

English



Fujitsu Software BS2000

FDDRL

Physical Backup and Restore in BS2000

User Guide

Valid for:
FDDRL V21.0A

Edition June 2021

Comments... Suggestions... Corrections...

The User Documentation Department would like to know your opinion on this manual. Your feedback helps us to optimize our documentation to suit your individual needs.

Feel free to send us your comments by e-mail to: bs2000.info@fujitsu.com.

Certified documentation according to DIN EN ISO 9001:2015

To ensure a consistently high quality standard and user-friendliness, this documentation was created to meet the regulations of a quality management system which complies with the requirements of the standard DIN EN ISO 9001:2015.

Copyright and Trademarks

Copyright © 2025 Fujitsu

All rights reserved.

Delivery subject to availability; right of technical modifications reserved.

All hardware and software names used are trademarks of their respective manufacturers.

Table of Contents

- FDDRL V21.0A 2021-06** **6**
- 1 Preface** **7**
 - 1.1 Objectives and target groups of the manual** **8**
 - 1.2 Summary of contents** **9**
 - 1.3 Changes since the last edition of the manual** **10**
 - 1.4 Notational conventions** **11**
- 2 Introduction to FDDRL** **12**
 - 2.1 Physical data saving** **13**
 - 2.2 Data saving functions** **14**
 - 2.2.1 Copying of disks (COPY-DISK, COPY-PUBSET) 15
 - 2.2.2 Saving of disks (DUMP-DISK, DUMP-PUBSET) 16
 - 2.2.3 Restoring of disks (RELOAD-DISK, RELOAD-PUBSET) 17
 - 2.3 Save entity** **18**
 - 2.3.1 Single disk 19
 - 2.3.2 Disk set 20
- 3 Calling and executing FDDRL** **22**
 - 3.1 Calling FDDRL** **23**
 - 3.2 Saving pubsets** **25**
 - 3.3 Restoring pubsets** **26**
 - 3.4 Saving the home pubset online** **27**
 - 3.4.1 Properties of the online save 28
 - 3.4.2 Prerequisites for an online save 29
 - 3.4.3 Executing the online save 31
 - 3.4.4 Prerequisites for checking the online save 33
 - 3.5 Saving Clones (SHC-OSD)** **34**
 - 3.6 Mirroring with DRV** **36**
 - 3.7 Automating FDDRL saves** **37**
 - 3.8 FDDRL requests, FDDRL jobs, FDDRL subtasks** **38**
 - 3.8.1 FDDRL request 39
 - 3.8.2 FDDRL job 41
 - 3.8.3 FDDRL subtasks 42
 - 3.9 Central request monitoring on the SE server** **44**
 - 3.10 Data compression** **45**
 - 3.11 Data encryption** **46**
 - 3.12 FDDRL output** **47**
 - 3.12.1 Output of the FDDRL jobs 48
 - 3.12.2 Status output on the FDDRL subtasks 49

3.12.3 Results log of the FDDRL jobs	50
3.12.4 Tape information and information on disk backup	53
3.12.5 Job variable for backup volumes	54
3.12.6 Messages	55
3.13 Security	56
3.14 Diagnostic aids	57
3.15 Error handling	58
3.15.1 Program abortion	59
3.16 Performance	60
4 Handling data volumes	61
4.1 Disks	62
4.1.1 Input disk	63
4.1.2 Output disk	64
4.1.3 Disk formats	65
4.1.4 Replacing defective pubset disks	66
4.2 Backup volumes	67
4.2.1 Premounting backup volumes	68
4.2.2 Tape error statistics	69
4.2.3 Protecting backup volumes	70
4.2.4 Compatibility of the backup volumes	71
4.3 Magnetic tape cartridge devices with stacker	72
4.4 Device management for FDDRL	73
5 Scope of supply and installation of FDDRL	74
5.1 Scope of supply of FDDRL	75
5.2 Installation of FDDRL	77
6 FDDRL statements	80
6.1 Overview of the FDDRL statements	81
6.2 Description of the FDDRL statements	84
6.2.1 COPY-DISK Copy disk	85
6.2.2 COPY-PUBSET Copy all disks of pubset	87
6.2.3 CREATE-JOBS Create FDDRL jobs	89
6.2.4 DELETE-REQUESTS Delete completed FDDRL requests	90
6.2.5 DUMP-DISK Save disk	92
6.2.6 DUMP-PUBSET Save all disks of pubset	95
6.2.7 END Terminate FDDRL	98
6.2.8 MODIFY-FDDRL-PARAMETERS Define execution parameters	99
6.2.9 RELOAD-DISK Restore disk	104
6.2.10 RELOAD-PUBSET Restore a pubset	108
6.2.11 SHOW-FDDRL-PARAMETERS Show the current FDDRL parameters	112
6.2.12 SHOW-FDDRL-STATUS Show running (foreign) FDDRL jobs	116

6.2.13 SHOW-FDDRL-TAPE-INFORMATION Show the tape information of an FDDRL tape	121
6.2.14 SHOW-REQUESTS Show running and completed FDDRL requests	127
6.2.15 START-JOBS Start FDDRL jobs	132
7 Duplicating FDDRL backup volumes with PERCON	134
8 Administering the volumes with MAREN	136
8.1 MAREN catalog	137
8.2 MAREN tape set	138
8.3 Creating an overview of the FDDRL backup volumes	140
8.4 Subsequently modifying the expiration date of a backup volume	141
8.5 Adjusting volume characteristics of earlier FDDRL versions	142
8.6 Adjusting volume characteristics of PERCON copies	143
8.7 Defining a location	144
9 Sample applications	145
9.1 Copying a private disk for backup	146
9.2 Copying a private disk for restoration	149
9.3 Saving an SM pubset to MTC	152
9.4 Restoring an SM pubset from MTC	156
9.5 Saving private disks	160
9.6 Restoring a private disk from a disk set save	163
9.7 Saving the home pubset from the mirror disks	166
9.8 Restoring a pubset on new disks	171
9.9 Saving the home pubset online	181
10 Glossary	188
11 Abbreviations	192
12 Related publications	194

Preface

FDDRL (**F**ast **D**isk **D**ump and **R**e**L**oad) is a BS2000 software product for saving the contents of BS2000 disks and pubsets. FDDRL supports both public and private disks. The pubsets can be either single-feature (SF) or system-managed (SM) pubsets.

FDDRL supports physical saving of disk contents to disks and magnetic tape cartridges (MTC). If the contents of a disk are accidentally destroyed, you can restore the entire disk from the backup volumes.

This manual describes version V21.0A of FDDRL.

Objectives and target groups of the manual

FDDRL is employed for system backup. It is intended for systems support staff for use in IT centers. This manual is also aimed at this target group.

Summary of contents

The manual is structured as follows:

- Chapters 1 through 5 describe the functions, execution, installation and invocation of FDDRL.
- Chapter 6 contains all FDDRL statements in alphabetical order.
- Chapter 7 describes how to copy FDDRL backups using the software product “PERCON”.
- Chapter 8 is compulsory reading if you are using the software product “MAREN” for MTC administration.
- Chapter 9 contains an example each of the COPY, DUMP and RELOAD functions.
- This is followed by various lists designed to help you in using this manual.

Additional product information

Current information, version and hardware dependencies, and instructions for installing and using a product version are contained in the associated Release Notice. These Release Notices are available online at <http://bs2manuals.ts.fujitsu.com>.

Changes since the last edition of the manual

This manual describes the functionality of FDDRL V21.0A. The following changes have been made since the last edition of the manual:

Functional changes

- Procedure *LIBRARY-ELEMENT(SYSPRC.FDDRL.210,MERGE-PAR) for merge of FDDRL parameter files.
- The procedures SYSPRC.FDDRL.200.CHECK-OLS and SYSPRC.FDDRL.200.CHECK-OLS.C has been removed and are now available as J and SYSJ elements in SYSPRC.FDDRL.210.

Notational conventions

Special characters and the notation used to in the command and statement formats. The current SDF metasyntax and the description of the SDF data types are contained in the "Commands" manual [9].

The following notational conventions are used in this manual:

- In application examples, user input appears in Gothic typeface. All input lines are completed with `↵` at alphanumeric terminals and with `EM DUE` at block terminals. Therefore these key pictograms are not shown at the end of the input lines.

Output from the operating system appears in Gothic typeface.

- Literature references in the text are given in abbreviated form in inverted commas. The full titles of all publications referred to, together with a brief description of each, can be found in [chapter "Related publications" \(fddrlen, #103\)](#) at the back of this manual.
- Cross-references within this manual include the chapter and, if necessary, also the section. References to subjects which are described in other manuals contain only the abbreviated title of the manual. You can then use the index in that manual to find the corresponding page.

! This symbol indicates a warning, which you **must** observe for reasons of system and operating security.

i This symbol draws attention to information of particular importance.

Introduction to FDDRL

The software product FDDRL (**F**ast **D**isk **D**ump and **R**e**L**oad) is used for the physical saving (dumping) and restoring (reloading) of disks and pubsets in a BS2000 system. FDDRL is used for system backup.

Table of Contents

- [Physical data saving \(FDDRL V21.0A 2021-06, #8\)](#)
- [Data saving functions \(FDDRL V21.0A 2021-06, #9\)](#)
 - [Copying of disks \(COPY-DISK, COPY-PUBSET\) \(FDDRL V21.0A 2021-06, #10\)](#)
 - [Saving of disks \(DUMP-DISK, DUMP-PUBSET\) \(FDDRL V21.0A 2021-06, #11\)](#)
 - [Restoring of disks \(RELOAD-DISK, RELOAD-PUBSET\) \(FDDRL V21.0A 2021-06, #12\)](#)
- [Save entity \(FDDRL V21.0A 2021-06, #13\)](#)
 - [Single disk \(FDDRL V21.0A 2021-06, #14\)](#)
 - [Disk set \(FDDRL V21.0A 2021-06, #15\)](#)

Physical data saving

Physical data saving involves the saving of all occupied blocks of a volume. FDDRL physically saves and restores the contents of whole disks in their physical order. All the data on a disk to be saved, including the volume labels, is transferred block by block in the same physical order onto other volumes, from where it can be restored if necessary.

The occupancy information of a disk (VTOC area) is determined using the F5 labels of the disk. FDDRL is able to identify that the F5 labels are not consistent in relation to the DMS catalog, for example, in the event that a pubset was not exported correctly. In this case, the disk is saved without using the occupancy information, or in other words, the disk is saved in its entirety.

If, for example, a disk has to be restored, it is not necessary to initialize the alternate disk with the desired VSN (volume serial number) beforehand. The alternate disk can be put into operation with the original VSN immediately after the FDDRL run.

FDDRL also retains the physical concatenations on a bootable disk, with the result that the copy can likewise be booted.

FDDRL cannot be used to save or restore individual files or to reorganize disks. These operations require software products for logical data saving such as "HSMS" (see the "HSMS" manual [6]) or "ARCHIVE" (see the "ARCHIVE" manual [1]). Alternatively BS2000 supports logical saving to so-called Snapsets at command level (see the "Introductory Guide to Systems Support" [5]).

FDDRL saves the (occupied) blocks of the disks in ascending order without knowing their file assignment.

Data saving functions

FDDRL offers the following two utilization models for disks:

- Copying of disks between disk devices (COPY statements) for saving or restoration.
- Backup of disks to magnetic tape cartridge (DUMP statements) and restoring disks from the save cartridge (RELOAD statements).

Copying of disks (COPY-DISK, COPY-PUBSET)

FDDRL copies the contents of a disk which need not belong to a pubset or the content of all disks of a pubset (input disk(s)), including all the volume labels, onto an equal number of other disks (output disk(s)). The input disk and the output disk must have the same device type.

You can use the COPY function for both saving and restoring one or more disks. The ACTION operand allows you to define which of these activities you want to execute:

- To save the original disk to the output disk, specify ACTION=*SAVE.
- To restore the original disk from the copy, specify ACTION=*RESTORE.

Note

The purpose of copying (save or restore) shows FDDRL which disk must be the source or target disk. This permits inadvertent switching of the disks with the same VSN to be recognized by the operator.

Save: Copying a newer disk to an older disk.

Restore: Copying an older disk to a newer disk.

The COPY-DISK statement is used to copy a single disk, the COPY-PUBSET statement to copy all disks of a pubset. The pubsets can be single-feature (SF) or system-managed (SM) pubsets.

Saving of disks (DUMP-DISK, DUMP-PUBSET)

FDDRL saves one or more disks (input disk(s)) to MTC. The BS2000 standard volume labels (SVLs) are also written to the MTC. These labels contain information about the VSN of the disk and the anchoring of any IPL files.

In the case of pubsets, either single disks are saved to independent MTC (see save entity "[Single disk](#)") or all disks are saved to common MTC (see save entity "[Disk set](#)").

The DUMP-DISK statement is used to save (also multiple) single disks, the DUMP-PUBSET statement to save all disks of a pubset.

Restoring of disks (RELOAD-DISK, RELOAD-PUBSET)

If necessary, one or more or all disks of the pubset can be restored from the backup volume. Each disk used for restoration has the same device type as the disk which was originally saved, and is overwritten by the contents of this disk. After restoration, the system addresses the newly restored disk under the VSN of the originally saved disk.

The RELOAD-DISK statement is used to restore (reload) one or more individual disks.

All the disks of a pubset are restored (reloaded) using the RELOAD-PUBSET statement.

