

ROPES

Remote Online Print Executive System Version 15.0

Utilities Guide

PREFACE

This publication contains information necessary for the installation, maintenance and operation of the Remote Online Print Executive System (ROPES), a proprietary program product used to manage a network of printers in the CICS interactive environment. It provides data processing managers, system programmers, application programmers and operating personnel with the information required to effectively use this product.

Information in this publication is subject to significant change.

THIS MANUAL IS PROVIDED FOR THE SOLE AND EXCLUSIVE USE OF THE CUSTOMER. THE MATERIAL CONTAINED IN THIS MANUAL IS CONFIDENTIAL AND SHOULD BE SO TREATED. PORTIONS OF THIS MANUAL MAY BE REPRODUCED FOR INTERNAL USE ONLY. THE COPYRIGHT NOTICE MUST APPEAR ON ALL COPIES.

| Fifth Edition (June 2008) [Edited with no significant changes in June, 2009)

| This edition applies to Version 15.0 of the program product Remote Online Print Executive System (ROPES) and to all subsequent versions and modifications until otherwise indicated in new editions or newsletters.

| © Copyright 1991-2009, Axios Products, Inc. All rights reserved.

Contents

Contents.	<u>3</u>
Introduction.	<u>7</u>
What This Manual Contains.	<u>7</u>
Related Publications.	<u>7</u>
General Information.	<u>9</u>
ROP EBRPU Batch Report Printing Utility Program.	<u>11</u>
Input and Output.	<u>11</u>
Control.	<u>11</u>
Job Control Statements.	<u>11</u>
Utility Control Statement.	<u>12</u>
ROPEBRPU Example.	<u>12</u>
ROPECPYG ROPES Screen Print Transaction.	<u>13</u>
Installation and Use.	<u>13</u>
Restrictions.	<u>13</u>
Error Handling.	<u>13</u>
ROPEKCRQ / ROPEZCxx ROPESPRT ROPES Screen Print Transaction.	<u>15</u>
Installation and Use.	<u>15</u>
Restrictions.	<u>15</u>
Error Handling.	<u>15</u>
ROPEBQUE Batch Queuing While ROPES Is Active Online.	<u>17</u>
Input and Output.	<u>17</u>
Control.	<u>17</u>
Job Control Statements.	<u>18</u>
Intermediate File Definition.	<u>18</u>
BQTX ROPES Batch Queue Transfer Transaction.	<u>19</u>
Table Entries.	<u>19</u>
ROPEBQOR Batch Queue Reorganization Utility.	<u>21</u>
Input and Output.	<u>21</u>
Control.	<u>21</u>
Job Control Statements.	<u>21</u>
Restrictions.	<u>22</u>
Utility Control Statements.	<u>22</u>
ROPEBQOR Examples.	<u>23</u>
ROPEUTIL ROPES Utility Program Driver.	<u>25</u>
Input and Output.	<u>25</u>
Control.	<u>25</u>
Job Control Statements.	<u>25</u>
Restrictions.	<u>26</u>
Utility Control Statements.	<u>27</u>

Alternate Facility File Resynchronization Function.	29
Function.	29
Data Sets.	29
Command Format.	29
REPORTS.	29
EXCLUDE.	29
MESSAGES.	29
ALTSYNChronize Utility Examples.	29
The Controls Utility Function.	31
Function.	31
Data sets.	31
Command Format.	31
PTYPE.	31
Options For The Screen-Handler Record Type.	31
DESC.	31
TRANPFX.	31
PROGPFX.	31
LANG.	31
MSGFILE.	31
HLPFILE.	32
ENVTABL.	32
COLORS.	32
Default Field Color.	32
Label Field Color.	32
Error Message Field Color.	32
Value Field Color.	32
Error Value Field Color.	32
Information Field Color.	32
Highlighted Field Color.	32
Reserved Field Color.	32
System Parameter Record Options.	32
DESC.	32
MROREGN.	32
TDQUE.	33
ITRNPFX.	33
MTERMID.	33
ACCDEST.	33
SECUPD.	33
SECEXIT.	33
INTRDRQ.	33
RFILES.	33
Alternate Queue.	33
Alternate RIB.	33
Batch Queue.	33
Job File.	33
Printer File.	34
Primary Spool.	34
Distribution Definition File.	34
Distribution Translation File.	34
Report Information File.	34
Report/Printer Index.	34
LPR Print Options File.	34
PF Key Parameter Record Options.	34
DESC.	34

PANEL.....	34
KEY1 To KEY12.	34
Logging Parameter Record Options.	34
DESC.....	34
LOGTYPE.....	34
TDQUE.	35
LOGEXIT.	35
Controls Utility Example.	35
The LOADHLP Utility Function.	37
Function.	37
Data sets.	37
Help Input Text And Markers.	37
Command Format.	38
LOADHLP Utility Example.	38
The Purge Utility Function.	39
Function.	39
Data sets.	39
Command Format.	39
REPORTS.	39
EXCLUDE.	39
MESSAGES.	39
Purge Utility Examples.	39
The Register Utility Function.	41
Function.	41
Data Sets.	41
Utility Control Statement Format.	41
PTYPE.	41
COMPANY.	41
ADDRESS.	41
CITY.	41
STATE.	41
ZIPCODE.	41
COUNTRY.	41
Register Utility Sample JCL.	42
The Apply Security Changes Utility Function.	43
Function.	43
Data Sets.	43
Utility Control Statement Format.	43
SECCODE.	43
Apply Security Utility Sample JCL.	43
The List Security Information Utility Function.	45
Function.	45
Data Sets.	45
Utility Control Statement Format.	45
SECINFO.	45
List Security Information Utility Sample JCL.	45
The JCL Load Utility Function.	47
Function.	47
Data Sets.	47

Utility Control Statement Format..	47
JOBS..	47
DESC..	47
LINES..	47
JCL Header Control Record..	48
JCL Loader Utility Sample JCL..	48
The Convert JCL File Utility Function..	49
Function..	49
Data Sets..	49
Utility Control Statement Format..	49
MEMBERS..	49
REPLACE..	49
Convert JCL File Utility Sample JCL..	49
Report To Printer Index Generation Function..	51
Function..	51
Input and Output..	51
Command Format..	51
MESSAGES..	51
Reindex Utility Example..	51
The ROPES External Interface Utility Functions..	53
Introduction..	53
Functions..	53
Data Sets..	53
Command Format..	53
ROPES External Interface Utility Examples..	55
ROPES External Interface Sample Output..	55
Report To PDF Conversion Function..	59
Function..	59
Input and Output..	59
Command Format..	59
Build PDF Utility Example..	59
Report To HTML Conversion Function..	61
Function..	61
Input and Output..	61
Command Format..	61
Build HTML Utility Example..	61
TCP/IP Translation Facility Default Setting..	63
Function..	63
Input and Output..	63
Command Format..	63
Translate Facility Defaults Utility Example..	63
Stand-By Facility SYSIDENT Utility..	65
Function..	65
Input and Output..	65
Command Format..	65
Stand-By Facility SYSIDENT Utility Example..	65
Index..	67

Introduction

What This Manual Contains

This manual is intended for programmers that will be using the ROPES batch and online utility programs and transactions.

Each chapter in the manual addresses one or more of these subjects.

Other ROPES related activities are documented in the other ROPES manuals. The Related Publications lists those manuals and summarizes their contents.

Just a brief note on a notation used in this manual. Wherever you see <hlq> in a data set name we are referring to the **high level** qualifier you choose to use when installing the ROPES data sets. You should always substitute that value when you actually refer to the data sets in JCL, and you will need to make this information available to any programmers or users that might need this information in the course of their work.

Related Publications

In addition to this manual, there are three other ROPES publications. Your installation was supplied with at least one copy of each of these manuals in Adobe Acrobat (PDF) format. You may print any copies you require for your company's use, but remember that this material is licensed to your company and may not be distributed outside of your organization.

General Information Manual

This manual contains an introduction to ROPES with an overview of the functionality ROPES provides.

Administrator's Guide

This manual is intended to guide the ROPES installer and administrator through the steps required to install ROPES, customize ROPES for your installation, establish and maintain the ROPES control files, archive report data, audit ROPES activities, account for ROPES usage,

secure ROPES, and tailor and use the ROPES exits.

User's Guide

This manual is intended for end-users of ROPES. It explains the various ROPES status displays, the ROPES commands and transactions used to control ROPES, the JES transfer facility used to retrieve output from JES for printing by ROPES, and it documents the report browsing transaction.

Programmer's Guide

This manual is intended for programmers that will be interfacing application programs to ROPES through the ROPES Application Programming Interface. The ROPES sample programs are also documented here.

Messages and Codes

This manual is intended for administrators, programmers and end users that will be using the various functions of ROPES. It contains a complete list of the ROPES messages and codes with descriptions and suggested user actions when appropriate.

General Information

This section contains a description of the ROPES Utility Programs provided to each installation to assist in performing certain common ROPES services. The instructions and documentation necessary to use and understand each utility are provided.

UTILITY USE

ROPEBRPU Retrieve and print reports on batch printers. See page [11](#).

ROPEKCRQ/ROPEZCxx

ROPECPYG Produce screen-image printouts using a function key. ROPECPYG is described on page [13](#). ROPEKCRQ is described on page [15](#).

ROPESPRT Produce screen-image printouts using a function key. See page [15](#).

ROPEBQUE Store print data from a batch job while CICS is active. See page [17](#).

BQTX Transaction BQTX is used to transfer data stored by ROPEBQUE to the active ROPES queue. See page [19](#).

ROPEBQOR Reorganize the ROPES queue in a batch job. See page [21](#).

ROPEUTIL A master utility driver program which invokes various utility functions. See page [25](#).

ROPEALTU Alternate Facility Report File Synchronization utility. Invoked by the utility driver program ROPEUTIL. See page [29](#).

ROPECTRL Controls utility module. Sets screen handler, and system parameter records. Invoked by the utility driver program ROPEUTIL. See page [31](#).

ROPEHLPL Help information load utility. This program is invoked by the utility driver program ROPEUTIL. See page [37](#).

ROPEPURG Purges report data and resets ROPES report and printer controls. Invoked by the utility driver program ROPEUTIL. See page [39](#).

ROPERGST Register company information utility. Invoked by the utility driver program ROPEUTIL. See page [41](#).

ROPEASEC Apply Security Information changes. Invoked by the utility driver program ROPEUTIL. See page [43](#).

ROPELSEC List Security Information. Invoked by the utility driver program ROPEUTIL. See page [45](#).

ROPEJCLL Convert Job Streams for the Job Submission Utility. See page [47](#).

ROPECVJF Converts the old job data file to the new job file format. See page [49](#).

ROPERPIB Build the Report-To-Printer Index file. See page [51](#).

EXCI Perform External Call Interface Functions. See page [53](#).

ROPEBPDF Batch report to PDF conversion. Invoked by the utility driver program ROPEUTIL. See page [59](#).

ROPEBHTM Batch report to HTML conversion. Invoked by the utility driver program ROPEUTIL. See page [61](#).

ROPETRAN Define the TCP/IP Translation Defaults. See page [63](#).

ROPESBYU Define the Stand-By Facility participants SYSIDENT values. See page [65](#).

ROPEBRPU Batch Report Printing Utility Program

ROPEBRPU can be used to perform the following functions:

- Retrieve report data stored on the ROPES queue
- Print one or more retrieved reports on batch printers

Input and Output

ROPEBRPU uses the following inputs:

- The ROPERIB data set, containing the report control information;
- The ROPEQBR data set, containing the report text;
- A sequential data set, containing control cards to direct the processing of reports.

ROPEBRPU produces the following outputs:

- A message data set, containing program-generated status messages;
- A report data set, containing the ROPES report in print image format.

ROPEBRPU provides a return code to indicate the results of program execution. The return codes and their meanings are:

- 00, which indicates successful completion.
- 04, which indicates that one or more reports was not found. The job continues.
- 08, which indicates that one or more errors were noted in the message data set. Processing for a specific report was terminated. The job step continues.
- 12, which indicates an uncorrectable error occurred. The job was terminated.
- 16, which indicates that the output data set is missing. The job was terminated.
- 20, which indicates the control data set was missing. The job was terminated.
- 24, which indicates the message data set was missing. The job was terminated.

Control

ROPEBRPU is controlled by job control statements and utility control statements. The job control statements are required to execute or invoke ROPEBRPU and to define the data sets that are used and produced by the program. The utility control statements are used to control the functions of ROPEBRPU.

Job Control Statements

[Table I](#) shows the job control statements necessary for using ROPEBRPU.

Table I ROPEBRPU JCL

Statement	Use
JOB	Initiates the job.
EXEC	Specifies the program name, or, if the job control statements reside in a procedure library, the procedure name.
SYSLST	Defines a sequential message data set. The data set can be written to a system output device, a tape volume, or direct access volume.
ROPERIB	Defines the Report Information data set.
ROPEQBR	Defines the Queue data set.
SYS006	Defines the output data set. It can define a sequential data set on a card punch, a printer, a tape volume or a direct access volume.
SYSIPT	Defines the control data set containing the utility controls.

Restrictions

- The SYSLST data set is required for each use of ROPEBRPU.
- The message data set has a logical record length of 133 bytes, and consist of fixed-length unblocked records with an ASA control character in the first byte of each record. The output data set has a logical record length of 255 bytes, and consist of fixed-length unblocked records with an ASA control character in the first byte of each record. The control data set contains 80-byte unblocked records.
- The requirements of ROPES batch processing must be met.

Utility Control Statement

The format of the utility control statement used to control ROPEBRPU is:

- Columns 1-8 - the name of the report to be processed.
- Columns 9-80 - unused; may be used for comments.

One control statement is required for each report to be processed.

ROPEBRPU Example

In this example, three reports (REPORT01, REPORT03, and REPORT05) are selected for batch printing from the ROPES queue.

OS Installations use this example:

```
//PRINTRPT JOB ...
//          EXEC PGM=ROPEBRPU
//SYSLST    DD SYSOUT=A
//SYS006    DD SYSOUT=A
//ROPERIB   DD DSN=ROPE.S.RIBFILE.BASE,DISP=OLD
//ROPEQBR   DD DSN=ROPE.QBRFILE.BASE,DISP=OLD
//SYSIPT    DD *
REPORT01
REPORT03
REPORT05
/*
```

DOS Installations use this example:

```
// JOB PRINTRPT ...
// DLBL IJSYSUC,"your VSAM user catalog",,VSAM
// DLBL ROPERIB,"ROPE.S.RIBFILE.BASE",,VSAM,
//          CAT=IJSYSUC
// DLBL ROPEQBR,"ROPE.S.QBRFILE.BASE",,VSAM,
//          CAT=IJSYSUC
// DLBL ROPESCL,"ROPE.S.CORE.IMAGE.LIBRARY",99/365
// EXTENT ,vvvvvv,1,0,aaaaa,11
// LIBDEF CL,SEARCH=ROPESCL
// ASSGN SYS005,SYSIPT
// ASSGN SYS006,SYS008
// EXEC ROPEBRPU,SIZE=100K
REPORT01
REPORT03
REPORT05
/*
/&
```

ROPECPYG ROPES Screen Print Transaction

This transaction is invoked by the Attach Exit of the CICS Terminal Control Program (DFHTCP or DFHZCP) when the terminal operator hits the installation designated function key. The transaction reads the full screen and writes an exact copy to a ROPES report.

