EMC® Mainframe Enablers TimeFinder Utility for z/OS

Version 8.0 and higher

Product Guide

REV 04

Published May 2017

Dell believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED “AS-IS”. DELL MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. USE, COPYING, AND DISTRIBUTION OF ANY DELL SOFTWARE DESCRIBED IN THIS PUBLICATION REQUIRES AN APPLICABLE SOFTWARE LICENSE.

EMC Corporation
Hopkinton, Massachusetts 01748-9103
1-508-1000 In North America 1-866-464-7381
www.EMC.com

CONTENTS

PREFACE

As part of an effort to improve its product lines, EMC periodically releases revisions of its software and hardware. Therefore, some functions described in this document might not be supported by all versions of the software or hardware currently in use. The product release notes provide the most up-to-date information on product features.

Contact your EMC representative if a product does not function properly or does not function as described in this document.

Note: This document was accurate at publication time. New versions of this document might be released on the EMC online support website. Check the EMC online support website to ensure that you are using the latest version of this document.

Audience

This document is intended for the host system administrator, system programmer, or operator who will be involved in managing or operating the VMAX system.

Related documentation

The following documents provide additional information about Mainframe Enablers:

- *Mainframe Enablers Release Notes*
- *Mainframe Enablers Installation and Customization Guide*
- *Mainframe Enablers Message Guide*
- *AutoSwap for z/OS Product Guide*
- *Consistency Groups for z/OS Product Guide*
- *ResourcePak Base for z/OS Product Guide*
- *SRDF Host Component for z/OS Product Guide*
- *TimeFinder SnapVX and zDP Product Guide*
- *TimeFinder/Clone Mainframe Snap Facility Product Guide*
- *TimeFinder Mirror for z/OS Product Guide*
- *TimeFinder Utility for z/OS Product Guide*

The following documents provide additional information:

- *EMC VMAX All Flash Product Guide* — Documents the features and functions of the VMAX All Flash arrays.
- *HYPERMAX OS for EMC VMAX All Flash and EMC VMAX3 Family Release Notes* — Describe new features and any known limitations.
- *EMC VMAX3 Family with HYPERMAX OS Product Guide* — Documents the features and functions of the VMAX3 100K, 200K, and 400K arrays.

- *EMC VMAX Family with Enginuity Product Guide — Documents the features and functions of the VMAX 10K, 20K, and 40K arrays.*
- *E-Lab™ Interoperability Navigator (ELN) — Provides a web-based interoperability and solution search portal. You can find the ELN at https://elabnavigator.EMC.com.*

Conventions used in this document

EMC uses the following conventions for special notices:

⚠ CAUTION

CAUTION, used with the safety alert symbol, indicates a hazardous situation which, if not avoided, could result in minor or moderate injury.

Note: A note presents information that is important, but not hazard-related.

IMPORTANT

An important notice contains information essential to software or hardware operation.

Typographical conventions

EMC uses the following type style conventions in this document:

Normal	Used in running (nonprocedural) text for: • Names of interface elements, such as names of windows, dialog boxes, buttons, fields, and menus • Names of resources, attributes, pools, Boolean expressions, buttons, DQL statements, keywords, clauses, environment variables, functions, and utilities • URLs, pathnames, filenames, directory names, computer names, links, groups, service keys, file systems, and notifications
Bold	Used in running (nonprocedural) text for names of commands, daemons, options, programs, processes, services, applications, utilities, kernels, notifications, system calls, and man pages Used in procedures for: • Names of interface elements, such as names of windows, dialog boxes, buttons, fields, and menus • What the user specifically selects, clicks, presses, or types
Italic	Used in all text (including procedures) for: • Full titles of publications referenced in text • Emphasis, for example, a new term • Variables
`Courier`	Used for: • System output, such as an error message or script • URLs, complete paths, filenames, prompts, and syntax when shown outside of running text
`Courier bold`	Used for specific user input, such as commands
`Courier italic`	Used in procedures for: • Variables on the command line • User input variables
< >	Angle brackets enclose parameter or variable values supplied by the user

[ ]	Square brackets enclose optional values
|	Vertical bar indicates alternate selections — the bar means “or”
{ }	Braces enclose content that the user must specify, such as x or y or z
...	Ellipses indicate nonessential information omitted from the example

Where to get help

EMC support, product, and licensing information can be obtained on the EMC Online Support site as described next.

Note: To open a service request through the EMC Online Support site, you must have a valid support agreement. Contact your EMC sales representative for details about obtaining a valid support agreement or to answer any questions about your account.

Product information

For documentation, release notes, software updates, or for information about EMC products, licensing, and service, go to the EMC Online Support site (registration required) at:

`https://support.EMC.com`

Technical support

EMC offers a variety of support options.

Support by Product — EMC offers consolidated, product-specific information on the Web at:

`https://support.EMC.com/products`

The Support by Product web pages offer quick links to Documentation, White Papers, Advisories (such as frequently used Knowledgebase articles), and Downloads, as well as more dynamic content, such as presentations, discussion, relevant Customer Support Forum entries, and a link to EMC Live Chat.

