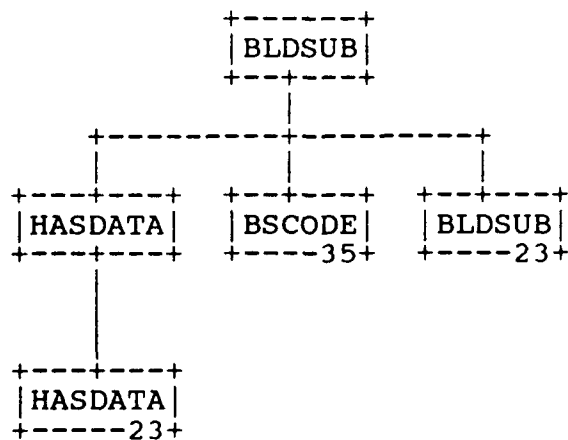
21



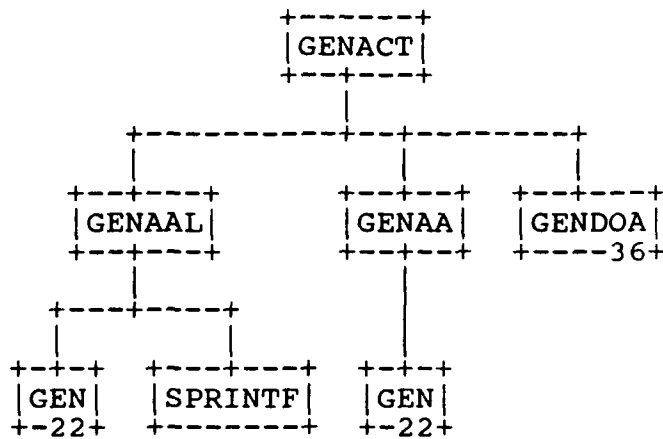
22



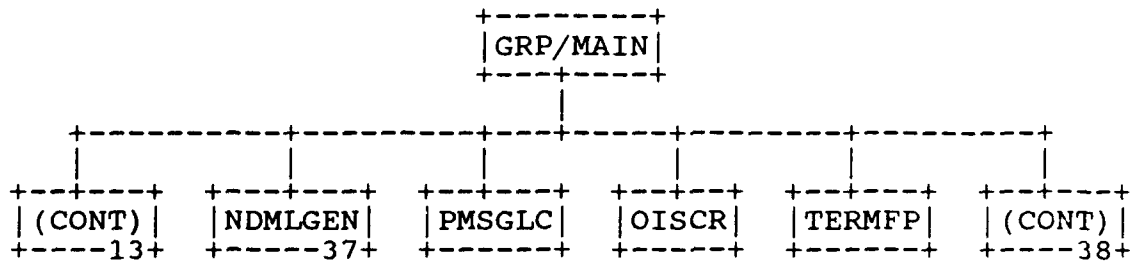
23



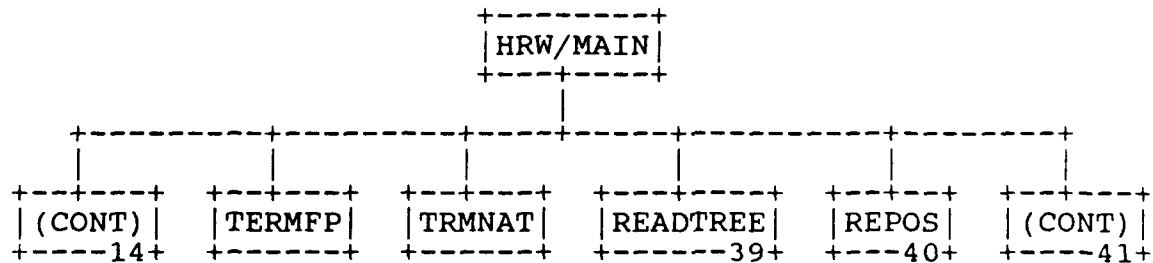
24



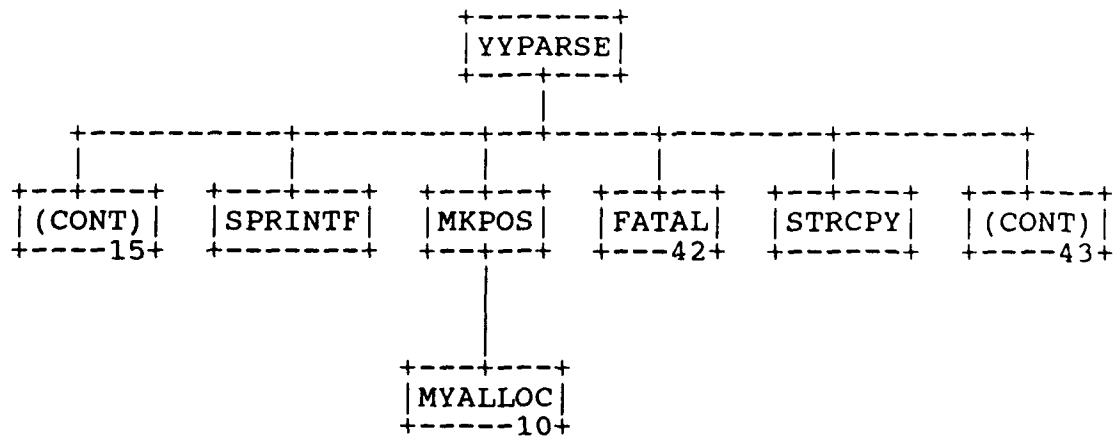
25



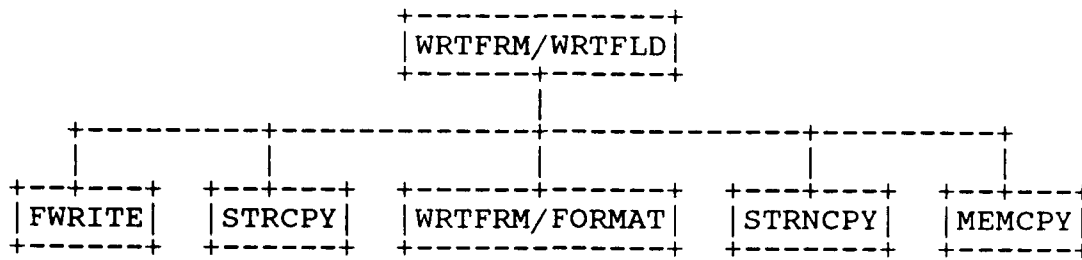
26



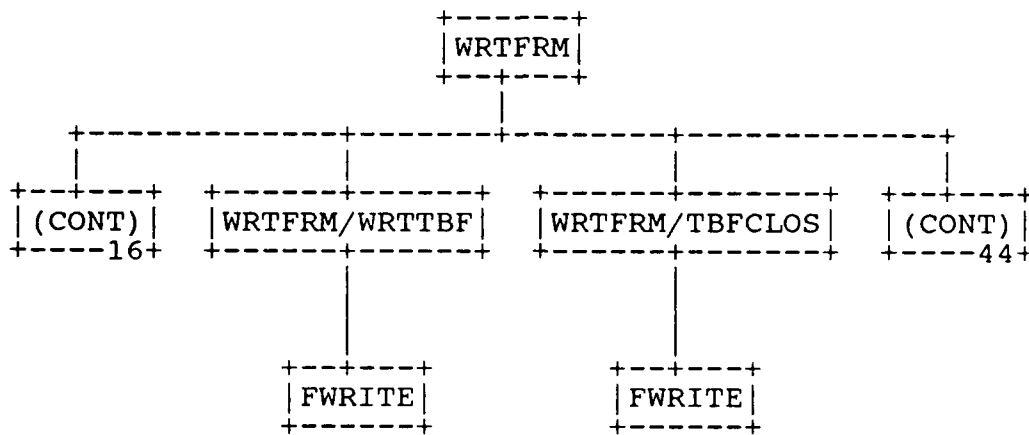
27



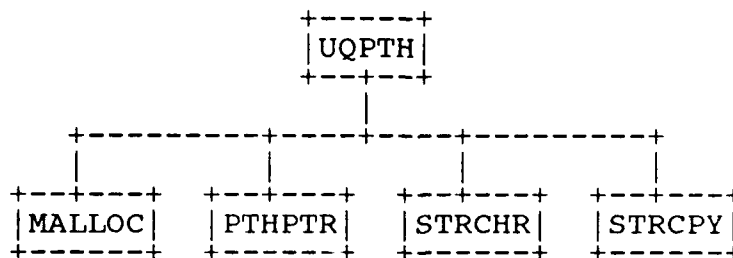
28

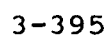


29

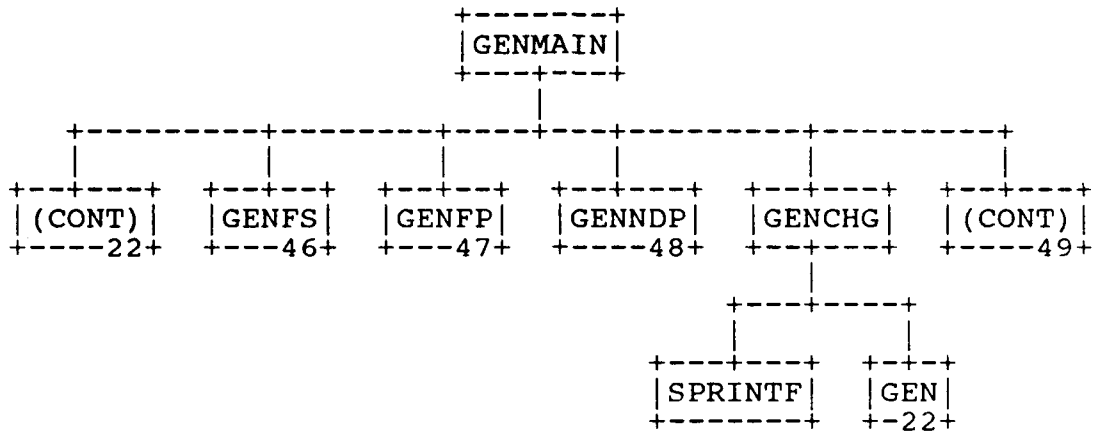


30

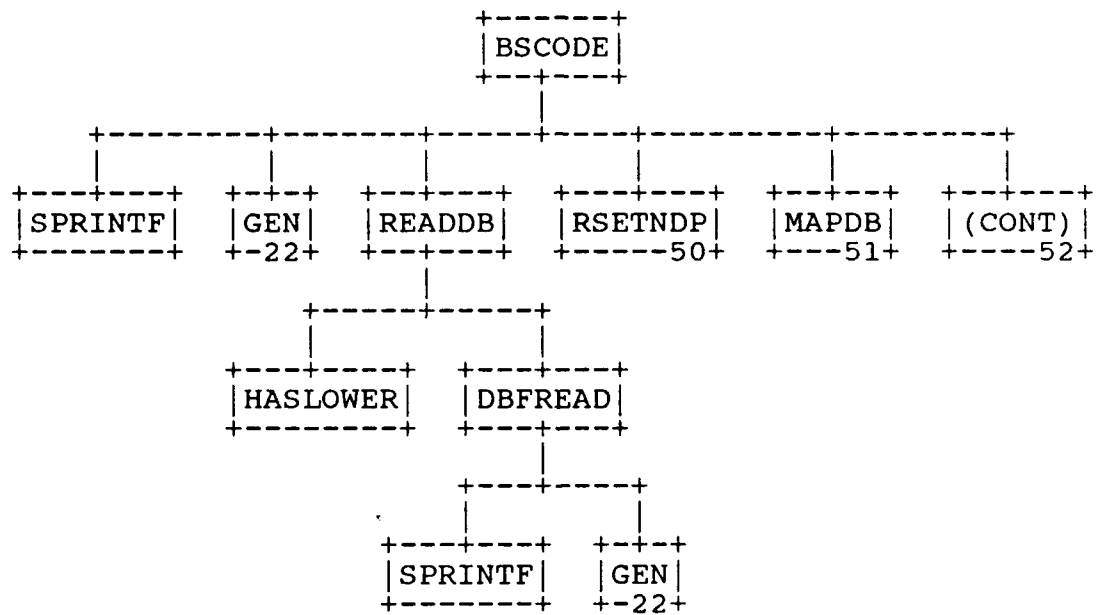




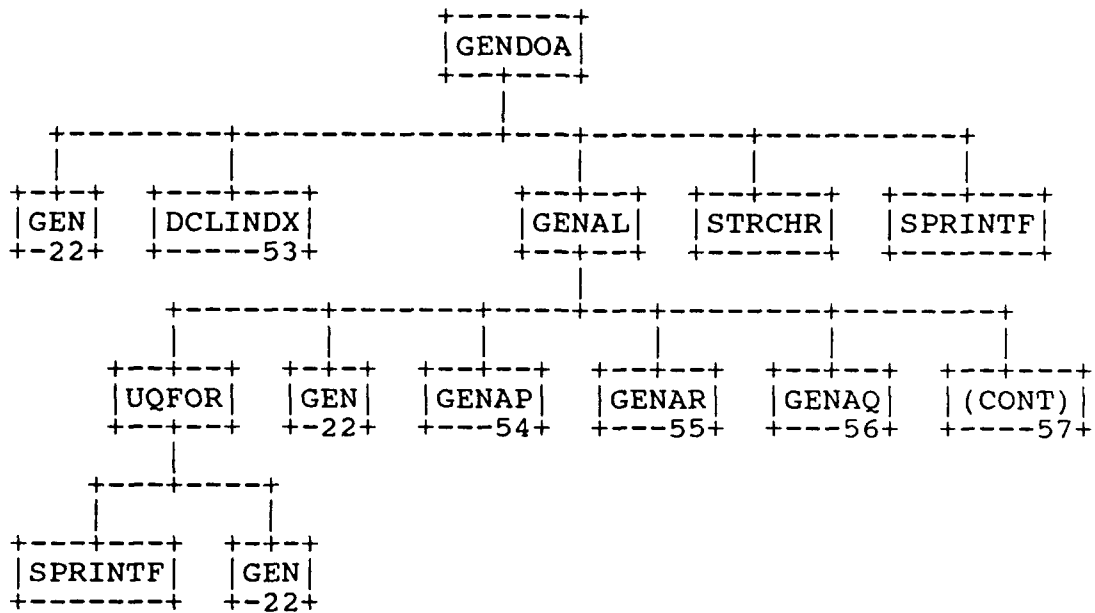
34



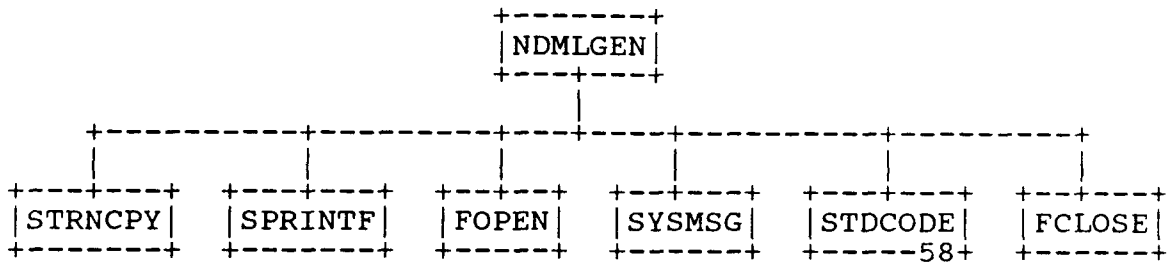
35



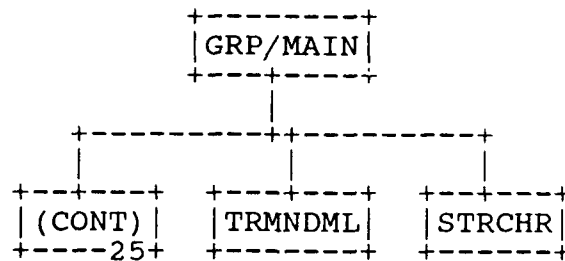
36



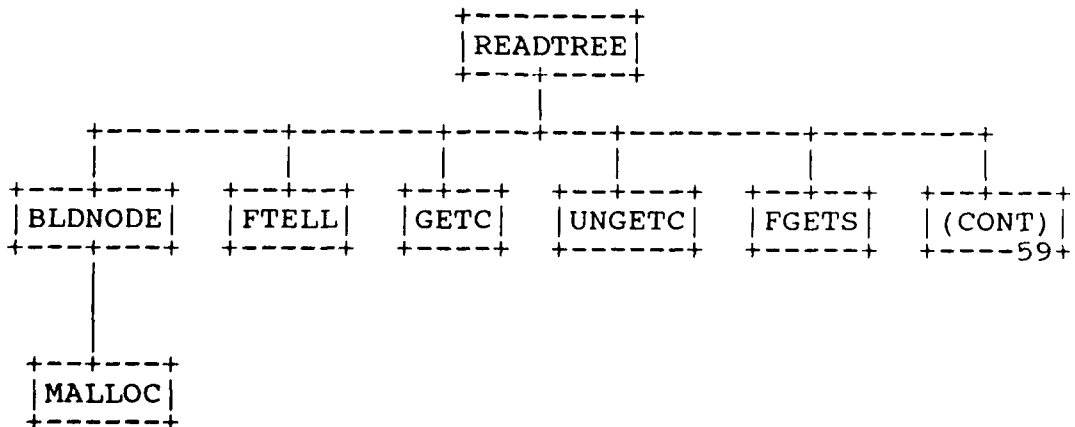
37



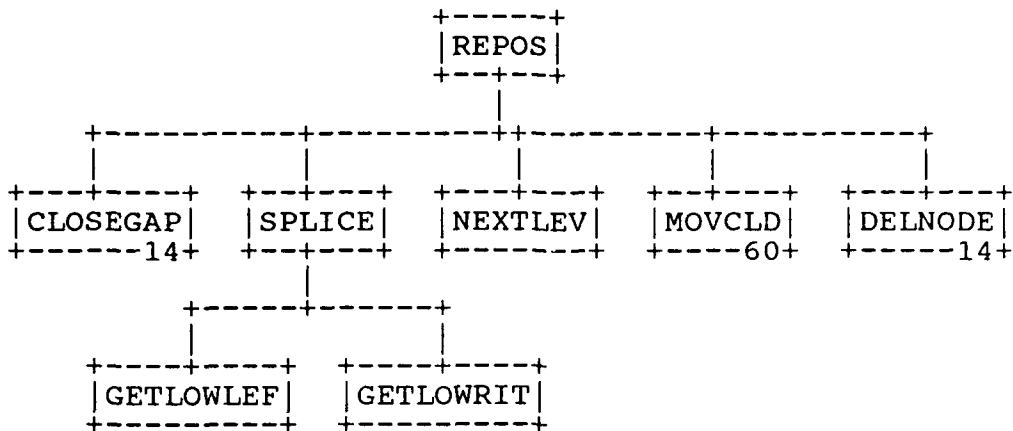
38



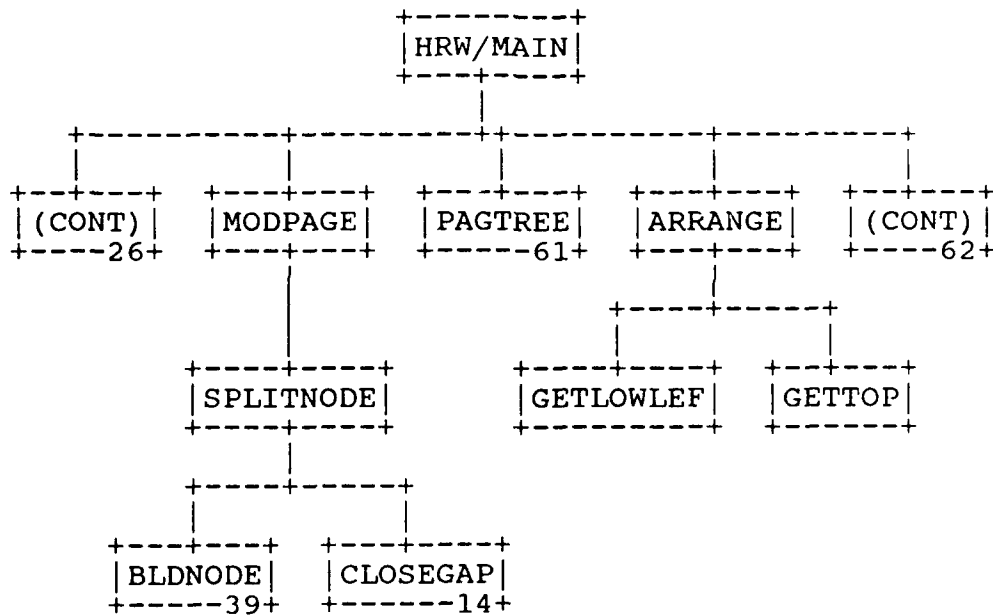
39



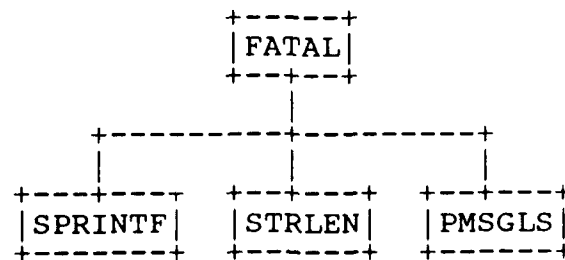
40



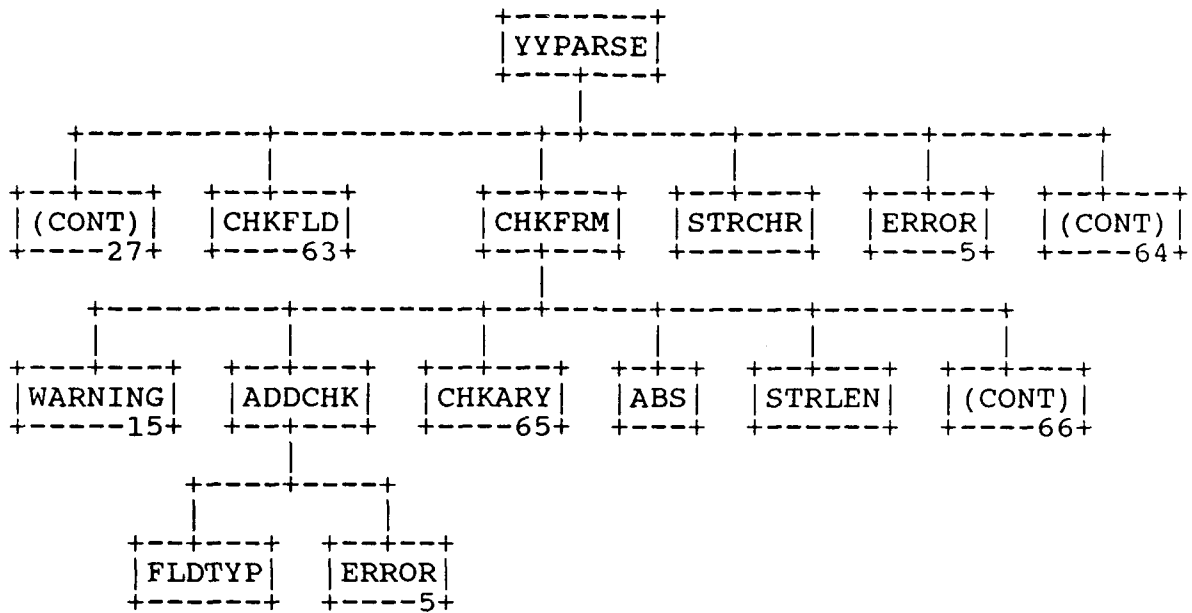
41



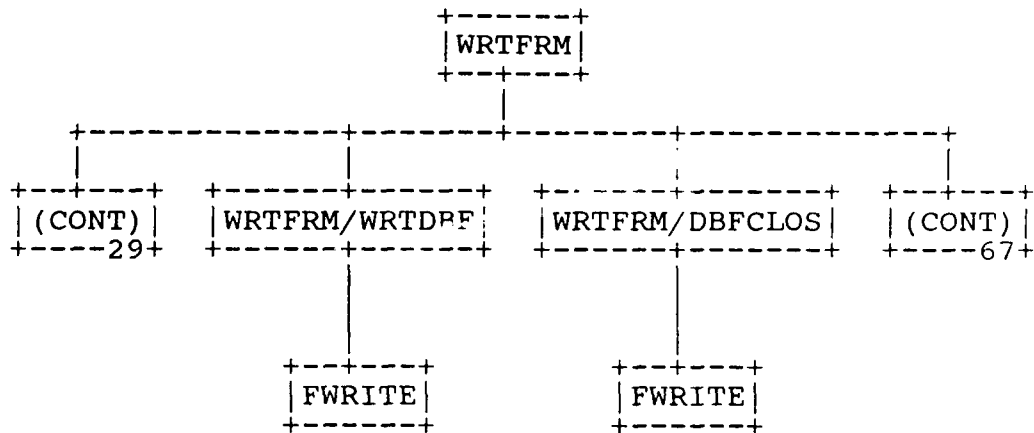
42



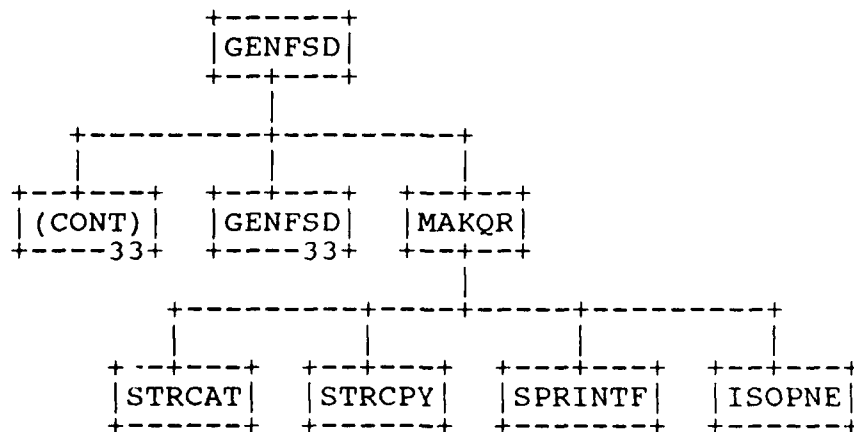
43



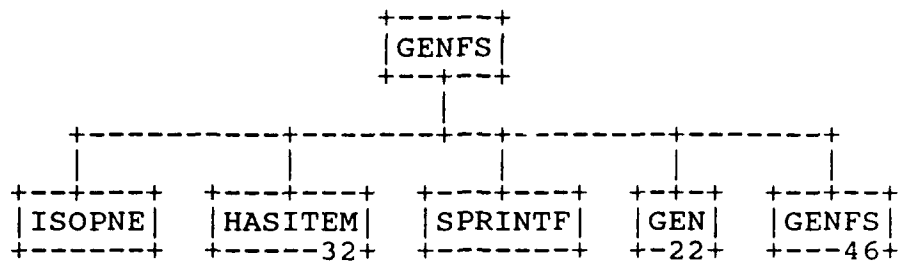
44



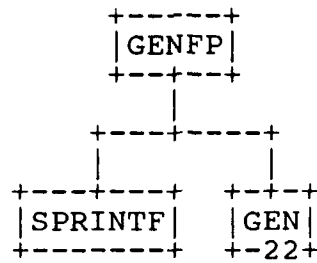
45



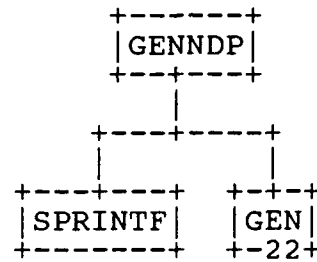
46



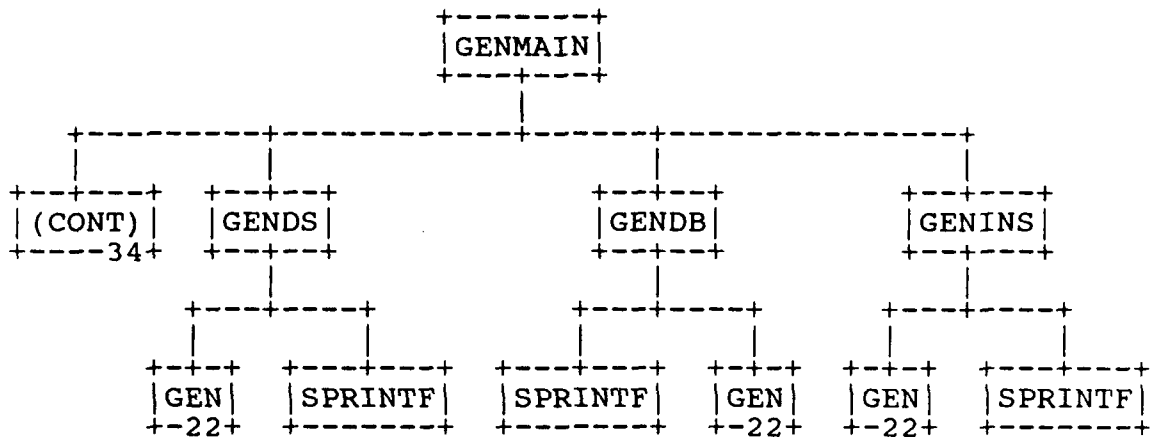
47