Installation and Use

ROPES source library member ROPECPYG contains the source program that supports this function. The user should examine this routine to determine the suitability of the title lines generated by the transaction. These may be modified by the installation.

The program constructs ROPES report names by concatenating the issuing terminal identifier to the prefix "COPY". The installation may modify this logic as required.

To use this utility, you must install the following Attach Exit in your Terminal Control Program. Either the CICS 1.4 or CICS 1.5 exit mechanism may be used. The CICS System Programmer's Guide describes the installation of these exits.

```

COPY DFHAID
* ROPES PA/PF PRINT TCP ATTACH EXIT
      SAVE (2,5)
      BALR 5,0
      USING *,5
      OC TCTTECA,TCTTECA ANY TASK?
      BNZ COPYRET YES, EXIT NOW
*
* CHANGE THE FOLLOWING COMPARE DEPENDING ON
* THE KEY TO BE USED FOR THE ROPES PF PRINT.
* THE KEY USED IN THIS EXAMPLE IS PA1.
*
      CLI TCTTEAID,DFHPA1 PA1 ENTERED?
      BNE COPYRET
      MVC TCAKCTI,ROPECPYT SET TRAN ID
      OI TCTTEKI,TCTTEOTI TRAN TO INIT
COPYRET DS 0H
      RETURN (2,5)
ROPECPYT DC CL4"COPY" TRAN ID
      DROP 5

```

You may change the default transaction code value.

Restrictions

This facility is available for non-conversational and pseudo-conversational transactions only.

Remote VTAM terminals that will use this function

must be specified as having an RUSIZE=2048 or CHNASSY=YES in their TCT entry, as full buffer read requests are issued.

Error Handling

All errors produce transaction ABENDS. The ABEND codes are documented in the source program.

ROPEKCRQ / ROPEZC_{xx} ROPESPRT ROPES

Screen Print Transaction

This transaction is invoked by the Request Exit of the Task Control Program when the terminal operator hits the designated function key. The transaction reads the full screen and writes an exact copy to a ROPES report.

Installation and Use

ROPES source library member ROPESPRT contains the source program that supports this function. The user should examine this routine to determine the suitability of the separator lines generated by the transaction. These may be modified by the installation.

The program reads and saves the screen image, then outputs a two line map at the bottom of the screen. The map source module ROPEPRT is in the ROPES source library and may be modified by your installation. The map provides an eight character data entry field into which the terminal operator must type the name of the report to receive the screen image.

When the name has been entered, the operator must press the ENTER key. The screen image is placed on the ROPES queue and a confirmation message is written back to the terminal in the mapped area at the bottom of the screen. The terminal operator should press the user-defined PA or PF key again to restore the original screen and the pseudo-conversational controls that may have been in effect when the screen print was initiated.

You must start the Task Control Request exit before this function can be used. To do this, execute the following command at a CICS terminal using the CECI function, execute the command in a program you have written or specify the option to enable the exit on the ROMT Auto-start Options menu.

One of two enable commands should be used depending on the CICS release you are running. For all CICS 2.1.2 installations and below the XKCREQ attach exit enable command must be used. For all CICS 3.1 through 3.3 installations the XZCATT exit

enable command must be used with program ROPEZCRQ. For CICS 4.1 and above, the XZCATT exit enable command must be used with program ROPEZC41. If you start the exit using the startup option under the Auto-start Options menu, then ROPES will start the appropriate exit for the CICS release you are running.

```
EXEC CICS ENABLE PROGRAM('ROPEKCRQ')
      EXIT(XKCREQ) START.
```

```
EXEC CICS ENABLE PROGRAM('ROPEZCRQ')
      EXIT(XZCATT) START.
```

```
EXEC CICS ENABLE PROGRAM('ROPEZC41')
      EXIT(XZCATT) START.
```

The source statements for the exit, ROPEKCRQ and ROPEZC_{xx}, are in the ROPES.SOURCE library. The distributed module uses function keys PF9 and PF21 to initiate the print operation and invokes the screen print program using transaction code RPRT. You may change the default transaction code value and function key assignments by modifying and reassembling module ROPEKCRQ or ROPEZC_{xx}.

Restrictions

This facility is available for non-conversational and pseudo-conversational transactions only.

Remote VTAM terminals that will use this function must be specified as having an RUSIZE=2048 or CHNASSY=YES in their TCT entry, as full buffer read requests are issued.

Error Handling

All errors produce transaction dumps. The dump codes are documented in the source program ROPESPRT.

ROPEBQUE Batch Queuing While ROPES Is Active Online

ROPEBQUE can be used to place report data on the ROPES Queue while ROPES is active under CICS. The routine will place the data for one or more reports on an intermediate data set. The ROPES BQTX Transaction can be invoked under CICS to complete the report queuing.

Input and Output

ROPEBQUE uses the following inputs:

- **FUNCTION** - an eight character constant or variable field specifying the required operation. The routine supports the following functions:

OPEN this function initiates processing by opening the intermediate data set. The user must place the one-to eight-character DD or the one-to seven-character DLBL name of the intermediate data set in the REPORT-NAME field;

WRITE this function stores the data in the REPORT-LINE field for the report named in the REPORT-NAME field. The user may intermix data for multiple reports by setting the appropriate report name value before each call; and

CLOSE this function will mark the end of the report named in the REPORT-NAME field. If the REPORT-NAME field is set to spaces the intermediate data set will be closed;

- **REPORT-NAME** - is an eight character constant or variable field used as described above; and
- **REPORT-LINE** - is a 133 character print line image. The first character must contain a ROPES carriage control character.

ROPEBQUE produces the following output:

- A queue data set, containing report data for spooling by the ROPES BQTX transaction.

ROPEBQUE provides a return code to indicate the results of request processing. The return codes (passed in general register 15 or special register RETURN-CODE) and their meanings are:

- 00, which indicates successful completion;
- 04, which indicates the intermediate file was not open when the WRITE or CLOSE function was requested, or was open when an OPEN function was requested;
- 08, which indicates that an I/O error occurred while processing the intermediate file;
- 12, which indicates that the intermediate file could not be opened; and
- 16, which indicates that the user did not specify a valid function.

Control

ROPEBQUE is controlled by the calling parameters passed when the service routine is invoked. ROPEBQUE is invoked as follows:

Assembler programs:

```

      .
      CALL ROPEBQUE, (FUNCTION, RPTNAME, RPTLINE)
      .
FUNCTION DS    CL8      FUNCTION CODE
RPTNAME  DS    CL8      REPORT NAME
RPTLINE  DS    CL133    REPORT LINE
      .

```

COBOL programs:

```

      .
77 FUNCTION-CODE PIC X(8).
77 REPORT-NAME  PIC X(8).
01 REPORT-LINE.
   05 CONTROL-CHARACTER PIC X(1).
   05 REPORT-DATA       PIC X(132).
      .
      CALL "ROPEBQUE" USING FUNCTION-CODE,
                          REPORT-NAME,
                          REPORT-LINE.

```

Job Control Statements

The user must provide the job control statements necessary to define the intermediate data set. The DD name must correspond to that used in the OPEN function.

NOTE: You must specify DA in the DD statement (OS) or DLBL statement (DOS) for this data set.

Intermediate File Definition

The user must define the intermediate file as a relative record direct access data set consisting of fixed length, 141 byte records. The data set must be available to ROPES online under CICS.

BQTX ROPES Batch Queue Transfer Transaction

FUNCTION: Transfer Data To ROPES

COMMAND: BQTX/data set/rptid

BQTX is the four character transaction code.

data set is the one- to eight-character symbolic name of the intermediate data set loaded by ROPEBQUE. The name must appear in the file control table (FCT).

rptid is the one- to eight-character report name of the report to be transferred. If omitted, all reports on the intermediate file will be transferred.

/ is the only valid delimiter between parameters. It should be omitted when it would be the last character in the command string. A blank in place of a delimiter terminates the command string.

In response to this command the specified report (or all reports) will be transferred from the ROPEBQUE-created intermediate data set to the ROPES queue. If the end of report indicator is not found in the intermediate file (written by ROPEBQUE in response to a CLOSE request), the operator will be given the choice of queuing the data already processed or of canceling the transfer.

Table Entries

One FCT entry is required for each data set to be used for batch queue transfer.

Example:

```
DFHFCT TYPE=DATASET,DATASET=dataset,
        ACCMETH=BDAM,SERVREQ=(GET,BROWSE),
        FILSTAT=(ENABLED,CLOSED),
        RECFORM=(FIXED,UNBLOCKED),
        BLKSIZE=141,LRECL=141,
        RELTYPE=HEX[,EXTENT=1 (DOS only)]
```


ROPEBQOR Batch Queue Reorganization Utility

Program ROPEBQOR can be used in a batch job to perform the following functions:

- Reorganize the ROPES Queue according to the Discard and Retention data values in the ROPERIB data set;
- Print report data on batch printers, based on a user-supplied retention time;
- Purge reports or parts of reports based on a user-supplied retention time; and
- Exclude one or more reports from any of the above operations.

Input and Output

ROPEBQOR uses these inputs:

- The ROPERIB data set, containing the report control information;
- The ROPEPCR data set, containing the printer status information;
- The ROPEQBR data set, containing the report text;
- The ROPERPI data set, containing the report to printer index; and
- An optional sequential data set containing control cards to direct the processing of reports.

ROPEBQOR produces these outputs:

- A message data set, containing program-generated status messages;
- An updated ROPERIB data set;
- An updated ROPEPCR data set;
- An updated ROPEQBR data set;
- One or more messages written to the console to allow automation tools to be aware of significant events;
- An optional report data set, containing the ROPES reports, in print image format, if the print option was selected; and
- An optional archive data set, containing the ROPEQBR file data records purged by the reorganization process.

ROPEBQOR provides a return code to indicate the results of program execution. The return codes and their meanings are:

- 00, which indicates successful completion;
- 04, which indicates that one or more reports were not found. The job step continues, ignoring the unknown reports;
- 08, which indicates that one or more errors were noted in the message data set. Processing ends and the job step is terminated; and
- 12, which indicates that the message data set was missing. The job step is terminated.

Control

ROPEBQOR is controlled by job control statements and utility control statements. The job control statements are required to execute or invoke ROPEBQOR and to define the data sets that are used and produced by the program. The utility control statements are used to control the functions of ROPEBQOR.

Job Control Statements

[Table I](#), [Table II](#) shows the job control statements necessary for using ROPEBQOR.

Table II ROPEBQOR JCL

Statement	Use
JOB	Initiates the job.
EXEC	Specifies the program name.
ROPEARC	Defines the ROPES Archive data set. The data set can be written to a tape or disk. This data set is required unless the \$NOARCH control statement is used, in which case this data set is optional and will be ignored if present.
ROPERIB	Defines the ROPES Report Information data set. This data set is required.
ROPEPCR	Defines the ROPES Printer Control data set. This data set is required.
ROPEQBR	Defines the ROPES Queue data set. This data set is required.
ROPERPI	Defines the ROPES Report to Printer index data set. This data set is required.
SYSIPT	Defines the control data set containing the utility control statements. This data set is optional.
SYSLST	Defines a sequential message data set. The data set can be written to a printer, a tape or a disk. Normally, this data set will be specified as a SYSOUT data set. This data set is required.
SYS006	Defines the print output data set. It can define a sequential data set on a printer, a tape or a disk. Normally, the data set will be defined as a SYSOUT data set. If the \$PRINT control statement is used, this data set is required,

Restrictions

- The ROPERIB, ROPEPCR, ROPEQBR, ROPERPI and SYSLST data sets are required for each use of ROPEBQOR.
- The SYSLST and SYS006 data sets have a logical record length of 133 bytes, and consist of fixed-length records with an ANSI control character in the first byte of each record. You may block the data sets if desired.
- The SYSIPT data set contains 80-byte card images.
- The ROPEARC data set contains variable length records with a maximum record size of 4100. The data set may be blocked if desired.

Utility Control Statements

The format of the utility control statements used to control ROPEBQOR is:

- Column 1 - contains the \$ character.
- Columns 2-10 - contains the function to be performed.
- Columns 20-28 - contains the parameter to be used with the function requested.

The valid functions are:

NOARCH This option causes the archiving of purged data to be suppressed. The archiving function is normally on and this function is required to suppress the recording of purged data. There is no parameter.

PRINT This option causes the printing of expired report data. The expiration time, expressed in hours, is entered in the parameter field of the control statement. If a value of 0000 is entered, the Retention Time value in the ROPERIB data set will be used for each report.

PURGE This option causes the purging (and optional archiving) of expired report data. The expiration time, expressed in hours, is entered in the parameter field of the control statement. If a value of 0000 is entered, the Retention Time value in the ROPERIB data set will be used for each report. This option, in

effect, changes the Discard value for all reports to 6 for this run only.

WTOMSG This option causes the batch reorganization utility to write certain messages to the console using the Write To Operator service. These messages may be trapped by your automation tools to take action as required. The messages echoed to the console (they are also written to the SYSLST data set) are: 2106, 2118, 2119 and 2129 (all are prefixed with "ROPES"). Message 2129 is new and acknowledges that the WTOMSG command was detected.

EXCLUDE This option causes the report named in the parameter field to be excluded from the PRINT and PURGE operations if they are specified. One EXCLUDE card is required for each report to be excluded. You may specify any number of EXCLUDE statements. Excluded reports will not be printed and will be purged according to their normal Discard and Retention Time assignments.

IGNORE This options causes the report named in the parameter field to be excluded from all processing. The report will not be reorganized, purged or printed during this run. One IGNORE card is required for each report to be excluded from all processing. You may specify any number of IGNORE statements.

The valid values for the parameter field of the control statement are:

Expiration

Time a four digit number, in the range of 0001 to 9999. This number indicates the number of hours that report data is to be retained on queue after its creation before it is considered eligible for processing by the PRINT and PURGE functions. A value of 0024 would indicate one day (24 hours). To hold data on queue for one week, a value of 0168 is required. Leading zeros are required, and help to prevent errors in preparing the data.

Report

Name a one- to eight-character string. The string names a report to be EXCLUDED or IGNORED. If the report is not found on the ROPERIB data set, the control statement will be ignored and a message will be issued.

Notes

If more than one NOARCH, PRINT or PURGE control statement is encountered, only the last will be in effect for the run.

If a report is named on multiple EXCLUDE or IGNORE statements, only the last occurrence will be in effect for the run.

If a report is named on both an EXCLUDE and an IGNORE statement, the IGNORE statement will override the EXCLUDE statement, regardless of the order of the statements in the Control data set.