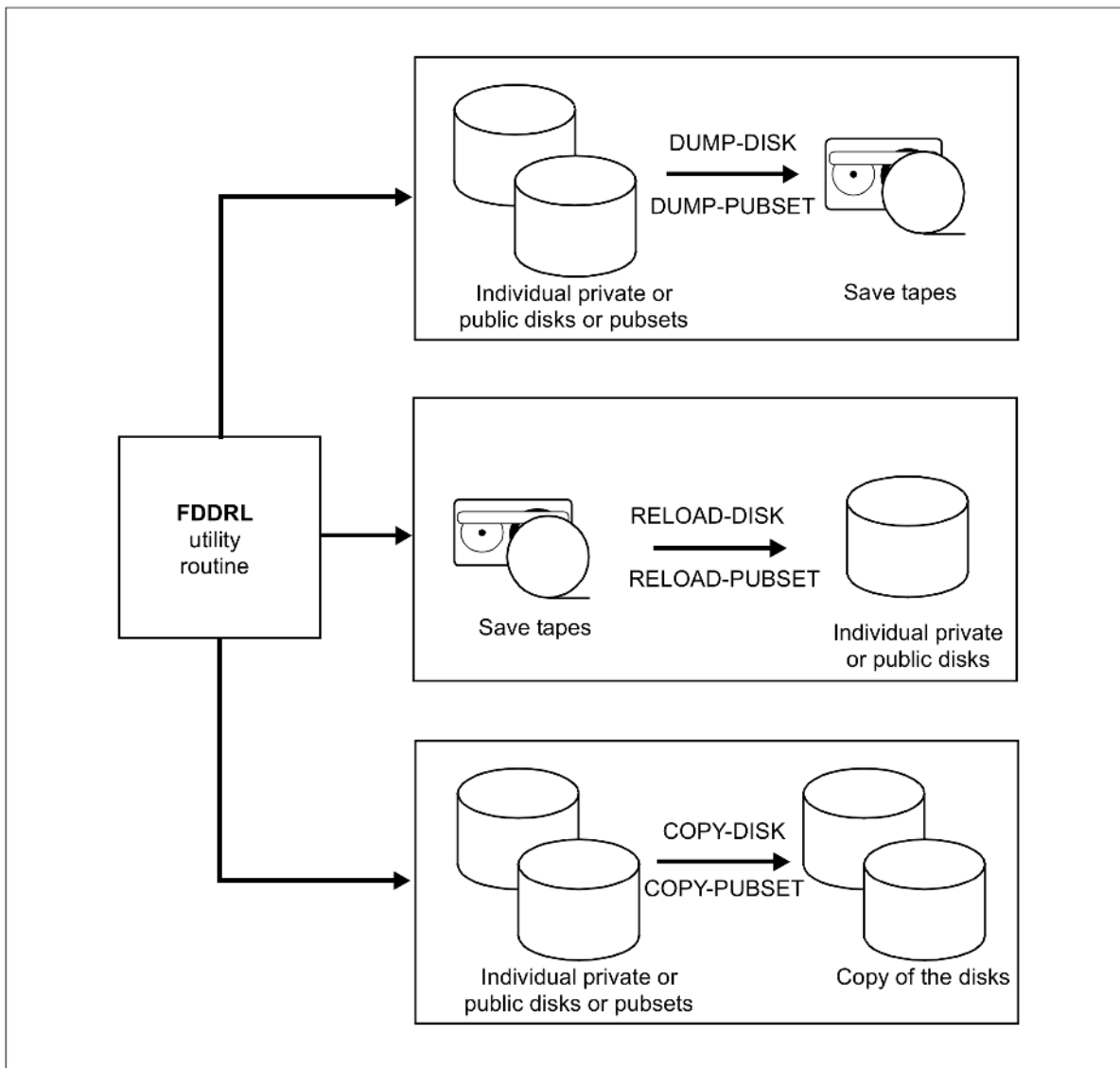


Figure 1: Data saving functions of FDDRL

Save entity

The data saving functions of FDDRL support two save entities

- **Single disk**
Single disk is a public or private disk.
- **Disk set**
A disk set consists either of all the disks of a pubset (multiple disk sets can also be set up in the case of DUMP-PUBSET) or of multiple private disks.

Irrespective of the save entity, the saves on MTC (tape sets) cannot be extended later.

Single disk

In all FDDRL versions up to V15.0 disk backups were performed on the basis of this save entity. Each individual disk forms a save entity which FDDRL saves independently of others on its own MTC (tape sets).

FDDRL generates the single disk save entity in the following cases:

- In the case of DUMP-DISK, precisely one disk is specified in the DISK operand.
- In the case of DUMP-PUBSET, the SAVE-ENTITY=*SINGLE-DISK operand is specified.

An FDDRL job is created for each disk.

Disk set

FDDRL also supports the disk set save entity. Here multiple disks form a save entity which FDDRL saves on common MTC (tape set) (disk transition without change of tape). The disk set save entity makes optimum use of the tape cartridge capacity and significantly reduces the number of cartridges used.

FDDRL generates the disk set save entity in the following cases:

- In the case of DUMP-DISK, more than one disk is specified in the DISK operand.
- In the case of DUMP-PUBSET, the SAVE-ENTITY=*DISK-SET operand is specified. The number of disk sets can also be specified (the default is NUMBER-OF-DISK-SETS=1).

An FDDRL job is created for each disk and for each disk set.

The disk set save entity requires at least V16 tape format (the FDDRL parameter TAPE-FORMAT=*STD presets V17 tape format). Disk set saves can therefore be restored only using FDDRL V16.0A or V17.0A or higher.

RELOAD-DISK enables all, some or single disks of a disk set save to be restored.

The entire pubset can be restored using the RELOAD-PUBSET statement. If it was only possible to restore some of the disks using RELOAD-PUBSET because of disruptions in operation, restoration of the rest of this pubset's disks can be performed later with RELOAD-DISK. The use of RELOAD-DISK instead of RELOAD-PUBSET prevents disks that have already been created successfully from being restored again.

With the disk set save entity, an FDDRL job is created for each disk and each disk set. Each disk set is processed by an FDDRL subtask (when PROCESS-JOBS= *MAINTASK, by the main task).

Number of disk sets with DUMP-PUBSET

With DUMP-PUBSET you use the NUMBER-OF-DISK-SETS operand to specify the number of disk sets and consequently the number of tape sets generated. In this way you control the size (number of tapes) of the tape sets and the parallelism required for the save. The number should not exceed the value specified by the FDDRL parameter TASK-LIMIT, otherwise tapes will be used for empty disk sets.

You should avoid more than 255 tapes being required for saving a disk set because MAREN can only accommodate a maximum of 255 tapes in a tape set.

If multiple disk sets are generated with DUMP-PUBSET (NUMBER-OF-DISK-SETS >1), the assignment of disk to disk set is not predictable because this is defined dynamically in the save run so as to ensure that an equal load is placed on the subtasks.

Example of saving a pubset using multiplexing (multiplexing level 2)

Pubset A, which consists of 5 disks, is saved to a tape set:

```
//DUMP-PUBSET PUBSET=A,SAVE-ENTITY=*DISK-SET
```

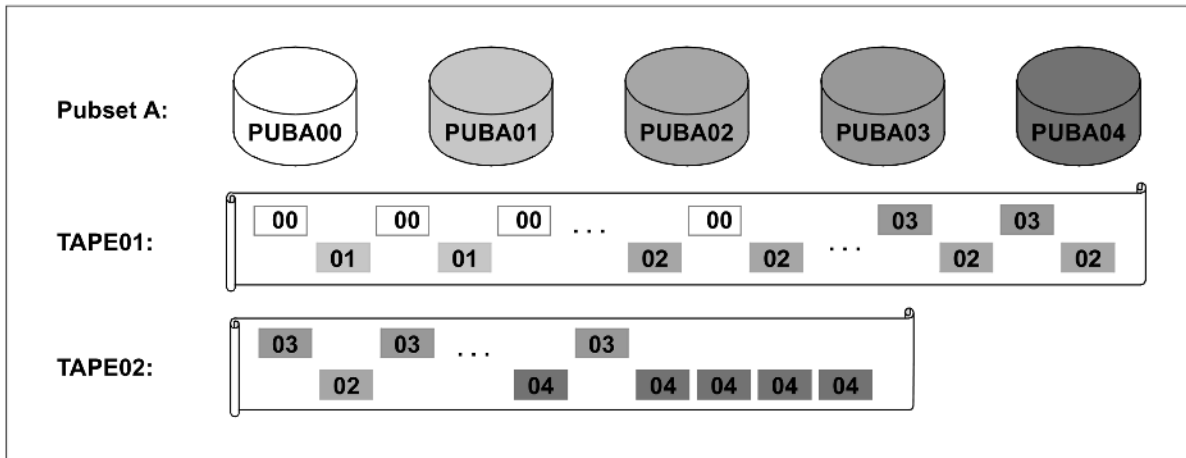


Figure 2: Saving a pubset as a disk set to a tape set using multiplexing level 2

Example of saving a pubset as disk sets to 2 tape sets

Pubset A, which consists of 5 disks, is saved to two tape sets:

```
//DUMP-PUBSET PUBSET=A,SAVE-ENTITY=*DISK-SET(NUM-OF-D-SETS=2)
```

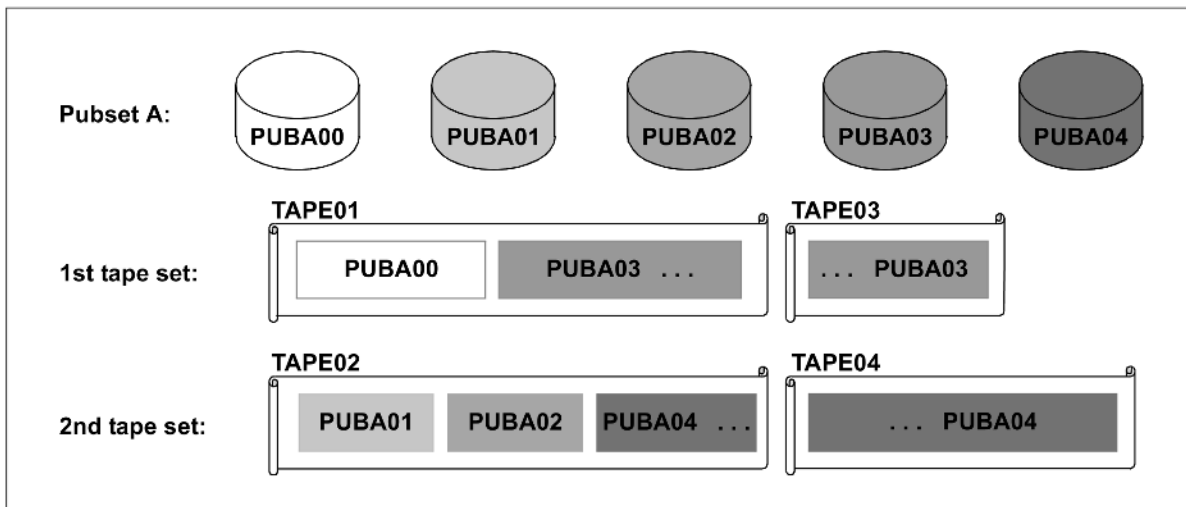


Figure 3: Saving a pubset as a disk set to 2 tape sets

In order to make the example more straightforward, the merging of the disk data through multiplexing is not shown.

Calling and executing FDDRL

This chapter provides important information on the invocation and execution of FDDRL.

Table of Contents

- [Calling FDDRL \(FDDRL V21.0A 2021-06, #17\)](#)
- [Saving pubsets \(FDDRL V21.0A 2021-06, #18\)](#)
- [Restoring pubsets \(FDDRL V21.0A 2021-06, #19\)](#)
- [Saving the home pubset online \(FDDRL V21.0A 2021-06, #20\)](#)
- [Saving Clones \(SHC-OSD\) \(FDDRL V21.0A 2021-06, #25\)](#)
- [Mirroring with DRV \(FDDRL V21.0A 2021-06, #26\)](#)
- [Automating FDDRL saves \(FDDRL V21.0A 2021-06, #28\)](#)
- [FDDRL requests, FDDRL jobs, FDDRL subtasks \(FDDRL V21.0A 2021-06, #29\)](#)
- [Central request monitoring on the SE server \(FDDRL V21.0A 2021-06, #33\)](#)
- [Data compression \(FDDRL V21.0A 2021-06, #34\)](#)
- [Data encryption \(FDDRL V21.0A 2021-06, #35\)](#)
- [FDDRL output \(FDDRL V21.0A 2021-06, #36\)](#)
- [Security \(FDDRL V21.0A 2021-06, #43\)](#)
- [Diagnostic aids \(FDDRL V21.0A 2021-06, #44\)](#)
- [Error handling \(FDDRL V21.0A 2021-06, #45\)](#)
- [Performance \(FDDRL V21.0A 2021-06, #47\)](#)

Calling FDDRL

The FDDRL utility routine is started with the `/START-FDDRL` command. This command belongs to the UTILITIES domain. It can only be called by users with TSOS privilege. FDDRL can be called in interactive or in batch mode. Messages are output to SYSOUT.

Each time the program starts, any system-specific adjustments to the FDDRL parameters made in the FDDRL parameter file are registered and displayed.

Statements are read from SYSDTA. Erroneous statements are rejected. In interactive mode they can be corrected immediately. In batch mode or procedure mode the spin-off mechanism is triggered.

If FDDRL is terminated because of an error, BS2000 issues message EXC0732. This message contains an insert in the form of a seven-character termination code assigned by FDDRL (FBDxxxx). This code identifies a message text which describes the cause of the error in more detail [chapter "Messages"](#). You can use an HTML application to find the FDDRL messages on the Manual Server (URL: <http://bs2manuals.ts.fujitsu.com>) and on the "BS2000 SoftBooks" DVD.