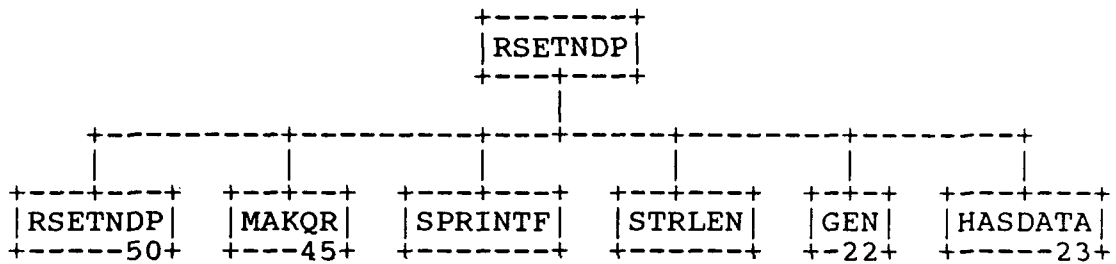
48



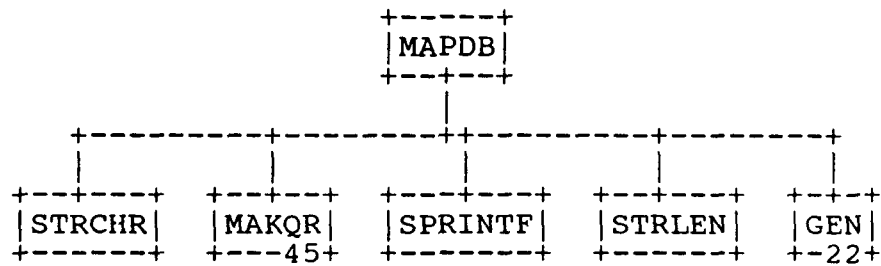
49



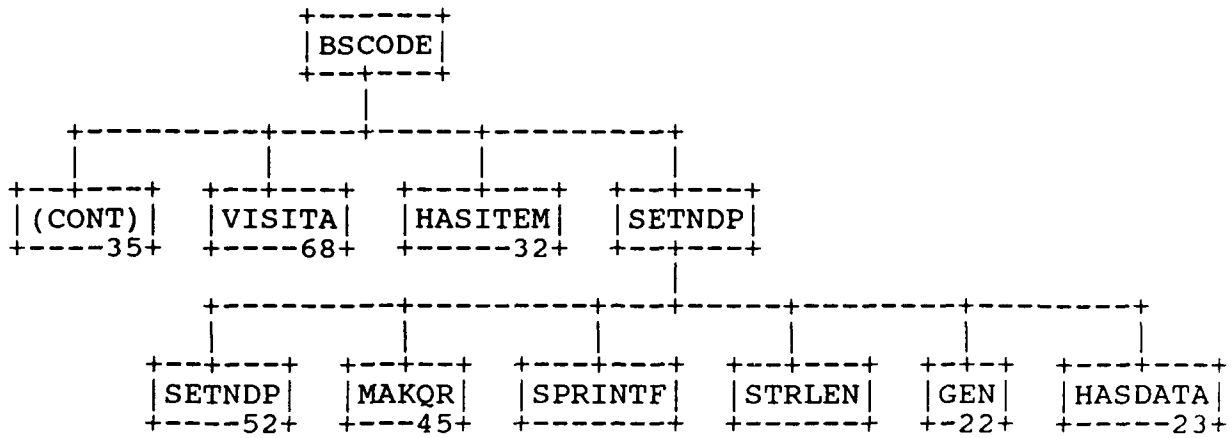
50



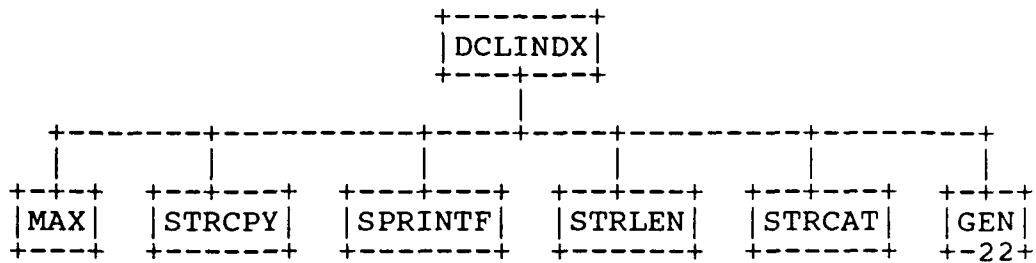
51



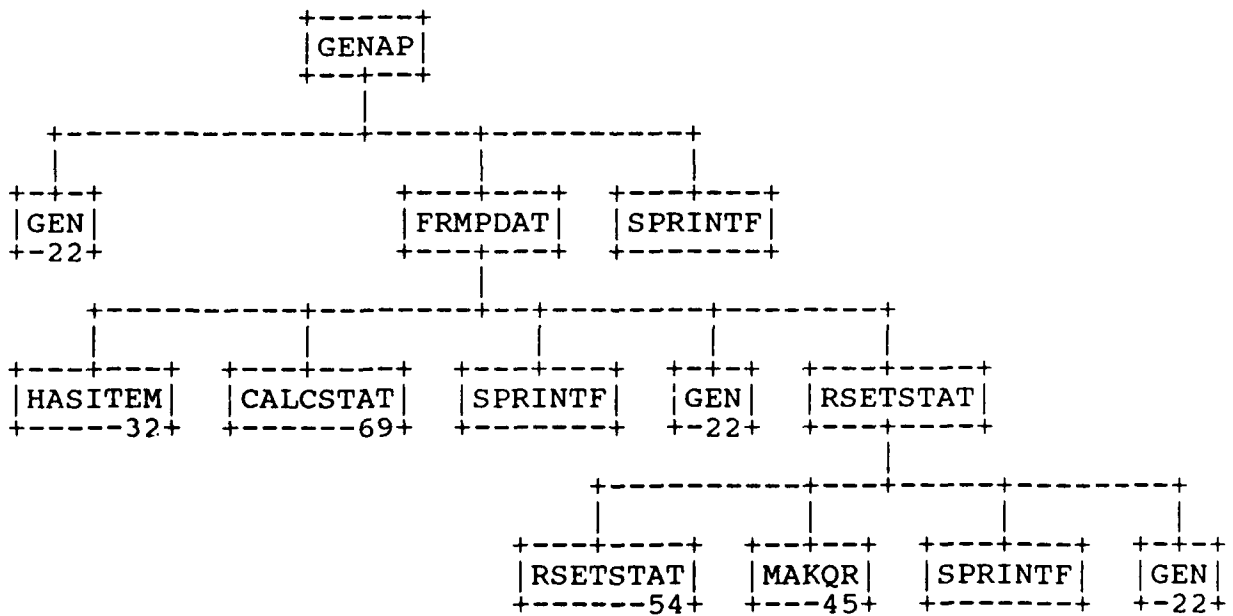
52



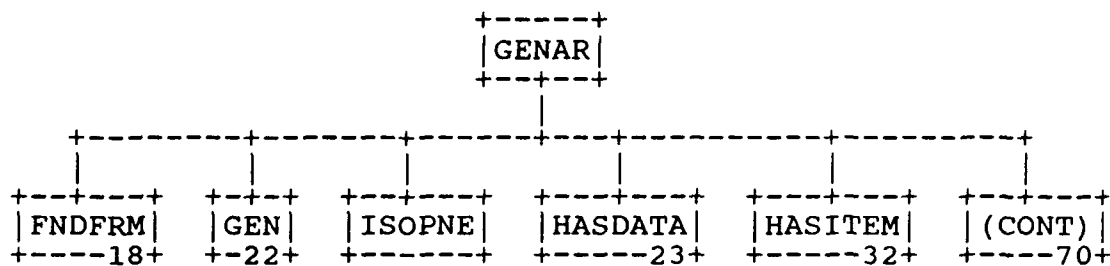
53



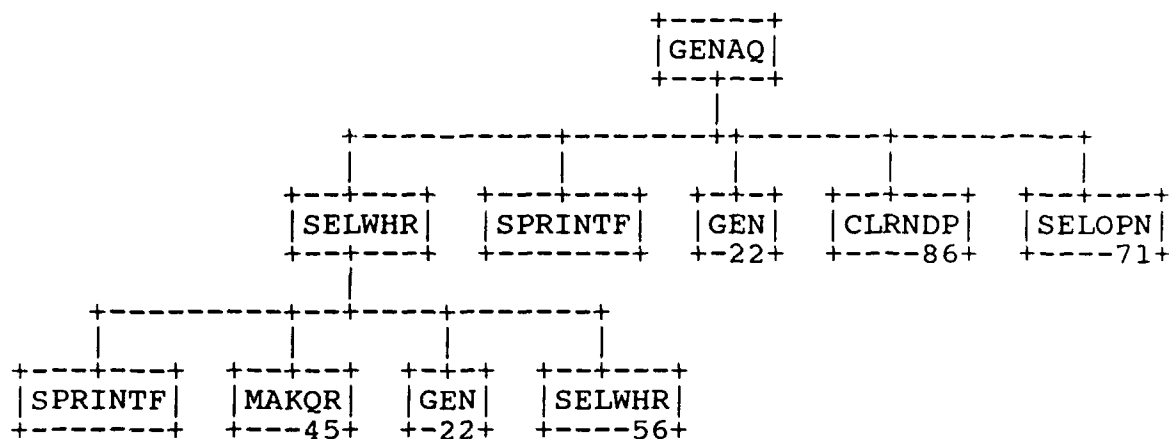
54



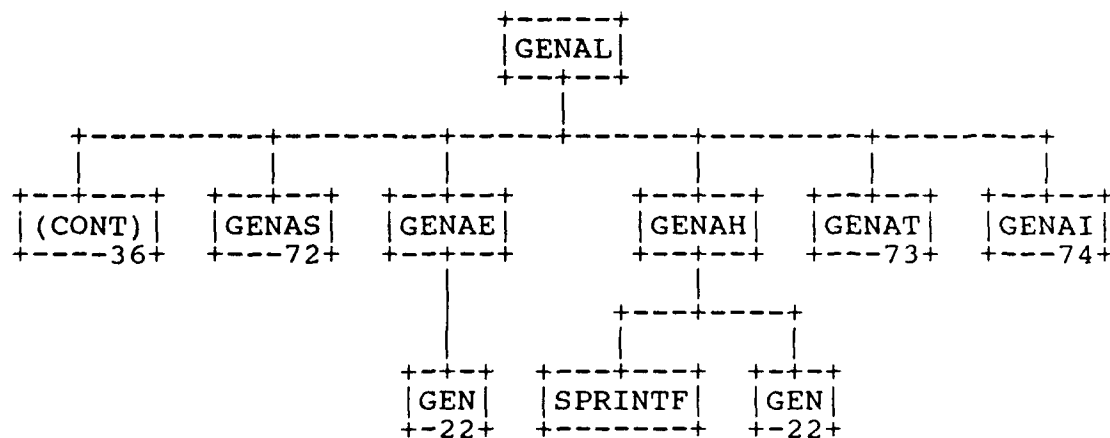
55



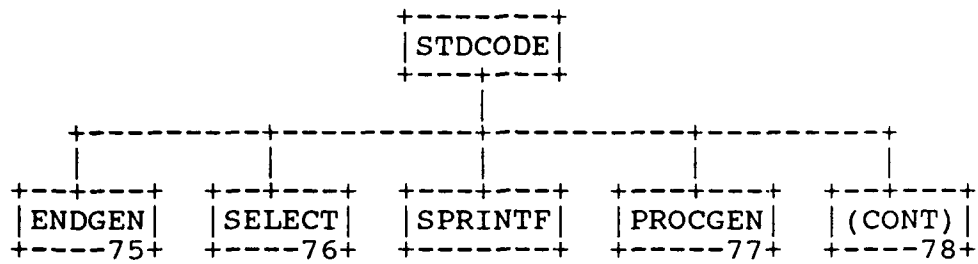
56



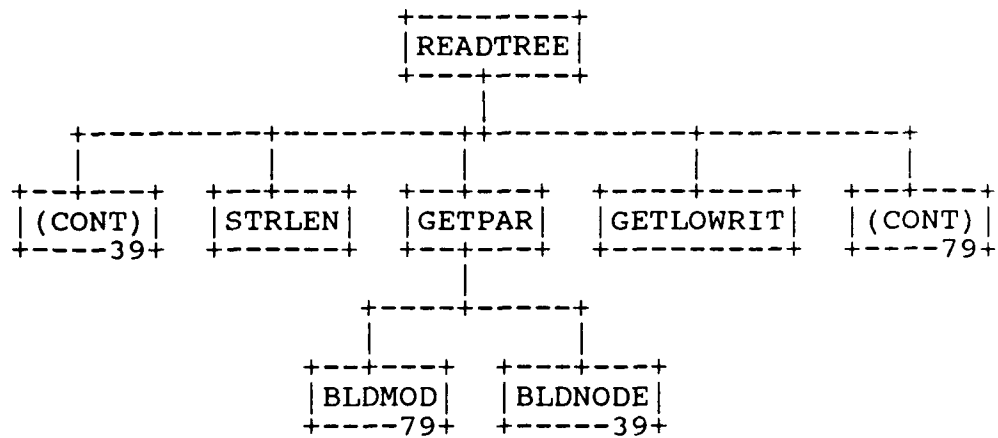
57



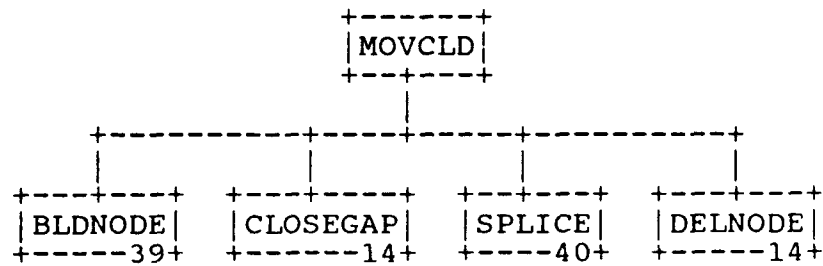
58



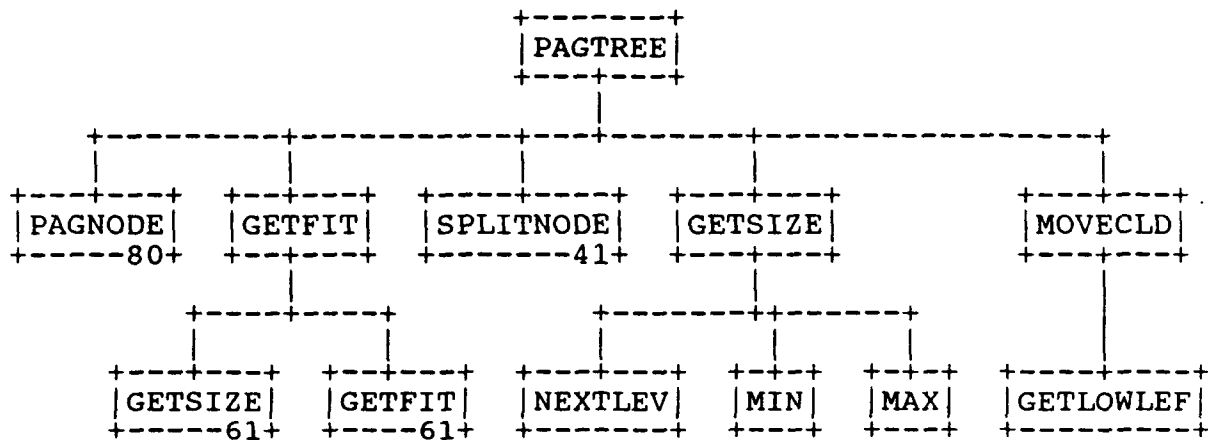
59



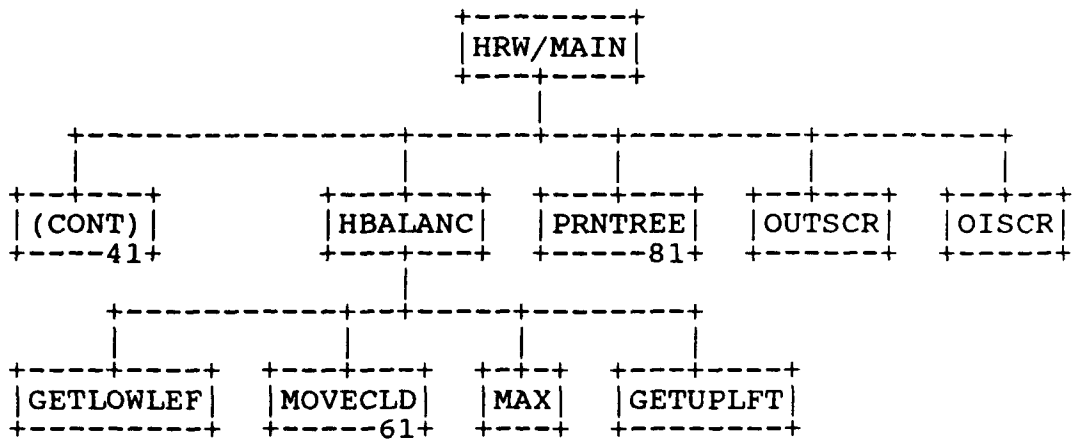
60



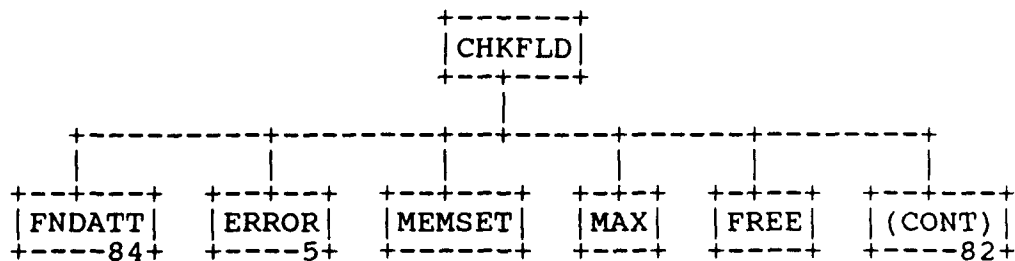
61



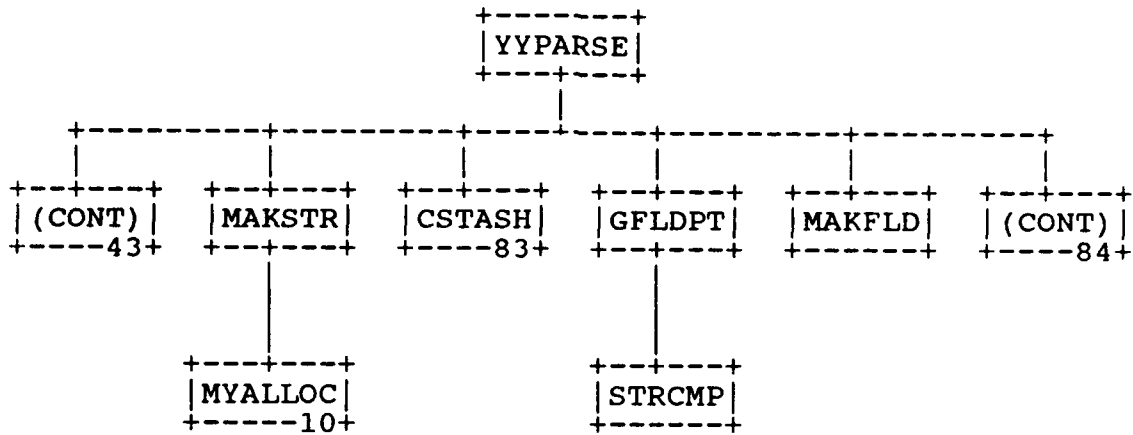
62



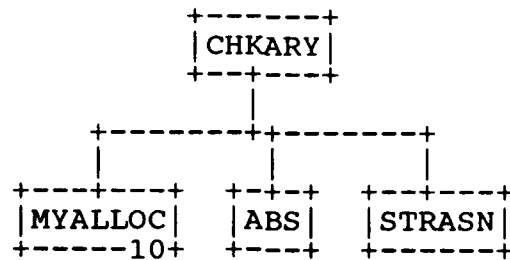
63



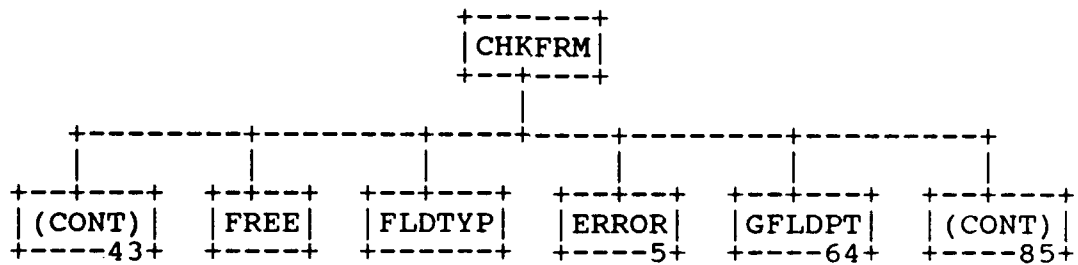
64



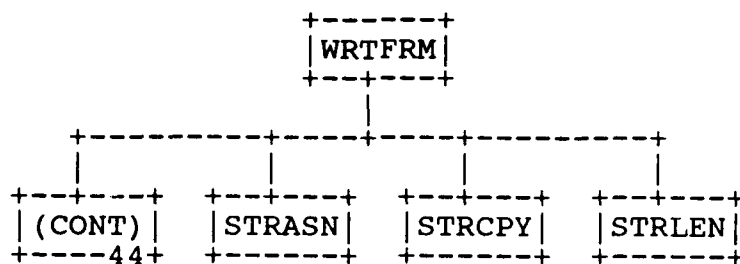
65



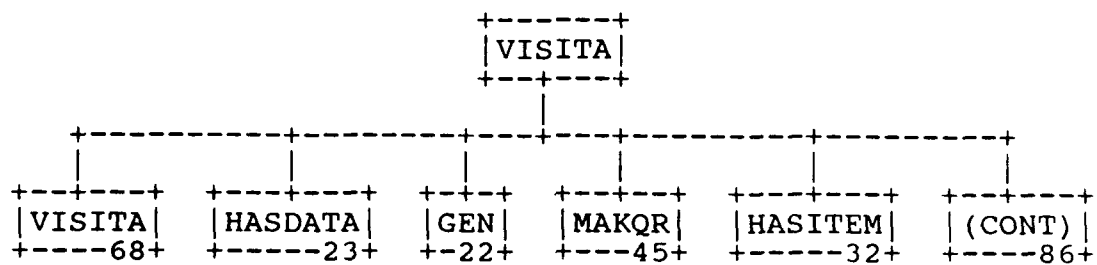
66



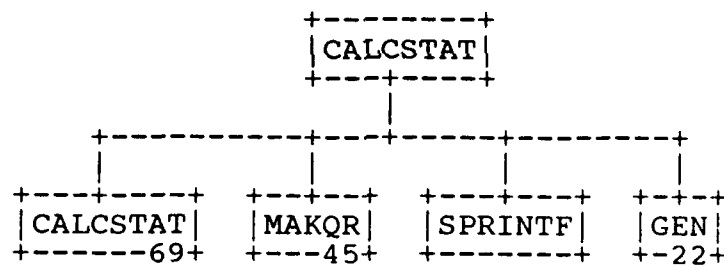
67



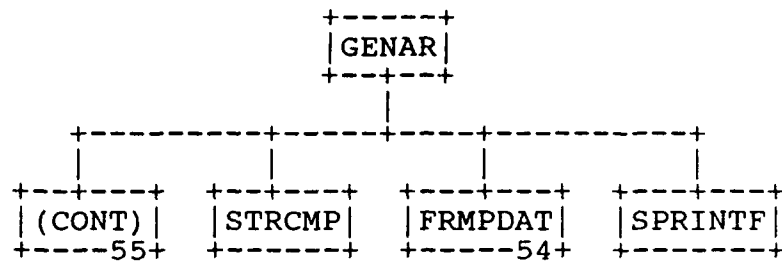
68



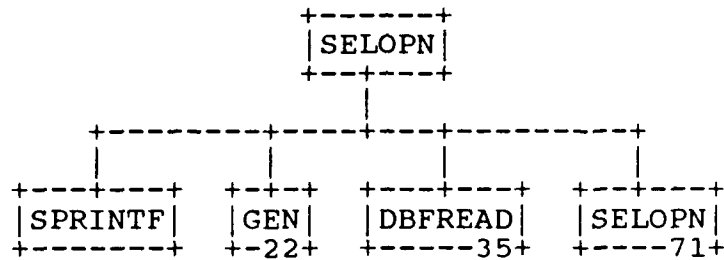
69



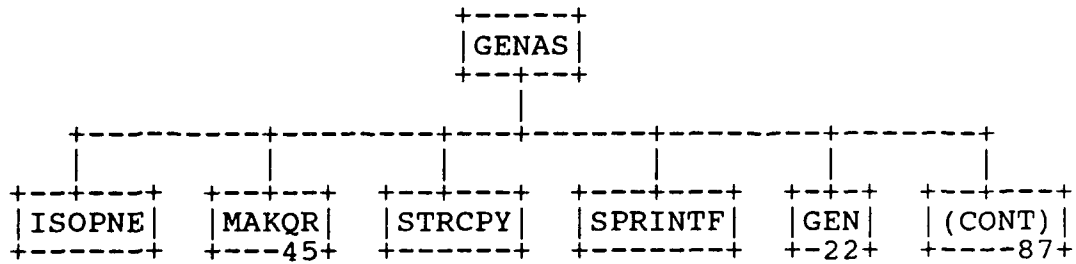
70



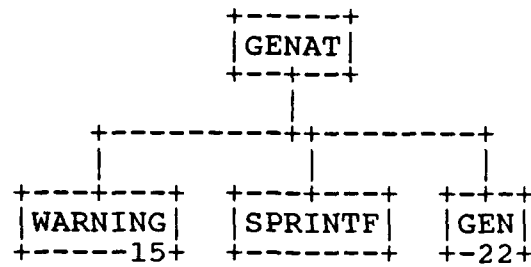
71



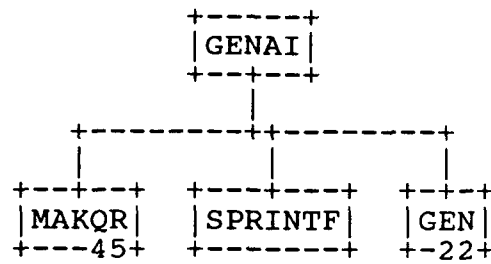
72