ROPEBQOR Examples

1. In this example, the ROPES Queue is to be reorganized using the Discard and Retention values in the ROPERIB data set. Archiving is to take place, and no special processing is required.

```
//REORG JOB
//      EXEC   PGM=ROPEBQOR
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//SYSLST  DD SYSOUT=A
//ROPEARC DD DSN=ROPES.ARCHIVE,DISP=OLD
//ROPERIB DD DSN=ROPES.RIBFILE.BASE,DISP=OLD
//ROPEPCR DD DSN=ROPES.PCRFILE.BASE,DISP=OLD
//ROPEQBR DD DSN=ROPES.QBRFILE.BASE,DISP=OLD
//ROPERPI DD DSN=ROPES.RPIFILE.BASE,DISP=OLD
//SYSIPT  DD DUMMY
```

2. In this example, the ROPES Queue is to be reorganized using a user-supplied expiration value of 5 days (0120 hours). No archiving is to be performed, and two reports are to be excluded from purge processing.

```
//REORG JOB
//      EXEC   PGM=ROPEBQOR
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//SYSLST  DD SYSOUT=A
//ROPERIB DD DSN=ROPES.RIBFILE.BASE,DISP=OLD
//ROPEPCR DD DSN=ROPES.PCRFILE.BASE,DISP=OLD
//ROPEQBR DD DSN=ROPES.QBRFILE.BASE,DISP=OLD
//ROPERPI DD DSN=ROPES.RPIFILE.BASE,DISP=OLD
//SYSIPT  DD *
$NOARCH
$PURGE           0120
$EXCLUDE         REPORT01
$EXCLUDE         REPORT03
/*
```

3. In this example, the ROPES Queue is to be reorganized using the Discard and Retention values in

the ROPERIB data set, but three reports are not to be processed. Archiving will occur.

```
//REORG JOB
//      EXEC   PGM=ROPEBQOR
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//SYSLST  DD SYSOUT=A
//ROPEARC DD DSN=ROPES.ARCHIVE,DISP=OLD
//ROPERIB DD DSN=ROPES.RIBFILE.BASE,DISP=OLD
//ROPEPCR DD DSN=ROPES.PCRFILE.BASE,DISP=OLD
//ROPEQBR DD DSN=ROPES.QBRFILE.BASE,DISP=OLD
//ROPERPI DD DSN=ROPES.RPIFILE.BASE,DISP=OLD
//SYSIPT  DD *
$IGNORE          REPORT01
$IGNORE          REPORT03
$IGNORE          REPORT05
/*
```

4. In this example, the ROPES Queue is to be reorganized using a user-supplied expiration value of 3 days (0072 hours). All reports that are over 2 days old (0048 hours) are to be printed. No archiving is to take place. One report is to be excluded from PRINT and PURGE processing, while another report is to be excluded from all processing. Significant messages will be written to the console.

```
//REORG JOB
//      EXEC   PGM=ROPEBQOR
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//SYSLST  DD SYSOUT=A
//ROPERIB DD DSN=ROPES.RIBFILE.BASE,DISP=OLD
//ROPEPCR DD DSN=ROPES.PCRFILE.BASE,DISP=OLD
//ROPEQBR DD DSN=ROPES.QBRFILE.BASE,DISP=OLD
//ROPERPI DD DSN=ROPES.RPIFILE.BASE,DISP=OLD
//SYSIPT  DD *
$NOARCH
$WTOMSG
$PRINT          0048
$PURGE          0072
$EXCLUDE        REPORT03
$IGNORE          REPORT05
/*
```

Note:

There is a sample of the JCL used to run this program supplied in the ROPES.SOURCE data set under the name RUNBQOR.

ROPEUTIL ROPES Utility Program Driver

Program ROPEUTIL is a ROPES Master Utility processor. ROPEUTIL performs control statement format checking and directs the appropriate utility function to perform the request. Using ROPEUTIL all utility functions use a similar control statement format, multiple utility functions can be accomplished in the same job step, and required JCL statements remain the same for most utility functions.

Input and Output

For most utility functions, ROPEUTIL will require the following inputs:

- A sequential card image data set containing control cards to direct the processing of one or more utility functions. This data set is always required.
- The ROPERIB data set, containing the report control information.
- The ROPEPCR data set, containing the printer status information.
- The ROPEQBR data set, containing the report text.

ROPEUTIL produces the following outputs:

- A message data set, containing utility status messages. This data set is always required.
- An updated ROPERIB data set.
- An updated ROPEPCR data set.
- An updated ROPEQBR data set.

Additional input or output data sets may be required for certain functions. The specific function descriptions will include information about these data sets.

ROPEUTIL provides a return code to indicate the results of program execution. The return codes and their meanings are:

- 00, which indicates successful completion;
- 04, which indicates that selected reports or printers were not found. The job step continues, ignoring the unknown resources;

- 08, which indicates that one or more errors were noted in the message data set. Processing ends and the job step is terminated; and
- 12, which indicates that the message data set was missing. The job step is terminated.

Control

ROPEUTIL is controlled by job control statements and utility control statements. The job control statements are required to execute or invoke ROPEUTIL and to define the data sets that are used and produced by the program. The utility control statements are used to control the functions of ROPEUTIL.

Job Control Statements

Table [III](#) shows the job control statements necessary for using ROPEUTIL.

The EXEC DD card specification can include the new PARM option "COMWSLEN". The COMWSLEN parameter option is required for all ROPES External Interface utility functions. The value for COMWSLEN may be specified in bytes up to a value of 31K. The value may also be specified as "nnK" or "nnk". Example: PARM='COMWSLEN=24K'. The default supplied by our JCL is 24K but may be increased to any value just under 32K.

Table III ROPEUTIL JCL

Statement	Use
JOB	Initiates the job.
EXEC	Specifies the program name and any PARM values.
STEPLIB	Specifies the ROPES load library data set.
ROPERIB	Defines the ROPES Report Information data set. This data set may be optional, depending on the utility function used. Please see the utility function description for data sets required.
ROPEPCR	Defines the ROPES Printer Control data set. This data set may be optional, depending on the utility function used.
ROPEQBR	Defines the ROPES Queue data set. This data set may be optional, depending on the utility function used.
ROPEIPT	Defines the control data set containing the utility control statements. This data set is required.
ROPELST	Defines a sequential message data set. The data set can be written to a printer, a tape or a disk. Normally, this data set will be specified as a SYSOUT data set. This data set is required.
ROPERPT	Defines a sequential message data set. The data set can be written to a printer, a tape or a disk. Normally, this data set will be specified as a SYSOUT data set. This data set is optional but required for all ROPES External Interface options.
SYSPRINT	Defines a sequential message data set. This data set is normally allocated to SYSOUT, but may be allocated to a printer, tape, or disk. This data set is optional, but required for all ROPES External Interface utility functions.
SYSMDUMP	Defines a sequential output listing data set for dump output. This data set is normally allocated to SYSOUT, but may be allocated to a printer, tape, or disk. This data set is optional but required for all ROPES External

Restrictions

- The ROPEIPT and ROPELST data sets are required for each use of ROPEUTIL. The ROPERIB, ROPEPCR, ROPEQBR and other data sets may be required, depending on which utility functions you request.
- The ROPELST data set has a logical record length of 133 characters, and consists of fixed-length records with an ANSI control character in the first character of each record. You may block the data set if desired.
- The ROPEIPT data set contains 80-character card images.

Utility Control Statements

The ROPEUTIL control statement format is flexible.

Please follow these rules when coding the control statements:

- Control statements start with a COMMAND and end with a period.
- A PARAMETER must follow a COMMAND or another PARAMETER.
- A left parenthesis must immediately follow a PARAMETER. The parenthesis begins the PARAMETER, which contains a single VALUE or VALUE LIST separated by commas. The PARAMETER ends with a right parenthesis.
- Any number of blanks may appear between PARAMETERS, provided a comma appears after the right parenthesis of the previous PARAMETER.
- A VALUE LIST may be continued in any column prior to column 72, provided a comma is placed at the end of the last value coded.
- No continuation character is required to continue control statements.
- Multiple control statements may be coded in one execution of the ROPEUTIL program. Control statements are processed in the order they occur.
- Statements that start with an asterisk ('*') in column 1 are treated as a comment. The line will be echoed in the CARDIN list and then ignored.

Termination of the control statements and execution of ROPEUTIL is triggered by the end-of-file on the ROPEIPT data set.

The format of all utility control statements is shown below:

```
COMMAND PARAMETER (value,value,...),
PARAMETER (value,value,...),
PARAMETER (value,value,...).
```

The number of parameters associated with a command depends on the utility function being used. This is also true concerning the values of the parameters.

Alternate Facility File Resynchronization Function

Function

This utility is used to read the ROPES RIB file and add any missing report entries to the ROPES Alternate Facility RIB file.

You can “clear” the Alternate Facility RIB file by first deleting all the report records (but not the control records at the beginning of the file, all of whose keys begin with a space). Then run this utility to “Add” all the reports back into the Alternate Facility. If you do this, you should also clear the Alternate Facility QBR file, as none of the records in that file will be referenced by the cleared ALR file and the space they occupy will be wasted.

Data Sets

The ROPERIB and ROPEALR files are required by the Resynchronization utility function.

Command Format

```
ALTSYNCh REPortS(@ALL|report,report..)
      [,EXCLude(report,report,...),]
      [,MESSAGES(option)|,MSGs(option)].
```

REPORTS

is a required parameter. The REPORTS parameter specifies the ROPES reports that are to be ALTSYNChronized. To select all reports, code “@ALL.” REPORTS may specify a single report name or a list of report names. Wild cards may be used.

EXCLUDE

is an optional parameter. EXCLUDE is used to specify ROPES reports that should be excluded from the operation. EXCLUDE must specify a single report name, or a list of report names. Wild cards may be used. Report names that are specified on both the EXCLUDE and REPORTS parameters will be **excluded** from processing.

MESSAGES (or MSGS)

requests detailed or summary messages. If

extended messages are requested, additional messages will be printed to indicate each report added to the Alternate Facility. If the MESSAGES parameter is omitted, extended messages will be suppressed.

The permitted values for the parameter fields of the control statement are:

report a one to eight-character string. The string names a ROPES report to be synchronized or EXCLUDED. A generic report name may be specified by using the asterisk (*) as a wild card character. If all reports are to be Synchronized then a single report name may be replaced with the constant **@ALL**. The **@ALL** parameter value must not be used with other report names. If a report is not found in the ROPERIB data set, the report will be ignored and a warning message will be issued.

option controls the type of messages issued by the ALTSYNChronize function. The values for **option** in the MESSAGES parameter are **VERBOSE** or **TERSE**. Any value other than these will cause extended messages to be turned off.

ALTSYNChronize Utility Examples

1. In this example, report controls for all ROPES reports found in the ROPERIB data set are to be ALTSYNChronized. Only summary messages will be issued.

```
//ALTSYNChJOB
//          EXEC PGM=ROPEUTIL
//STEPLIB   DD DSN=ROPES.LOADLIB,DISP=SHR
//ROPERIB   DD DSN=ROPES.RIBFILE.BASE,DISP=SHR
//ROPEALR   DD DSN=ROPES.RIBALTF.BASE,DISP=SHR
//ROPELST   DD SYSOUT=A
//ROPEIPT   DD *
ALTSYNCh REPORTS (@ALL) .
//
```

2. In this example, all ROPES reports will be ALTSYNChronized except for those reports named by the EXCLUDE parameter value list. Reports RPT001 through RPT010 will be excluded from the process.

```
//ALTSYNChJOB
//      EXEC PGM=ROPEUTIL
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//ROPERIB DD DSN=ROPES.RIBFILE.BASE,DISP=SHR
//ROPEALR DD DSN=ROPES.RIBALTF.BASE,DISP=SHR
//ROPELST DD SYSOUT=A
//ROPEIPT DD *
ALTSYNCh REPORTS (@ALL),EXCLUDE (RPT001,RPT002,
RPT003,RPT004,RPT005,
RPT006,RPT007,RPT008,RPT009,RPT010) .
//
```

3. In this example, two separate ALTSYNCh control statements are used to process different groups of ROPES reports. Each ALTSYNCh statement will cause a separate execution of the function. The first control statement will cause reports “RPT001” through “RPT010” to be processed. The second control statement will cause all reports beginning with characters “INVEN” to be processed, excluding reports “INVEN20”, and “INVEN21”.

```
//ALTSYNChJOB
//      EXEC PGM=ROPEUTIL
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//ROPERIB DD DSN=ROPES.RIBFILE.BASE,DISP=SHR
//ROPEALR DD DSN=ROPES.RIBALTF.BASE,DISP=SHR
//ROPELST DD SYSOUT=A
//ROPEIPT DD *
RESYNCh REP (RPT001,RPT002,RPT003,RPT004,RPT005,
RPT006,RPT007,RPT008,RPT009,RPT010) .
ALTSYNCh REP (INVEN*),MSGS (VERBOSE) ,
EXCL (INVEN20,INVEN21) .
//
```

Note there are some differences in this example. The abbreviated versions of the command and parameters have been used. In the second control statement, the second parameter “EXCL” has been continued on the next line, and detailed messages will be issued. The short form of the MESSAGES parameter (MSGS) has been used to request extended messages. A wild card has been used to perform a generic selection of reports.

The Controls Utility Function

Function

The controls utility function initializes or changes system parameter records in the parameter file. The parameter types that can be initialized or changed are; Screen Handler Options, ROPES System Options, ROPES PF Key Definitions, and Maintenance Logging Controls.

Data sets

In addition to the ROPELST and ROPEIPT data sets, the CONTROLS utility function requires access to the ROPEPRM data set. This file is a VSAM KSDS file containing parameter records of different types. These parameter records establish certain global options for ROPES. The ROPEPRM file is created and initialized by the AMSCNTL and REPROJCL job members run as part of the installation process.

Command Format

Each **CONTROLS** statement defines a different parameter record type. All record types are stored in the ROPES parameter file.

A parameter keyword common to the four **CONTROLS** statements is the **PTYPE** parameter. The **PTYPE** parameter specifies the type of parameter record being defined. Each parameter is described below.

PTYPE

Specifies the parameter record type being processed. Four possible values can be coded for the **PTYPE** parameter. These values are; "SCREEN-HANDLER", "SYSPARMS", "LOGGING", and "PFKEY". **PTYPE** is a required parameter of the **CONTROLS** statement.

Options For The Screen-Handler Record Type

The following parameter keyword options apply to

the "SCREEN-HANDLER" parameter record type only.

DESC

Initializes the description field of the parameter record to the text value specified. The **DESC** parameter value may be up to 40 characters long.

TRANPFX

Specifies the transaction prefix that will be used for ROPES online transactions. The **TRANPFX** parameter value must be 2 characters long, and must specify the first two characters of any ROPES transaction identification. The default value for **TRANPFX** is "RO". Changing this default will require a change to the **PCT** definitions for all of the **REPORT** distribution, and system status display transactions.