EMC Live Chat — Open a Chat or instant message session with an EMC Support Engineer.

eLicensing support

To activate your entitlements and obtain your VMAX license files, visit the Service Center on `http://support.EMC.com`, as directed on your License Authorization Code (LAC) letter emailed to you.

For help with missing or incorrect entitlements after activation (that is, expected functionality remains unavailable because it is not licensed), contact your EMC Account Representative or Authorized Reseller.

For help with any errors applying license files through Solutions Enabler, contact the EMC Customer Support Center.

If you are missing a LAC letter, or require further instructions on activating your licenses through the Online Support site, contact EMC's worldwide Licensing team at licensing@emc.com or call:

- *North America, Latin America, APJK, Australia, New Zealand: SVC4EMC (800-782-4362) and follow the voice prompts.*
- *EMEA: +353 (0) 21 4879862 and follow the voice prompts.*

Your comments

Your suggestions will help us continue to improve the accuracy, organization, and overall quality of the user publications. Send your opinions of this document to:

VMAXContentFeedback@emc.com

CHAPTER 1
Introduction

This chapter covers the following topics:

EMC Mainframe Enablers and EMC TimeFinder Utility

EMC® TimeFinder® Utility is one of the EMC Mainframe Enablers. The EMC Mainframe Enablers include the following components that you can use to monitor and manage your storage:

- ResourcePak® Base for z/OS
- SRDF® Host Component for z/OS
- AutoSwap for z/OS
- Consistency Groups for z/OS
- TimeFinder SnapVX
- Data Protector for z Systems (zDP™)[1]
- TimeFinder®/Clone Mainframe SNAP Facility
- TimeFinder/Mirror for z/OS
- TimeFinder Utility

When you install the Mainframe Enablers kit, you install the software for all the components.

Licensing

Refer to the following documents for information about licensing:

- *Mainframe Enablers Installation and Customization Guide*
- *VMAX All Flash Product Guide*
- *VMAX3 Family Product Guide*
- *VMAX Family Product Guide*

Introduction to TimeFinder Utility

With TimeFinder Utility, you can relabel devices with new volsers and update the catalogs by renaming and recataloging both VSAM and non-VSAM datasets.

TimeFinder Utility enables you to perform the following operations in a production or simulation run:

- Assign a different volser to a device CUU
- Mass change the HLQs (high-level qualifiers) of datasets
- Recatalog the datasets to an existing or a new catalog

You can also execute IDCAMS[2] statements when required.

[1] zDP requires TimeFinder SnapVX but is a separately licensed product.

[2] IDCAMS stands for IDC Access Method Services. IDC is the IBM product code for VSAM.

Typically, you use TimeFinder Utility with the following products:

- EMC TimeFinder/Mirror
- EMC TimeFinder/Clone Mainframe SNAP Facility

TimeFinder Utility extends the processing of Business Continuance Volumes (BCVs) and standard devices. For example, it can be used to eliminate duplicate volume names in the mainframe environment.

Note: The *TimeFinder/Mirror for z/OS Product Guide* provides more information about BCVs and standard devices.

Limitations

- The following datasets are not processed:
  - Multi-volume datasets, if the volumes are not relabeled with a RELABEL statement or by the TimeFinder/Mirror SPLIT command with a VOLID(volser,E) option.

    Note: For information on the SPLIT command, refer to *TimeFinder/Mirror for z/OS Product Guide*.

  - Non-VSAM datasets that have never been opened.
  - Key range VSAM datasets
  - IDF data space and IDF datasets, as they have different formats in comparison with VSAM clusters
  - Volumes and datasets under the GSM environment
  - Open Edition HFS datasets
- A target dataset requires the original dataset to be a non-VSAM or a VSAM dataset cataloged to an EDF catalog.
- For VSAM datasets, the original catalog and the catalog after RENAME must be different, and must have catalog names of the same length.

CHAPTER 2
Getting Started

This chapter covers the following topics:

Running TimeFinder Utility

Sample JCL

The sample JCL for the TimeFinder Utility is supplied in the SMFE*vrm*.SAMPLIB members TFSAMP04 and TFSAMP05.

The JCL to run TimeFinder Utility is as follows:

```
//EMCTFU   EXEC PGM=EMCTFU,REGION=4M
//SYSOUT   DD SYSOUT=*
//TFINPUT  DD *
.
TimeFinder Utility command statements
.
/*
//TFMODEL  DD *
IDCAMS statement
/*
```

EXEC parameters

`PGM`

The main program of TimeFinder Utility is EMCTFU.

`REGION`

The region size of 4M is recommended.

DD statements

`TFINPUT`

Used to specify TimeFinder Utility commands.

Note: For information on available commands, refer to "Command Reference" on page 25.

TFINPUT must be 80-byte fixed length.

`TFMODEL`

Used to specify IDCAMS statements. This DD is optional.

Note: For information on IDCAMS statements, refer to "IDCAMS model statements" on page 35.

TFMODEL must be 80-byte fixed length.

Messages and return codes

Some messages you receive from TimeFinder Utility can be returned with a warning (W), an error (E), or a serious error message class (for example, E with a return code of 12). The message class you receive with these messages depends on the return code encountered.

The job step return codes can range from zero (0) to eight (8). Zero would be the least serious, as in an informational message. Eight would be an error.