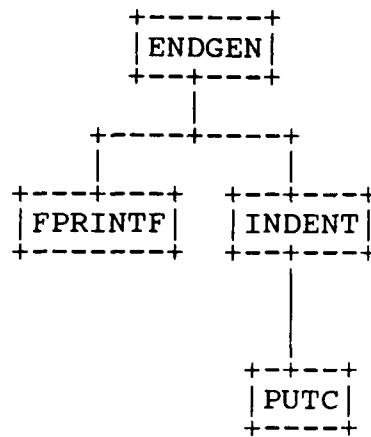
73



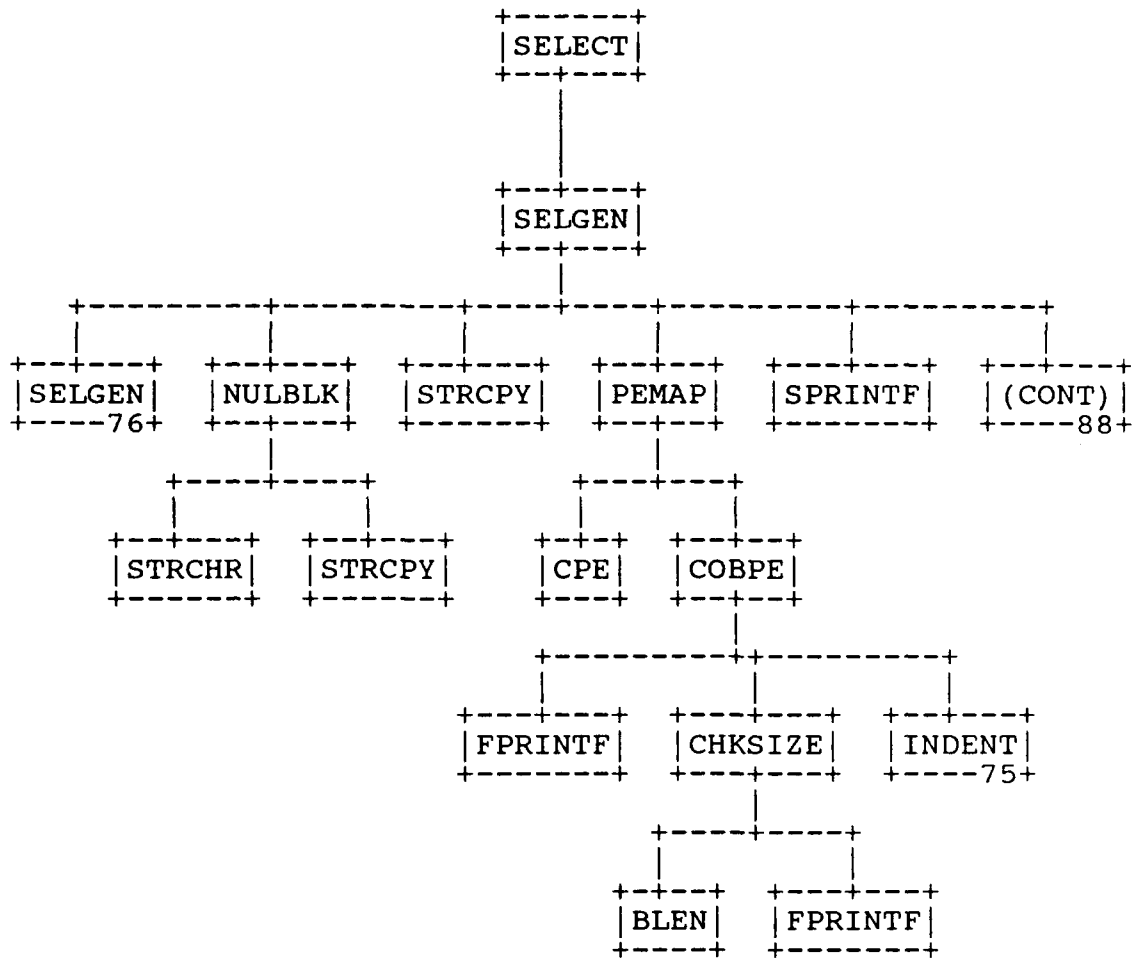
74



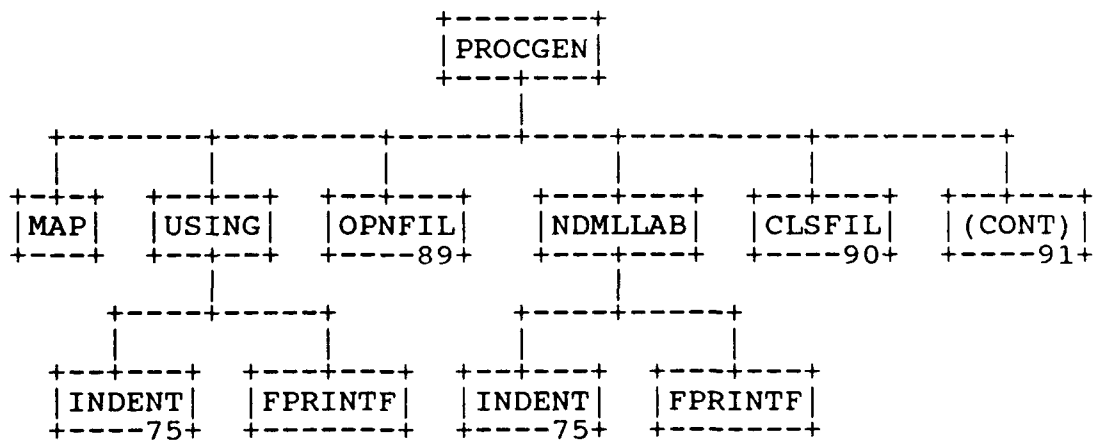
75



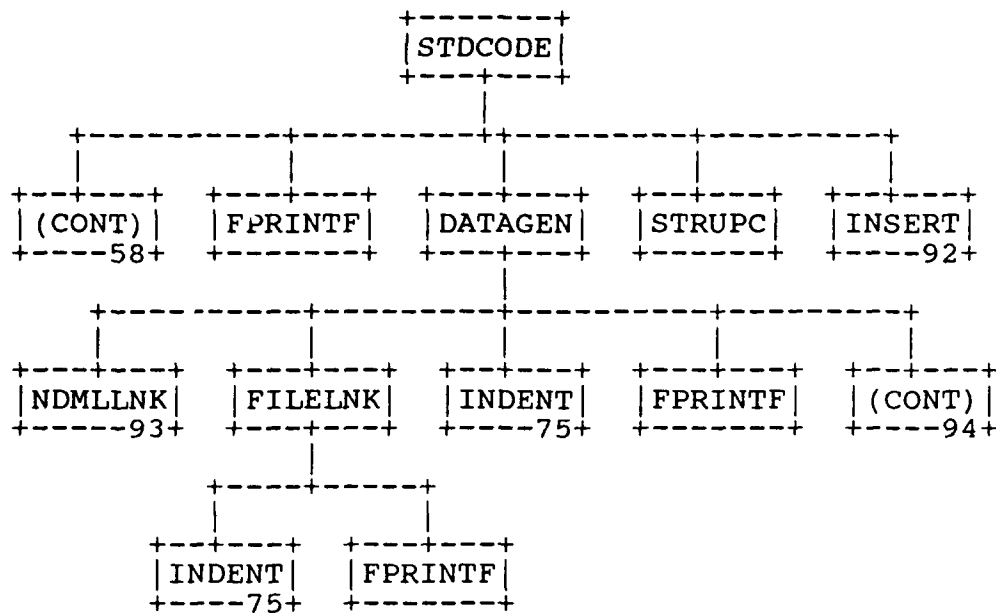
76



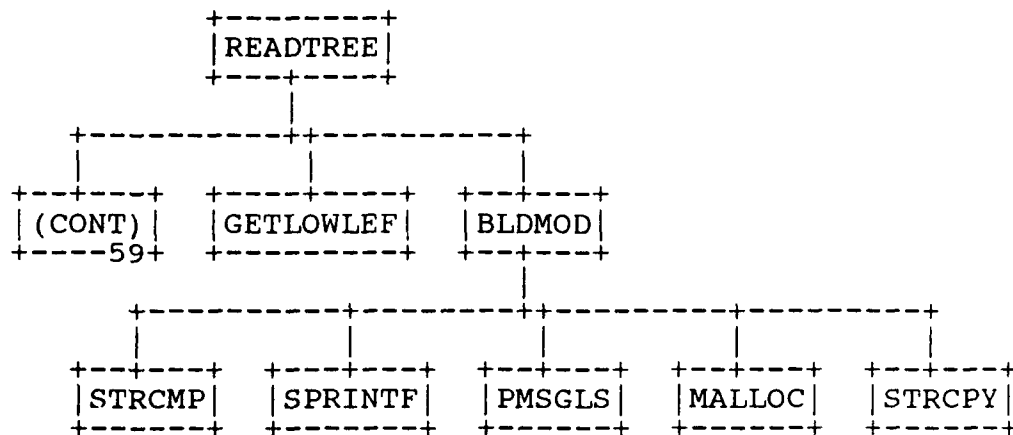
77



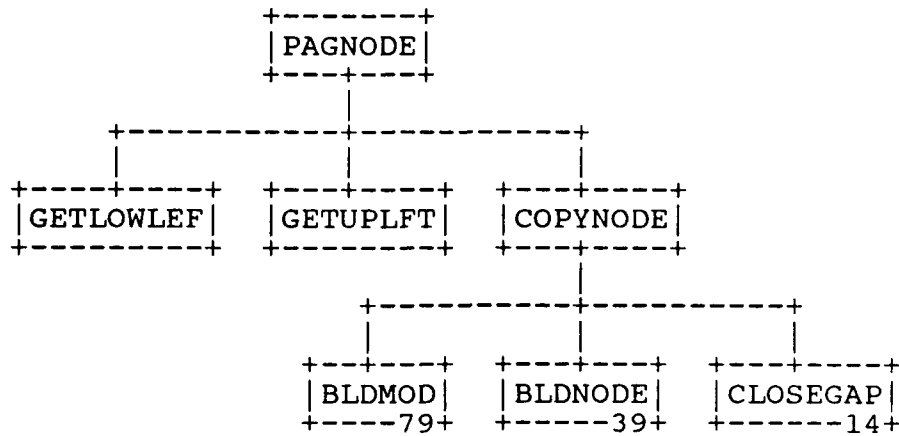
78



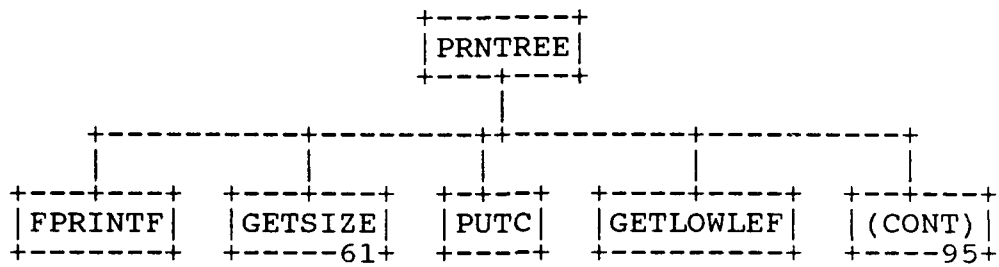
79



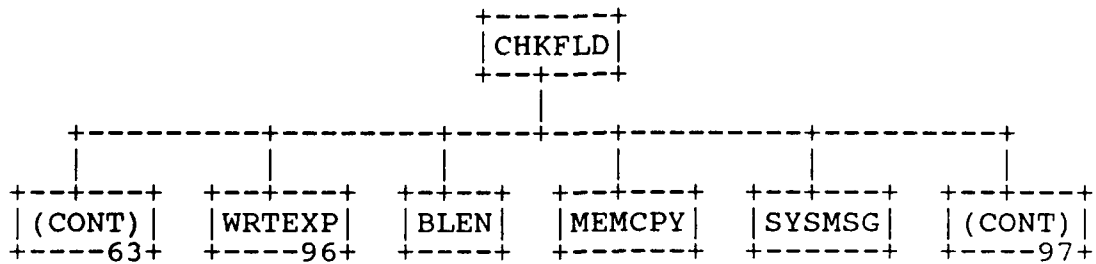
80



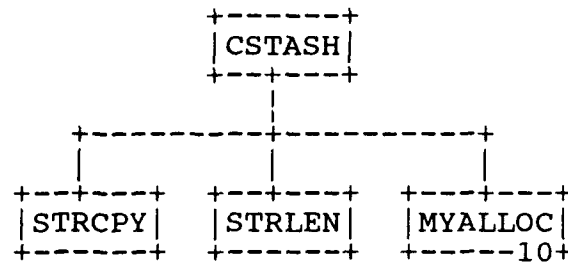
81



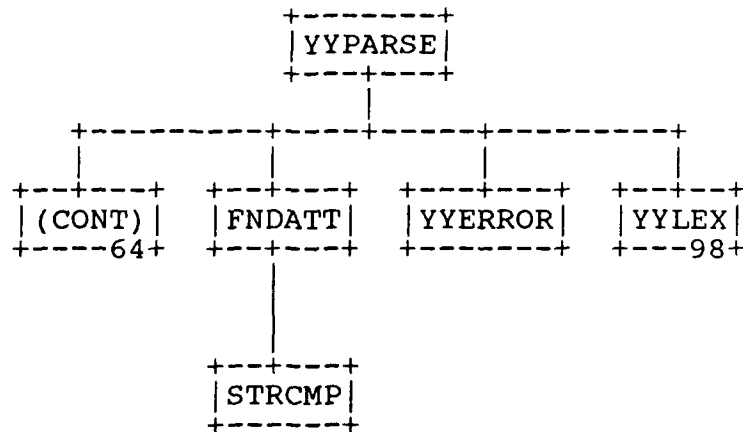
82



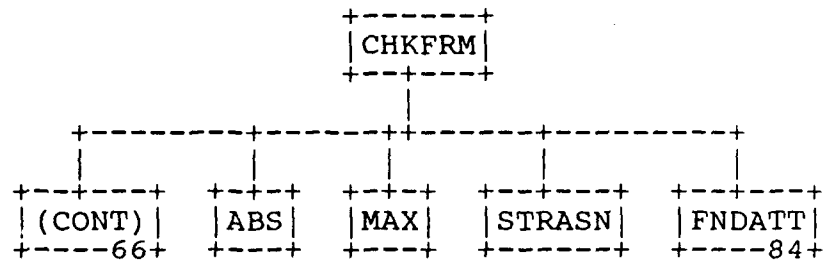
83

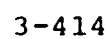


84

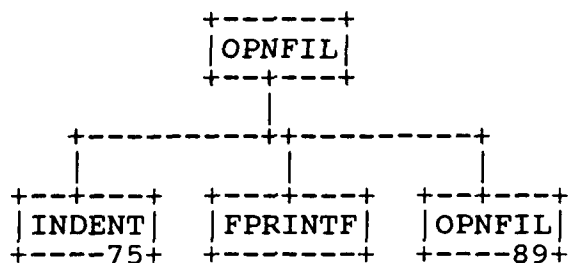


85

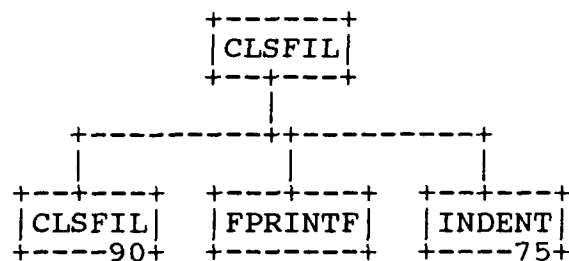




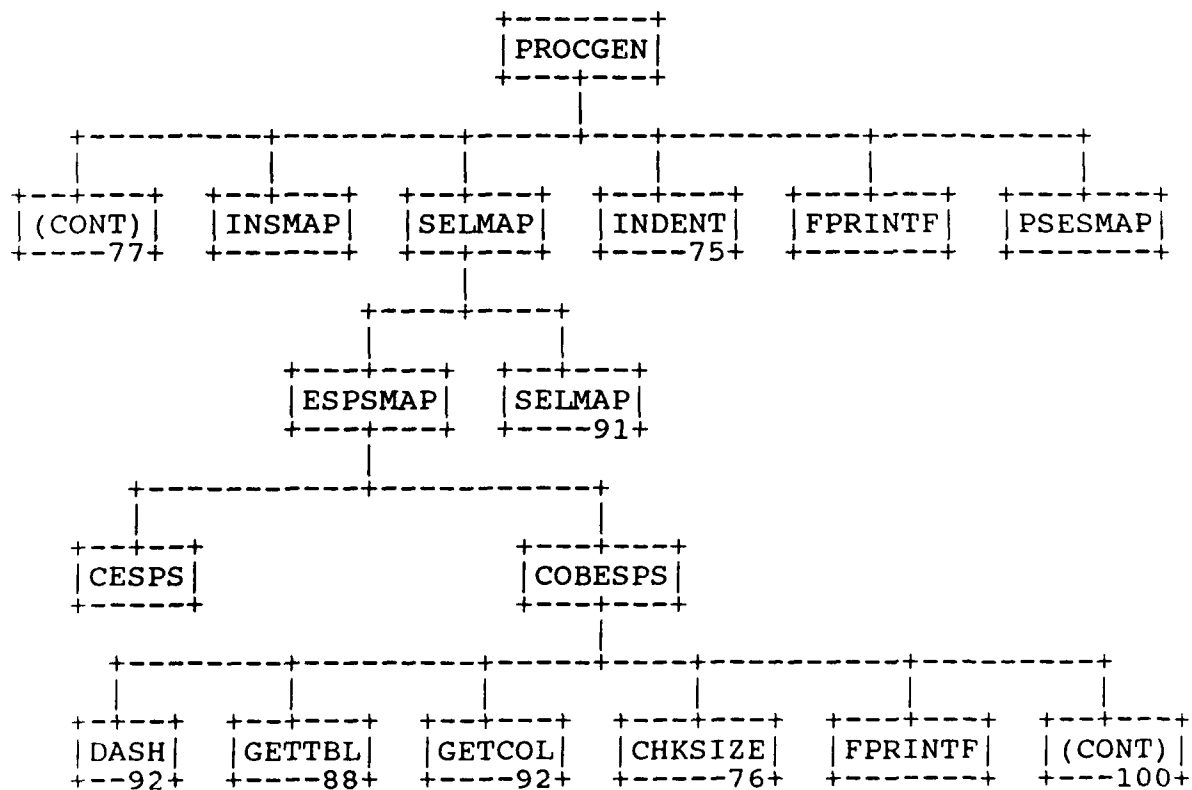
89



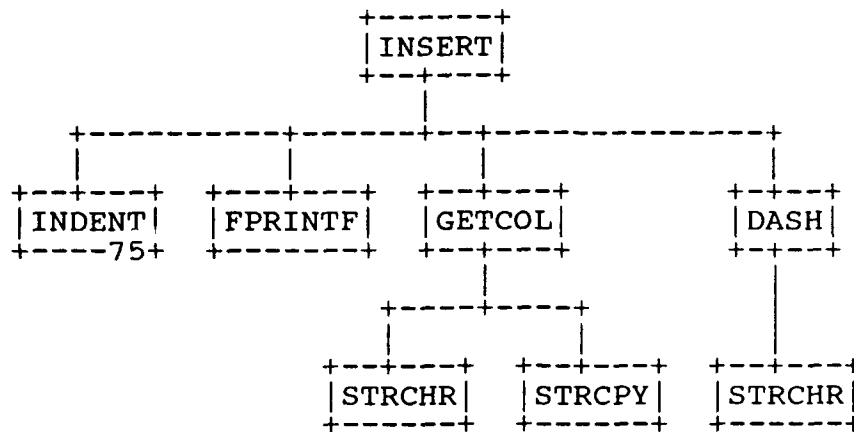
90



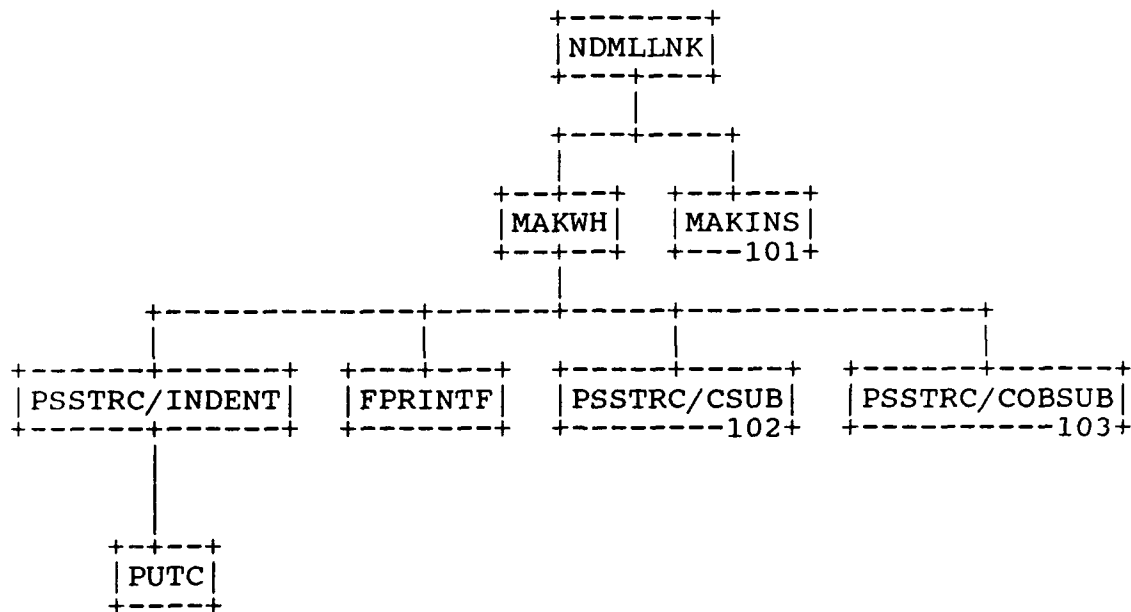
91



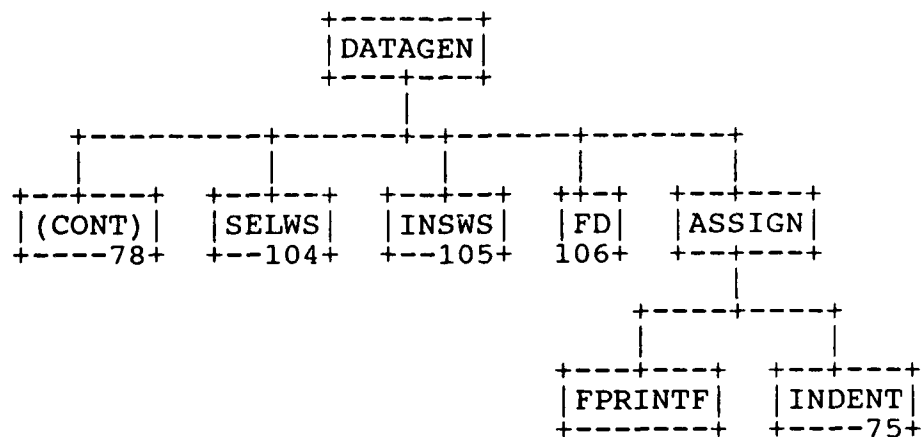
92



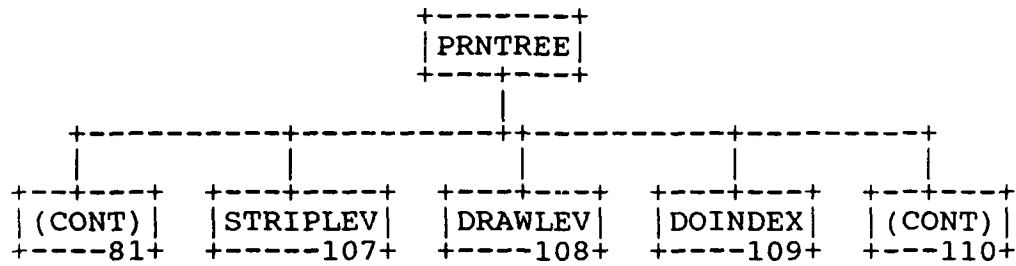
93



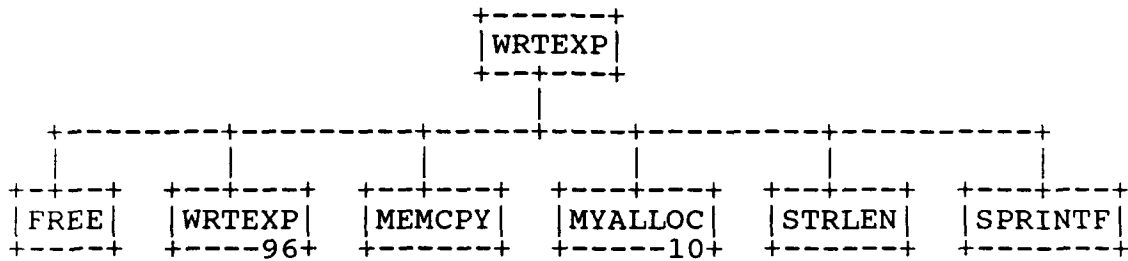
94



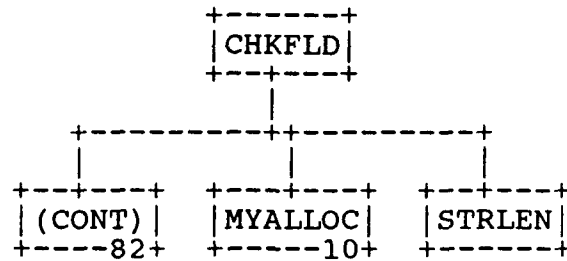
95



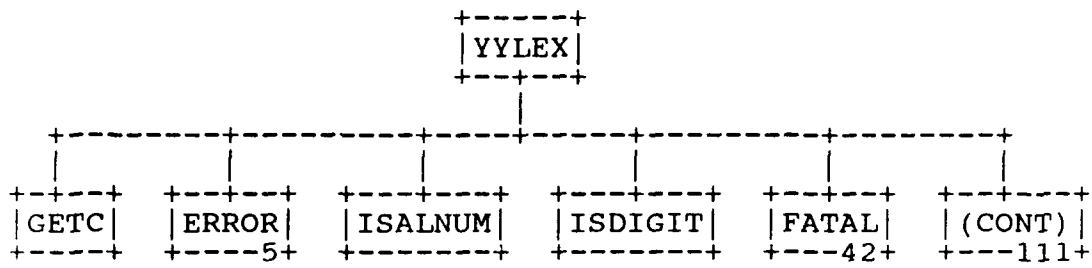
96



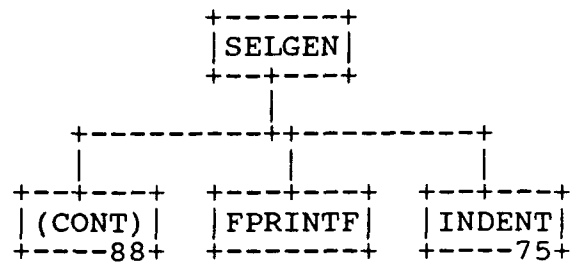
97



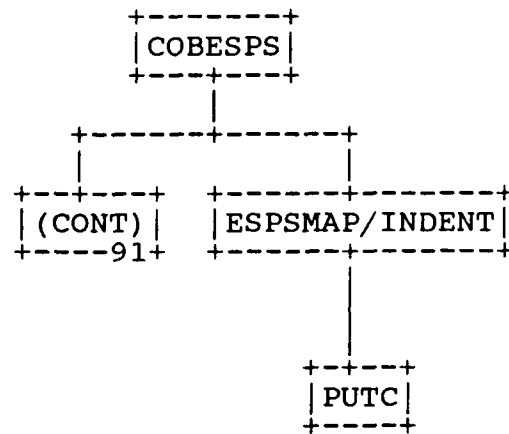
98



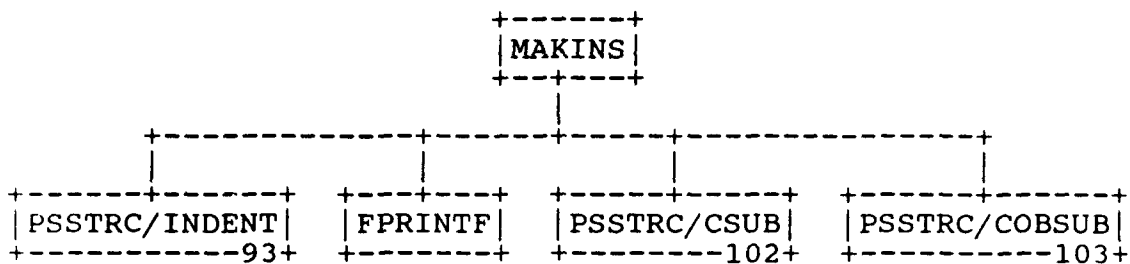
99