PROGPFX

Specifies the program prefix that will be used for ROPES CICS programs. The **PROGPFX** parameter value must be 3 characters long, and specifies the first 3 characters of a ROPES program name. The default for **PROGPFX** is "R51". Changing this value will require changes to be made to the **PPT** definitions for all Report Distribution, and System Status Display programs.

LANG

Specifies the language code that will be used when messages, titles, and screens are displayed at a CICS terminal. The only acceptable value for the language option is "001" which specifies **English**. The only language supported at this time is English. When other languages are supported, messages, titles, and whole screens will change to a different language by simply changing this parameter value.

MSGFILE

Specifies the full name of the ROPES message file. The default message file name is "ROPEMSG". The name specified must point to a valid **FCT** definition for the ROPES message file.

HLPFILE

Specifies the full name of the ROPES help file. This parameter option is set to “ROPEHLP” by default. The name specified here must point to a valid FCT definition for the help file.

ENVTABL

Specifies the suffix name of the ROPES Screen Handler environment table. The full table name is achieved by adding the program prefix value to the beginning of the environment table suffix name. The environment table suffix is limited to 5 characters, and should be set to a value that points to a valid Screen Handler environment table. ROPES uses the value “ENVTB” by default. This value should not be changed unless you have established a customized version of the environment table, and used a different name.

COLORS

Specifies the default colors that will be used for various types of input and display map fields. This parameter option directs the Screen Handler program to associate colors with panel field types. The color values that may be coded for this parameter option are positional. There are eight (8) color positions, and each one represents a different type of input or display field. The colors are entered as hexadecimal values. The colors and their field types are described below in order of their positions.

Default Field Color - Specifies the color that will be used if no color has been directly specified for the panel field. The default value set for this field is null or binary zeros. This will cause the default BMS color to be used.

Label Field Color - Specifies the color that will be used for a **label** type panel field. Label type panel fields are descriptive fields, or name fields that immediately precede an input field. The default color for label type fields is **blue** (Hex value “F1”).

Error Message Field Color - Specifies the color that will be used for an error message panel field. If the Screen Handler program must issue an error message, it will change the message field to this color. The error message field color is **red** (F2) by default.

Value Field Color - Specifies the color that will be used for a value field, or input panel field.

The default color for this type of field is **green** (F4).

Error Value Field Color - Specifies the color that will be used for a value (input) field that is in error. The Screen Handler program highlights any input field that is in error in this color. The default color value for this type of panel field is **yellow** (F6).

Information Field Color - Specifies the color that will be used for an information type field. An information field is a heading, an options list, or some other panel field which contains information that assists the operator in using an input field. The default value for an information field is **neutral** or **white** (F7).

Highlighted Field Color - Specifies the color that will be used for a highlighted panel field. A highlighted field may be a heading, or a title field which marks the area of a screen as serving a unique purpose. The default value for a highlighted field is **turquoise** (F5).

Reserved Field Color - This color is not associated with a field type at this time, but is reserved for establishing a new field type if necessary. The default color assigned to this field is **pink** (F3).

System Parameter Record Options

The following parameter options are associated with the SYSPARMS type parameter record. The SYSPARMS parameter record contains a number of global system options for ROPES. At the present time this parameter record controls options for the Report Distribution, Job Editor/Submit Facility, and ROPES System Status Display only. The parameter option fields are described below.

DESC

Specifies a description that will be associated with the parameter record. The description parameter may contain text up to 40 characters in length.

MROREGN

Specifies the MRO region number ROPES is installed under. This option tells ROPES which CICS region the print task should run in. If ROPES is running in the same CICS region as the Terminal Owning Region (TOR), then this

parameter value should be set to “01”. If the printer task should run in another region, then you must specify the number of the region here. This parameter must also match a printer transaction identification defined in the remote or local region. The printer task transaction identification is normally R#01.

TDQUE

Specifies the extra-partition transient data queue name ROPES can use to route error and activity messages. The value for this parameter must specify a 4 character transient data queue name defined in the CICS Destination Control Table (DCT).

ITRNPFX

Specifies the prefix name used for ROPES internally initiated, or auto-initiated transactions. An example of this type of ROPES transaction would be the printer task transaction name (R#01). The ITRNPFX value must be 2 characters in length, and would specify the prefix name (first two characters) of any auto-initiated ROPES transaction.

MTERMID

Specifies the master terminal identification for ROPES. This parameter must contain a valid 4 character terminal identification. The terminal name specified should be defined in the CICS Terminal Control Table, or in the CSD file.

ACCDEST

Specifies the name of an extra-partition transient data queue that will be used to receive ROPES activity accounting records. The parameter value must not exceed 4 characters in length, and should specify the name of a valid extra-partition destination.

SECUPD

Specifies whether security checks will be activated for file updates, and other potentially secured functions. Setting a value of ‘Y’ (Yes) for this keyword parameter causes the program named by the **SECEXIT** parameter to be called. Setting a value of ‘N’ (No) causes the call to the security exit to be skipped.

SECEXIT

Specifies the full name of the security exit program. By default this program is named ‘R51RSECX’. As provided, this program

performs an “EXEC CICS QUERY SECURITY” to determine if the user has access to the resource or function. *This requires the use of an external CICS security management program like RACF.* Please refer to the ‘ROPES Administrator’s Guide’ and the chapter on ‘Security’ for more information regarding the use and implementation of this security exit.

INTRDRQ

Specifies the name of the Transient Data Extra-partition queue that is allocated to the Internal Reader. The Transient Data Queue name must be 4 characters in length. This extra-partition queue is used by the job submission facility under the JCL Editor transaction ROJB. Please consult the Administrators Guide in the chapter ‘Installation Steps’ under ‘STEP 3. Add the Required Entries into the CICS Resource Definitions’ for more information on the definition of the extra-partition destinations required by ROPES. The default transient data queue name is JESR.

RFILES

Specifies the names of all the files that will be used by ROPES. The list of file names specified by this parameter cannot exceed 10 names. Each file name must not exceed 8 characters. Each file name has its position within the list. The order of this list is presented in the descriptions below.

Alternate Queue - Specifies the name of the alternate ROPES spool file. The default name provided is “ROPEALQ”.

Alternate RIB - Specifies the name of the alternate Report Information Block (RIB) file. The default name provided for this file is “ROPEALR”.

Batch Queue - Specifies the name of the Batch Queue file which is used by the BQTX transaction for transferring a report from a BDAM file to the ROPES spool. Please ROPEBQUE in this section for more information about the ROPEBQUE facility. The default name provided for this file is “ROPEBQX”.

Job File - Specifies the name of the JCL (Job) file which is to be used with the ROPES job submit facility, and JCL editor. The default name provided for this file is “ROPEJOB”.

Printer File - Specifies the name of the file which contains the ROPES printer definitions. The default name provided for this parameter is "ROPEPCR".

Primary Spool - Specifies the name of the ROPES primary spool file. This file contains the actual report data from which the printer tasks retrieve report data. The default name provided for this parameter is "ROPEQBR".

Distribution Definition File - Specifies the name of the file which contains report distribution definitions. The default name provided for this file is "ROPERDDD".

Distribution Translation File - Specifies the name of the file which contains report distribution translation tables. The default name provided for this parameter is "ROPERDTD".

Report Information File - Specifies the name of the file which contain ROPES report definitions. The default value provided for this parameter is "ROPERIB".

Report/Printer Index - Specifies the name of the file which contains all the associations of reports assigned to printers. The default value provided for this parameter is "ROPERPI".

LPR Print Options File - Specifies the name of the ROPES LPR Print Options file. This file contains the LPR Printer/Report options that will be used to send report output to Line Printer Daemon servers over the Internet or intra-net. The default value for this parameter is ROPELPRO.

PF Key Parameter Record Options

The parameter options described below are all associated with the PF Key parameter record type. The PF Key parameter record defines the PF keys that can be used with a given ROPES panel. This parameter record is only used with the Report Distribution, System Status Display, and Job submission facilities in this release. PF Key control statements are provided for all of the required ROPES screens in the job member RUNBUILD located in the ROPES distribution source data set.

DESC

Specifies the description for the PF key parameter record. The description text may be

up to 40 characters in length, and should describe this parameter record as being a PF key type parameter record.

PANEL

Specifies the panel **mapset** name that this PF key parameter record is associated with. The panel name must be the name of an existing **mapset**, and can be up to 7 characters in length.

KEY1 To KEY12

Specifies the PF key label, number, and hexadecimal AID values for up to 12 PF keys. The KEY1 parameter defines the PF key values for PF key 1, KEY2 defines the values for PF key 2, and so on up to parameter KEY12. For each "KEYnn" parameter there are four positional values. These positional values are; label name, PF Key label number, primary AID value, and secondary AID value. For example:

KEY1(l1l1,nn,a1,a2) where;

l1l1 = label,

nn = function number,

a1 = primary AID value, and

a2 = secondary AID value.

Logging Parameter Record Options

The parameter options described below are associated with the "Logging Parameter Record". In this release the Logging Parameter Record controls logging for the Report Distribution System only.

DESC

Specifies the description for the logging parameter record. The description value may be up to 40 characters in length.

LOGTYPE

Specifies whether record maintenance logging will be performed, and the destination of the logging records. The "LOGTYPE" parameter value may be 1 character long, and may be one of three possible choices. The first choice is "N" which specifies that no logging will be performed. This is the default. The second choice is "T" which specifies that logging will be performed, and log records will be written to extra-partition transient data. The third choice is "U" which specifies that logging will be performed by a **user exit**. The user exit will be

called to determine the destination and handling of the logging records.

TDQUE

Specifies the name of the extra-partition transient data queue if LOGTYPE was set to "T". The transient data queue name may be up to 4 bytes, and should specify the name of an extra-partition queue defined in the Destination Control Table (DCT). The default value for this parameter option is "CSMT". Please consult the "Auditability And Accounting" section of the Administrator's Guide for an in depth discussion on the logging exit program, and the logging destination.

LOGEXIT

Specifies the name of a user exit program which will be invoked to handle processing of log records. Requirements of the user exit program are described in the chapter "Auditability And Accounting". The "LOGEXIT" parameter must specify a full program name up to 8 characters in length.

Controls Utility Example

The figure shown below shows an example of the JCL and control statements required to execute the CONTROLS utility function properly. The control statements that are required by the RUNBUILD JCL for ROPES installation are not fully depicted by this example. If you wish to see the complete set of CONTROLS statements please examine the JCL in member RUNBUILD located in the ROPES distribution source data set.

```

//          JOB (***) PLEASE PLACE YOUR JOB CARD HERE (***)
//*+-----+
//*| THIS STEP CREATES THE PARAMETER RECORDS REQUIRED BY THE |
//*| SCREEN HANDLER PROGRAM.                               |
//*+-----+
//CONTRLS EXEC PGM=ROPEUTIL
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//ROPELST DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//ROPEPARM DD DSN=ROPES.PRMFILE.BASE,DISP=SHR
//ROPEIPT DD *
CONTROLS PTYPE (SCREEN-HANDLER) ,
        DESC (SCREEN HANDLER PROGRAM CONTROL OPTIONS) ,
        TRANPFX (RO) , PROGPFX (R51) , LANG (001) , MSGFILE (ROPEMSG) ,
        HLPFILE (ROPEHLP) , ENVVTAB (ENVVTB) , COLORS (0,1,2,4,6,7,5,3) .

CONTROLS PTYPE (SYSPARMS) ,
        DESC (ROPES GENERAL SYSTEM CONTROLS) ,
        MROREGN (01) , TDQUE (CSMT) , ITRNPFX (R#) , MTERMID (TTTT) ,
        SECUPD (N) , SECEXIT (R51RSECX) , INTRDRQ (INTR) ,
        ACCDEST (CSMT) , RFILES (ROPEALQ, ROPEALR, ROPEBQX,
        ROPEJOB, ROPEPCR, ROPEQBR, ROPERDDD, ROPERDTD, ROPERIB, ROPERPI) .

CONTROLS PTYPE (LOGGING) ,
        DESC (MAINTENANCE LOGGING OPTIONS) ,
        LOGTYPE (N) , TDQUE (CSMT) .

CONTROLS PTYPE (PFKEY) ,
        DESC (PFKEY DEFINITION FOR SCREEN R51HLP0) ,
        PANEL (R51HLP0) , KEY1 (HELP, 01, F1, C1) , KEY3 (EXIT, 03, F3, C3) ,
        KEY4 (RETN, 04, F4, C4) , KEY7 (PREV, 07, F7, C7) ,
        KEY8 (NEXT, 08, F8, C8) , KEY10 (TBOC, 10, 7A, 4A) .

        . . . CARDS CONTINUED . . .

```

Figure 1 Sample Controls Utility JCL

The LOADHLP Utility Function

Function

The help information load utility loads the ROPES help file with help text. The help file is a VSAM KSDS file. The help loader utility reads the pre-formatted help text from a physical sequential file, and stores it in the VSAM cluster by key. This permits context sensitive help requests to be satisfied with direct access. All help information is converted until the end-of-file condition is raised on the ROPEHLPI file.

Data sets

In addition to the ROPELST and ROPEIPT files, the LOADHLP utility function requires the ROPEHLPF, and ROPEHLPI files.

The ROPEHLPF file is a VSAM KSDS file allocated during the installation of ROPES. This file will ultimately contain the help information online ROPES transactions will use.

The ROPEHLPI file contains the formatted input help text that will be converted and loaded by the LOADHLP function. The ROPEHLPI input file is a physical sequential file with a maximum record length of 78 characters. The input text file contains help text, and “**help markers**” which are used to mark chapters, headings, and context sensitive help locations. The help text input file for ROPES is provided with the installation materials, and is installed by the RUNBUILD job member. The format and details of the help text markers is described below.

Help Input Text And Markers

The input help text file must contain fixed length records. The logical record length must be 78. Any block size which is a factor of 78 can be used.

The help text must be modified to contain chapter, heading, and field header entry markers. These markers define the location of chapter titles, heading titles, and sub-heading titles which are used by the help loader utility to build a table of contents. **Only one marker is permitted to a line of help text.**

Markers must be placed in the text in a manner which surrounds the chapter or heading titles. Each marker is surrounded by a pair of less than/greater than symbols (“<” “>”). In most cases a pair of markers is required to surround a heading. The markers which can be used are described in the table below.

Each marker shown in the table below must be surrounded by a pair of less than and greater than symbols (“<” “>”).

1. **CH or ch** - Marks a new chapter.
2. **H1 or h1** - Marks a heading level 1.
3. **H2 or h2** - Marks a heading level 2.
4. **H3 or h3** - Marks a heading level 3.
5. **H4 or h4** - Marks a heading level 4.
6. **HE or he** - Marks a field heading entry.
7. **NP or np** - Marks a new page.

Chapter markers and heading markers are placed around the chapter and heading text as in the sample below.

< marker > Heading Text < marker >

Field heading entries (“**He**”) are used where context sensitive help is required for a screen field. These markers are coded differently than the chapter and heading title markers. The field heading entry marker uses a single marker in the beginning of the entry, and includes the mapset name and the field’s help name. The field entry marker must be coded starting in column 1 of the text record. The help name is assigned to the panel field when the map is being created. An example of this type of marker entry appears below. The word “marker” shown below must be replaced with letters “**he**”.