START-FDDRL	Alias: FDDRL
VERSION = *STD / <product-version 3..7 without-man>	
MONJV = *NONE / <filename 1..54 without-gen-vers>	
CPU-LIMIT = *JOB-REST / <integer 1..32767 seconds>	

VERSION = *STD / <product-version 3..7 without-man>

Version number of FDDRL in the format mm.n[a[so]], e.g. 21.0 or 21.0A or 21.0A00 (see also the description of the SDF data types in the "Commands" manual [9]).

The default is *STD, i.e. the currently installed version of FDDRL is loaded.

MONJV = *NONE / <filename 1..54 without-gen-vers>

This operand is only available if the "Job Variables" software product is being used (see also the "Job Variables" manual [8]).

Name of the monitoring job variable which is to monitor the FDDRL run. If the monitoring job variable does not yet exist, a new one is created.

When the FDDRL run is completed, the monitoring job variable assumes one of the following values:

\$T 0000 FDDRL run terminated normally, no error
\$A 3030 FDDRL run terminated abnormally, error occurred

CPU-LIMIT = *JOB-REST / <integer 1..32767 seconds >

Maximum CPU time in seconds that FDDRL (main task) may take to execute. If FDDRL exceeds the specified time during execution, in interactive mode it is interrupted and a message is output. The user can then request a dump, abort or continue the program. If the time is exceeded in batch mode, FDDRL is terminated.

CPU-LIMIT = *JOB-REST

If the job was started with a time limit (depending on the job class), the remaining CPU time is used as a time limit for FDDRL. Otherwise, FDDRL runs without a time limit.

Saving pubsets

As well as individual private or public disks, FDDRL can also save complete pubsets. The advantage of saving pubsets as opposed to saving individual disks is that FDDRL independently determines all disks of the pubset, thus guaranteeing a complete and consistent backup. The statements COPY-PUBSET and DUMP-PUBSET are available for saving pubsets.

FDDRL supports the saving of complete SF and SM pubsets. You do not need to distinguish between these two pubset types, as FDDRL determines the type itself by evaluating the MRS catalog:

- A conventional SF pubset can be recorded in the master catalog but need not be.
- An SM pubset must be entered in the master catalog with the name and device type of the control volume set. The entry must be up to date and complete. If an SM pubset is not entered in the master catalog, FDDRL assumes when analyzing the statement that it is dealing with an SF pubset. If this is not the case, the error will not be detected until the FDDRL job is generated (CREATE-JOBS statement).

To determine all volume sets of an SM pubset, a so-called mini import is performed on the control volume set. Volume sets with the status “defined only” or “defect” are not included in the save.

FDDRL checks the time stamps of all disks of a volume set to make sure that they belong to that set (consistency check). This involves the requesting of all disks of the volume set before processing and the reading of their BS2000 standard labels.

FDDRL can only process a pubset if it has exclusive access to it. All disks on the pubset must be attached. The pubset may not be imported. An imported pubset can be saved when consistent copies of all disks exist (created by [Saving Clones \(SHC-OSD\)](#) or [Mirroring with DRV](#)).

Saving the home pubset

Disk mirroring with DRV or SHC-OSD allows you to split the mirrored disks of the home pubset during live operation. FDDRL can save the mirror disks which have been split in this way in the same manner as the pubset's original disks. This means that the home pubset is saved during BS2000 operation.

An online save can be started with FDDRL for the unmirrored home pubset during ongoing operation. The prerequisites and how the save is performed are described in the [chapter "Saving the home pubset online"](#).

If an online save with FDDRL is not possible, the home pubset can only be saved offline. In this case the active system must be terminated in order to perform the save. The disks can then be attached to another system or standby system and saved using FDDRL.

Adjusting the size of volumes using SPACEOPT

When a volume is copied and restored with FDDRL, the capacity of the saved volume is always taken over, even when the output disk had a higher capacity beforehand.

Following restoration, a mixture created in this way can be removed by adjusting the size with the SPACEOPT command ADAPT-PUBSET-SIZE. This function is only available for disks of a pubset.

A pubset which was enlarged by adjusting the size with /ADAPT-PUBSET-SIZE should be imported and exported again before being saved so that it is backed up on the basis of a consistent F5 label and the save scope to the allocated PAM pages is thus reduced.

Restoring pubsets

For consistency reasons, pubsets should always be saved and restored in their entirety.

FDDRL can guarantee the completeness and consistency if you save a pubset using DUMP-PUBSET and the SAVE-ENTITY=*DISK-SET operand and restore it using the RELOAD-PUBSET statement.

If a pubset was saved using DUMP-PUBSET and SAVE-ENTITY=*SINGLE-DISK or each individual pubset was saved using DUMP-DISK, when you restore this pubset you must restore each individual disk with RELOAD-DISK. Here you yourself must ensure that you restore the pubset completely and consistently (i.e. of the basis of a uniform save status).

The tapes which belong to a save can be ascertained from the save log, from FDDRL job variables or on the basis of the file name using the tape management system MAREN.

Pubsets can only be restored when they have been exported with EXCAT.

Saving the home pubset online

With the online save FDDRL offers the option of saving the home pubset during ongoing operation. An online save can be considered when the home pubset has no mirror disks which are available for a save and it is not possible or desirable to interrupt system operations to save the home pubset.

The online save is called in the DUMP-PUBSET or COPY-PUBSET statement using the PUBSET=*HOME(...) operand. The following control options are available here:

- Specification of an alternative default pubset (DEFAULT-PUBSET operand)

As the home pubset is also always the default pubset for the user ID TSOS, the SYSOUT logs of the FDDRL subtasks and the job variables for recording the tape VSNs are also contained on the home pubset.

Consequently the SYSOUT log is disabled when an online save takes place and the job variables are not updated during the runtime, but only at the end of the save.

A different pubset can optionally be specified for the logs and job variables. This permits logging and updating of the job variables to continue during the online save.

- Time limit for the write lock (TIME-LIMIT operand)

The write lock which is set for the home pubset during the save is canceled when the save terminates, and at the latest when the specified time limit is reached (default: 5 minutes).

Properties of the online save

When the save starts, FDDRL stops all write I/Os for all disks of the pubset at the same time. The disks of the pubset are then as consistent with each other as mirror disks after splitting. As soon as a disk has been saved, FDDRL cancels the write lock for precisely this disk. Only when all disks have been saved does FDDRL cancel the write lock for the pubset.

During the write lock all the tasks in the system which execute write operations on the home pubset are placed in a wait state (VPASS) until the online save has been completed. Access to the PAGING file and MSCF access to the watchdog file are excluded from this.

FDDRL uses the F5 label from a storage area of the Allocator to save the disk, and consequently only the disk's allocated areas are saved.

FDDRL reduces the save scope by excluding disk areas with large files which are unimportant for the restoration (SYS.PAGING, SNAPFILE, SLEDFILE) from the save.

The time limit (TIME-LIMIT operand) determines the maximum time the write lock will apply for the home pubset. 1 to 120 minutes can be set; the default value is 5 minutes. If the time limit is reached before the end of the online save, FDDRL cancels the write lock for the home pubset, but continues the save. In this case the save will be inconsistent and unusable for restoration. However, the information about the time and space requirements of the save can be evaluated for subsequent saves.

Effects of the write lock

Tasks are also locked after a file has been read if the access date is updated in the file catalog (ACCESS-DATE) when the file is closed (e.g. when reading with `/SHOW-FILE`).

Indirect lock

Tasks can also be locked indirectly during the write lock because they request services from server tasks which are themselves locked. The indirectly locked tasks as a rule then wait in a communication bourse.

CONSLOG logging

Console messages are initially collected in an I/O buffer and then written to the CONSLOG file by the CLOG task block by block. As it is not possible to write the CONSLOG file during the online save, the CLOG task waits (with VPASS in PPAM). All messages for CONSLOG being buffered.

When more than 32 messages have been buffered per task, the message processing (TYPIO) also places the reporting task in a wait state (VPASS in the ECTYP module). Systems support should therefore as far as possible restrict the message output to console during an online save (e.g. switch off BCAM monitoring).

Prerequisites for an online save

Some system-specific prerequisites must be observed to ensure that the online save is as free as possible from error. The validation procedure supplied with FDDRL ([chapter "Prerequisites for checking the online save"](#)) checks whether the following prerequisites are provided and, if necessary, suggests measures to comply with requirements:

- Write caching with DAB may not be enabled for the home pubset.
It must be terminated before the save so that all data can be written to the disks. After the save it can be switched on again.
- In the case of a mirrored home pubset only the copied mirror pubset should be saved instead of the online save (see [chapter "Saving Clones \(SHC-OSD\)"](#)).
- BCAM status JV
If a BCAM status JV for sign-of-life monitoring exists, BCAM updates the JV every 60 seconds. In this case BCAM is blocked by the write lock during the online save and accepts no further connections. You must check whether the status JV is actually required by applications:
 - If the status JV is not required, it should be deleted (`/MODIFY-HOST-ATTR STATUS-JV=*NONE`).
 - If the status JV is required, it should be moved to another pubset.
- BCAM monitoring
In the case of BCAM monitoring on the console, a large number of messages are output to the console and logged accordingly in the CONSLOG file. BCAM monitoring should consequently be switched off at least during the online save (`/BCMOFF`).
- MAREN (MARENCP, MARENUCP)
If tape swapping is required during the online save with `//DUMP-PUBSET`, it must be ensured that MAREN is not blocked by write I/Os on the home pubset. The files concerned should be moved to another pubset.
MARENCP can be blocked if the following files reside on the home pubset:
 - MARENCAT (volume catalog)
 - MARENLOG (logging file)
 - MARENLMF/CP (location manager file)
 - SYSOUT file if MARENCP was started in diagnostic modeMARENUCP can be blocked if the following files reside on the home pubset:
 - MARENLOG (logging file)
 - MARENLMF/CP (location manager file)
 - SYSOUT file (all messages connected with the console application are logged here)The console application of MARENUCP (`$MARENUCP`) should not use more routing codes than those actually required (A, E, G, P, T).
- ROBAR-CL
If tape swapping occurs during the online save, the save via ROBAR-CL can be blocked. To prevent this blockade, the following files must be moved to another pubset:
 - SYSOUT file (all messages connected with the console application are logged here)
 - Trace file (if the ROBAR trace was enabled at the start for ROBAR-CL)The console application of ROBAR-CL should not use more routing codes than those actually required (A, E, G, N, T, U).
- Databases
In the case of databases (SESAM) the SYSOUT and logging files should not reside on the home pubset.

Restrictions for the FDDRL run

The following settings exist for the FDDRL run when an online save takes place:

- The online save can only be performed with subtasks (FDDRL parameter PROCESS-JOBS=*BY-SUBTASKS).
- All disks of the pubset must be saved at the same time. To permit this, the FDDRL parameter TASK-LIMIT and the NUMBER-OF-DISK-SETS operand must have the appropriate settings:
 - In the case of DUMP-PUBSET with SAVE-ENTITY=*SINGLE-DISK and COPY-PUBSET the TASK-LIMIT must be the same as the number of the home pubset's disks.
 - In the case of DUMP-PUBSET with SAVE-ENTITY=*DISK-SET the NUMBER-OF-DISK-SETS and TASK-LIMIT are calculated as follows:
 - > Divide the number of disks by 4 and round up the result.

This results in, for instance:

- at least 1 disk set in the case of 1 to 4 disks
- at least 2 disk sets in the case 5 to 8 disks

At startup all subtasks wait at a synchronization point for the simultaneous setting of the write lock (HOLD-IO) for all disks of the pubset. To ensure that all subtasks reach this synchronization point, each subtask must be started and receive the necessary resources. The following requirements must be met to permit this:

- Every subtask can be started. No restrictions may exist (e.g. through the job class limit).
- With //DUMP-PUBSET a tape device is available to each subtask and a suitable volume can be mounted.

Executing the online save

When CREATE-JOBS is used, FDDRL initially checks whether all the requirements for executing the online save have been satisfied.

When START-JOBS is used the main task starts all the subtasks required. In addition, FDDRL starts its own monitoring task (Online Save Control Task = OLSC task) with the job name \$FBDOLSC, and this monitors the time limit.

Each subtask requests the resources (disk devices, tape devices and initial volumes). If the home pubset is set for DEFAULT-PUBSET (default setting), the SYSOUT log is disabled. The subtask then waits for the write lock to be set by the main task.

The main task monitors the startup of the subtasks:

- If a subtask does not reach the synchronization point for setting the write lock, the run is aborted (e.g. insufficient resources).
- When all the subtasks have reached the synchronization point, the main task sets the write lock for all disks of the home pubset.

Each subtask begins by performing the save. After a disk has been saved, the subtask cancels the write lock for precisely this disk.

During the save the main task monitors the progress of the save operations for the subtasks and ensures that the time limit is observed. The main task executes no outputs while the write lock applies. Nor is the status log created during the online save.

It is not possible to interrupt the main task with K2 (BREAK). When the main task is aborted using the /CANCEL-JOB command, FDDRL cancels the write lock and terminates all subtasks which are still running.

The monitoring task also ensures that the time limit is observed. When the monitoring task is aborted, the write lock is canceled. However, the save jobs continue to run.

As soon as all save jobs have been completed, any write lock which still exists for the disks of the pubset is canceled. The subtask for monitoring the time limit is terminated.

Handling blockades

An FDDRL task is blocked when an unexpected event (e.g. program error with system dump) triggers a write I/O on the home pubset during the write lock. The following measures cause FDDRL to recognize a blockade and to cancel the write lock for the home pubset:

- The FDDRL main task monitors the progress of the save operations for the subtasks. If no progress can be detected for one minute, the write lock is canceled prematurely.
- The FDDRL main task and the monitoring task monitor compliance with the time limit specified for the save (TIME-LIMIT operand, default: 5 minutes). The write lock is canceled at the latest when the time limit is reached. However, the online save is continued to the end so that the user can ascertain the scope and duration of the save. The save itself is, however, inconsistent and unusable.