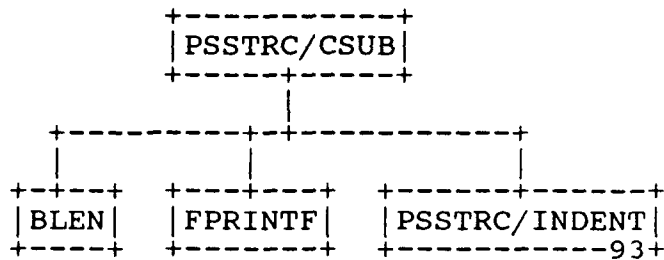
100



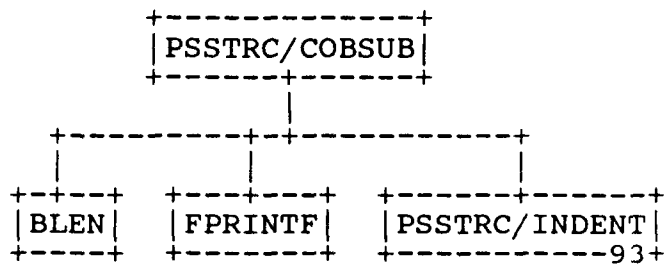
101



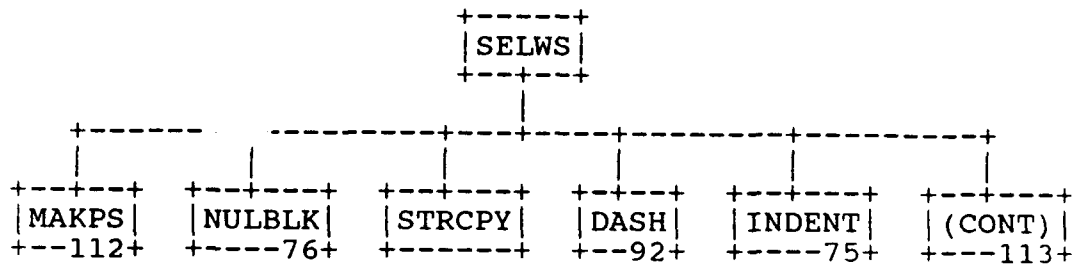
102



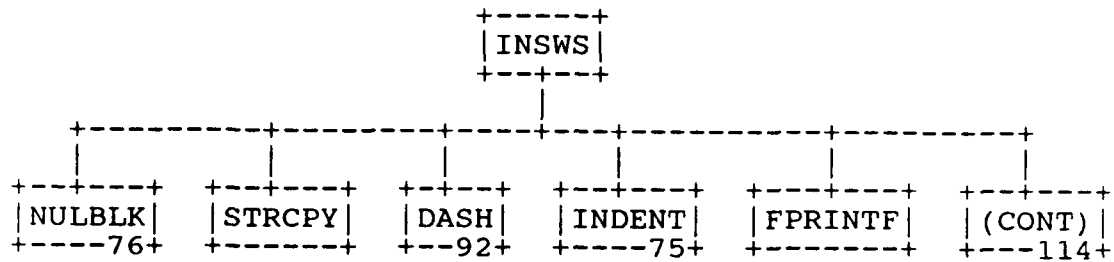
103



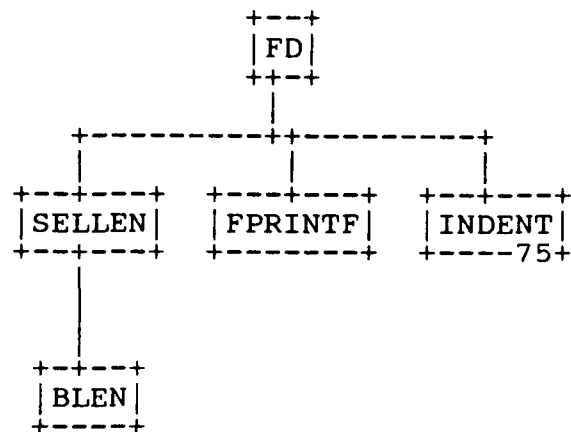
104



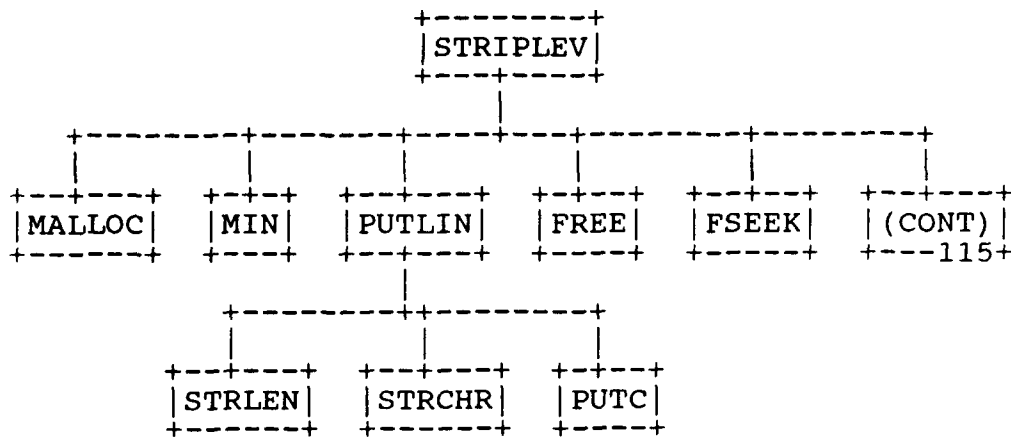
105



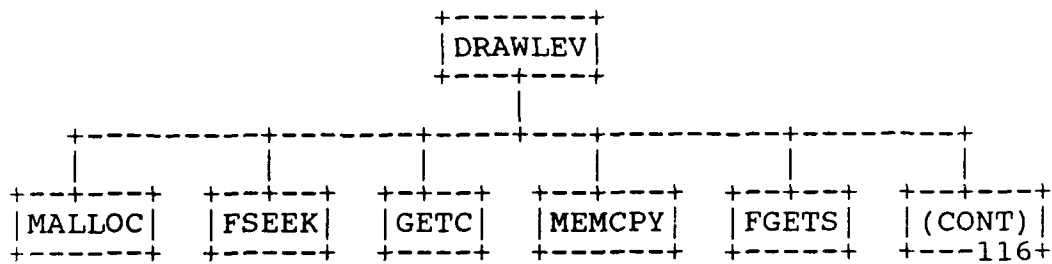
106



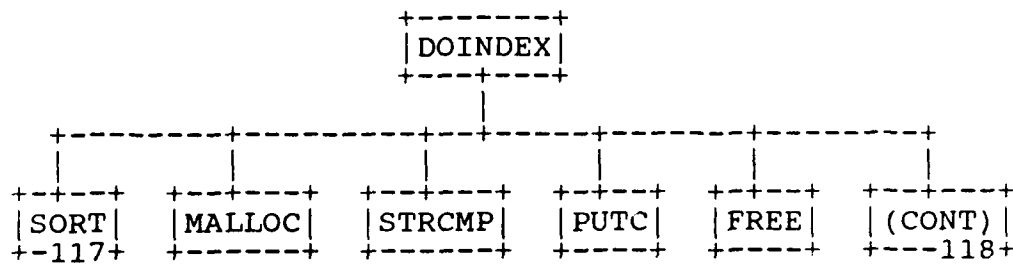
107



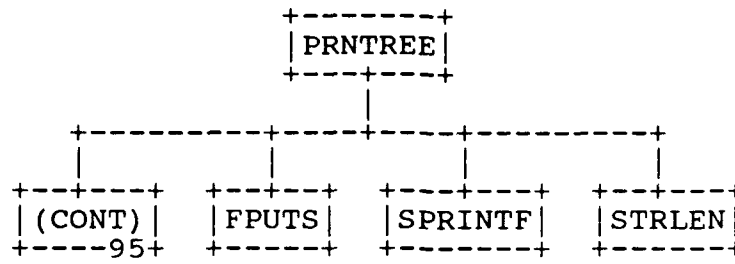
108



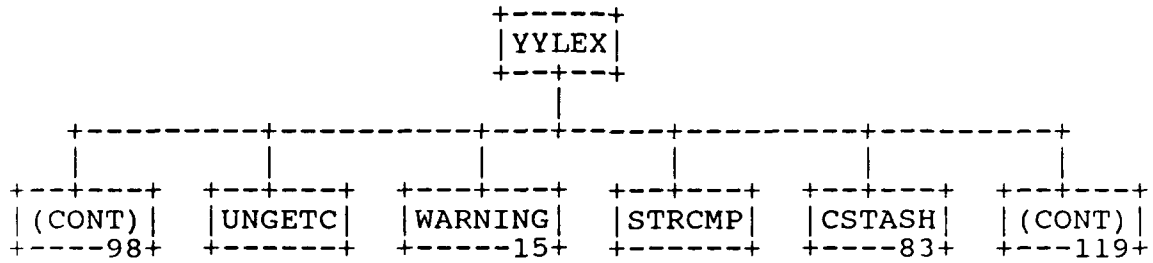
109



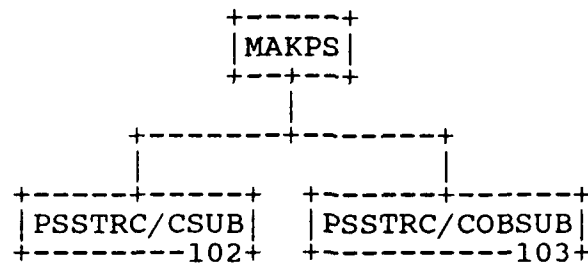
110



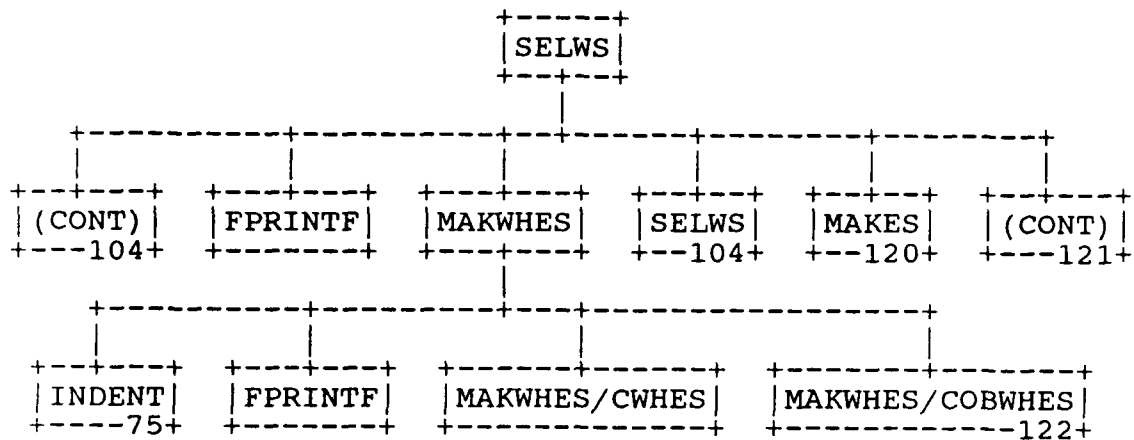
111



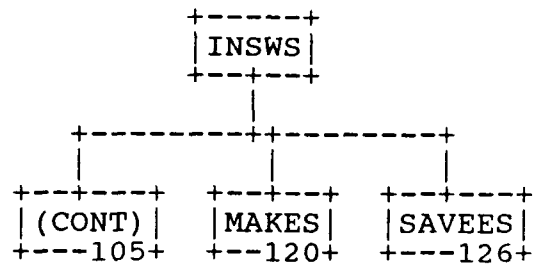
112



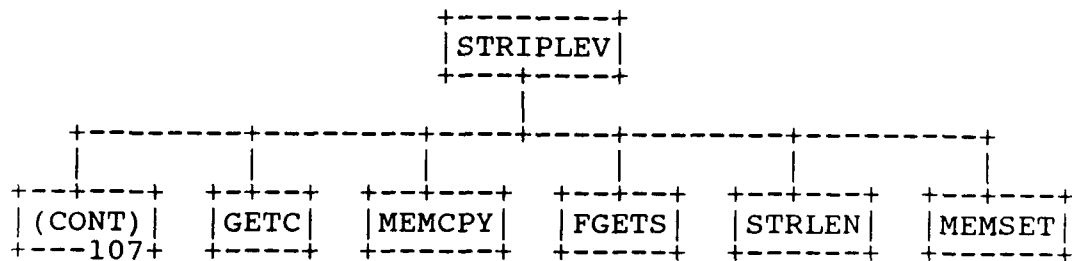
113



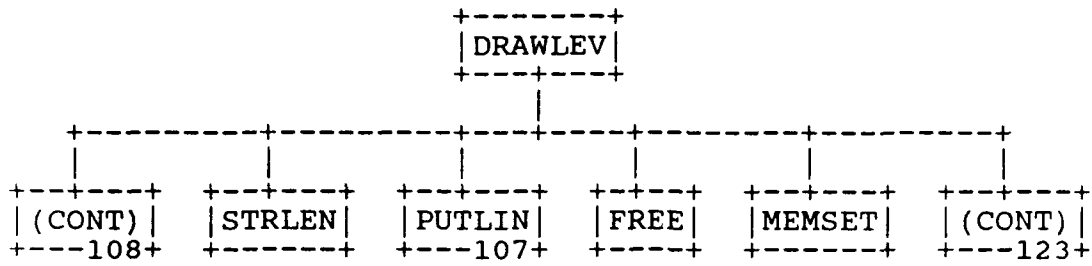
114



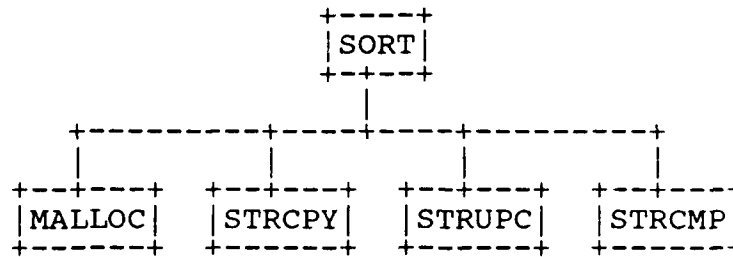
115



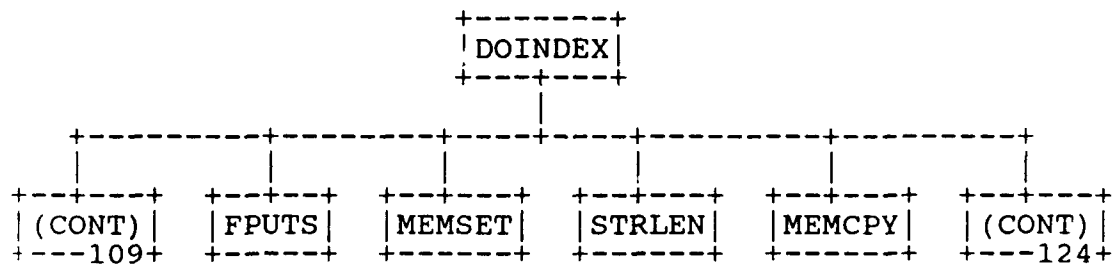
116



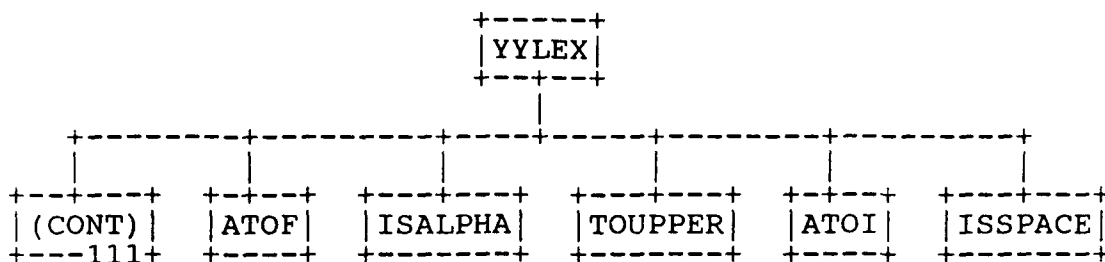
117



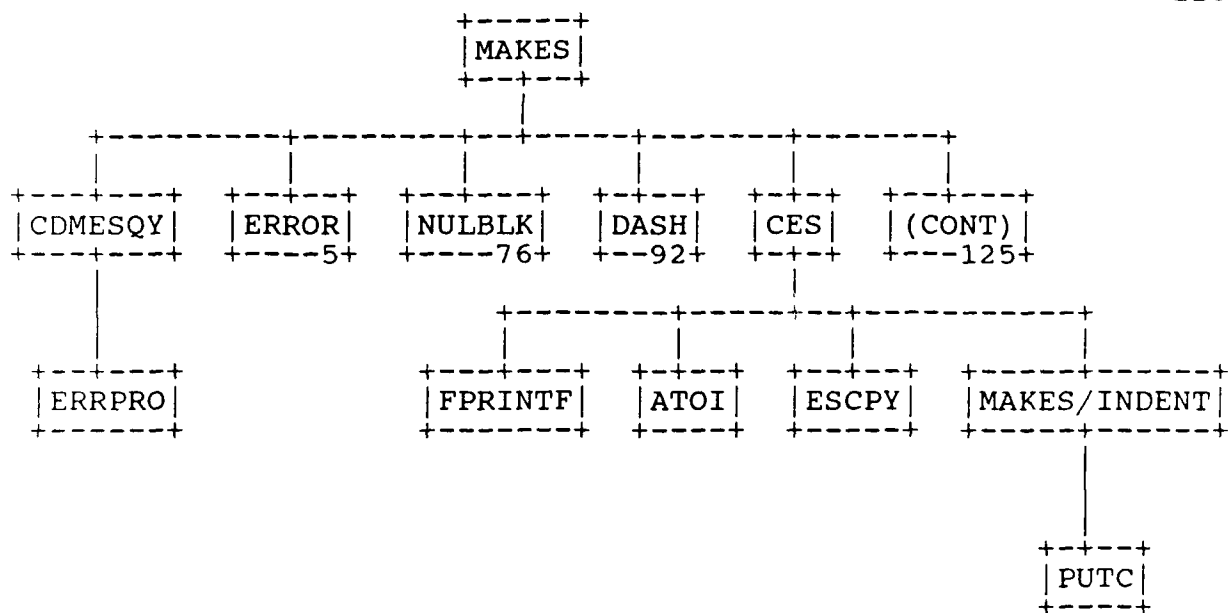
118



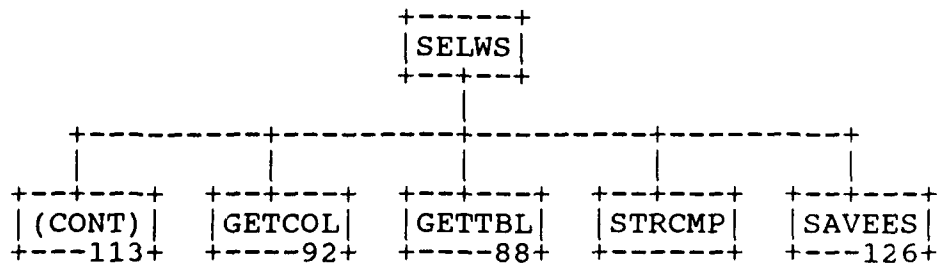
119



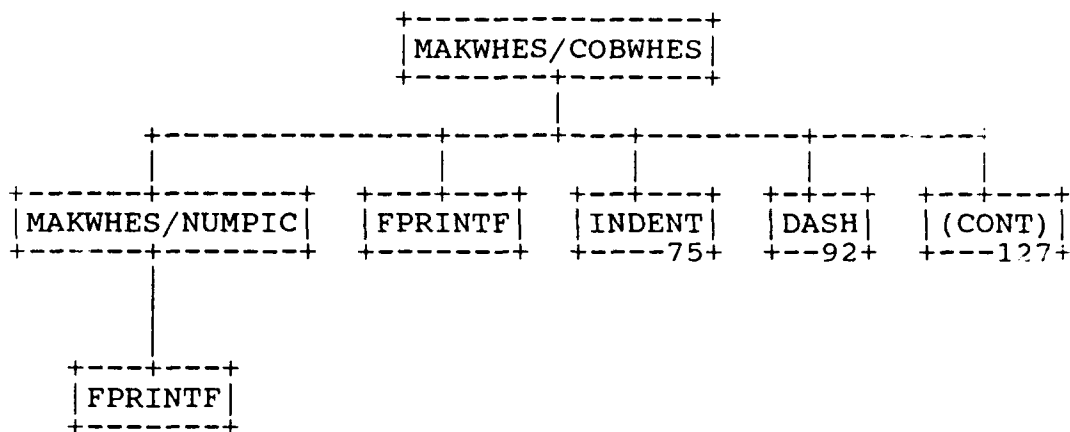
120



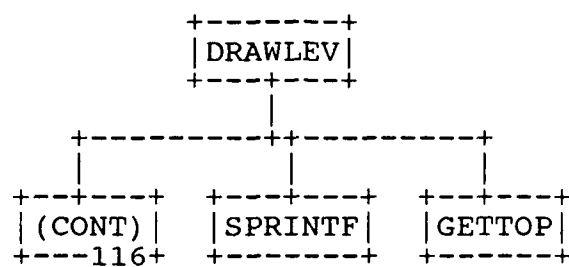
121



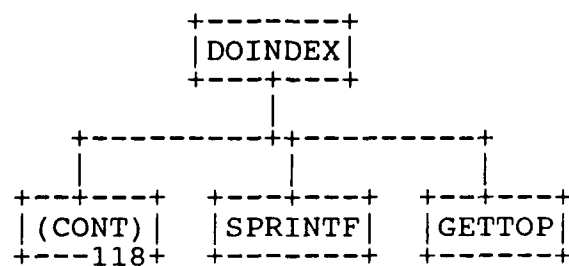
122



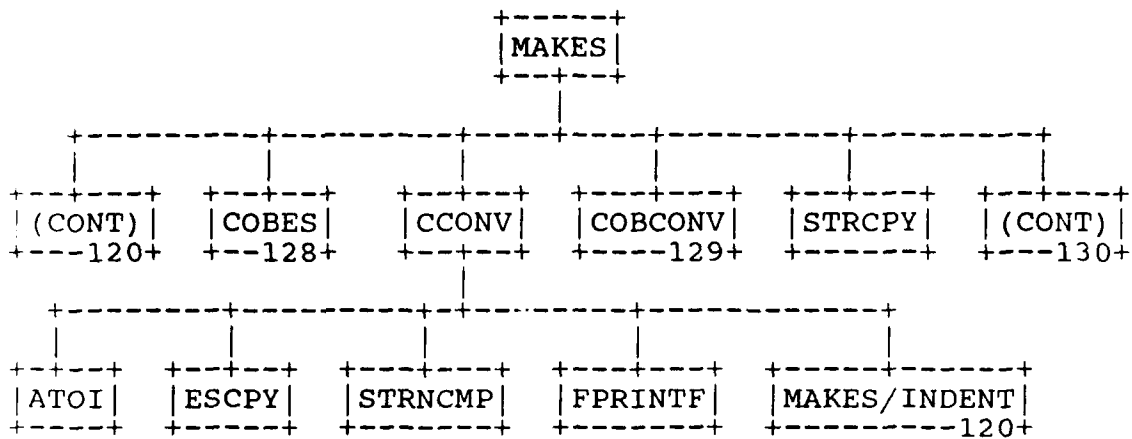
123



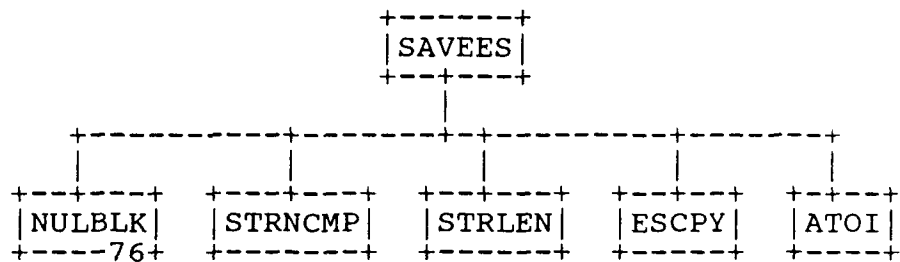