< marker > R51RD01OPTIONS

R51RD01 is the name of the mapset, and OPTIONS is the help name assigned to this context sensitive help entry. The help text for this field usually appears immediately after the field heading entry marker.

The new page marker is specified as a single marker at the location where the new page is required. This marker causes an additional page break which can be used when paging through help, or locating a page while using the ROPES online help facility.

Command Format

The utility control statement which invokes the help loader utility is the “LOADHLP” utility control statement. The utility control command may also be shortened to “LDHLP”. This utility control statement has only one parameter which is “LPP”. A sample of the statement is shown below.

LOADHLP LPP(11).

The “LPP” parameter specifies the number of lines that the on-line help facility will use as a scroll value for one page. This number does not actually have to agree with the number of lines in an actual page of text. The value for LPP must be numeric, and must have a value in the range of 1 to 255.

LOADHLP Utility Example

An example of the load help utility JCL is shown in the figure below. In this example, the scroll value or page size is 11 lines per page. This is the same number of lines that appear on the online help display screen. So one page of help text is equal to 1 screen of help text. A version of this JCL member has been provided in the ROPES.SOURCE library, in member RUNLDHLP.

```
//          JOB (***) PLEASE PLACE YOUR JOB CARD HERE (***)
//**+-----+
//**| THIS STEP LOADS YOUR ROPES HELP FILE WITH HELP INFORMATION |
//**| FROM THE HELP TEXT INPUT FILE.                               |
//**|                                                               |
//**| Please consult the Programmer's Guide for information about |
//**| format of the input to the help load utility.                |
//**+-----+
//LOADHLP EXEC PGM=ROPEUTIL
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//ROPELST DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//ROPEHLPF DD DSN=ROPES.HLPFILE.BASE,DISP=SHR
//ROPEHLPF DD DSN=ROPES.HLPFILE.TEXT,DISP=OLD,VOL=SER=ROPE80,UNIT=CART,
//          LABEL=(18,SL,,IN)
//ROPEIPT DD *
//          LOADHLP LPP(11).
//**
//
```

Figure 2 Sample LOADHLP Utility JCL

The Purge Utility Function

Function

Performs a purge of all reports or selected reports, and automatically resets report information for all associated printers.

Data sets

The ROPERIB, ROPEPCR, and ROPEQBR files are required by the Purge utility function.

Command Format

```
PURge REpOrts(report,report,...),
      EXCLude(report,report,...),
      MESSAGEs(option).
```

REPORTS

is a required parameter. The REPORTS parameter specifies the ROPES reports that are to be purged. To select all reports, REPORTS must specify “@ALL”. REPORTS may specify a single report name or a list of report names. Wild cards may be used.

EXCLUDE

is an optional parameter. EXCLUDE is used to specify ROPES reports that should be excluded from the purge operation. EXCLUDE must specify a single report name, or a list of report names. Wild cards may be used. Report names that are specified on both the EXCLUDE and REPORTS parameters will be **excluded** from purge processing.

MESSAGES (or MSGS)

requests detailed or summary messages. If extended messages are requested, additional messages will be printed to indicate each printer reset to reflect the cleared report, the number of Queue records deleted from the ROPEQBR file, and the number of each Queue record out of range with the control information in the Report Information Block. If the MESSAGES parameter is omitted, extended messages will be suppressed.

The permitted values for the parameter fields of the control statement are:

report a one to eight-character string. The string names a ROPES report to be PURGED or EXCLUDED. A generic report name may be specified by using the asterisk (*) as a wild card character. If all reports are to be purged then a single report name may be replaced with the constant @ALL. The @ALL parameter value must not be used with other report names. If a report is not found in the ROPERIB data set, the report will be ignored and a warning message will be issued.

option controls the type of messages issued by the PURGE function. The values for OPTION in the MESSAGES parameter are **VERBOSE** or **TERSE**. Any value other than these will cause extended messages to be turned off.

Purge Utility Examples

1. In this example, report data for all ROPES reports found in the ROPERIB data set is to be purged. All report data is purged from the ROPEQBR data set, and report information for all associated printers in the ROPEPCR file is reset to reflect the purge. Only summary messages will be issued.

```
//PURGE JOB
//      EXEC PGM=ROPEUTIL
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//ROPERIB DD DSN=ROPES.RIBFILE.BASE,DISP=SHR
//ROPEPCR DD DSN=ROPES.PCRFILE.BASE,DISP=SHR
//ROPEQBR DD DSN=ROPES.QBRFILE.BASE,DISP=SHR
//ROPELST DD SYSOUT=A
//ROPEIPT DD *
PURGE REPORTS (@ALL) .
//
```

2. In this example, all ROPES reports will be purged from the ROPES Queue except for those reports named by the EXCLUDE parameter value list. Reports RPT001 through RPT010 will be excluded from the purge. All printers associated with cleared reports are updated to reflect the purge of report data.

```
//PURGE JOB
//      EXEC PGM=ROPEUTIL
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//ROPERIB DD DSN=ROPES.RIBFILE.BASE,DISP=SHR
```

```
//ROPEPCR DD DSN=ROPES.PCRFILE.BASE,DISP=SHR
//ROPEQBR DD DSN=ROPES.QBRFILE.BASE,DISP=SHR
//ROPELST DD SYSOUT=A
//ROPEIPT DD *
PURGE REPORTS (@ALL), EXCLUDE (RPT001, RPT002,
RPT003, RPT004, RPT005,
RPT006, RPT007, RPT008, RPT009, RPT010) .
//
```

3. In this example, two separate PURGE control statements are used to purge different groups of ROPES reports. Each purge statement will cause a separate execution of the purge function. The first purge control statement will cause reports “RPT001” through “RPT010” to be purged from the ROPES Queue. The second purge control statement will cause all reports beginning with characters “INVEN” to be purged, excluding reports “INVEN20”, and “INVEN21”. All printers associated with cleared reports are updated to reflect the purge of report data.

```
//PURGE JOB
// EXEC PGM=ROPEUTIL
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//ROPERIB DD DSN=ROPES.RIBFILE.BASE,DISP=SHR
//ROPEPCR DD DSN=ROPES.PCRFILE.BASE,DISP=SHR
//ROPEQBR DD DSN=ROPES.QBRFILE.BASE,DISP=SHR
//ROPELST DD SYSOUT=A
//ROPEIPT DD *
PUR REP (RPT001, RPT002, RPT003, RPT004, RPT005,
RPT006, RPT007, RPT008, RPT009, RPT010) .
PUR REP (INVEN*), MSGS (VERBOSE),
EXCL (INVEN20, INVEN21) .
//
```

Note there are some differences in this example. The abbreviated versions of the command and parameters have been used. In the second control statement, the second parameter “EXCL” has been continued on the next line, and detailed messages will be issued. The short form of the MESSAGES parameter (MSGS) has been used to request extended messages. A wild card has been used to perform a generic selection of reports.

The Register Utility Function

Function

The register utility function registers your company information with ROPES. This utility function is executed as part of the installation of ROPES. ROPES uses your company information along with a security code, provided by Axios, to grant your company use of the product. The register utility grants four days access from the date the register utility is run. This period allows you time to implement the security code provided by Axios which grants you access to the product on your CPU. If you do not have the security code, you must obtain it from Axios.

Data Sets

The register utility uses the following files as input files.

ROPEIPT - This file contains the utility control statements for the register utility function. This file is a physical sequential file containing fixed length 80 byte records. This file is a required file.

The following files are produced as output from the register utility function.

ROPELST - This file is the utility output message file. This file is a physical sequential file containing fixed length 133 byte records. This file is normally allocated to SYSOUT. This is a required file.

ROPEPARM - This file is the ROPES parameter file. The parameter file is a VSAM KSDS file. Attributes for this file can be obtained from the AMSCNTL member located in the distribution source data set. This is a required file for this utility function.

Utility Control Statement Format

The register utility control statement can be seen in the JCL example after the DD statement 'ROPEIPT'. The utility control statement uses the 'REGISTER' utility command. The register command is used with

seven other utility parameter options. All seven parameters are required. The register parameters are described below.

PTYPE

Specifies the parameter type record being created or updated. The parameter record type is 'COMPANY'. For the REGISTER utility function PTYPE may not change, and must always be set to 'COMPANY'.

COMPANY

Specifies the company name which ROPES has been licensed to. The company name may occupy up to 30 characters, and must not contain any periods or other special characters except for the hyphen.

ADDRESS

Specifies the address of the company ROPES has been licensed to. The address may be up to 30 characters in length, and should not contain any special characters except for the hyphen.

CITY

Specifies the city of the company ROPES has been licensed to. The city name may be up to 20 characters long, and should not contain any special characters except for the hyphen.

STATE

Specifies the state of the company ROPES has been licensed to. The state may not exceed 2 characters in length, and should be a valid state abbreviation.

ZIPCODE

Specifies the zip code of the company ROPES has been licensed to. The zip code is a 10 digit field which contains the primary zip code, and the ZIP PLUS 4 code. A hyphen separates the primary zip code and the zip plus 4 number. If the zip plus 4 number is not known, leave the hyphen and zip plus 4 portion of the field blank.

COUNTRY

Specifies the country of the company ROPES has been licensed to. The country name may be

up to 20 characters in length, and should not contain special characters except the hyphen.

Register Utility Sample JCL

A sample of the register utility JCL follows.

```
// JOB (** PLEASE PLACE YOUR JOB CARD HERE **)
/*+-----+
/*| REGISTER THE COMPANY NAME           |
/*+-----+
//REGISTR EXEC PGM=ROPEUTIL
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//ROPELST DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//ROPEPARM DD DSN=ROPES.PRMFILE.BASE,DISP=SHR
//ROPEIPT DD *
REGISTER PTYPE(COMPANY),
          COMPANY(** COMPANY NAME **),
          ADDRESS(NUMBER-STREET ADDRESS),
          CITY(CITY NAME),STATE(XX),ZIPCODE(NNNNN-NNNN),
          COUNTRY(COUNTRY NAME).
/*
//
```

The Apply Security Changes Utility Function

Function

The Apply Security Changes utility function installs or changes expiration dates and authorized CPU information in ROPES. This utility function is executed as part of the maintenance of ROPES. ROPES uses the expiration date and CPU information to grant your company use of the product. This program processes security code strings provided by Axios which grants you access to the product on your CPU. If you do not have the security code, you must obtain it from Axios.

Data Sets

The Apply Security Changes utility uses the following files as input files.

ROPEIPT - This file contains the utility control statements for the utility function. This file is a physical sequential file containing fixed length 80 byte records. This file is a required file.

The following files are produced as output from the utility function.

ROPELST - This file is the utility output message file. This file is a physical sequential file containing fixed length 133 byte records. This file is normally allocated to SYSOUT. This is a required file.

ROPEPARM - This file is the ROPES parameter file. The parameter file is a VSAM KSDS file. Attributes for this file can be obtained from the AMSCNTL member located in the distribution source data set. This is a required file for this utility function.

Utility Control Statement Format

The utility control statement can be seen in the JCL example after the DD statement 'ROPEIPT'. The utility control statement uses the 'APPLYSEC' utility command. The APPLYSEC command is used with one other utility parameter. The parameter is required. The APPLYSEC parameter is described

below.

SECCODE

Specifies the security string value. The security string is a string of characters provided by Axios to add or change authorized CPUs and to set or remove the product expiration date.

The security string consists of letters, numbers and a special character which specify the product expiration date and authorized CPUs. This information is encoded.

Apply Security Utility Sample JCL

A sample of the utility JCL follows.

```
// JOB (***) PLEASE PLACE YOUR JOB CARD HERE (***)
// *-----+
// *| EDIT COMPANY SECURITY INFORMATION      |
// *-----+
//APPLYSEC EXEC PGM=ROPEUTIL
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//ROPELST DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//ROPEPARM DD DSN=ROPES.PRMFILE.BASE,DISP=SHR
//ROPEIPT DD *
//APPLYSEC SECCODE(02A44#113A300001030307).
/*
//
```


The List Security Information Utility Function

Function

The List Security Information utility function lists information installed with the Register and Apply Security Changes utilities.

Data Sets

The List Security Information utility uses the following files as input files.

ROPEIPT - This file contains the utility control statements for the utility function. This file is a physical sequential file containing fixed length 80 byte records. This file is a required file.

The following files are produced as output from the utility function.

ROPELST - This file is the utility output message file. This file is a physical sequential file containing fixed length 133 byte records. This file is normally allocated to SYSOUT. This is a required file.

ROPEPARM - This file is the ROPES parameter file. The parameter file is a VSAM KSDS file. Attributes for this file can be obtained from the AMSCNTL member located in the distribution source data set. This is a required file for this utility function.

the values, separated by commas. The valid values are:

ACCOUNT - which lists the information supplied by the Register Utility;

CPUS - which lists the CPU identification of the processor on which this job is running;

AUTHCPUS - which lists the CPUs for which ROPES has already been authorized; and

EXPDATE - which lists the product expiration date.

List Security Information Utility Sample JCL

A sample of the List Security Information utility JCL follows.

```
// JOB (***) PLEASE PLACE YOUR JOB CARD HERE (***)
// *+-----+
// *| LIST COMPANY SECURITY INFORMATION |
// *+-----+
// LISTSEC EXEC PGM=ROPEUTIL
// STEPLIB DD DSN=ROPES.LOADLIB, DISP=SHR
// ROPELST DD SYSOUT=*
// SYSUDUMP DD SYSOUT=*
// ROPEPARM DD DSN=ROPES.PRMFILE.BASE, DISP=SHR
// ROPEIPT DD *
// LISTSEC SECINFO(ACCOUNT, CPUS, AUTHCPUS, EXPDATE) .
/*
//
```

Utility Control Statement Format

The List Security Information utility control statement can be seen in the JCL example after the DD statement 'ROPEIPT'. The utility control statement uses the 'LISTSEC' utility command. The LISTSEC command is used with one other utility parameter. The parameter is required. The parameter is described below.

SECINFO

Specifies the type of information to be displayed. There are four valid values for this parameter, and you may code one or more of

The JCL Load Utility Function

Function

The JCL load utility loads one or more JCL input streams from a physical sequential file, and stores them in the ROPES job file. The job file is a VSAM KSDS file defined by the AMSCNTL job which was executed as part of the ROPES installation process.

Once JCL is loaded into the job file it can be accessed by the ROPES JCL editor function under CICS. The transaction used to invoke the JCL editor is ROJB.

Please note ROJB is the new version of the JCL editor transaction, and it replaces the RJOB transaction of earlier releases. The new transaction (ROJB) also uses a new version of the JCL stream VSAM file. JCL streams in the old file must be converted/copied to the new file using ROPES utility function 'CONVERT'. The convert utility is described later in this chapter.

Data Sets

The JCL load utility uses the following files as input files.