Restrictions

During the write lock all tasks throughout the system which execute write operations on the home pubset are placed in a wait state.

Assumed read operations in a file also trigger a write operation in the catalog when Close is called. Consequently these tasks, too, must wait for the end of the write lock.

It is not possible to start a subsystem during the write lock.

The high I/O load caused by FDDRL obstructs all read accesses to the home pubset.

Administration of the POSIX file systems is not possible during an online save. Write accesses to POSIX file systems are suspended and only executed after the write lock has been reset.

Prerequisites for checking the online save

Before starting the online save, you can use the validation procedure to check whether all requirements for performing the online save are satisfied. The validation procedure is available as an S procedure in two format:

- ***LIBRARY-ELEMENT(SYSPRC.FDDRL.210,CHECK-OLS,J)**
S procedure in text format. The predefined procedure can be edited and customized using an editor. The chargeable subsystem SDF-P is required to execute the validation procedure.
- ***LIBRARY-ELEMENT(SYSPRC.FDDRL.210,CHECK-OLS,SYJ)**
Object procedure which is created from the original text format using COMPILE-PRODEDURE. SDF-P is not required to execute the procedure. No customization is possible with this format.

You call the validation procedure with the CALL-PROCEDURE command. You can specify the following parameters in the PROCEDURE-PARAMETERS=*PARAMETERS(...) operand:

Parameter der Prüfprozedur CHECK-OLS

SYSOUT = *PRIMARY / <filename 1..54 without-gen-vers>

,INFORMATION = MIN / MED / MAX

Description of the procedure parameters

SYSOUT = *PRIMARY / <filename 1..54 without-gen-vers>

Determines the output target for the SYSOUT outputs of the procedure run.

INFORMATION = MIN / MED / MAX

Controls the scope of the information output.

INFORMATION = MIN

The checks that have been performed and the results are displayed.

INFORMATION = MED

Output takes place as with INFORMATION=MIN. The following information is also listed:

- All tasks which occupy the pubset
- All open files on the pubset

INFORMATION = MAX

Output takes place as with INFORMATION=MED. The relevant commands and their results are also displayed. This output is used for diagnostic purposes.

Saving Clones (SHC-OSD)

The software product SHC-OSD (Storage Host Component) offers various clone functionalities. More information can be found in the chapter "Local replication with clones" in the manual „SHC-OSD“ [14]

With SHC-OSD it is possible to create consistent clone units of a pubset during operation. Suspending all I/Os on all affected units of a pubset during the split ensures data consistency across all disks in the pubset. This can then be used, among other things, to create an FDDRL backup of the home pubset without interrupting operations. In addition, the clone units are supplied with the current F5 label, so that FDDRL only has to save the allocated areas of the disk during the backup.

The procedure for creating the clone units differs depending on the replication function:

QuickOPC-Clone on ETERNUS DX/AF

You can find more information on this in the QuickOPC (ETERNUS DX / AF) chapter in the “SHC-OSD” [14].

At certain times, QuickOPC creates clone units from the original units with the commands:

```
/START-CLONE-SESSION UNIT=*BY-PUBSET(PUBSET=<catid>,NEW-PUBSET=*SAME / *SPECIAL-VSN  
/ <catid>, HOLD-IO=*UNTIL-ACTIVATED, WAIT=*UNTIL-SYNCHRONIZATION
```

or

```
/RESTART-CLONE-SESSION UNIT=*BY-PUBSET(PUBSET=<catid>,NEW-PUBSET=*SAME / *SPECIAL-  
VSN / <catid>, HOLD-IO=*UNTIL-ACTIVATED), WAIT=*UNTIL-SYNCHRONIZATION
```

Before the clone can be used (by FDDRL), it must be in the `SPLIT` status.

The parameter `WAIT=*UNTIL-SYNCHRONIZATION` waits for the end of synchronization and simplifies the processing of procedures if an FDDRL backup is then carried out.

EC-Clone on ETERNUS DX/AF

More information can be found in the chapter "Equivalent Copy (EC, ETERNUS DX / AF)" in the manual "SHC-OSD" [14].

By executing the `/START-CLONE-SESSION` command, the synchronization of the clone pair is started implicitly. After the initial synchronization has ended, the pair is in the `SYNCHRONIZED` status.

In this status, the clone units can be separated using the `/ACTIVATE-CLONE` command:

```
/ACTIVATE-CLONE UNIT=*BY-PUBSET (PUBSET=<catid>, NEW-PUBSET=*SAME / *SPECIAL-VSN /  
<catid>, HOLD-IO=*UNTIL-ACTIVATED
```

After the clone pair has been activated, it enters the `SPLIT` state.

Parameter `NEW-PUBSET=*SAME / *SPECIAL-VSN / <catid>`

The `NEW-PUBSET` parameter of the `/START-CLONE-SESSION`, `/RESTART-CLONE-SESSION` and `/ACTIVATE-CLONE` statements can be used to determine which VSNs are assigned to the cloned disks.

FDDRL backup of clone units with `SPECIAL-VSN (NEW-PUBSET=*SPECIAL-VSN)`

For an FDDRL backup, it is recommended to use the parameter `NEW-PUBSET=*SPECIAL-VSN`. This gives the clone units special VSNs. The existence of disks with identical VSNs is avoided.

With the FDDRL statements `DUMP-DISK` and `DUMP-PUBSET` and the operand `SPECIAL-VSN=*YES`, FDDRL requests these disks renamed with special VSNs for processing. The operand `SPECIAL-VSN=*YES` is used for disk allocation and indicates that these disks are additional disk units specially renamed by SHC-OSD. The disks are backed up as if they had their original VSN. They can be restored under the original VSN.

FDDRL backup of clone units with renaming (`NEW-PUBSET=<catid>`)

Renaming the pubset with the parameter `NEW-PUBSET=<catid>` for FDDRL backup is not recommended. Only SF pubsets can be renamed explicitly; But even there, dealing with renamed pubsets creates additional work when managing the backups and restoring (PVSREN). After the explicit renaming, FDDRL saves the pubset with a new catalog ID. It should be noted that the old catalog ID is used in files of applications (e.g. procedures).

FDDRL backup of clone units without renaming (`NEW-PUBSET=*SAME`)

A separation without renaming creates disks with an identical VSN. On the one hand, this means that with the FDDRL backup, the mounting request must be answered individually for each disk. On the other hand, even after a crash during the separation, the operator has to distinguish the disks with double VSN during `IMPORT-PUBSET` processing.

Write caching with DAB must be ended before the disconnection so that all data is written to the disks.

! Parameter HOLD-ID

The `HOLD-IO=*UNTIL-ACTIVATED` parameter is important so that all clone units in a pubset have a consistent status.

The `WAIT=*UNTIL-SYNCHRONIZATION` parameter waits for the synchronization to be completed and thus simplifies the processing of procedures if an FDDRL backup is then performed.

Mirroring with DRV

DRV (Dual Recording by Volume) is a method of recording in which the BS2000 input/output system maintains its data on two disks. The DRV procedure is also referred to as dual mode.

The interfaces for DRV are provided by the software product DRV zur Verfügung. For further details, please refer to the “DRV” manual [3].

On the other hand, with the SRV procedure (Single Recording by Volume) input and output of data can only be carried out on a single disk. As a result, the SRV procedure is also referred to as mono mode.

FDDRL always occupies disks using SPECIAL occupancy. As a result, disks are not saved and restored as DRV disk pairs, but rather always in the form of single disks. The DRV parameters as set in the Standard Volume Label are also saved and restored.

DRV disks can be saved using the following methods:

1. Saving without DRV operation

DRV disks are usually saved in the same way as SRV:

- > Export pubsets.
Take private disks out of operation using `/SET-DISK-PARAMETER ASSIGN- TIME=*USER.`
- > Then only save one disk of each disk pair.
- > After the save has been completed, return DRV disks to live operation, dual mode is then possible again.

2. Saving consistent disk statuses with a short period of interrupted operation

The applications of a DRV disk are only interrupted for a short period of time. One of the disks continues to run in mono mode, while the other disk of the pair is saved.

- > Take the DRV disk pair out of operation.
- > Deactivate dual mode (`/STOP-DRV-DUAL-MODE`).
- > Put the disk or pubset into mono mode operation.
- > Save the split disks; You can rename pubsets beforehand using PVSREN.
- > After the save is completed, reactivate dual mode again (`/START-DRV-DUAL-MODE`). This automatically starts a DRV restoration.

3. Splitting and saving disks during live operation

It is possible to split disks during live operation. However, when you do this, a crash-consistent data status is only created for private disks and single disk pubsets. For pubsets with more than one disk it is not sensible to split disks during live operation, since the inconsistency of the F5 labels that may result from saving the disk may require the entire disk to be saved, including the areas that have not been allocated.

- > After saving, use `/START-DRV-DUAL-MODE` to start dual mode again. A DRV restoration of the disk will be necessary as a result.

Automating FDDRL saves

Through automatic recognition of volumes, FDDRL saves can be performed such that the operator no longer has to respond to mount messages. Instead, an informative message is simply output on the console and processing is resumed.

With **disks** the following steps must be taken:

- Mount the disks before your request
- Specify the VSN of the output disks in the `OVERWRITE-DISK` operand of the `COPY-DISK`, `COPY-PUBSET` or `RELOAD-DISK`, `RELOAD-PUBSET` statement. (The user must specify the VSN of the input disk anyway.)

With **MTC devices** the following steps must be taken:

- In the case of MTC devices with an automatic mounting aid, either a MAREN scratch pool must be available and `TAPE=*ANY` be specified for the output tapes or the VSNs must be specified explicitly in the `TAPE` operand.
- In the case of MTC devices without an automatic mounting aid, the tapes/cartridges must have been mounted beforehand, and their VSNs must be specified in the `TAPE` operand.

Before MTC devices are assigned, they are reserved using a `/SECURE-RESOURCE-ALLOCATION` command. You can control the wait time for reservation using the FDDRL parameter `SECURE-WAIT-TIME`.

Specifying a wait time of more than 3 minutes prevents an operator dialog. A wait time which is sufficiently long provides the opportunity for a MTC device to be freed up in the meantime, thereby preventing the FDDRL job from being aborted due to no free MTC device being available.

FDDRL requests, FDDRL jobs, FDDRL subtasks

- [FDDRL request](#)
- [FDDRL job](#)
- [FDDRL subtasks](#)

FDDRL request

The term FDDRL request was introduced with FDDRL V19.0A.

FDDRL requests are created by executing the FDDRL statements for the FDDRL functions (COPY, DUMP, RELOAD).

One request is created per statement and object name (pubset, disk or disk set). A request can have three statuses: accepted, started or completed. Intermediate statuses are not managed.

A request is defined by the following parameters:

REQUEST-NAME	A request name comprises the '3-character alias name (ALIAS-NAME) of the statement and the object name, e.g. DMP-ABCD, DMD-ABCD . 0, DMD#WORK01
REQUEST-DATE REQUEST-TIME	Date and time the request was accepted (entry of the START-JOBS statement)
STATUS	ACCEPTED, STARTED, COMPLETED
SUB-STATUS	This is only defined for the COMPLETED status: OK or WITH-ERRORS
REPORT-FILE	Name of the report file (for the COMPLETED status only).
TSN	TSN of the main task; used internally to check aborted runs.
TERM-DATE/ TERM-TIME	Termination time (end of an FDDRL request) The duration of a request can be determined from the difference between the termination time and the start time.

You can display running and completed FDDRL requests using the `//SHOW-REQUESTS` statement. Here you can filter the requests which are to be displayed according to attributes and also define the required output medium (SYSOUT, SYSLST and/or S variable).

When FDDRL is running on a Server Unit of an SE server, the requests and your status changes are reported to the SE Manager.

When an FDDRL request ends, a report file is created by default (`REPORT-FILE=*STD`). When it is running on a Server Unit of an SE server, by default (`MONITORING=*STD`) a report file in PDF format is also created which can be displayed in the SE Manager.

Request file

The FDDRL requests are stored in a system-wide request file with the name `SYSDAT.FDDRL.210.REQUESTS`. If the file does not yet exist when a request is created, it is created automatically. The request file is specific to the FDDRL version. When you change the FDDRL version, it is not envisaged that the requests from other FDDRL versions should be taken over.

By default, in each FDDRL run all FDDRL requests which are older than 40 days are deleted from the request file. When required, you can adjust this age limit using the `KEEP-REQUESTS` parameter in the FDDRL parameter file.

However, the `//DELETE-REQUESTS` statement also enables you to delete **completed** requests manually from the request file.

FDDRL job

An FDDRL job describes the processing parameters for each save entity. FDDRL jobs are defined from an FDDRL request. One FDDRL job is created per disk and DISK-SET.

FDDRL jobs are also created by the `//CREATE-JOBS` statement. The created jobs are output to SYSOUT or SYSLST in the "REQUESTED FDDRL JOBS" table (see [chapter "Output of the FDDRL jobs"](#)).

The `//START-JOBS` statement enters the FDDRL requests in the request file with the ACCEPTED status. It then starts the FDDRL jobs, setting the status of the requests to STARTED.

Displaying the job status

If the FDDRL jobs are executed under subtasks, the status of the own jobs can be displayed under the main task. The `STATUS-OUTPUT-PERIOD` parameter can be used to define an output interval for this purpose.

The FDDRL statement `//SHOW-FDDRL-STATUS` enables foreign FDDRL jobs (in the same system) to be displayed. In comparison to `//SHOW-REQUESTS`, the current processing progress of each FDDRL job at disk level can also be displayed here.

This also allows the system administration to monitor FDDRL jobs which were started in batch mode or by other FDDRL users.