The maximum return code to continue processing is defined by the MAXRC parameter described. If MAXRC is not set, TimeFinder Utility always returns the highest return code encountered as the job step return code.

Note: For a list of messages, refer to the *Mainframe Enablers Message Guide.*

CHAPTER 3
Operations

This chapter covers the following topics:

Updating catalogs

The process of updating the catalog includes two basic types of operations:

- Relabeling devices
- Renaming and recataloging datasets, which includes a preliminary phase to select the required datasets and catalogs.

Defining execution settings enables you to govern the process.

To update the catalog, include the statements you want into the TimeFinder Utility job JCL and run the job, as described in "Running TimeFinder Utility" on page 14.

Refer to "Command sequence example" on page 34 for an explained example.

For a detailed description of TimeFinder Utility processing, refer to "Processing flow" on page 20.

Relabeling devices

Table 1 lists operations for relabeling devices.

Table 1 Relabeling devices

Operation	Control
Assign a new volser to the device (relabeling)	RELABEL command
Define the online/offline status of a relabeled device	RELABEL command, OFFLINE keyword

Renaming and recataloging datasets

Table 2 lists operations for selecting datasets.

Table 2 Selecting datasets

Operation	Control
Select a device/volume on which datasets reside	PROCESS command
Select the type of datasets: VSAM or non-VSAM	PROCESS command, BOTH|VSAM|NON-VSAM keyword

Table 3 lists operations for selecting catalogs.

Table 3 Selecting/creating catalog

Operation	Control
Create a new catalog	CATALOG command, NEW keyword
Select an existing catalog	CATALOG command
Define a default catalog	CATALOG command, DEFAULT keyword
Select a catalog to add renamed datasets	RENAME command, CATALOG keyword
Select a catalog to obtain information for multi-volume datasets, GDGs[a], path names	SOURCECATALOG command

[a] GDG stands for Generation Data Groups.

Table 4 lists operations for renaming and recataloging datasets.

Table 4 Renaming datasets

Operation	Control
Uncatalog datasets before recataloging	CATALOG command, CLEANUP keyword
Mass change the HLQ of datasets (renaming) and recatalog the datasets	RENAME command

Defining execution settings

Table 5 lists operations for defining execution settings.

Table 5 Defining execution settings

Operation	Control
Do a simulation run	SIMULATE command
Set the maximum job step return code to continue processing	MAXRC command
Enable/disable debugging	DEBUG command
Select types of messages written to the log	MSG command
Redirect messages to the operator console	MSG command

Processing flow

Figure 1 shows an overview of the TimeFinder Utility processing flow.

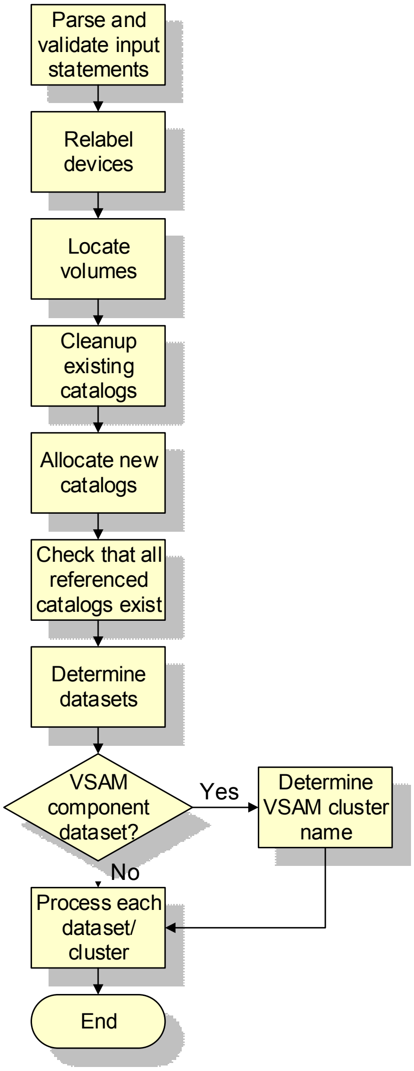
Parse and validate input statements
Relabel devices
Locate volumes
Cleanup existing catalogs
Allocate new catalogs
Check that all referenced catalogs exist
Determine datasets
VSAM component dataset?
Yes
Determine VSAM cluster name
No
Process each dataset/ cluster
End

Figure 1 Processing flow

The steps are as follows:

1. Parse and validate all input statements.
2. Relabel devices as directed by the RELABEL statements.
3. Locate all volumes defined by the PROCESS statements.
4. Perform all CATALOG CLEANUP requests.
5. Allocate all new catalogs, as directed by the CATALOG NEW statements.
6. Ensure that all other referenced catalogs exist.
7. Read in the dataset list from all volumes defined by the PROCESS statements.
8. Determine which datasets are selected for processing.