ROPEIPT - This file contains the utility control statements for the utility function. This file is a physical sequential file containing fixed length 80 byte records. This file is a required file.

ROPEJCL - This file allocates the input job stream file which may be a physical sequential file, or a PDS member. One or more JCL streams separated by '%%NAME=' header control records may be loaded. The input file must be fixed blocked, and must have a logical record length of 80 bytes.

The following files are produced as output from the utility function.

ROPELST - This file is the utility output message file. This file is a physical sequential file containing fixed length 133 byte records. This file is normally allocated to SYSOUT. This is a required file.

ROPEJOB - This file is the ROPES job maintenance file. The job maintenance file is

a VSAM KSDS file. Input JCL streams are stored in this file. Attributes for this file can be obtained from the AMSCNTL member located in the distribution source data set. This is a required file for this utility function. *The JCL loader utility works with the new version of the JCL maintenance file only!*

Utility Control Statement Format

The JCL loader utility control statement can be seen in the JCL example after the DD statement 'ROPEIPT'. The utility control statement uses the 'LOADJCL' utility command. The LOADJCL command is used with three utility parameter keywords. The **JOBS**, and **LINES** keyword parameters are required. The **DESC** parameter is required if only one JCL stream is loaded. These keyword parameters are described below.

JOBS

Specifies a single job will be loaded, or a group of job streams will be loaded. If a single job will be loaded from the ROPEJCL input file, then an 8 character job name must be entered as the parameter value. If a group of jobs is specified, then the parameter value must be set to 'ALL'. When using the 'ALL' specification you must also include JCL header records which are described below. The **JOBS** parameter is a required parameter.

DESC

Specifies the description of a single JCL job stream provided a job name has been supplied for the JOBS parameter, and not the value 'ALL'. The description parameter may contain a description up to 40 characters in length.

LINES

Specifies the maximum number of lines that will be contained in the largest JCL job stream included in this run of the JCL loader utility. This parameter controls the size of the edit work area obtained when using the JCL editor under CICS. The value supplied for this parameter should be approximately 20 percent larger than the largest number of lines contained in a single JCL job stream. This parameter is a required

parameter.

A sample of the utility control statement appears in the sample utility JCL figure below.

JCL Header Control Record

The JCL header control record is required when the 'JOBS' keyword parameter specifies 'ALL' JCL streams in the ROPEJCL file will be processed. The JCL header control record specifies the name and description of each JCL job stream. The JCL header record must be located just before each job stream included in the ROPEJCL file. A sample of the JCL header control record is shown below.

%%NAME=jobname description

The '%%NAME=' constant must be coded in columns 1 through 7, the job name must be placed in columns 8 through 15, and the description must be placed in columns 17 through 46. The jobname must be 8 characters in length and should be padded with blanks if necessary, and the description must not exceed 40 characters in length.

JCL Loader Utility Sample JCL

A sample of the JCL loader utility JCL follows.

```
// JOB (***) PLEASE PLACE YOUR JOB CARD HERE (***)
// *+-----+
// *| EXECUTE JCL LOADER UTILITY. |
// *+-----+
//LOADJCL EXEC PGM=ROPEUTIL
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//ROPELST DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//ROPEJCL DD DSN=JCL.INPUT.FILE,DISP=SHR
//ROPEJOB DD DSN=ROPES.JCLFILE.BASE,DISP=SHR
//ROPEIPT DD *
LOADJCL JOBS(ALL),LINES(1000).
/*
//
```


The Convert JCL File Utility Function

Function

The convert JCL file utility converts/copies one or more JCL streams from the old JCL job maintenance file to the new job maintenance file. JCL job stream records are copied/converted from the old JCL record format to the new format.

Both JCL job maintenance files are VSAM KSDS files. The original job maintenance file (old format) comes from a release of ROPES prior to ROPES version 6.0.2. The old file is allocated to CICS by the DD name '**ROPEJOB**'. The new job maintenance file is created by the AMSCNTL member in the ROPES source distribution file, and is allocated to CICS by DD name '**ROPEJOB**'.

Once JCL streams have been converted into the new job file they can be accessed by the new ROPES JCL editor transaction ROJB under CICS.

Please note ROJB is the new version of the JCL editor transaction, and it replaces the RJOB transaction of earlier releases.

Data Sets

The JCL CONVERT utility uses the following files as input files.

ROPEIPT - This file contains the utility control statements for the utility function. This file is a physical sequential file containing fixed length 80 byte records. This file is a required file.

ROPEJCL - This file allocates the input job stream file which contains the JCL streams to be converted. This file is a VSAM KSDS file created by a release of ROPES prior to version 6.0 PTF level 2.

The following files are produced as output from this utility function.

ROPELST - This file is the utility output message file. This file is a physical sequential file containing fixed length 133 byte records. This file is normally allocated to SYSOUT. This is a required file.

ROPEOJCL - This file is the new ROPES job maintenance file. It is the target file of the convert utility, and will contain the reformatted JCL streams for this release of ROPES. The new job maintenance file is a VSAM KSDS file created by the AMSCNTL member in the source distribution library. Please refer to the installation instructions for more information on this file.

Utility Control Statement Format

The JCL CONVERT utility control statement can be seen in the JCL example after the DD statement 'ROPEIPT'. The utility control statement uses the 'CONVERT' utility command. The CONVERT command is used with two utility parameter keywords. The keywords are **MEMBERS**, and **REPLACE**. The **MEMBERS** keyword is required, and the **REPLACE** keyword is optional.

MEMBERS

Specifies a list of selected member names, or the value '**ALL**'. If multiple member names are coded, they must be separated by commas. The member names selected will be converted and then stored in the new job maintenance file referenced by DD name ROPEJOB.

REPLACE

Specifies selected JCL members will replace existing JCL members in the target file (ROPEJOB). The possible values for this keyword parameter is either '**YES**', or '**NO**'.

A sample of the utility control statement appears in the sample utility JCL figure below.

Convert JCL File Utility Sample JCL

Samples of the Convert utility JCL are shown below.

```
// JOB (***) PLEASE PLACE YOUR JOB CARD HERE (***)
// *+-----+
// *| CONVERT ALL JCL MEMBERS WITHOUT REPLACING. |
// *+-----+

```

```
//CONVERT EXEC PGM=ROPEUTIL
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//ROPELST DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//ROPEIJCL DD DSN=ROPES.OLD.JOBFILE,DISP=SHR
//ROPEOJCL DD DSN=ROPES.JCLFILE.BASE,DISP=SHR
//ROPEIPT DD *
  CONVERT MEMBERS (ALL) ,REPLACE (NO) .
/*
//

// JOB (***) PLEASE PLACE YOUR JOB CARD HERE (***)
//*+-----+
//*| CONVERT SELECTED MEMBERS AND REPLACE THEM.|
//*+-----+
//CONVERT EXEC PGM=ROPEUTIL
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//ROPELST DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//ROPEIJCL DD DSN=ROPES.OLD.JOBFILE,DISP=SHR
//ROPEOJCL DD DSN=ROPES.JCLFILE.BASE,DISP=SHR
//ROPEIPT DD *
  CONVERT MEMBERS (MEMBR1,MEMBR2,MEMBR3,MEMBR4,
    MEMBR5,MEMBR6,MEMBR7) ,REPLACE (YES) .
/*
//
```

Report To Printer Index Generation Function

Function

This utility function is used to scan the ROPERIB and ROPEPCR data sets and use the data found to generate the ROPERPI data set. It also generates a report that indicates all reports defined to ROPES and the printers to which they have been assigned.

Input and Output

The ROPERIB data set, containing the report control information; and the ROPEPCR data set, containing the printer status information are both required for this function.

A newly created, or replaced, ROPERPI data set is produced by this function. The previous restriction requiring that the ROPERPI data set be deleted and redefined prior to execution of this utility has been removed in Version 13.

This utility function dynamically invokes the system SORT utility. The sort program, named SORT, must be available in the job's JOBLIB, the step's STEPLIB, or in the system LINKLST library list. A SYSOUT data set must be provided for SORT program messages.

Command Format

REINDEX [MESSAGES(option)] MSGS(option).

MESSAGES (or MSGS)

requests detailed or summary messages. If extended messages are requested, additional messages will be printed to indicate each printer reset to reflect the cleared report, the number of Queue records deleted from the ROPEQBR file, and the number of each Queue record out of range with the control information in the Report Information Block. If the MESSAGES parameter is omitted, extended messages will be suppressed.

The permitted values for the parameter fields of the control statement are:

option controls the type of messages issued by the PURGE function. The values for OPTION in the MESSAGES parameter are **VERBOSE** or **TERSE**. Any value other than these will cause extended messages to be turned off.

Reindex Utility Example

```
//BUILDRPI EXEC PGM=ROPERUTIL
//STEPLIB DD DSN=ROPES.LOADLIB,DIS
           P=SHR
//ROPERIB DD DSN=ROPES.RIBFILE.BAS
           E,DISP=SHR
//ROPEPCR DD DSN=ROPES.PCRFILE.BAS
           E,DISP=SHR
//ROPERPI DD DSN=ROPES.RPIFILE.BAS
           E,DISP=SHR
//ROPELST DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//ROPEIPT DD *
           REINDEX MESSAGES (VERBOSE) .
```


The ROPES External Interface Utility Functions

Introduction

The ROPES External Interface uses the standard utility command driver established in an earlier release of ROPES. Utility functions under the utility command driver execute program ROPEUTIL. The functions and programs of the ROPES External Interface are defined to the utility driver program (ROPEUTIL) in two internal tables. These internal tables define the utility control statement syntax and the utility programs that are executed as a result of entering the utility control statements. The utility command associated with the ROPES External Interface is "XPLINK". There are three possible major functions which can be performed by the XPLINK command. They are FUNCTION(OPENLINK), FUNCTION(SENDREQ), and FUNCTION(CLOSELINK).

The 'OPENLINK' function establishes the ROPES External Interface session using the CICS External Interface (EXCI). The 'SENDREQ' function sends the utility request to the ROPES External Interface server program in CICS. Finally, the 'CLOSELINK' function terminates communication over the MRO link using the EXCI calls.

The 'OPENLINK' function must be performed before any 'SENDREQ' functions may be executed. The 'CLOSELINK' function must be performed after an 'OPENLINK' function has been executed. Multiple 'SENDREQ' functions may be performed before the 'CLOSELINK' function. The number of 'SENDREQ' utility function statements are not limited.

Please read on now for detailed information about the JCL requirements and utility control statement syntax.

Functions

There are two utility functions provided by the ROPES External Interface at this time. The first function lists detailed information about ROPES printers. The second function lists detailed information about ROPES reports. One or more printers/reports may be listed by these utility

functions. The content and details produced by these utility functions can serve as diagnostic information for some ROPES problems and are not provided by other online displays within ROPES.

Data Sets

The ROPEIPT, ROPELST, and ROPERPT data sets are required by the ROPES External Interface utility functions. In addition, the CICS.SDFHEXCI load library must be added to the STEPLIB concatenation for the batch job, and the SYSMDUMP DD statement should be added in the event an abend occurs which produces a dump. A sample of the JCL required to run these utility functions can be found in the ROPES distribution source library in member 'RUNEXCI'.

Command Format

```
XPLINK FUNCTION(function),
  [OPTIONS(applid,xtran,serverp|[,userid]),]
  [REQUEST(request text),]
  [PRINTER(printerid|generic|ALL),]
  [REPORT(reportid|generic|ALL),]
```

FUNCTION

The FUNCTION keyword parameter specifies the major type of function to be performed. Three possible values may be coded for the FUNCTION parameter::

OPENLINK

Opens and initializes the ROPES External Interface and opens a link to CICS. This function must be performed before the SENDREQ and CLOSELINK functions. The OPTIONS keyword must also be specified with the OPENLINK function.

SENDREQ

Sends a utility request or command to the ROPES External Interface server program in CICS. The value of the REQUEST parameter specifies the type of request to be sent. Other keyword parameters will be required for use with the SENDREQ function, but this will

depend on the type of request being made.

CLOSELINK

Closes and deallocates the ROPES External Interface link to CICS. This function must be performed last. No other keyword parameters are used with this function.

OPTIONS

The OPTIONS keyword parameter specifies option values required to establish the link between the ROPES External Interface batch client program and the CICS server program. Four positional parameters may be specified. The first three positional parameters are required, and the last positional parameter is optional. The positional parameter values are:

applid

Specifies the CICS applid for the CICS region that will receive the utility requests. This is a required parameter value and is the first positional value.

xtran

Specifies the ROPES External Interface mirror transaction id that will be associated with the server program. The default for this parameter is "ROXP". This is a required parameter value and is the second positional value.

serverp

Specifies the ROPES External Interface server program name that will be associated with the server mirror transaction. The default value for this parameter is "ROPEXPLS". This is a required parameter value and is the third positional value.

userid

Specifies the userid that will be associated with the link to CICS using the CICS External Interface. The default user is automatically set to the batch client program name (ROPEXPLR). There is generally no need to change the default user unless there are specific security reasons for doing so. Therefore, the userid parameter is optional and in most cases should be omitted unless there are specific security reasons for changing it. The userid parameter value is the fourth positional value.

REQUEST

The REQUEST parameter keyword specifies the type of utility request to be performed. The REQUEST value is in text format and may be up

to 20 characters in length. Two possible utility REQUEST values may be used at this time. The two utility REQUEST values are:

LIST PRINTERS

Directs the ROPES External Interface server program to produce a list of printer details for one or more printers specified by the PRINTER parameter keyword. Detailed definition and status information about the printer or printers is produced and sent back to the ROPES External Interface batch client program which sends the output to DD statement ROPERPT. The PRINTER parameter keyword is required if this REQUEST type is used.

LIST REPORTS

Directs the ROPES External Interface server program to produce a list of report details for one or more reports specified by the REPORT parameter keyword. Detailed definition and status information about the report(s) is produced and set back to the ROPES External Interface batch client program which sends the output to DD statement ROPERPT. The REPORT parameter keyword is required if this REQUEST type is used.

Additional REQUEST types will be made available in future updates and releases of ROPES.

PRINTER

Specifies the printer or printers which will be selected and included in the printer detail report. Three possible value types may be coded for the PRINTER keyword parameter but only one may be coded. They are:

printerid

Specifies a specific 4 character ROPES printer id. If a specific printer id is used, then no additional printer ids may be coded. Detailed information about the single printer id will be produced.

generic

Specifies a partial or generic printer id by coding an asterisk at some point beyond the first position of the printer id to be selected. For example, if you wish to generate a detail report for printers that begin with the characters "PJ", you would code "PJ*" for the generic printer id. Possible permutations of this would be: "P*", "PJ*", and "PJ0*". Note that coding an asterisk in the first position is invalid.

ALL

Specifies that all printers must be selected for the detailed printer report.