`//SHOW-FDDRL-STATUS` only displays requests which are running. Completed requests are displayed using the `//SHOW-REQUESTS` statement.

FDDRL subtasks

If the FDDRL parameter `PROCESS-JOBS=*BY-SUBTASKS` is set (default), the FDDRL jobs are executed under FDDRL subtasks. The subtasks for executing the FDDRL jobs are started by the `//START-JOBS` statement.

You can specify the degree of parallelism using the FDDRL parameter `TASK-LIMIT`.

A separate FDDRL subtask is generated for each save entity (single disk or disk set). The subtasks process the jobs of their save entity:

- With the single disk save entity, precisely one disk is processed.
- With the disk set save entity, each subtask processes multiple disks; the jobs for disk processing are assigned dynamically to each subtask.

The FDDRL subtasks have the following attributes:

- FDDRL subtasks are treated like batch tasks in BS2000, but without `SYSDTA` and `SYSCMD` environments.
- FDDRL subtasks receive the same user ID (`TSOS`) and account number as the FDDRL main task.
- FDDRL subtasks inherit the password table of the main task. The passwords of tapes can be specified under the main task (`/ADD-PASSWORD`).
- The job class of the FDDRL subtask is the same as that of the FDDRL main task. If this job class is not permitted for batch tasks, the FDDRL subtask is started under the default job class for batch tasks.
- The task category of the FDDRL subtask is the same as that of the FDDRL main task. If this task category is not permitted for batch tasks, the default category of the job class is used.
- The job name of the FDDRL subtask is controlled by the value of the FDDRL Parameter `JOB-NAME`:
 - By default (`JOB-NAME=*STD`), the job name of the FDDRL subtask is generated either from the VSN of the disk or from the name of the disk set, depending on the save entity:
A VSN is preceded by the "\$" character. Each illegal character in the job name, such as a period "." or hyphen "-", is replaced by the "#" character.

Exception

If the job name of the main task is called "MARENTST", the subtask is also given this job name (support for MAREN test mode).

Examples

For the disk `ABC.01` the job name `$ABC#01` is created. You can display the job status for this FDDRL subtask using the following command: `/SHOW-JOB-STATUS JOB-NAME=$ABC#01`.

The pubset FDL is restored using `SAVE-ENTITY=*DISK-SET`. You can display the job status for this FDDRL subtask using the following command: `/SHOW-JOB-STATUS JOB-NAME=FDL#1`.

- If specified `JOB-NAME=*SAME` the FDDRL subtask get the same name as the maintask.
 - If a jobname is specified (`JOB-NAME=<name 1..8>`), the FDDRL subtask gets this name
- The FDDRL subtask receives the task priority that was specified with the `RUN-PRIORITY` parameter. The permitted limit values of the job class are automatically respected.
 - If you have specified the behavior of FDDRL V18.0 with the `REPORT-FILE=*NONE` operand in the `//MODIFY-FDDRL-PARAMETERS` statement, the outputs of the FDDRL subtasks and their messages to console are logged to `SYSOUT`. After subtask start, `SYSOUT` is assigned to the following file:

-
- S.FBD.<stsn>.<yyyy-mm-dd>.<mtsn>.<vsn> (single disk save entity)
 - S.FBD.<stsn>.<yyyy-mm-dd>.<mtsn>.<dset> (disk set save entity)

with the following meanings:

- <stsn> is the task sequence number (TSN) of the subtask
- <mtsn> is the task sequence number (TSN) of the main task
- <vsn> is the volume serial number of the disk
- <dset> disk set name

- If the FDDRL parameter `REPORT-FILE=*STD` is set, the outputs of the subtasks and their messages on the console are transferred to the log file of the FDDRL job at the end, the original SYSOUT files of the subtasks are then deleted.
- The FDDRL subtasks are monitored by the FDDRL main task during their execution. Their status can be output periodically in accordance with the `STATUS-OUTPUT-PERIOD` parameter (see [chapter "Status output on the FDDRL subtasks"](#)).
- If FDDRL processing is performed under subtasks, disk and tape devices may not be reserved by the user under the main task with the `/SECURE-RESOURCE-ALLOCATION` command since this reservation cannot be passed on to the subtasks. Devices reserved in this way would be blocked for the subtasks.

Adjusting the number of FDDRL subtasks

Precisely one FDDRL subtask is generated per save entity (single disk or disk set). The FDDRL parameter `TASK-LIMIT` specifies how many of these FDDRL subtasks are to be active simultaneously. (Additionally, the number of parallel FDDRL subtasks can be limited by the job class limit for all batch jobs of their class)

By default, the task limit is set to four FDDRL subtasks. However, you can adjust this value to your requirements within the range 1 through 16 using the `TASK-LIMIT` parameter of the `//MODIFY-FDDRL-PARAMETERS` statement. You may need to adjust the value for the following reasons:

- To control system utilization – especially with respect to input/output.
- The number of parallel save runs is adjusted to match the available tape devices. Tape devices must therefore remain free so that
 - they can be used for premounting backup volumes (see [chapter "Automating FDDRL saves"](#)).
 - any volume initializations that become necessary can be carried out.
 - they can be “reserved” for other applications.
- To permit multiple disk sets to be saved to their own tape sets in parallel when the disk set save entity is used, the task limit must at least be equal to the number of disk sets.

Central request monitoring on the SE server

The **BS2000 Backup Monitor** is integrated as an SE Management application in the **Applications** main menu of the **SE Manager**. The BS2000 Backup Monitor provides information on the status of the save requests which were requested in the BS2000 systems of the SE server with the software products HSMS and FDDRL. For completed requests the BS2000 Backup Monitor provides an execution log in PDF format. The execution log is created independently of the logging specification when logging is requested in the BS2000 system.

The following requests are monitored:

- Backup
- Restoring data
- Migration (HSMS only)
- Archival (HSMS only)
- Copying backup files, including reorganization runs of archives (HSMS only)
- Exporting and importing files (HSMS only)

The display of a BS2000 system's FDDRL requests is controlled using the FDDRL parameter `MONITORING`. By default `MONITORING=*NO` is set, i.e. monitoring is disabled. To switch the monitoring on permanently, the parameter in the parameter file must be set to `MONITORING=*STD`.

Data compression

The aim of data compression is to improve utilization of the capacity of the volumes and consequently to reduce the number of MTC required for a saving data. FDDRL recognizes two types of data compression:

- **Hardware compression**
Hardware compression is performed by the MTC device (i.e. the device controller compresses the data and optimizes blocking). It is performed automatically when the corresponding volume type is used (e.g. TAPE-C4). The data saved in this way can only be read on the corresponding MTC devices.
- **Software compression**
Software compression was offered by FDDRL up to V15.0A if hardware compression was not possible (but had to be specified with the operand `DATA-COMPRESSION=*YES`). Since state-of-the-art MTC devices work with hardware compression by default, software compression is becoming increasingly less important for the MTC devices in use and is no longer offered in FDDRL V16.0A and higher. If this compression type is encountered during restoration, FDDRL automatically decompresses the data.

Irrespective of the type of compression, backup volumes can be duplicated without restrictions by means of PERCON (see [chapter "Duplicating FDDRL backup volumes with PERCON"](#) and the "PERCON" manual [11]).

Data encryption

To ensure that data can be protected against unauthorized access when volumes are transported, it can be encrypted when it is saved on LTO4 drives.

Encryption is requested using the volume type:

- With TAPE-U2, TAPE-U3, TAPE-U4, TAPE-U5, TAPE-U6, TAPE-U7, TAPE-U8 the data is saved in **unencrypted** form.
- With TAPE-U4E, TAPE-U5E, TAPE-U6E, TAPE-U7E the data is saved in **encrypted** form.

High-performance encryption and decryption is performed by the MTC device.

The functions for using volume encryption in BS2000 are provided by the optional subsystem MAREN V12.0 and higher.

FDDRL output

Information on the FDDRL jobs and FDDRL subtasks is output.

Table of Contents

- [Output of the FDDRL jobs \(FDDRL V21.0A 2021-06, #37\)](#)
- [Status output on the FDDRL subtasks \(FDDRL V21.0A 2021-06, #38\)](#)
- [Results log of the FDDRL jobs \(FDDRL V21.0A 2021-06, #39\)](#)
- [Tape information and information on disk backup \(FDDRL V21.0A 2021-06, #40\)](#)
- [Job variable for backup volumes \(FDDRL V21.0A 2021-06, #41\)](#)
- [Messages \(FDDRL V21.0A 2021-06, #42\)](#)

Output of the FDDRL jobs

After the //CREATE-JOBS statement has been issued and after FDDRL has ascertained all the disks of a pubset or the disks of a disk set, a list of all FDDRL jobs is output. As a result, users can check and, if required, correct their jobs before they are actually executed.

Example

```
REQUESTED FDDRL JOBS          FDDRL V21.0A00          2020-06-25 13:46:54
STATEMENT                      REQ-NAME
DISK      DEV-TYPE I-DEV >O-DEV O-DISK FORM      DISK-DATE
DISK-SET TAPE-DEV LOCATION (TAPES)
-----
DUMP-PUBSET          SBZ3          DMP-SBZ3
SBZ3.0   D3435   EE1A                      2020-06-25 13:42:30
SBZ3.1   D3435   EE1B                      2020-06-25 13:42:30
SBZ3.2   D3435   EE1C                      2020-06-25 13:42:30
SBZ3-1   TAPE-C4
```

Descriptions of the fields

STATEMENT FDDRL statement with catid or VSN or MN of the (first) disk

DISK VSN of the disk

DEV-TYPE Device type of the disk

I-DEV Mnemonic device name (MN) of the input disk

O-DEV Mnemonic device name (MN) of the output disk

O-DISK OVERWRITE-DISK operand, VSN of the output disk

FORM Indicates whether the output disk is being formatted (YES/NO)

DISK-DATE Time stamp:

- of the input disks (DUMP-PUBSET, COPY-PUBSET)
- of the original disk (RELOAD-PUBSET)

In the case of RELOAD-DISK, blanks are displayed since the value is not yet known at this point.

DISK-SET Name of the disk set

TAPE-DEV Volume type of MTC

LOCATION Location of MTC

TAPES VSN of the backup volumes (MTC)

Status output on the FDDRL subtasks

While the FDDRL jobs are handled by FDDRL subtasks, the FDDRL main task outputs information on its FDDRL subtasks after a specified interval (value of the STATUS-OUTPUT-PERIOD parameter). This allows the user to follow the progress.

Example

```
ACTUAL FDDRL JOB STATUS          FDDRL V21.0A00      2020-09-16 10:51:06
STATEMENT
  DISK/-SET  TSN  FORM  JOB-STA      SEL-HPS  PROCESS-HPS    %
  -----  -
DUMP-PUBSET          SBZ3
SBZ3.0      04HT    ACTIVE    1037840    388000    37%
SBZ3.1      04HT    ACTIVE    1030960    387920    37%
SBZ3.2      04HT    ACTIVE     983600    387920    39%
SBZ3-1      04HT    ACTIVE                    1163840
SUMMARY: INACTIVE = 0    ACTIVE = 4    TERMINATED = 0    TOTAL = 4
-----
```

Descriptions of the fields

STATEMENT	FDDRL statement with catid or VSN or MN of the (first) disk
DISK/-SET	VSN of the disk or name of the disk set
TSN	Task sequence number of the FDDRL subtask
FORM	TSN of the batch job which formats the output disk
JOB-STA	Status of the FDDRL subtasks ACTIVE: FDDRL subtask is running INACT.: FDDRL subtask has not yet been started Once they have been terminated, the FDDRL subtasks are listed only in the summary field.
SEL-HPS	Number of disk blocks to be transferred (HP: physical half-page, 2KB)
PROCESS-HPS	Anzahl der bereits verarbeiteten Plattenblöcke (HPs)
%	Number of disk blocks already processed (HPs) Processing progress in percent

Results log of the FDDRL jobs

When processing is completed, a log of all FDDRL jobs is output. This displays the used tape cartridges and output disks and provides a final report on the correctness and completeness of the processing. The tapes and output disks used are displayed.

Example

```

FINAL FDDRL JOB PROTOCOL          FDDRL V21.0A00          2020-09-16 10:52:36
STATEMENT                          REQ-NAME  REQ-DATE  REQ-TIME
DISK  DEV-TYPE  IN  >  OUT          DISK-DATE  TSN      HPS      TERM-MSG
D-SET  F-NAME                                TSN      HPS      TERM-MSG
      TAPE-DEV (TAPES)
-----
DUMP-PUBSET          SBZ3          DMP-SBZ3  2020-09-16 10:50:06
SBZ3.0 D3435      EE1A >SBZ3-1 2020-09-16 10:46:45      1037840
SBZ3.1 D3435      EE1B >SBZ3-1 2020-09-16 10:46:45      1030960
SBZ3.2 D3435      EE1C >SBZ3-1 2020-09-16 10:46:45      983600
SBZ3-1 FDDRL.P.SBZ3.2020-09-16.105000-1          04HT  3052400
      TAPE-C4 (SW0111,SW0112,SW0114)
-----

```

Descriptions of the fields

REQ-NAME	Request name
REQ-DATE	Start date of the request
REQ-TIME	Start time of the request
STATEMENT	FDDRL statement with catid or VSN or MN of the (first) disk
DISK	VSN of the disk
DEV-TYPE	Device type of the disk
IN	Mnemonic device name (MN) of the input disk
OUT	Mnemonic device name (MN) of the output disk or reference to the disk set (name of the disk set)
DISK-DATE	Time stamp of the disk or NONE With disks of a pubset, an identical time stamp guarantees a consistent save or restoration.
TSN	Task sequence number of the task which has processed the job
HPS	If the FDDRL job was terminated successfully, the number of transferred disk blocks is output (HP: physical half-page, 2K).
TERM-MSG	If the FDDRL job was terminated abnormally, the termination code is output instead of the number of HPs.
D-SET	Name of the disk set
F-NAME	File name in the MAREN catalog
TAPE-DEV	Volume type of the MTC
TAPES	VSN of the backup volumes (MTC) (maximum 255 or 512). The backup MTC are only output after a successful run.