9. Process each selected dataset/cluster in the alphabetic order:

   For non-VSAM datasets:

   a. Determine the number and order of volumes for the dataset.
   b. For GDG datasets, ensure that the GDG base exists in the target catalog.
      - If the GDG base exists in the target catalog: proceed to step c.
      - If the GDG base does not exist in the target catalog and SOURCECATALOG GDG=YES:

        TimeFinder Utility looks up the GDG base in the source catalog. If the GDG base exists in the source catalog, TimeFinder Utility uses it to execute the GDGBASE model statement and creates the GDG base in the target catalog. If the GDG base does not exist in the source catalog, TimeFinder Utility executes the GDGDFLT model statement to create a default GDG base in the target catalog.
      - If the GDG base does not exist in the target catalog and SOURCECATALOG GDG=NO:

        TimeFinder Utility executes the GDGDFLT model statement to create a default GDG base in the target catalog.
   c. Rename the dataset in the VTOC and VTOC index on each volume.
   d. Catalog the dataset into the target catalog.

   For VSAM datasets:

   a. Determine the number and order of volumes for the index component (if present) and for the data component.
   b. Rename the component datasets in the VTOC and VTOC index on each volume.
   c. Recatalog the cluster into the target catalog.
   d. If a source catalog is available and VSAM path processing is enabled (SOURCECATALOG PATH=YES): obtain the path definitions for the cluster and add VSAM paths to the target catalog.
   e. For each alternate index that exists for this cluster and is selected for processing:
      - Determine the number and order of volumes for the index component.
      - Determine the number and order of volumes for the data component.
      - Rename the component datasets in the VTOC and VTOC index on each volume.
      - Recatalog the alternate index into the target catalog.
      - If a source catalog is available and VSAM path processing is enabled (SOURCECATALOG PATH=YES): obtain the path definitions for the cluster and add VSAM paths to the target catalog.
10. Processing completed.

Processing of multi-volume datasets

Two modes of processing are available for multi-volume datasets. The mode is selected by setting the SOURCECATALOG MVOL parameter to YES or NO.

SOURCECATALOG MVOL=YES

While processing a dataset, TimeFinder Utility performs a catalog lookup. If the dataset is not cataloged, TimeFinder Utility checks the VTOC entries for all of the volumes being processed for all of the segments of the dataset.

If all of the dataset segments are in the list of volumes being processed, TimeFinder Utility catalogs the dataset. If all of the dataset segments are not in the list of volumes being processed, TimeFinder Utility does not catalog the dataset, and processing continues.

If the dataset is cataloged, TimeFinder Utility uses the list of volumes for the dataset currently cataloged to determine whether this is a multi-volume dataset.

If the currently cataloged dataset resides on a single volume, TimeFinder Utility assumes that the dataset being processed is also a single volume dataset, and processing continues. However, if the currently cataloged dataset is a multi-volume dataset, TimeFinder Utility assumes that the dataset being processed is also a multi-volume dataset.

In fact, TimeFinder Utility uses the volume list from the currently cataloged dataset to determine the order and list of volumes for the dataset being processed. If this dataset segment does not map to the first volume in the volume list, TimeFinder Utility stops processing this dataset segment.

If this dataset segment does map to the first volume in the volume list, TimeFinder Utility checks that each of the original volumes maps to a volume currently being processed. If any of the original volumes dord not map to a volume currently being processed, TimeFinder Utility stops processing this dataset.

After TimeFinder Utility has determined all of the new volumes, it continues to process the dataset. TimeFinder Utility processes all of the dataset segments.

SOURCECATALOG MVOL=NO

In this case, TimeFinder Utility does not reference the source catalog to determine the number and sequence of dataset extents. TimeFinder Utility searches all volumes to be processed and locates all extents for a dataset.

For non-VSAM datasets, TimeFinder Utility uses the DS1VOLSQ field in the format 1 DSCB to determine the order of the volumes. In addition, TimeFinder Utility uses the flag DS1IND80 in field DS1DSIND to determine that it has found all of the necessary volumes.

Note: If a program creating the non-VSAM multi-volume dataset abends, the DS1IND80 flag is not set in the DS1DSIND field in the format 1 DSCB. Because the flag is not set, TimeFinder Utility is unable to determine that all of the necessary volumes have been found.

For VSAM datasets, these fields are not maintained and TimeFinder Utility uses another method. VSAM file information is maintained in the VVDS on each volume. TimeFinder Utility uses VVDS to determine the order of the volumes and whether all volumes are present.

Minimizing executing time

When you use TimeFinder Utility to process and rename large numbers (thousands) of datasets, the execution time can be very long (in the range of hours). You can take the following steps to minimize the execution time:

- Specify a temporary or empty user catalog in the CATALOG statement to eliminate the time spent on the CLEANUP phase.
- Specify a SOURCECATALOG statement that includes the DEFAULT=NO and DIRECT=YES parameters.
- Implement VLF for the user catalog that is specified in the CATALOG statement to be used by TimeFinder Utility processing. (See the IBM document, *MVS Initialization and Tuning Reference,* for details.) SYS1.PARMLIB member COFVLFxx should include CLASS NAME IGGCAS and the specified user catalog as an eligible major name (EMAJ).
- Review your GRS environment. Run time can be shorter in a GRS STAR configuration than in a RING configuration when in a SYSPLEX and there are more than two systems. In a RING configuration, check GRSCNFxx parmlib member for RESMIL and ACCELSYS parameter specifications. Reducing these to smaller values speeds up run time. Also, verify that all systems in the GRS RING have the same settings (when the settings do not match, the highest value is used).