REPORT

Specifies the report or reports which will be selected and included in the detailed report information output. Three possible value types may be coded for the REPORT keyword parameter but only one may be coded. They are:

reportid

Specifies one specific 8 character ROPES report id. If a specific report id is used, then no additional report ids may be coded. Detailed information about the single report id will be produced.

generic

Specifies a partial or generic report id by coding an asterisk at some point beyond the first position of the report id to be selected. For example, if you wish to generate a detail report for reports that begin with "EXPEN", you would code "EXPEN*" for the generic report id. Note that coding an asterisk in the first position is invalid.

ALL

Specifies that all reports must be selected for the detailed report information generated.

ROPES External Interface Utility Examples

1. In this example, detailed status and resource definition information will be produced for printer "PR01". The details will be obtained and generated from the ROPES files allocated to CICS, and then sent back to the batch client program for output.

```
//ROPEEXCI EXEC PGM=ROPEUTIL,
//          PARM='COMWSLEN=24K'
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//          DD DSN=CICS.SDFHEXCI,DISP=SHR
//ROPELST DD SYSOUT=*
//ROPERPT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSMDUMP DD SYSOUT=*
//ROPEIPT DD *
XPLINK FUNCTION(OPENLINK),
  OPTIONS(CICS,ROXP,ROPEXPLS).
XPLINK FUNCTION(SENDREQ),
  REQUEST(LIST PRINTERS),
  PRINTER(PR01).
XPLINK FUNCTION(CLOSELINK).
/*
//
```

2. In this example, detailed status and resource definition information will be produced for all reports that begin with the characters "EXPEN". The details will be obtained and generated from the ROPES files allocated to CICS, and then sent back to the batch client program for output.

```
//ROPEEXCI EXEC PGM=ROPEUTIL,
//          PARM='COMWSLEN=24K'
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//          DD DSN=CICS.SDFHEXCI,DISP=SHR
//ROPELST DD SYSOUT=*
//ROPERPT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSMDUMP DD SYSOUT=*
//ROPEIPT DD *
XPLINK FUNCTION(OPENLINK),
  OPTIONS(CICS,ROXP,ROPEXPLS).
XPLINK FUNCTION(SENDREQ),
  REQUEST(LIST REPORTS),REPORT(EXPEN*).
XPLINK FUNCTION(CLOSELINK).
/*
//
```

3. In this example, detailed status and resource information will be produced for all printers and reports that begin with "INVEN". The details will be obtained and generated from the ROPES files allocated to CICS, and then sent back to the batch client program for output. Note in the example below that there are two 'SENDREQ' utility control statements. The number of SENDREQ utility controls statements that may be coded is unlimited.

```
//ROPEEXCI EXEC PGM=ROPEUTIL,
//          PARM='COMWSLEN=24K'
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//          DD DSN=CICS.SDFHEXCI,DISP=SHR
//ROPELST DD SYSOUT=*
//ROPERPT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSMDUMP DD SYSOUT=*
//ROPEIPT DD *
XPLINK FUNCTION(OPENLINK),
  OPTIONS(CICS,ROXP,ROPEXPLS).
XPLINK FUNCTION(SENDREQ),
  REQUEST(LIST PRINTERS),PRINTER(ALL).
XPLINK FUNCTION(SENDREQ),
  REQUEST(LIST REPORTS),REPORT(INVEN*).
XPLINK FUNCTION(CLOSELINK).
/*
//
```

ROPES External Interface Sample Output

Figure [Figure 3](#) on page [57](#) is a sample of the output produced when listing printer and report detail information using the ROPES External Interface.

The fields of the 'Printer Details' report can be matched up with the record description for the Printer Control Record (PCR). The record description for the PCR can be found in the distribution source

library under member ROPEPCR. Under the field "SVC FLAGS (SITDM)" in the sample report, five 1 byte flag fields have been combined in the output. These flags correspond with the "Service Request Flags" which can be found in the ROPEPCR record layout and include fields PCRSTART, PCRINTRP, PCRTEST, PCRDRAIN, and PCRSTMOD. Otherwise, most fields in the 'Printer Detail' report are fairly self explanatory or can be looked up in the ROPEPCR record layout.

The fields of the 'Report Details' sample are described by the record layout for the Report Information Block or RIB. The RIB record layout can be found in the Ropes Distribution Source library in member ROPERIB. One field in the sample output named "FLAGS:" combines six 1 byte flag fields which are located in the ROPERIB record description starting at field RIBAPEND and ending with field RIBINTER. Descriptions for these six flag fields can be found in the record layout for the RIB record (ROPERIB). Other fields in the sample "Report Detail" output are relatively self explanatory.

Information displayed by the 'Printer Details' and 'Report Details' reports can be used to verify and debug some error conditions that can arise out of conflicting data conditions between the RIB and PCR records. Some of the information contained in these reports have not been available in any other display or report in ROPES. These reports will also provide a useful audit trail of the resources currently defined in your ROPES system.

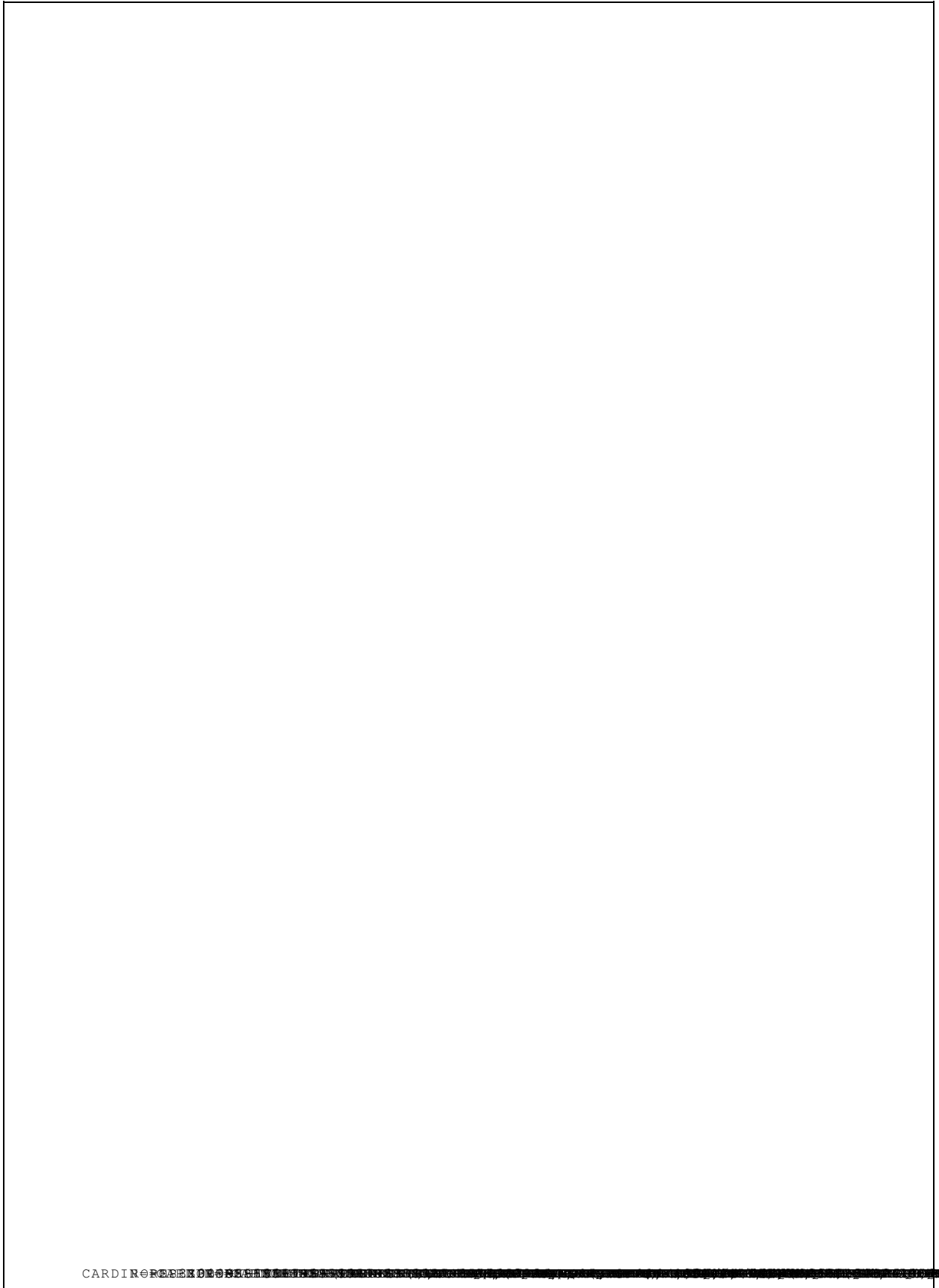


Figure 3 ROPES EXCI Example Output

Report To PDF Conversion Function

Function

This utility function is used to convert ROPES reports into PDF (Adobe® Acrobat® Portable Document Format) files. You can transmit these files to end users or store these files on your web server for access through the internet or your intranet.

We have provided an online sample program and mapset as well, which you may investigate and examine. The ROPES Source Library member ROPEOPDF contains the program and member ROPEPDF contains the mapset.

Input and Output

The ROPERIB data set, containing the report control information and the ROPEQBR data set, containing the report text are both required for this function. You may also include the ROPES library containing the ROPES Forms Control Blocks to provide controls for the conversion process. You may also provide input in a sequential data set or library member.

A newly created data set is produced by this function. The generated PDF is a Variable Length data set and is already in Ascii coding, so transfers of the data set should be binary.

Command Format

```
MAKEPDF {REPORT(reportid)|INFILE(INPUT)},
  OUTFILE(NEWPDF),
  USEFCB(RO$FPDFM).
```

REPORT

This parameter specifies the name of the ROPES report to be processed. This parameter is required, unless input is from a file, in which case this parameter is not permitted.

INFILE

This parameter specifies the DD Name of the DD statement that defines the input sequential data set or library member to be processed. The input data set must be defined with fixed or variable length records, and must be defined

with ASA or machine carriage control characters (RECFM=FA | FBA | FM | FBM | VA | VBA | VM | VBM). This parameter is required if the REPORT parameter is omitted, and is not permitted if the REPORT parameter is coded.

OUTFILE

This parameter specifies the DD name of the data set defined to receive the generate PDF. This parameter is required

USEFCB

This parameter is optional. If used, it specifies the name of the ROPES Forms Control Block (FCB) module (RO\$Fxxxx) to be used to obtain the formatting controls. If this parameter is omitted, the FCB assigned to the report is used. If the FCB does not contain the TYPE=PDFCNTL definitions, default values are used, as follows:

Orientation	Landscape
Page Dimensions	
Width	8.5 inches
Height	11 inches
Lines Per Inch	6
Font	6 point
Margins	
Top	1 inch
Bottom	1 inch
Left	1 inch
Right	1 inch

Build PDF Utility Example

```
//RUNBPDF JOB (ROPES), 'MAKE A PDF',
//          CLASS=A,MSGCLASS=X,NOTIFY=&SYSUID
//ALLOC    EXEC PGM=IEFBRI4
//SYSUT1   DD DSN=&SYSUID..WORKPDF,DISP=(MOD,CATLG),
//          UNIT=SYSDA,SPACE=(TRK,(1,1)),
//          DCB=(RECFM=VB,LRECL=259,BLKSIZE=6194)
//RUNBPDF EXEC PGM=ROPEUTIL
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//ROPERIB DD DSN=ROPES.RIBFILE.BASE,DISP=SHR
//ROPEQBR DD DSN=ROPES.QBRFILE.BASE,DISP=SHR
//ROPELST DD SYSOUT=*
//SYSUT1   DD DSN=&SYSUID..WORKPDF,DISP=(OLD,PASS)
//NEWPDF   DD DSN=YOUR.PDF,DISP=(NEW,CATLG),UNIT=SYSDA,
//          SPACE=(TRK,(1,1)),
//          DCB=(RECFM=VB,LRECL=300,BLKSIZE=27998)
//ROPEIPT DD *
MAKEPDF REPORT (REPORT01),OUTFILE(NEWPDF),USEFCB(RO$FPDFM) .
/*
```

Source library member RUNBPDF contains sample
JCL for your use.

Report To HTML Conversion Function

Function

This utility function is used to convert ROPES reports into HTML format files. You can transmit these files to end users or store these files on your web server for access through the internet or your intranet.

We have provided an online sample program and mapset as well, which you may investigate and examine. The ROPES Source Library member ROPEOHTM contains the program and member ROPEHTM contains the mapset.

Input and Output

The ROPERIB data set, containing the report control information and the ROPEQBR data set, containing the report text are both required for this function. You may also include the ROPES library containing the ROPES Forms Control Blocks to provide controls for the conversion process. You may also provide input in a sequential data set or library member.

A newly created data set is produced by this function. The generated HTML is a Variable Length data set and is already in Ascii coding, so transfers of the data set should be binary.

Command Format

```
MAKEPDF {REPORT(reportid)|INFILE(INPUT)},
        OUTFILE(NEWHTM),
        USEFCB(RO$FHTML).
```

REPORT

This parameter specifies the name of the ROPES report to be processed. This parameter is required, unless input is from a file, in which case this parameter is not permitted.

INFILE

This parameter specifies the DD Name of the DD statement that defines the input sequential data set or library member to be processed. The input data set must be defined with fixed or variable length records, and must be defined with ASA or machine carriage control

characters (RECFM=FA | FBA | FM | FBM | VA | VBA | VM | VBM). This parameter is required if the REPORT parameter is omitted, and is not permitted if the REPORT parameter is coded.

OUTFILE

This parameter specifies the DD name of the data set defined to receive the generate HTML. This parameter is required

USEFCB

This parameter is optional. If used, it specifies the name of the ROPES Forms Control Block (FCB) module (RO\$Fxxxx) to be used to obtain the formatting controls. If this parameter is omitted, the FCB assigned to the report is used. If the FCB does not contain the TYPE=HTMCNTL definitions, default values are used, as follows:

Font Size	Medium
Font Face	Courier New
Banner	No
Banner Size	X-large
Foreground color	Black
Background color	White
Title	None
Image	None

Build HTML Utility Example

```
//RUNBHTM JOB (ROPES), 'MAKE AN HTML',
//          CLASS=A,MSGCLASS=X, NOTIFY=&SYSUID
//ALLOC   EXEC PGM=IEFBR14
//SYSUT1  DD DSN=&SYSUID..WORKHTM, DISP=(MOD,CATLG),
//          UNIT=SYSDA,SPACE=(TRK,(1,1)),
//          DCB=(RECFM=VB,LRECL=259,BLKSIZE=6194)
//RUNBPDF EXEC PGM=ROPEUTIL
//STEPLIB DD DSN=ROPES.LOADLIB, DISP=SHR
//ROPERIB DD DSN=ROPES.RIBFILE.BASE, DISP=SHR
//ROPEQBR DD DSN=ROPES.QBRFILE.BASE, DISP=SHR
//ROPELST DD SYSOUT=*
//SYSUT1  DD DSN=&SYSUID..WORKHTM, DISP=(OLD,PASS)
//NEWPDF  DD DSN=YOUR.PDF, DISP=(NEW,CATLG),UNIT=SYSDA,
//          SPACE=(TRK,(1,1)),
//          DCB=(RECFM=VB,LRECL=300,BLKSIZE=27998)
//ROPEIPT DD *
MAKEPDF REPORT(REPORT01),OUTFILE(NEWHTM),USEFCB(RO$FHTML).
/*
```

Source library member RUNBHTM contains sample JCL for your use.