Output to S variable

Output information	Name of the S variable	T	Contents
Output identification	var(*list).ID	S	FBDFFJP
Alias of the statement	var(*list).STMT-NAME	S	<c-string 3..3>
Object type	var(*list).OBJECT-TYPE	S	PUBSET DISK UNIT
Object name	var(*list).OBJECT-NAME	S	<catid> <vsn> (<vsn>,...) <mn> (<mn>,...)
Request name	var(*list).REQ-NAME	S	<c-string 5..10>
Device type of the tapes	var(*list).TAPE-DEV-TYPE	S	<c-string 2..8> "
Location of the tapes	var(*list).LOCATION	S	*STD <c-string 1..8> "
Expiration date	var(*list).RETENT-PERIOD	S	<0..32767 days>
FDDRL tape format or " if no tape is used	var(*list).TAPE-FORM	S	*V17-FORMAT *V16-FORMAT *V15-FORMAT "
Name of the report file	var(*list).REPORT-FILE	S	<string 43..54> *NONE
Number of disks (1..n)	var(*list).NUM-OF-DISKS	S	<c-string 1..4>
Number of disk sets (0..9) '0': no disk set	var(*list).NUM-OF-DISK-SETS	S	<c-string 1..1>
Termination message in the event of errors	var(*list).TERM-MSG	S	<c-string 7..7> "
VSN of the disk	var(*list).DISK(*list).DISK	S	<c-string 1..6>
Device type of the disk	var(*list).DISK(*list).DEV-TYPE	S	<c-string 1..8>
<mn> of the input disk in the case of COPY /DUMP	var(*list).DISK(*list).INPUT-UNIT	S	<c-string 2..4>
	var(*list).DISK(*list).OUTPUT-UNIT	S	<c-string 2..4>

<mn> of the output disk in the case of COPY /RELOAD			
Time stamp of the disk (date)	var(*list).DISK(*list).DATE	S	<yyyy-mm-dd>
Time stamp of the disk (time)	var(*list).DISK(*list).TIME	S	<hh:mm:ss>
TSN of the processing task (main task / subtask)	var(*list).DISK(*list).TSN	S	<c-string 1..4>
Disk set with which the disk was saved	var(*list).DISK(*list).DISK-SET	S	<c-string 3..6> "
Processed HPs	var(*list).DISK(*list).PROCESS-HPS	S	<c-string 0..10> "
Save scope of the disk	var(*list).DISK(*list).SEL-BLOCKS	S	*USED-BLOCKS *ALL-BLOCKS
MAREN file name	var(*list).DISK(*list).F-NAME	S	<c-string 1..41> "
Number of MTCs	var(*list).DISK(*list).NUM-OF-TAPES	S	<c-string 1..2> 0
VSN list of MTCs	var(*list).DISK(*list).TAPE(*list).	S	<c-string 1..6>
Termination message in the event of errors	var(*list).DISK(*list).TERM-MSG	S	<c-string 7..7> "
Name of the disk set	var(*list).DISK-SET(*list).DISK-SET	S	<string 3..6> "
VSN list of the disks	var(*list).DISK-SET(*list). DISK(*list)	S	list of <c-string 1..6>
TSN of the processing task (main task / subtask)	var(*list).DISK-SET(*list).TSN	S	<c-string 1..4> "
Processed HPs	var(*list).DISK-SET(*list). PROCESS-HPS	S	<c-string 1..10> "
MAREN file name	var(*list).DISK-SET(*list).F-NAME	S	<c-string 1..41>
Number of MTCs	var(*list).DISK-SET(*list).NUM-OF-TAPES	S	<c-string 1..2>
VSN list of MTCs	var(*list).DISK-SET(*list).TAPE(*list)	S	list of <c-string 1..6> "
Termination message in the event of errors	var(*list).DISK-SET(*list).TERM-MSG	S	<c-string 7..7> "

Tape information and information on disk backup

The tape information "FDDRL TAPE INFORMATION" displays not only the properties of the tape but also the disk properties of a single disk backup or of all disks of a DISK-SET save.

This information is automatically output to SYSOUT when a tape is rejected during mount processing .

You can also request it explicitly with the `//SHOW-FDDRL-TAPE-INFORMATION` statement.

This means that you can obtain information on the tape save and can mount the disks required for RELOAD-DISK or RELOAD-PUBSET (device type, format, capacity).

With DUMP-DISK detailed information is output explaining why the tape was rejected.

Output examples and the meaning of the information output can be found in [chapter "SHOW-FDDRL-TAPE-INFORMATION Show the tape information of an FDDRL tape"](#)

Job variable for backup volumes

During the save, FDDRL stores the VSNs of the backup volumes in a job variable. This job variable can then be used to specify the volumes of the last backup when restoration takes place.

You can use this function if you create a job variable with a particular name before the save run. You can also utilize this function without using MAREN.

With one exception, the name of the job variable corresponds to the MAREN file name:

- In the `//DUMP-DISK` statement you specify the MAREN file name explicitly in the `FILE-NAME` operand, or when `FILE-NAME=*STD` is specified FDDRL forms a standard file name.
- In the case of the `//DUMP-PUBSET` statement, FDDRL forms a standard file name.

If standard names are assigned for the save (see also the [chapter "MAREN tape set"](#)), FDDRL expects the job variable under the following name:

- For the single disk save entity
 - `FDDRL.D.<vsn_1..6>` in `//DUMP-DISK`
 - `FDDRL.P.<catid_1..4>.<vsn_6..6>` in the case of `//DUMP-PUBSET`
- For the disk set save entity
 - `FDDRL.D.<vsn_1..6>-1` with the first VSN specified in `//DUMP-DISK`
 - `FDDRL.P.<catid_1..4>-<disk-set-no>` in `//DUMP-PUBSET`
As an exception to the rule, here the name is not the same as the MAREN file name because the date and time of the backup are not known when the job variable is created.

At the start of the backup, FDDRL deleted the job variable value. The first character in the job variable is “#” if the backup has not been completed or is aborted.

Messages

All FDDRL messages begin with a 7-character message code. This message code comprises three characters for the message class and four characters for the error code. The message class for FDDRL is "FBD".

When FDDRL is running in batch mode, messages which require a response from the operator are output at the console.

When FDDRL is running in interactive mode, all messages are directed to SYSOUT.

You can query the message text and optionally also the meaning and response texts for a message code with the `/HELP-MSG-INFORMATION` command.

You can also use an HTML application to find the FDDRL messages on the Manual Server (URL: <http://bs2manuals.ts.fujitsu.com>) and on the "BS2000 SoftBooks" DVD.

Security

With the `//COPY-PUBSET` and `//DUMP-PUBSET` statements, a save run is automatically initiated for each disk of the pubset. The consistency of the disks and their membership of the pubset is guaranteed by the time stamp. The user must make sure that all disks are attached and accessible when processing begins so that all disks of the pubset are saved with a consistent status.

During saving, MTC from the same save run which have already been used are protected against overwriting. If there are several save runs, the MTC can only be protected against overwriting during these runs if `RETENTION-PERIOD > 1` is specified.

If unrecoverable errors occur on the output device during a save run, the save can only be completed correctly on an alternate volume.

In a restoration run (`//RELOAD-DISK`), FDDRL ensures that only MTC with the correct save version of a disk are used for restoration.

In a restoration run (`//RELOAD-DISK` and `//COPY-DISK`, `//COPY-PUBSET` with `ACTION=*RESTORE`), permanent errors on the input device can only be skipped within the context of indispensable restrictions and with loss of the affected data.

Error messages during tape processing may not necessarily be due to faulty tape cartridges. The errors could, for example, be caused by dirty read/write heads on the MTC device. In any event, an increased incidence of errors should certainly be investigated. Only MTC of the highest quality should be used for save runs.

Diagnostic aids

If an error occurs, the following documents should be produced and sent to the technical support staff:

- System dump
- SYSOUT log of the FDDRL main task and the FDDRL subtasks
- REP file
- Input data

Backup volumes (MTC) used in the run which led to the error should be retained until the error has been eliminated, in case they are requested for diagnostic purposes by technical support staff.

Error handling

If errors occur in device operation or device requests and in I/O operations, FDDRL attempts recovery. If this fails, the operator receives a message that must be answered; otherwise FDDRL aborts the FDDRL job.

Messages requiring an operator response

These messages include:

- Errors in device/volume allocation
- Unrecoverable input/output errors

Depending on the situation, FDDRL offers the following response options:

- *Repeat (RETRY)*
e.g. in the case of input/output errors, device INOP
- *Skip (CONTINUE)*
e.g. in the case of disk input errors (parity, block length, undefined) and unreadable user labels, i.e. UTLs (user trailer labels), EOF (end-of-file labels) or EOV (end-of-volume labels)
- *Skip the current error and all future errors (CONTINUE ALWAYS)*
e.g. in the case of disk and tape input errors (parity)
- *Restart (RESTART)*
e.g. in the case of input/output errors which may be due to device problems that might be remedied by using a different device.
For *disks*, the volume is released and requested again; then input/output is repeated. For *MTC*, the volume is requested again, allocated and then rewritten from the beginning.
- *Abort FDDRL jobs (TERMINATE)*
The FDDRL job is aborted. The allocated devices are released and the program terminates. In the log the abortion is indicated by the termination code.

Program abortion

FDDRL aborts the program if, for example, input/output errors are discovered while reading important metadata (labels), or in the case of program errors. How FDDRL behaves when a program is aborted depends on the type of processing:

- Processing without FDDRL subtasks (`PROCESS-JOBS=*BY-MAINTASK` parameter)
Abnormal termination (`/CANCEL-PROGRAM`, `/CANCEL-JOB`) of an FDDRL job occurs immediately, irrespective of other FDDRL jobs.
- Processing with FDDRL subtasks (`PROCESS-JOBS=*BY-SUBTASKS(. . .)` parameter)
 - If an FDDRL subtask is aborted with the command `/CANCEL-JOB`, termination occurs immediately. Neither the FDDRL main task nor all other FDDRL jobs are affected.
 - If the FDDRL subtask which monitors an online save is aborted with the `/CANCEL-JOB` command, the write lock for the home pubset is canceled. However, the save jobs continue to run. The save itself is flagged as inconsistent and cannot be used for any restore operation.
 - If an FDDRL program in the main task is aborted, for example with the command `/CANCEL-PROGRAM`, the abortion is postponed until all currently active FDDRL subtasks have terminated. Inactive FDDRL subtasks, however, are no longer activated.
 - If the FDDRL main task is aborted with `/CANCEL-JOB`, it first aborts all currently active FDDRL subtasks and then terminates itself. In the case of an online save the FDDRL main task also releases the write lock for the home pubset.

Performance

System backups generally involve very large quantities of data. FDDRL offers a number of functions for speeding up the transfer of data and thus reducing the time required for the saves:

- When saving a disk, FDDRL normally transfers only those areas which are marked as occupied in the F5 label of the disk. However, if the saving of the disk was preceded by an abnormally terminated BS2000 session, the whole disk is saved automatically.
- In the case of V15 and V16 tape format (see the TAPE-FORMAT operand in statement, "[MODIFY-FDDRL-PARAMETERS Define execution parameters](#)") ,FDDRL works with large blocks provided the tape devices support this (as of TAPE-C3).
- In order to speed up the disk side to the higher data transfer rate of the tape devices in the case of dump and reload, FDDRL utilizes parallel access to disk devices by means of PAV (Parallel Access Volumes). In the case of V16 tape format with disk sets, FDDRL uses multiplexing by accessing two disks at a time for a tape device. In the case of the V17 tape format with disk sets, FDDRL uses multiplexing by accessing up to four disks at a time for a tape device.

FDDRL can perform its functions under separate tasks (FDDRL subtasks). The `TASK-LIMIT` parameter allows you to preset the degree of parallelism (see also [chapter "FDDRL requests, FDDRL jobs, FDDRL subtasks"](#)). By means of the `RUN-PRIORITY` parameter, you can also specify for each FDDRL subtask the task priority within the job class.

When FDDRL is used, two different cases can be distinguished with respect to throughput and performance:

- When BS2000 is running with a small number of applications, an FDDRL run should be performed as quickly as possible and without regard for the response time behavior.
In this case, you can set the `TASK-LIMIT` parameter up to the limits of the available tape devices in order to achieve a good throughput rate from the high degree of parallelism. With four or more parallel runs, however, noticeably worse response time behavior is to be expected.
You can also set a high `RUN-PRIORITY` (low value).
- During normal BS2000 operation, an FDDRL run should be performed without impairing the response time or the performance.
You must take the other applications into account when setting the `TASK-LIMIT` parameter; above all, sufficient tape devices must be kept free. The `RUN-PRIORITY` parameter must be set low (high value).

To allow efficient control of resource utilization (tape devices, CPU time, task limit) during an FDDRL run as well, you should set up a separate job class for FDDRL.

- The throughput can be optimized if one MTC controller with its own channel is available for every two disks.
- Software compression with the `//DUMP-DISK` and `//DUMP-PUBSET` statements uses a considerable amount of CPU resources. Therefore, if restrictions in response time behavior are not acceptable, you should not use software compression.

Each new tape format promises better performance than its predecessor. The improvement affects both the tape and the disk side.