124



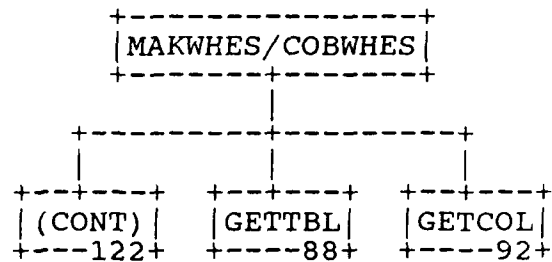
125



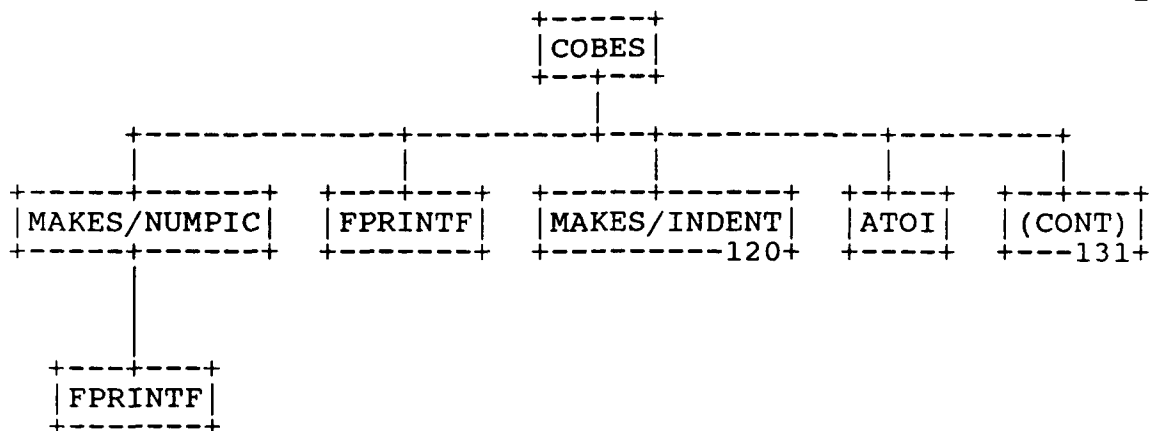
126



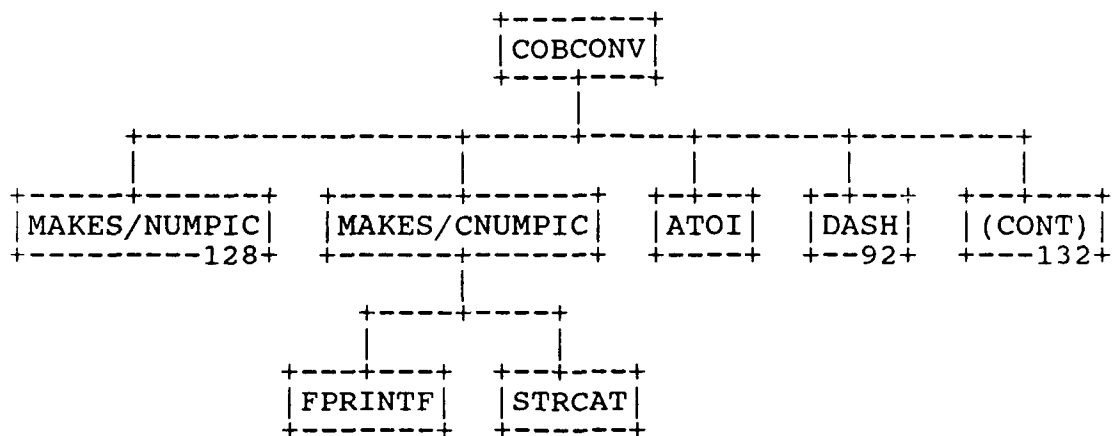
127



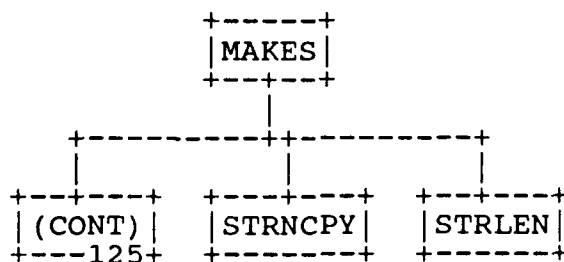
128



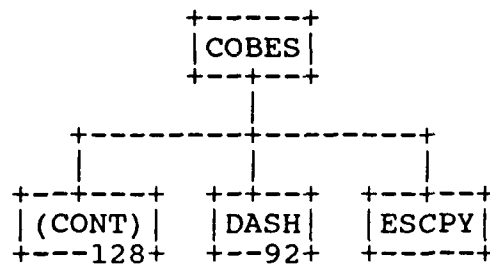
129



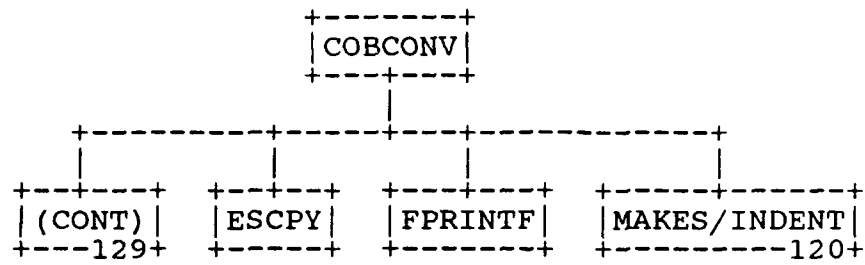
130



131



132



ABS		ESCPY	
ACTRSV.....	21	ESPSMAP.....	91
ADDCHK.....	43	ESPSMAP/INDENT...	100
ADDFRM		FATAL.....	42
ARRANGE.....	41	FCLOSE	
ASSIGN.....	94	FD.....	106
ATOF		FGETS	
ATOI		FILELNK.....	78
BLDMOD.....	79	FLANCI.....	6
BLDNODE.....	39	FLDRSV.....	20
BLDSUB.....	23	FLDTYP	
BLN		FNDATT.....	84
BSCODE.....	35	FNDFRM.....	18
CALCSTAT.....	69	FOPEN	
CALLOC		FPRINTF	
CCONV.....	125	FPUTS	
CDMESQY.....	120	FREE	
CES.....	120	FRMPDAT.....	54
CESPS		FRNTND.....	4
CHKARY.....	65	FSEEK	
CHKFLD.....	63	FTELL	
CHKFRM.....	43	FWRITE	
CHKGRP.....	86	GDATA	
CHKSIZE.....	76	GEN.....	22
CLOSEGAP.....	14	GENAA.....	24
CLRNDP.....	86	GENAAL.....	24
CLSFIL.....	90	GENACT.....	24
COBCONV.....	129	GENAE.....	57
COBES.....	128	GENAH.....	57
COBESPS.....	91	GENAI.....	74
COBPE.....	76	GENAL.....	36
COPFLD		GENAP.....	54
COPYNODE.....	80	GENAQ.....	56
CPE		GENAR.....	55