TCP/IP Translation Facility Default Setting

Function

This utility function is used to supply the ROPES default translation settings to be used when processing LPR or Direct Socket printing. You can run this utility at any time to change the default options in use. A change to the default options will not change the settings in any ROLI panel records that were updated or created in Version 14.

Input and Output

The ROPEPRM data set, containing the Translation Facility control information is required for this function.

Command Format

```
TRANSLATE
  USEFOR(LPR|DIRECT),
  INPROG(C05|C15|A2E|CTR),
  INTABLE(table name),
  OUTPROG(C04|C14|E2A|CTR)
  OUTTABLE(table name)
  EZACICTR(S|M).
```

USEFOR

This parameter specifies the purpose of the default settings being provided. Choose either LPR or DIRECT. If you use both methods, then run the utility once for each method, as shown in the example, below. This parameter is required.

INPROG

This parameter specifies the input translation method to be used. Choose from one of the four values in the list. This parameter is required.

OUTPROG

This parameter specifies the output translation method to be used. Choose from one of the four values in the list. This parameter is required

INTABLE

This use of this parameter depends on the choice of input translation method and the mode of the

EZACICTR program generation. If A2E is used, it specifies the name of the EZACICTR translate table to be used for input. If CTR is used, it specifies the name of the user provided external translation table to be used. A sample user translate table has been supplied in the ROPES source library, in member ROPETCTR. If this parameter is omitted, and the EZACICTR mode is SINGLE, then the default table will be used. If the EZACICTR mode is MULTIPLE, an error will occur when printing is attempted.

OUTTABLE

This use of this parameter depends on the choice of output translation method and the mode of the EZACICTR program generation. If E2A is used, it specifies the name of the EZACICTR translate table to be used for input. If CTR is used, it specifies the name of the user provided external translation table to be used. A sample user translate table has been supplied in the ROPES source library, in member ROPETCTR. If this parameter is omitted, and the EZACICTR mode is SINGLE, then the default table will be used. If the EZACICTR mode is MULTIPLE, an error will occur when printing is attempted.

EZACICTR

This parameter specifies the mode setting used when EZACICTR was installed. Specify either S for Single or M for Multiple. This setting determines how EZACIA2E and EZACIE2A are called. If set incorrectly, an error will occur when printing is attempted.

Translate Facility Defaults Utility Example

```
//RUNTRAN JOB (ROPES), 'SET TRANDATA',
// CLASS=A,MSGCLASS=X,NOTIFY=&SYSUID
//+-----+
/*| UPDATE THE TRANSLATION FACILITY DATA
//+-----+
//TRAN EXEC PGM=ROPEUTIL
//STEPLIB DD DSN=ROPES.LOADLIB,DISP=SHR
//ROPELST DD SYSOUT=*
//ROPEPRM DD DSN=ROPES.PRMFILE.BASE,DISP=SHR
//SYSUDUMP DD SYSOUT=X
//ROPEIPT DD *
TRANSLATE USEFOR (LPR), INPROG (C05), OUTPROG (C04)
TRANSLATE USEFOR (DIRECT), INPROG (C05), OUTPROG (C04)
//
```

| The example will set the defaults to the values that
| correspond to the TCP/IP LPR or Direct Socket
| printing translation values in use prior to ROPES
| V14.0. Source library member RUNTRAN contains
| sample JCL for your use.
|

Stand-By Facility SYSIDENT Utility

Function

This utility function is used to define the CICS Application Owning Regions (AORs) that are to participate in the automatic transfer of data from the ROPES Stand-By Facility data sets to the primary facility data sets. Please refer to the Administrator's Guide for information about setting up and using the Stand-By Facility.

The SYSIDENT values used by the Stand-By facility are the 1 - 4 character values used to specify the CICS SYSIDENT value specified in the DFHSITxx table or in the SYSIDENT= SYSIN override.

Input and Output

The ROPEPRM data set, containing the System Identifier record(s) is required for this function.

Command Format

```
STANDBY CYCLE(hhmmss),
  ADDIDENT(sysid1,sysid2,...sysidn),
  DELIDENT(sysid1,sysid2,...sysidn).
```

CYCLE

This is the facility cycle time. After each full pass through the SYSIDENT list, the facility will schedule itself to resume after the specified interval and the current task will terminate. If the interval is 000000, the facility will not recycle, but will instead run once and then terminate without scheduling itself to be restarted.

ADDIDENT

This is the list of SYSIDENTs (maximum 2000 in any one execution) to be added to the table of defined systems. A total of 4000 entries can be contained in the ROPEPRM file, but to exceed 2000 entries, the utility must be run more than once.

DELIDENT

This is the list of SYSIDENTs (maximum of 2000 in any one execution) to be removed from

the table of defined systems. To remove more than 2000 entries, run the utility more than once.

Stand-By Facility SYSIDENT Utility Example

```
//RUNSBYU JOB (ROPE), 'SET STANDBY ',
//      CLASS=A,MSGCLASS=X,NOTIFY=&SYSUID
//*+-----+
//*| UPDATE THE STANDBY FACILITY DATA |
//*+-----+
//SBYU      EXEC PGM=ROPEUTIL
//STEPLIB DD DSN=ROPE.LOADLIB,DISP=SHR
//ROPELST DD SYSOUT=*
//ROPEPRM DD DSN=ROPE.PRMFILE.BASE,DISP=SHR
//SYSUDUMP DD SYSOUT=X
//ROPEIPT DD *
STANDBY CYCLE(003000),ADDIDENT(S630,AR01,AR02),
        DELIDENT(S0W1) .
//
```

Source library member RUNSBYU contains sample JCL for your use.

Index

%%NAME..	47 , 48
ACCDEST..	33 , 36
Accounting..	33
Acrobat..	7
Address..	41 , 42
Adobe..	7
Alternate	
Facility..	9 , 29
Archive..	7 , 21 , 23 , 24
Attach..	13 , 15
Audit Trail..	56
AUTO ..	15 , 33
BOTtom..	15 , 59
BUFSIZE..	57 , 58
Carriage Control..	17 , 59 , 61
Change..	2 , 13 , 15 , 31 , 32 , 41 , 43 , 54 , 63
CITY..	41 , 42
Class..	57 , 59 , 61 , 63 , 65
Clear..	29
COLORS..	32 , 36
Command..	15 , 19 , 23 , 27 , 29-31 , 38-41 , 43 , 45 , 47 , 49 , 51 , 53 , 59 , 61 , 63 , 65
COMPANY..	7 , 9 , 41-43 , 45
CONTROLS..	9 , 12 , 15 , 29 , 31 , 32 , 34-36 , 39 , 47 , 51 , 55 , 59 , 61
CONTROLS PTYPE..	36
Controls Utility..	9 , 31 , 35 , 36
Convert JCL File Utility..	49
JCL Header Control Record..	48
Copies..	2 , 7 , 49 , 57
COUNTRY..	41 , 42
Data Sets..	7 , 11 , 21 , 22 , 25 , 26 , 29 , 31 , 37 , 39 , 41 , 43 , 45 , 47 , 49 , 51 , 53 , 65
DATE..	41 , 43 , 45
DCT..	33 , 35
DEPTH..	35
DESC..	31 , 32 , 34 , 36 , 47
Destination Control Table..	33 , 35
Direct Socket..	63 , 64
Discard..	21 , 23
DISP..	12 , 23 , 24 , 29 , 30 , 36 , 38-40 , 42 , 43 , 45 , 48 , 50 , 51 , 55 , 57-59 , 61 , 63 , 65
Distribution..	31 , 32 , 34 , 35 , 41 , 43 , 45 , 47 , 49 , 53 , 55 , 56
distribution definition..	34
Enter..	15
ENVTABL..	32 , 36
Error..	11 , 13 , 15 , 17 , 32 , 33 , 56 , 63
EXCI..	9 , 53 , 57
EXCLUDE..	21 , 23 , 24 , 29 , 30 , 39 , 40
Exit..	13 , 15 , 33-36 , 57 , 58
EZACIA2E..	63
EZACICTR..	63
EZACIE2A..	63
FCB..	57 , 59 , 61

FCT.	19 , 31 , 32
First.	12 , 17 , 22 , 26 , 29-31 , 33 , 34 , 40 , 53-55 , 57 , 58
FORM.	30 , 40 , 57
Forms Control Block.	59 , 61
Functions.	7 , 9 , 11 , 17 , 21-23 , 25 , 26 , 33 , 53
HEADER.	37 , 47 , 48
HLPFILE.	32 , 36 , 38
Hold.	23 , 57
HTML.	9 , 61
INDEX.	9 , 21 , 34 , 51 , 67
Installation.	2 , 7 , 9 , 13 , 15 , 31 , 35 , 37 , 41 , 47 , 49
INTRDRQ.	33 , 36
ITRNPFX.	33 , 36
JCL Loader Utility.	47 , 48
JES.	7
JESR.	33
Job.	9 , 11 , 12 , 18 , 21 , 23-25 , 29-34 , 36-40 , 42 , 43 , 45 , 47-50 , 53 , 59 , 61 , 63 , 65
JOBNAME.	48
JOBS.	47 , 48
KEY1.	34 , 36
KEY10.	36
KEY12.	34
KEY2.	34
KEY3.	36
KEY4.	36
KEY7.	36
KEY8.	36
Label.	32 , 34 , 38
LANG.	31 , 36
Last.	19 , 23 , 27 , 54 , 57 , 58
Left.	27 , 59
LIN S.	13 , 15 , 38 , 47 , 48 , 57-59
Load Help Utility.	38
LOAD LP.	37 , 38
LOGEXIT.	35
logging.	31 , 34-36
Logging Exit.	35
LOGTYPE.	34-36
LPP.	38
LPR.	34 , 63 , 64
Maintenance.	2 , 31 , 34 , 36 , 43 , 47 , 49
MEMBERS.	31 , 49 , 50
Menu.	15
mount.	57
MRO.	32 , 53 , 57
MROREGN.	32 , 36
MSGFILE.	31 , 36
MTERMID.	33 , 36
Next.	30 , 36 , 40
Overview.	7
PANEL.	32 , 34 , 36 , 37 , 63
PCR.	55 , 56
PCT.	31
PDF.	7 , 9 , 59 , 61
PF21.	15

PF9.	15
PFKEY.	36
PPT.	31
Prefix.	13 , 31-33
Prev.	36
Print.	1 , 2 , 7 , 9 , 11 , 13 , 15 , 17 , 21-24 , 32 , 34
Printer ID.	54
Priority.	57
PROGPFX.	31 , 36
Program Prefix.	31 , 32
Programming.	7
PTYPE.	31 , 36 , 41 , 42
QBR.	29 , 57 , 58
Queue Reorganization.	21
R51RSECX.	36
RACF.	33
REGISTER.	9 , 17 , 41 , 42 , 45
Register Utility.	41 , 42 , 45
Remove.	43 , 65
REPLACE.	49 , 50
Report	
Line.	17
Name.	17 , 19 , 29 , 39
Parameter.	54 , 59 , 61
Report Distribution.	31 , 32 , 34
REPORTS.	9 , 11 , 12 , 17 , 19 , 21 , 23-25 , 29 , 30 , 34 , 39 , 40 , 51 , 53-57 , 59 , 61
Reset.	39 , 51
Retention Time.	21-23
RFILES.	33 , 36
RIB.	29 , 33 , 56 , 57
Right.	27 , 59
ROJB.	33 , 47 , 49
ROPEALQ.	36
ROPEALR.	29 , 30 , 36
ROPEARC.	21-24
ROPEBPDF.	9
ROPEBQOR.	9 , 21-24
ROPEBQUE.	9 , 17 , 19 , 33
ROPEBRPU.	9 , 11 , 12
ROPECPYG.	9 , 13
ROPECVJF.	9
ROPEHLP.	36
ROPEIPT.	25-27 , 29-31 , 36-43 , 45 , 47-51 , 53 , 55 , 59 , 61 , 63 , 65
ROPEJCLL.	9
ROPEJOB.	36 , 47-49
ROPEKCRQ.	9 , 15
ROPELST.	25 , 26 , 29-31 , 36-43 , 45 , 47-51 , 53 , 55 , 59 , 61 , 63 , 65
ROPEMSG.	36
ROPEPCR.	21-26 , 36 , 39 , 40 , 51 , 56
ROPEPRM.	31 , 63 , 65
ROPEPURG.	9
ROPEQBR.	11 , 12 , 21-26 , 36 , 39 , 40 , 51 , 59 , 61
ROPERDDD.	36
ROPERDTD.	36
ROPERIB.	11 , 12 , 21-26 , 29 , 30 , 36 , 39 , 40 , 51 , 56 , 59 , 61

ROPERPI.	21-24 , 36 , 51
ROPERPIB.	9
ROPERPT.	25 , 53-55
ROPESPRT.	9 , 15
ROPEUTIL.	9 , 25-27 , 29 , 30 , 36 , 38-40 , 42 , 43 , 45 , 48 , 50 , 53 , 55 , 59 , 61 , 63 , 65
RUNBUILD.	34 , 35 , 37
SAVE.	13
Screen Handler Options.	31
Screen Handler Program.	32 , 36
Screen Print.	13 , 15
SCREEN-HANDLER.	31
SECEXIT.	33 , 36
SECUPD.	33 , 36
Security.	9 , 33 , 41 , 43 , 45 , 54
Security String.	43
Send.	34
SINGLE.	27 , 29 , 37 , 39 , 47 , 54 , 55 , 63
Socket.	63 , 64
Standby.	65
Start.	15 , 27
Status.	7 , 11 , 21 , 25 , 31 , 32 , 34 , 51 , 54 , 55
Store.	9 , 59 , 61
SUBJECT.	2
Suffix.	32 , 57
TCP/IP.	9 , 63 , 64
TDQUE.	33 , 35 , 36
Terminate.	65
TIME.	21-23 , 31 , 32 , 41 , 53 , 54 , 63 , 65
TOP.	57 , 59
TRANPFX.	31 , 36
transaction.	7 , 9 , 13 , 15 , 17 , 19 , 31 , 33 , 47 , 49 , 54
transaction prefix.	31
Transfer.	7 , 9 , 19 , 65
Transient Data.	33-35
Translate.	63
Translation.	9 , 34 , 63 , 64
Up.	25 , 31 , 32 , 34 , 35 , 41 , 42 , 47 , 54-56 , 65
Utility.	7 , 9 , 11-13 , 21-23 , 25-27 , 29 , 31 , 35-39 , 41-43 , 45 , 47-49 , 51 , 53-55 , 57 , 59 , 61 , 63 , 65
Web.	59 , 61
WTOMSG.	23 , 24
ZIPCODE.	41 , 42