As far as possible, you should therefore always use the latest tape format (set using the FDDRL parameter `TAPE-FORMAT=*STD`).

Handling data volumes

FDDRL supports all volumes that are also supported by the various versions of BS2000. All volumes to be processed by FDDRL must have valid labels:

- The BS2000 disks must be formatted and initialized according to the valid BS2000 standards, i.e. they must have been prepared with the VOLIN utility routine (see the “Utility Routines” manual [2]). When mnemonic device names are specified, FDDRL automatically ensures that the formatting is correct.
- The MTC must be initialized with the INIT utility routine (see the “Utility Routines” manual [2]) in accordance with DIN 66029.

The syntax of all statements from FDDRL is geared toward the VSNs of the volumes. As a rule, therefore, volumes are assigned via their VSNs. If volumes which have been made known to FDDRL by their VSNs in a statement are already mounted before they are requested for the first time, the BS2000 operating system recognizes the volumes automatically. In this case, only a purely informative message is output, i.e. there is no mount request.

Note on variable-sized disks

Disks of the same type can have been generated with different capacities. These disks can be copied or restored only when the output disk is at least as large as the input disk or original disk. However, this operation means that the usable capacity of the larger disk is reduced to the capacity of the smaller disk.

In a RAID disk subsystem you can use the SPACEOPT utility routine to readjust the capacity to the actual size of the logical volume (see the `/ADAPT-PUBSET-SIZE` command in the “SPACEOPT” manual [15]).

Disks

- Input disk
- Output disk
- Disk formats
- Replacing defective pubset disks

Input disk

The input disk is the disk or one of the disks which FDDRL is to copy or save. In the case of `//COPY-DISK` and `//DUMP-DISK` statements, it is defined by the `DISK` operand, while with the `//COPY-PUBSET` and `//DUMP-PUBSET` statements it is defined by the `PUBSET` operand.

With the `//COPY-DISK`, `//COPY-PUBSET`, `//DUMP-DISK` and `//DUMP-PUBSET` statements you cannot specify the device type of the input disk. By default it is set to the general disk type "STDDISK", unless a clear special device type can already be determined when the statement is entered: with `//COPY-PUBSET` and `//DUMP-PUBSET`, FDDRL attempts to ascertain the special device type from the master catalog or on the basis of the attached devices. If this is not successful, the input disk is requested with the general disk type "STDDISK" and the special device type is registered after the assignment.

The input disk must have a valid BS2000 standard volume label (SVL), otherwise it cannot be saved. The disk must be exclusively available to FDDRL: it must be attached, but its catalog may only be imported if disk copies were split in the event of disk mirroring with SHC-OSD.

FDDRL supports the saving of a complete SM pubset as long as it does not comprise more than 255 individual disks. Any attempt to save an individual volume set is rejected, as restoration of only this volume set would produce an inconsistent SM pubset.

Output disk

The output disk is the disk or one of the disks to which FDDRL copies or restores. It is defined using the `OVERWRITE-DISK` operand in the `//COPY-DISK`, `//COPY-PUBSET`, `//RELOAD-DISK` or `//RELOAD-PUBSET` command.

The output disk must have the same device type and be formatted in the same way as the input disk or the saved original disk. Only if the output disk is specified with its device mnemonic (MN) FDDRL is able to initialize and format it in advance when required. Otherwise FDDRL accepts the output disk only if it is already installed and has the correct format. Disks of type D3475-8F must be configured and formatted in X2000.

Assignment of the output disk is controlled by means of the `OVERWRITE-DISK` operand. The following options exist for this:

- `OVERWRITE-DISK=*SAME-VSN`
The output disk must have the same VSN as the input disk or the disk which was originally saved to tape. With `//COPY-DISK` and `//COPY-PUBSET`, this means that two disks with the same VSN are online. If this is the case, FDDRL checks whether the time stamps of the input and output disks are compatible with the copy direction selected via the `ACTION` operand. For example, in a run with `ACTION=*SAVE`, the input disk must be newer than the output disk (see the `ACTION` operand).
- `OVERWRITE-DISK=<vsn 1...6>`
The VSN of the output disk is expressly specified. It cannot be reassigned by the operator.
- `OVERWRITE-DISK=*UNIT(UNIT=...)`
The output disk is specified by means of its device mnemonic (MN). It must either have the same MN as the input disk (`UNIT=*SAME-UNIT`) or an explicitly specified MN. If necessary, FDDRL employs the `VOLIN` utility routine to initialize and format the disk before it is used. FDDRL ensures that the operations are serialized here.
- `OVERWRITE-DISK=*BY-OPERATOR`
The operator is responsible for assigning the correct VSN.

The operand value `*BY-OPERATOR` should not be used during normal operation, since an operator error may cause foreign disks to be overwritten inadvertently. `*BY-OPERATOR` is intended exclusively for cases where the VSN is not known. This situation can occur if a previous `COPY-DISK`, `COPY-PUBSET`, `RELOAD-DISK` or `RELOAD-PUBSET` run was not terminated correctly. In this case, the SVL still contains a temporary VSN which has been assigned by FDDRL.

Temporary VSN

During restoration, the output disk is assigned a temporary VSN by FDDRL. This consists of the start time (hhmm) and the last two characters of the original VSN. After restoration has been completed, the output disk is assigned the VSN of the input disk or of the saved disk.

If the restoration run is aborted, the output disk retains the temporary VSN. In this way FDDRL prevents the disk which has not been restored completely from being used inadvertently. When the restoration run is repeated, the output disk's temporary VSN, its MN or the value `*BY-OPERATOR` must be specified in the `OVERWRITE-DISK` operand.

Disk formats

FDDRL supports disks with the formats KEY, NK2 and NK4.

FDDRL only ever copies or restores onto a disk which has the same format as the original disk, i.e. from KEY disks only to KEY disks, from NK2 disks only to NK2 disks, and from NK4 disks only to NK4 disks. In other words, FDDRL does not convert one format to another.

Before copying or restoring, FDDRL checks whether the format of the output disk matches that of the input disk. If the formats do not match, FDDRL proceeds as follows in accordance with the specification in the `OVERWRITE-DISK` operand:

- If a VSN or `*SAME-VSN` is specified, FDDRL rejects the output disk. The copy/restore run will not be started or will be aborted. You can then convert the output disk to the correct format by means of the VOLIN utility routine (see the “Utility Routines” manual [2]).
- If an MN or `*SAME-UNIT` is specified, FDDRL formats and initializes the disk automatically before it is used again.
- If `*BY-OPERATOR` is specified, FDDRL offers another mount option.

When accessing a disk to be processed, FDDRL automatically recognizes what format it has. It is therefore not necessary to specify the format.

There are various different allocation units, depending on the format. An allocation unit is the smallest possible unit of storage space that the system makes available when creating or extending a file. The following allocation units are supported for the formats KEY, NK2 and NK4 in BS2000:

Format	Allocation unit
KEY	6
NK2	6, 8, 64
NK4	8, 64

FDDRL always assigns the output disk the same allocation unit as the input disk.

If the output disk deviates from the input disk with regard to the DMS PAM keys or the allocation unit, the operator is requested to confirm the change. If no confirmation is given, the copy/restore run will be aborted or, if the output disk was requested with `OVERWRITE-DISK=*BY-OPERATOR`, FDDRL offers another mount option.

i The format of the disk must be distinguished from the definition which was specified when the disk was initialized with VOLIN using the FORMAT operand (KEY, NK2, NK4). This is independent of whether or not space is provided on the disk for writing a PAM key. It is therefore perfectly feasible for a disk to be defined for use by DMS as an NK disk even if the formatting of the disk allows a PAM key to be written.

Replacing defective pubset disks

If, during operation or FDDRL saves, pubset disks prove to have partial disk defects, these disks should either be reformatted (VOLIN utility routine, see the “Utility Routines” manual [2]) or replaced.

The description below is not relevant for EMC disk subsystems because the integrated RAID technology means that no disk defects occur on these.

The extent of the disk defects can be determined in various ways:

- by a large defect-garbage file or by reaching the extent limit in this file (DMS060C).
- by a large number of EXC0857 messages when the disk is accessed or when it is saved using FDDRL.

Procedure for replacing disks

- Identify the disks affected by the defect using the following command: `/SHOW-FILE-ATTR SELECT=*BY-ATTR (STATUS=*PAR(DEFECT-REPORTED=*YES))`. A list of files with defects that have already been detected is then displayed. The defective blocks are also referenced by the extent list in the defect-garbage file.
- If you have logical saves of these files, restore the files with these saves.
- Export the pubset.
- Use FDDRL to copy the disk to a new disk or to save the disk to tape; reformat the disk with VOLIN (FDDRL formats the disk automatically when the MN is specified) and use the save to restore it.
When the defective disks are read by FDDRL the I/O errors are skipped with “CONTINUE” or “CONTINUE ALL”. The numbers of the defective blocks are output (EXC0857, FBD3106, FBD3108); make a record of these numbers.
- Import the pubset with the new or reformatted disk.
- Using the block numbers you recorded, identify the files affected by the defect. The following command can be used to do this: `/SHOW-BLOCK-TO-FILE-ASSIGNMENT`
- Restore any files that are now still affected by means of their logical saves or by some other means.

Backup volumes

Disks can be saved in two different ways: either by creating an exact physical copy on another disk of the same format. Or by copying the contents of a disk to MTC. If necessary, FDDRL can then use these MTC to create a physical copy of the saved volume on a disk of the original device type (restoration of the disk).

For backup, FDDRL supports MTC of the types which are supported by the BS2000 OS DX version used. The currently valid list can be taken from the overview of the supported hardware configuration for a BS2000 OS DX version (see the "System Installation" manual [13] or the relevant Release Notice).

The type of backup volume is defined via the TAPE-DEVICE-TYPE operand of the //DUMP-DISK, //DUMP-PUBSET, //RELOAD-DISK and //RELOAD-PUBSET statements.

FDDRL accepts as backup volumes only those volumes which have been initialized in accordance with DIN 66029 and whose retention period has expired. This enhances data protection and allows the FDDRL backup volumes to be archived by means of the standard procedure specific to the computer center.

You can copy FDDRL backup volumes (MTC) using the software product PERCON (see [chapter "Duplicating FDDRL backup volumes with PERCON"](#) or the "PERCON" manual [11]). You can also restore disks as from the original backups from the backup MTC copied with PERCON.

If you are using the software product MAREN, MAREN administers all FDDRL backup MTC (see [chapter "Administering the volumes with MAREN"](#)).

Premounting backup volumes

For both DUMP-DISK and RELOAD-DISK runs with devices which have no robot (i.e. which only have a stacker), the backup volumes can be premounted in such a way that no operator intervention is necessary during the FDDRL run. For this purpose, the VSNs of the backup volumes have to be specified with the TAPE operand (also in the correct sequence in the case of RELOAD-DISK).

See also [chapter "Automating FDDRL saves"](#).

Tape error statistics

FDDRL compiles tape error statistics for the MTC used in DUMP-DISK, DUMP-PUBSET, RELOAD-DISK and RELOAD-PUBSET. At the end of each processing run, FDDRL reports how many errors (DATA ERRORS) occurred during processing of that particular volume.

Error messages during tape processing may not necessarily be due to faulty MTC. The errors could, for example, be caused by dirty read/write heads on the MTC device. In any event, an increased incidence of errors should certainly be investigated.

Only high-quality volumes (i.e. with less than 20 errors) should be used for save runs.

Protecting backup volumes

All backup volumes created during an FDDRL run (between the START-JOBS statement and the end of the job), are part of the same save run. They all receive the same date and time ID. A volume can only be used once during a save run, except in the case of a restart (see [section "Restart"](#) below) or if the volumes are used again for the same disk following an aborted run. A retention period can be issued for the volumes of a save run; the backup volumes must not be overwritten during this period. All backup volumes to which data is written are assigned the user-specified retention period in the HDR1 label. The retention period is specified as follows:

```
//MODIFY-FDDRL-PARAMETERS RETENTION-PERIOD=<integer 0..32767 days>
```

A backup volume can be protected against overwriting for up to 32767 days. Unless otherwise specified, the retention period is 0 days, but even then a backup volume cannot be overwritten during the same run.

The retention period is used to calculate the EXPIRATION-DATE. It is recorded in the MAREN catalog if MAREN is used.

! **WARNING!** Risk of backup volumes being overwritten. A retention period of 1 day may not be sufficient for FDDRL runs taking place shortly before midnight, since the time of day is not taken into account when the expiration date is calculated.

Restart

If an unrecoverable write error is discovered while writing or reading on MTC, FDDRL issues a message inquiring whether a restart should be performed. If a restart is desired, FDDRL goes to the beginning of the aborted volume and writes the same data again.

If the MTC device caused the error, the same volume can be remounted onto another MTC device and written again. (This is the only case in which a tapeMTC can be written twice in the same save run.)

If the volume itself was the error source, another volume should be mounted. If the VSN was specified via the TAPE operand, the new volume must have the same VSN as the defective one.

Compatibility of the backup volumes

Disk compatibility

Data transfer between disks (with COPY-DISK or DUMP-DISK/RELOAD-DISK) is only possible if both disks have the same device type.

With disks of the same type but different sizes, data can only be transferred to the larger of the two. This means that some of the space on this larger disk will remain unused.

In mirrored-disk operation with RAID1 disks, if the device type is the same, data can also be transferred with COPY-DISK or DUMP-DISK/RELOAD-DISK between disks with and without mirroring.

MTC compatibility

Backup volumes (MTC) which has been created with a different FDDRL version from the one currently used can only be read by the latter version if the corresponding tape type (TAPE-DEVICE-TYPE) is supported by the other (older or newer) FDDRL version.