CSTASH.....	83	GENAS.....	72
CTLRSV.....	20	GENAT.....	73
DASH.....	92	GENBEG.....	22
DATAGEN.....	78	GENCHG.....	34
DBFREAD.....	35	GENDB.....	49
DCLINDX.....	53	GENDOA.....	36
DELFLD		GENDS.....	49
DELNODE.....	14	GENFP.....	47
DOINDEX.....	109	GENFS.....	46
DRAWLEV.....	108	GENFSD.....	33
ENDGEN.....	75	GENINS.....	49
ERROR.....	5	GENMAIN.....	22
ERRPRO		GENNDP.....	48

GENPAG.....13	MAKWHEs/NUMPIC...122
GETC	MALLOc
GETCOL.....92	MAP
GETFILE.....5	MAPDB.....51
GETFIT.....61	MAX
GETLOWLEF	MEMCMP
GETLOWRIT	MEMCPY
GETPAR.....59	MEMSET
GETPTH.....19	MIN
GETSIZE.....61	MKINC.....32
GETTBL.....88	MKPOS.....27
GETTOP	MLPFRM.....18
GETUPLFT	MODPAGE.....41
GFLDPT.....64	MOVCLD.....60
GRP/MAIN.....2	MOVECLD.....61
HASDATA.....23	MYALLOC.....10
HASITEM.....32	NDMLGEN.....37
HASLOWER	NDMLLAB.....77
HBALANC.....62	NDMLLNK.....93
HRW/MAIN.....3	NEXTLEV
INDENT.....75	NULBLK.....76
INITAL	OISCR
INITFP	OPNFIL.....89
INSERT.....92	OUTSCR
INSMAP	PAGNODE.....80
INSRSV.....21	PAGTREE.....61
INSWS.....105	PEMAP.....76
ISALNUM	PMSGLC
ISALPHA	PMSGLS
ISDIGIT	PRINTF
ISOPNE	PRNT.....1
ISSPACE	PRNTREE.....81
MAKACT.....10	PROCGEN.....77
MAKES.....120	PSESMAP