CHAPTER 4
Command Reference

This chapter covers the following topics:

TimeFinder Utility commands

TimeFinder Utility commands are specified in the TFINPUT DD of the TimeFinder Utility JCL, as shown in "Running TimeFinder Utility" on page 14.

All syntax, keywords and parameters, must be in uppercase.

You can also include comment lines by using the following format:

```
/* comment */
```

Syntax conventions

The syntax conventions are as follows:

- Keywords appear in uppercase and must be spelled exactly as shown.
- Variables appear in lowercase and italics (for example, *column_name*). They represent user-supplied names or values.
- If punctuation marks, parentheses, arithmetic operators, or other such symbols are shown, enter them as part of the syntax.
- Default values are underlined. For example, `YES|NO` means that `NO` is the default value.

CATALOG

Determines the catalog to be processed or creates a new catalog.

Note: No ALIAS is created with the catalog.

Syntax

```
CATALOG catalog_name[,NEW][,VOLSER=volser]
[,DEFAULT][,CLEANUP]
```

Parameters

catalog_name

A one-to-44 character name of the catalog. This is a required parameter.

If you type one of the following words as the *catalog_name:*

- SYSCTLG
- SYSTEM_CATALOG
- SYSTEMCATALOG

TimeFinder Utility performs a lookup to determine the appropriate catalog, and the dataset is then cataloged into the system catalog structure.

Note: You cannot use the SYSCTLG keyword with the CLEANUP parameter. If you want to specify CLEANUP, add additional CATALOG statements for specific catalogs.

`CLEANUP`

Uncatalogs all existing catalog references to datasets on a volume selected for processing before recataloging datasets.

The CLEANUP parameter allows you to prepare existing catalogs and make them reusable.

`DEFAULT`

Sets the catalog identified by *catalog_name* as the default catalog for all RENAME commands that have no catalog specified. This parameter is optional.

`NEW`

Creates a new catalog on the volume defined by the VOLSER parameter.

This parameter is optional.

`VOLSER=`*`volser`*

Determines the volume on which to create the catalog. This parameter is required when you use the NEW keyword.

IMPORTANT

The volume you specify must be online to the TimeFinder Utility job step creating the catalog.

If *volser* is “SMSVOL,” TimeFinder Utility allocates the catalog without specifying a volser.

Note: Normally, the NEWCATLG model statement is used to allocate a new catalog. However, if you specify “SMSVOL,” the SMSCATLG model statement is used. For information on the IDCAMS model statements, refer to “IDCAMS model statements” on page 35.

Examples

- To define the CATALOG.USER catalog as the default catalog for this run:

  ```
  CATALOG CATALOG.USER,DEFAULT
  ```

- To define the system catalog as the default catalog for this run:

  ```
  CATALOG SYSCTLG,DEFAULT
  ```

- To cleanup volumes of the CATALOG.MVSICF.USER catalog:

  ```
  CATALOG CATALOG.MVSICF.USER,CLEANUP
  ```

- To create a catalog named MYCATALOG on the volume OP1234:

  ```
  CATALOG MYCATALOG,NEW,VOLSER=OP1234
  ```

DEBUG

Enables debug reporting during the processing.

Syntax

```
DEBUG
```

MAXRC

Sets the maximum allowed return code of a job step to continue processing.

You can specify any value from 0 to 8. The default value is 0.

Syntax

```
MAXRC={n}
```

Parameters

n

The maximum allowed return value.

- When *n* is 0 (zero), TimeFinder Utility treats any message as an error and returns the highest return code encountered as the job step return code. The returned message is an error.
- When *n* is from 1 to 8:
  - If you receive a return code that is not higher than *n*, the job step return code is set to 0. TimeFinder Utility treats any message as a warning and continues processing.
  - If the return code is higher than *n*, the job step return code is set to that value, and TimeFinder Utility treats any message as an error and stops processing.

If you receive a return code of 12, that value is greater than any possible MAXRC value, TimeFinder Utility treats the message as a serious error and stops processing.

MSG

Enables you to select types of messages written to the output log file. You can suppress individual messages or redirect messages to the operator console. You can also suppress warning messages about reserved[1] datasets and datasets that are not selected for processing.

IMPORTANT

Place the MSG statement(s) at the beginning of the input stream before any other statements so that the MSG statement(s) can be processed first.

[1] A reserved dataset is not eligible for processing. Catalogs, VVDS, and VTOC index datasets are reserved datasets.

Syntax

```
MSG [RESERVED=YES|NO][,UNMATCHED=YES|NO]
[,ID=message_id][,PRINT=YES|NO][,CONSOLE=YES|NO]
[,ROUTCDE=routing_code][,DESC=descriptor]
```

Parameters

`CONSOLE=YES|NO`

Determines whether to issue the message defined by the ID parameter to the operator console. This parameter is optional.

`DESC=`*`descriptor`*

Specifies a particular descriptor code to be used when sending a message to the operator console. Default is no descriptor code. This parameter is optional.

`ID=`*`message_id`*

Specifies a complete message ID for special processing. The message can be suppressed (PRINT=NO) or redirected to the operator console (CONSOLE=YES). This parameter is optional.

Note: To manipulate multiple message IDs, specify each message ID in a separate MSG statement. If you specify the same message ID multiple times, only the first occurrence is used.