MAKES/CNUMPIC...129	PSSTRC/COBSUB....103
MAKES/INDENT....120	PSSTRC/CSUB.....102
MAKES/NUMPIC....128	PSSTRC/INDENT....93
MAKFLD	PTHPTR
MAKINS.....101	PUTATT
MAKINT.....10	PUTC
MAKPS.....112	PUTCUR
MAKQR.....45	PUTLIN.....107
MAKSTR.....64	READDB.....35
MAKWH.....93	READTREE.....39
MAKWHEs.....113	REPOS.....40
MAKWHEs/COBWHEs..122	RSETNDP.....50
MAKWHEs/CWHEs	RSETSTAT.....54

RWEXPD.....17	WRTFRM/TBFCLOS....29
RWOPN.....12	WRTFRM/WRTDBF....44
RWSP/FIXFRM.....17	WRTFRM/WRTFLD....28
SAVEES.....126	WRTFRM/WRTTBF....29
SELECT.....76	WRTFRM/WRTTXT....16
SELGEN.....76	YYERROR
SELLEN.....106	YYLEX.....98
SELMAP.....91	YYPARSE.....6
SELOPN.....71	
SELRSV.....31	
SELWHR.....56	
SELWS.....104	
SETNDP.....52	
SORT.....117	
SPLICE.....40	
SPLITNODE.....41	
SPRINTF	
STATRSV.....20	
STDCODE.....58	
STRASN	
STRCAT	
STRCHR	
STRCMP	
STRCPY	
STRIPLEV.....107	
STRLEN	
STRNCMP	
STRNCPY	
STRSPN	
STRUPC	
SYSMSG	
TERMFP	
TOUPPER	
TRGRSV.....21	
TRMNAT	
TRMNDML	
UNGETC	
UQFOR.....36	
UQPTH.....30	
USING.....77	
VISITA.....68	
WARNING.....15	
WINRSV.....19	
WRTEXP.....96	
WRTFRM.....11	
WRTFRM/DBFCLOS....44	
WRTFRM/FORMAT	

3.11 Program Listings Comments

This information is contained in the Module Descriptions in section 3.10.

SECTION 4

QUALITY ASSURANCE PROVISIONS

4.1 Introduction and Definitions

"Testing" is a systematic process that may be preplanned and explicitly stated. Test techniques and procedures may be defined in advance, and a sequence of test steps may be specified. "Debugging" is the process of isolation and correction of the cause of an error.

"Antibugging" is defined as the philosophy of writing programs in such a way as to make bugs less likely to occur and when they do occur, to make them more noticeable to the programmer and the user. In other words, as much error checking as is practical and possible in each routine should be performed.

4.2 Computer Programming Test and Evaluation

The quality assurance provisions for test consists of the normal testing techniques that are accomplished during the construction process. They consist of design and code walk-throughs, unit testing, and integration testing. These tests are performed by the design team. Structured design, design walk-through and the incorporation of "antibugging" facilitate this testing by exposing and addressing problem areas before they become coded "bugs."