`PRINT=YES|NO`

Determines whether to write the message defined by the ID parameter to the log. This parameter is optional.

Note: Unless you specify PRINT=NO, a message directed to the operator console is also written to the log.

`RESERVED=YES|NO`

Determines whether to issue warning messages about reserved datasets. This parameter is optional.

`ROUTCDE=`*`routing_code`*

Specifies a particular routing code to be used when sending a message to the operator console. Default is no routing code. This parameter is optional.

`UNMATCHED=YES|NO`

Determines whether to issue warning messages about datasets that are not selected for processing. This parameter is optional.

Examples

```
MESSAGES RESERVED=NO,UNMATCHED=NO
```

```
MESSAGES ID=BCVU010I,CONSOLE=YES,DESC=3
```

PROCESS

Defines a unit for processing. For multi-volume datasets, specify each volume in a separate PROCESS statement.

IMPORTANT

If PROCESS statements are present, then RENAME statements are required.

Syntax

```
PROCESS CUU=device,VOLSER=volser[,BOTH|VSAM|NON-VSAM]
```

Parameters

BOTH|VSAM|NON-VSAM

Specifies type of datasets to be processed: only VSAM datasets, only non-VSAM datasets, or both VSAM and non-VSAM datasets. This parameter is optional.

CUU=*device*

Specifies a device on the VMAX system. The device must be online.

VOLSER=*volser*

Specifies the volume serial number of the device.

Note: If you supply both volser and CUU, they must point to the same device.

Example

```
PROCESS CUU=F000,VSAM
```

RELABEL

Relabels a device with a new volser and brings it online, unless otherwise instructed by the OFFLINE parameter.

Relabeling updates the following information:

- The VVDS and VTOC index names
- The VVDS name entry
- The DSCBs

If a RELABEL statement is specified for a device that is online and already has the NEW-VOLSER (such as would occur in an accidental rerun), the RELABEL command is ignored, and a message is written to the log.

Syntax

```
RELABEL CUU=device,OLD-VOLSER=old,
NEW-VOLSER=new[,OFFLINE]
```

Parameters

CUU=*device*

The device to be relabeled.

`NEW-VOLSER=`*`new`*

The new volser of the device.

`OFFLINE`

Leaves the relabeled device offline. This parameter is optional.

If a PROCESS statement for the new volser is also present in the run, the OFFLINE parameter is ignored.

`OLD-VOLSER=`*`old`*

The current volser of the device.

Example

```
RELABEL CUU=F000,OLD-VOLSER=TSO001,NEW-VOLSER=BCV001
```

RENAME

Changes the high-level qualifier (HLQ) of datasets and recatalogs the datasets.

If no catalog is specified, then the default catalog is used. If there is no default catalog, the program terminates.

If RENAME statements are present, PROCESS statements are required.

To use the renamed datasets, execute the DEFINE ALIAS command first.

Syntax

```
RENAME old_hlq,new_hlq[,CATALOG=catalog]
```

Parameters

`CATALOG=`*`catalog`*

The catalog to which the newly renamed datasets are added. This parameter is optional.

If you type one of the following words as the *catalog_name:*

- SYSCTLG
- SYSTEM_CATALOG
- SYSTEMCATALOG

TimeFinder Utility performs a lookup to determine the appropriate catalog, and the dataset is then cataloged into the system catalog structure.

`new_hlq`

The new HLQ to be used for all datasets that match the *old_HLQ* parameter.

The *new_hlq* replaces the matching *old_hlq* characters. If the *old_hlq* or *new_hlq* ends in an asterisk, then the strings are used as is. Otherwise, a period is added to the string to make it an index level.

When *new_hlq* is the same as *old_hlq*, the datasets are only recataloged.

`old_hlq`

The current HLQ of all datasets to be renamed.

Note: If you use PROTECT ALL in RACF, the *old_hlq* must be defined in RACF.

Examples

- To rename datasets beginning with the string “SYS1.” by replacing “SYS1.” with “SYS1BCV.”:

```
RENAME SYS1,SYS1BCV
```

- To rename datasets beginning with the string “SYS” by replacing “SYS” with “SYSBCV”:

```
RENAME SYS*,SYSBCV*
```

- To catalog all BAP datasets on volumes being processed in the system catalog structure:

```
RENAME BAP,CATALOG=SYSTEM_CATALOG
```

SIMULATE

Used to perform simulation runs.

SIMULATE does not change catalogs, but creates VVDSs or VTOCs.

If both RELABEL and PROCESS statements are included in a simulation run for the same device, the RELABEL command is simulated and the PROCESS statement is bypassed, because the device is not actually available until the RELABEL completes successfully.

Errors may occur during the actual RELABEL, or PROCESS operations may not be detected when SIMULATE is included in the job. Though no error occurred while processing a job with a SIMULATE statement, errors can be returned when the job is executed with SIMULATE.

Syntax

```
SIMULATE
```

SOURCECATALOG

Determines whether to use the source catalog of datasets to resolve issues regarding multi-volume datasets, GDGs, and path names.

Syntax

```
SOURCECATALOG|SRCCAT DEFAULT|DFLT=YES|NO
[,GENERATIONDATAGROUP|GDG=YES|NO][,MVOL=YES|NO]
[,PATH=YES|NO][,DIRECT=YES|NO]
```

Parameters

`DEFAULT|DFLT=YES|NO`

Sets the default option for whether to use the source catalog or not.

“YES” means that issues regarding multi-volume datasets, GDGs, and path names are resolved based on the information from the source catalog.

This setting can be overridden for each specific area by optional parameters.

`DIRECT=YES|NO`

Determines whether TimeFinder Utility can directly access the source catalog during cleanup and the target catalog when cataloging datasets. This parameter is optional.

`GDG=YES|NO`

Determines whether to use the source catalog when creating GDG base entries in the target catalog. This parameter is optional.

When set to YES, the GDGBASE model statement is used to create GDG base entries based on the information from the source catalog. When set to NO, the GDGDFLT model statement is used to create a default GDG base.

Note: For information on IDCAMS model statements, refer to “IDCAMS model statements” on page 35.

`MVOL=YES|NO`

Determines whether to use the source catalog to obtain a list of volumes for a multi-volume dataset. This parameter is optional.

When set to YES, the list of volumes is taken from the source catalog. When set to NO, the list of volumes can be created as follows:

- For VSAM datasets — based on the information found in the VVDS.
- For non-VSAM datasets — based on the DS1VOLSQ field in the format 1 DSCB.

`PATH=YES|NO`

Determines whether to use the source catalog to define VSAM paths for a VSAM dataset in the target catalog. This parameter is optional.

When set to YES, TimeFinder Utility takes existing path definitions from the source catalog. When set to NO, TimeFinder Utility does not process paths.

Examples

```
SOURCECATALOG DEFAULT=NO,GDG=YES
```

```
SRCCAT MVOL=NO,GDG=NO,PATH=YES
```

Command sequence example

The following JCL example illustrates the use of the TimeFinder Utility command statements.

```
//JOB
//EMCTFU  EXEC PGM=EMCTFU,REGION=4M
//SYSOUT  DD SYSOUT=*
//TFINPUT DD *
RELABEL  CUU=100,OLD-VOLSER=TSO000,NEW-VOLSER=BCV000
RELABEL  CUU=101,OLD-VOLSER=TSO001,NEW-VOLSER=BCV001
RELABEL  CUU=102,OLD-VOLSER=TSO002,NEW-VOLSER=BCV002
PROCESS  VOLSER=BCV000
PROCESS  VOLSER=BCV001,VSAM
PROCESS  VOLSER=BCV002,NON-VSAM
CATALOG  CATALOG.DEFAULT,DEFAULT
CATALOG  CATALOG.TEMP,NEW,VOLSER=TS0001
RENAME  SYS,SYSBCV,CATALOG=CATALOG.TEMP
RENAME  TSO,TSO.ABC,CATALOG=CATALOG.TEMP
RENAME  USER,USERBCV
RENAME  USER1,USER1
/*
```

In this example, TimeFinder Utility completes the following steps:

1. Change volsers for devices identified with CUUs 100, 101, 102 from TSO000, TSO001, TSO002 to BCV000, BCV001, BCV002.

2. Select datasets:

   - All datasets on volume BCV000
   - VSAM datasets on volume BCV001
   - Non-VSAM datasets on volume BCV002

3. Select catalogs:

   - Select the existing catalog named CATALOG.DEFAULT to be used as the default catalog for all RENAME commands that have no command reference.
   - Create a new catalog named CATALOG.TEMP on volume TSO001.

3. Perform the following operations with the selected datasets:

   - For datasets beginning with “SYS”: replace “SYS” with “SYSBCV” and add the datasets to the catalog CATALOG.TEMP.
   - For datasets beginning with “TSO”: replace “TSO” with “TSO.ABC” and add the datasets to the catalog CATALOG.TEMP.
   - For datasets beginning with “USER”: replace “USER” with “USERBCV” and add the datasets to the CATALOG.DEFAULT catalog.
   - For datasets beginning with “USER1”: add the datasets to the CATALOG.DEFAULT catalog.

IDCAMS model statements

In special cases, you may need to provide an IDCAMS statement to recatalog datasets. Typically, this is not required unless a password needs to be supplied to complete the operation.

IDCAMS statements are specified in the TFMODEL DD statement of the TimeFinder Utility job JCL.

IMPORTANT

Each TimeFinder Utility TFMODEL DD statement may contain only one IDCAMS statement. Multiple IDCAMS statements are not rejected by TimeFinder Utility, but IDCAMS does not successfully process the command.

Syntax and parameters

```
=name
line1 -
line2 -
line_n
=name
line1 -
line2 -
line_n
```

Where:

`=name`

A string of up to 8 characters. Current valid names are:

`CTLGNV`

Catalog a non-VSAM dataset using its old name (non-SMS).

`RECATNV`

Recatalog a non-VSAM dataset using its old name (SMS).

`RECATIX`

Recatalog the VSAM indexed cluster using its old name.

`RECATAI`

Recatalog the VSAM alternate index using its old name.

`RECATCL`

Recatalog the VSAM cluster using its old name.

`RECATPG`

Recatalog a VSAM page dataset using its old name.

`LISTCTLG`

Catalog listing for cleanup.

`UNCATLG`

Remove entries from the catalog during cleanup.

`NEWCATLG`

Create a new ICF catalog (non-SMS).

`SMSCATLG`

Create a new ICF catalog (SMS).

`DEFNPATH`

Define a path for the cluster or an alternate index.

`GDGDFLT`

Default model to create a GDG base.

`GDGBASE`

Define a GDG base using information from the source catalog.

`line#-`

The lines following the *=name* card make up the IDCAMS statement.

Repeating blanks are eliminated. Columns 1-72 are parsed. If a dash (-) is encountered, parsing of that line is terminated.

Substitution values

The following embedded substitution values are supported:

%BASENAME%	The base cluster name associated with an alternate index.
%CATALOGNAME%	The catalog name from the RENAME statement or the default catalog.
%CLUSTERNAME%	For a VSAM dataset, its cluster name.
%DATANAME%	For a VSAM dataset, the data component name.
%DEVICELIST%	For a non-VSAM dataset, the list of devices containing the dataset. The order is the same as the %VOLUMELIST%.
%DSNAME%	The old name of the dataset.
%FILEDDNAME%	The DDNAME of the first volume in the %VOLUMELIST%. The presence of the DDNAME parameter in IDCAMS statements improves performance.
%GDGATTR%	GDG attributes: SCRATCH or NOSCRATCH or EMPTY or NOEMPTY.
%GDGBASE%	The GDG base name.
%GDGLIMIT%	GDG generation limit number.
%INDEXNAME%	For a VSAM indexed dataset, the index component name.
%INDEXTYPE%	For a VSAM dataset, its index type .
%INDEXVOLUMELIST%	For a VSAM KSDS or AIX dataset, the list of volumes containing the index component.
%NEWNAME%	The new name of the dataset.
%PAGEATTR%	Page dataset attributes: blank or SWAP.
%VOLUMELIST%	For a non-VSAM dataset, the list of volumes containing the dataset.

Default statements

```
=CTLGNV
 DEFINE NONVSAM (NAME ('%DSNAME%') DEVICETYPE (%DEVICELIST%) -
  VOLUMES (%VOLUMELIST%) ) CATALOG ('%CATALOGNAME%')
=RECATNV
 DEFINE NONVSAM (NAME ('%DSNAME%') RECATALOG -
  DEVICETYPE (%DEVICELIST%) VOLUMES (%VOLUMELIST%) ) -
  CATALOG ('%CATALOGNAME%')
=RECATIX
 DEFINE CLUSTER (NAME ('%DSNAME%') -
  %INDEXTYPE% RECATALOG FILE(%FILEDDNAME%) ) -
  DATA (NAME ('%DATANAME%') VOLUMES (%VOLUMELIST%) ) -
  INDEX (NAME ('%INDEXNAME%') VOLUMES (%INDEXVOLUMELIST%) ) -
  CATALOG ('%CATALOGNAME%')
=RECATCL
 DEFINE CLUSTER (NAME ('%DSNAME%') -
  %INDEXTYPE% RECATALOG FILE(%FILEDDNAME%) -
  VOLUMES (%VOLUMELIST%) ) DATA (NAME ('%DATANAME%') ) -
  CATALOG ('%CATALOGNAME%')
=RECATAI
 DEFINE ALTERNATEINDEX (NAME ('%DSNAME%') -
   RELATE('%BASENAME%') RECATALOG FILE(%FILEDDNAME%) ) -
  DATA (NAME ('%DATANAME%') VOLUMES (%VOLUMELIST%) ) -
  INDEX (NAME ('%INDEXNAME%') VOLUMES (%INDEXVOLUMELIST%) ) -
  CATALOG ('%CATALOGNAME%')
=RECATPG
 DEFINE PAGESPACE (NAME ('%DSNAME%') -
   RECATALOG FILE(%FILEDDNAME%) %PAGEATTR% -
   VOLUMES (%VOLUMELIST%) ) CATALOG (''%CATALOGNAME%'')'
=LISTCTLG
 LISTCAT VOLUME CATALOG ('%CATALOGNAME%')
=UNCATLG
 DELETE '%DSNAME%' NOSCRATCH PURGE CATALOG ('%CATALOGNAME%')
=NEWCATLG
 DEFINE USERCATALOG -
  ( NAME('%CATALOGNAME%') CYLINDERS(1,1) VOL(%VOLUMELIST%) -
   ICFCATALOG)
=SMSCATLG
 DEFINE USERCATALOG -
  ( NAME('%CATALOGNAME%') CYLINDERS(1,1) VOL(*) -
   ICFCATALOG)
=DEFNPATH
 DEFINE PATH (NAME ('%DSNAME%') -
   PATHENTRY('%BASENAME%') ) CATALOG ('%CATALOGNAME%')
=GDGDFLT
 DEFINE GENERATIONDATAGROUP (NAME ('%GDGBASE%') -
   LIMIT(255) NOEMPTY NOSCRATCH ) CATALOG ('%CATALOGNAME%')
=GDGBASE
 DEFINE GENERATIONDATAGROUP (NAME ('%GDGBASE%') -
   LIMIT(%GDGLIMIT%) %GDGATTR% ) CATALOG ('%CATALOGNAME%')
```