If the MTCs are administered by MAREN, they can only be used for restoration if the archive attribute CLOSE-INDICATOR=CLOSED is entered; for the reloading of older FDDRL saves, this indicator itself must be set (see [chapter "Adjusting volume characteristics of earlier FDDRL versions"](#)).

Magnetic tape cartridge devices with stacker

MTC devices can be equipped with a stacker which can hold several cartridges. BS2000 supports these stackers for pre-mounting cartridges in FDDRL runs when AUTOMATIC mode is set on the stacker. The stackers can be used as follows:

- When mounting cartridges, the system prefers MTC devices with stackers. When processing continuation tapes, the MTC device already used is retained in the device suggestion of the mount request. The operator need only confirm the suggestion.
- The TAPE operand of the DUMP statements specifies the VSNs of the cartridges to be used. The cartridges are inserted in the stacker of the MTC device in the same order. When one cartridge has been filled with data, the system waits for the stacker to load another one before requesting the next cartridge. If the desired VSN matches that of the next cartridge in the stacker, FDDRL accepts the cartridge automatically.
- The cartridges to be used are not specified in the TAPE operand (TAPE=*ANY with the DUMP statements or TAPE=*BY-OPERATOR with the RELOAD statements). A stacker is loaded with cartridges. In the subsequent mount request (NKVT013), the operator can enter the mnemonic name of the MTC device together with the suffix "-ALL". The system now expects all continuation cartridges to be automatically loaded onto the same MTC device and it assigns them without any further mount messages.

Device management for FDDRL

In FDDRL there are numerous links to Nucleus Device Management (NDM), the BS2000 device management facility.

- Data volumes are allocated the usage mode "SPECIAL". This is shown by `USE=SPECIAL` in the mount requests (NKVD013, NKVT013) and in the output of the commands `/SHOW-DISK-STATUS` and `/SHOW-TAPE-STATUS`.
- FDDRL allocates data volumes (including private disks) as "task-exclusive". For this reason they must be free before they are allocated (`PHASE=ONLINE`). Also, they cannot be requested by other tasks while they are being used by FDDRL.
- Disks are allocated as "system-exclusive" (NKVD013) with `SYS-ALLOC=EXCL`. The system ID of this system's home pubset is entered in the system allocation log of the standard volume label so that simultaneous use of the disk by other systems is prevented. The type of allocation is shown in `/SHOW-DEVICE-STATUS` and `/SHOW-DISK-STATUS (VOL-A=EXCL)`.
- When processing private disks with FDDRL you should note the following: if the disk was mounted using `/SET-DISK-PARAMETER` or `/SET-DISK-DEFAULTS` with the attribute `ASSIGN-TIME=*OPERATOR`, the private disk is occupied by DMS as soon as it is switched online. In this case the disk must be released using the command `/SET-DISK-PARAMETER VOL=<vsn>,ASSIGN-TIME=*USER`.
- If FDDRL processing is performed under subtasks, disk and tape devices may not be reserved under the main task with the `/SECURE-RESOURCE-ALLOCATION` command since this reservation cannot be passed on to the subtasks. Devices reserved in this way would be blocked for the subtasks. Tape devices which are required are reserved by the FDDRL subtask itself. Here the subtask waits for the period specified in the FDDRL parameter `SECURE-WAIT-TIME`.
- In the case of FDDRL tapes, a `VOLUME-MOVE (/CHANGE-TAPE-MOUNT)` is possible in spite of the SPECIAL allocation so that hardware errors can be avoided.

Scope of supply and installation of FDDRL

FDDRL V21.0A will run under BS2000 OS DX version 1.0 or higher.

An appropriate earlier FDDRL version must be used for older BS2000 versions.

Scope of supply of FDDRL

The release unit FDDRL V21.0A contains the following files (release items):

- **SYSLNK.FDDRL.210**
Object module library for dynamic loading of the FDDRL components on S servers; TASKLIB for duplication of FDDRL backup tapes using PERCON
- **SKMLNK.FDDRL.210**
Object module library for loading the TU and TPR units on SUx86; TASKLIB for duplication of FDDRL backup tapes using PERCON
- **SYSTEMS.FDDRL.210**
Message file in MSGMAKER format
- **SYSNRF.FDDRL.210**
NOREF file for dynamic loading via DSSM
- **SYSPAR.FDDRL.210**
Parameter file for system-specific settings of the FDDRL parameters
- **SYSPRC.FDDRL.210**
Library containing SDF procedures
 - \$FBDVLN,J: internal S-procedure to perform a VOLIN run for disk formatting
 - CHECK-OLS, J: S-procedure to check the preconditions for an online save
 - CHECK-OLS, SYSJ: compiled S-procedure to check the preconditions for an online save (This variant is executable without SDF-P)
 - MERGE-PAR, J: S-procedure to merge a customer specific parameter file with the current parameter file
 - MERGE-PAR, SYSJ: compiled S-procedure to merge a customer specific parameter file with the current parameter file (This variant is executable without SDF-P)
- **SYSRMS.FDDRL.210**
File containing object corrections for the FDDRL modules
- **SYSSDF.FDDRL.210**
Subsystem syntax file containing the FDDRL statements
- **SYSSII.FDDRL.210**
File containing information on the structure and installation of FDDRL (for IMON)
- **SYSSSC.FDDRL.210**
File containing declarations for the BS2100 subsystem catalog

With FDDRL a SYSSII file is supplied for use under IMON. This file contains information on the structure and the installation of the product. It includes details of the product (release unit) and the release items. Each release item has a logical name which is used for the installation. The logical name, which is version-independent, is the part of the file name which describes the function it performs.

For more information about IMON, see the "IMON" manual [7].

Structure of the release unit FDDRL V21.0A:

Release item	Logical name
SYSLNK.FDDRL.210	SYSLNK
SKMLNK.FDDRL.210	SYSLNK
SYSTEMS.FDDRL.210	SYSTEMS
SYSNRF.FDDRL.210	SYSNRF
SYSPAR.FDDRL.210	SYSPAR
SYSPRC.FDDRL.210	SYSPRC
SYSRMS.FDDRL.210	SYSRMS
SYSSDF.FDDRL.210	SYSSDF
SYSSII.FDDRL.210	SYSSII
SYSSSC.FDDRL.210	SYSSSC
SYSREP.FDDRL.210	SYSREP

Installation of FDDRL

FDDRL can be installed with IMON under any user ID (see [chapter "Scope of supply of FDDRL"](#)).

The FDDRL subsystem catalog (SYSSSC.FDDRL.210) is entered in the BS2000 subsystem catalog.

The name of the syntax file (SYSSDF.FDDRL.210) is entered in the SDF parameter file. The syntax file is activated by SDF. It contains the FDDRL statements and the `/START-FDDRL` command.

The name of the message file (SYSMES.FDDRL.210) is entered in the MIP parameter file. The MIP parameter file is activated by MIP (Message Improvement Processing).

FDDRL parameters can be set on a system-specific basis in the SYSPAR.FDDRL.210 parameter file. If the parameter file already exists, it is not overwritten, but installed under the name SYSPAR.FDDRL.210.NEW. This ensures that any system-specific settings made for an earlier installation are retained.

FDDRL parameter

FDDRL knows the following parameters; the meaning of the parameters is described in the statement [MODIFY-FDDRL-PARAMETERS Define execution parameters](#).

parameter	standard value	possible values
TASK-LIMIT	4	<INTEGER_1..16>
RUN-PRIORITY	*STD	*STD / *OWN / <INTEGER_128..255>
STATUS-OUTPUT-PERIOD	5	<INTEGER_0..60>
SECURE-WAIT-TIME	60	*NO / <INTEGER_0..604800 seconds>
JOB-NAME	*STD	*STD / *SAME / <NAME_1..8>
PROCESS-JOBS	*BY-SUBTASKS	*BY-SUBTASKS / *BY-MAINTASK
RETENTION-PERIOD	0	<INTEGER_0..32767>
LOCATION	*STD	*STD / <ALPHANUM-NAME_1..8>
TAPE-FORMAT	*STD	*STD / *V17-FORMAT / *V16-FORMAT / *V15-FORMAT
REPORT-FILE	*STD	*STD / *NONE
MONITORING	*NO	*NO / *STD
KEEP-REQUESTS	40	<INTEGER_1..32767> / *YES

Adapting the parameters in the FDDRL parameter file

The FDDRL parameter file shows all FDDRL parameters with their possible and default values. The character '#' introduces a comment up to the end of the line. If you want to change the default values for the parameters, remove the '#' character at the beginning of the relevant line and enter the desired parameter value.

Like the operands in the `// MODIFY-FDDRL-PARAMETERS` statement, the parameters can also be specified in one line.

Merge of FDDRL parameter files

You can use the merge procedure **MERGE-PAR** to add system specific parameter of an existing FDDRL installation to the current parameter file. Thereby current system specific parameters are added at the end of the new installed parameter file (comment lines are not added) and are stored under the file name SYSPAR.FDDRL.210 by default. An existing parameter file is saved to the file name SYSPAR.FDDRL.210.SAV in advance.

You may call the merge procedure by the command `/CALL-PROCEDURE *LIB-EL(SYSPRC.FDDRL.210,MERGE-PAR)`.

With Operanden `PROCEDURE-PARAMETERS=*PARAMETERS(...)` you may specify the following parameters:

Parameter of the procedure MERGE-PAR

SYSPAR-NEW = *STD / <filename 1..54 without-gen-vers>

,SYSPAR-OLD = *STD / <filename 1..54 without-gen-vers>

,SYSPAR-OUT = *STD/ <filename 1..54 without-gen-vers>

Description of procedure parameter

SYSPAR-NEW = *STD / <filename 1..54 without-gen-vers>

Specifies the new installed parameter file

SYSPAR-NEW = *STD

Standard value is:

- SYSPAR.FDDRL.210.NEW for an update installation
- SYSPAR.FDDRL.210 for a new installation

SYSPAR-OLD = *STD / <filename 1..54 without-gen-vers>

Specifies the already installed system specific parameter file

SYSPAR-OLD = *STD

Standard value is:

- SYSPAR.FDDRL.210 for an update installation
- SYSPAR.FDDRL.200 for a new installation

SYSPAR-OUT = *STD / <filename 1..54 without-gen-vers>

Specifies the new created parameter file with system specific parameters

SYSPAR-OUT = *STD

The standard value is SYSPAR.FDDRL.210 for an update installation as well as a new installation.

Emergency system and FDDRL syntaxfiles

In the emergency system for the initial installation FDDRL is already pre-installed. However, in this system, FDDRL is restricted to the RELOAD function. A smaller SDF syntax file is also installed (SYSSDF.FDDRL.210.RELOAD). This means that the emergency system can be used for restoration purposes if a disaster occurs (all disks are destroyed).

If this emergency system is to be used as a standby system for FDDRL saves (e.g. of the home pubset), the original FDDRL syntax file (SYSSDF.FDDRL.210) must be installed on this system.

FDDRL statements

This chapter provides an overview of the FDDRL statements as well as notes on their formats and the syntax used. This is followed by a description of all statements in alphabetical order.

Table of Contents

- [Overview of the FDDRL statements \(FDDRL V21.0A 2021-06, #65\)](#)
- [Description of the FDDRL statements \(FDDRL V21.0A 2021-06, #66\)](#)
 - [COPY-DISK Copy disk \(FDDRL V21.0A 2021-06, #67\)](#)
 - [COPY-PUBSET Copy all disks of pubset \(FDDRL V21.0A 2021-06, #68\)](#)
 - [CREATE-JOBS Create FDDRL jobs \(FDDRL V21.0A 2021-06, #69\)](#)
 - [DELETE-REQUESTS Delete completed FDDRL requests \(FDDRL V21.0A 2021-06, #70\)](#)
 - [DUMP-DISK Save disk \(FDDRL V21.0A 2021-06, #71\)](#)
 - [DUMP-PUBSET Save all disks of pubset \(FDDRL V21.0A 2021-06, #72\)](#)
 - [END Terminate FDDRL \(FDDRL V21.0A 2021-06, #73\)](#)
 - [MODIFY-FDDRL-PARAMETERS Define execution parameters \(FDDRL V21.0A 2021-06, #74\)](#)
 - [RELOAD-DISK Restore disk \(FDDRL V21.0A 2021-06, #75\)](#)
 - [RELOAD-PUBSET Restore a pubset \(FDDRL V21.0A 2021-06, #76\)](#)
 - [SHOW-FDDRL-PARAMETERS Show the current FDDRL parameters \(FDDRL V21.0A 2021-06, #77\)](#)
 - [SHOW-FDDRL-STATUS Show running \(foreign\) FDDRL jobs \(FDDRL V21.0A 2021-06, #78\)](#)
 - [SHOW-FDDRL-TAPE-INFORMATION Show the tape information of an FDDRL tape \(FDDRL V21.0A 2021-06, #79\)](#)
 - [SHOW-REQUESTS Show running and completed FDDRL requests \(FDDRL V21.0A 2021-06, #80\)](#)
 - [START-JOBS Start FDDRL jobs \(FDDRL V21.0A 2021-06, #81\)](#)

Overview of the FDDRL statements

The following FDDRL statements are available:

FDDRL statement	Alias	Meaning
COPY-DISK	CPD	Copy a disk to another disk
COPY-PUBSET	CPP	Copy all disks of a pubset to disks
CREATE-JOBS	CRJ	Create the FDDRL jobs from the specified FDDRL statements
DELETE-REQUESTS	DLR	Delete completed FDDRL requests from the request file
DUMP-DISK	DMD	Save a disk to MTC
DUMP-PUBSET	DMP	Save all disks of a pubset to MTC
END		Terminate the FDDRL utility routine
MODIFY-FDDRL-PARAMETERS	MFP	Define the execution parameters for the FDDRL run
RELOAD-DISK	RLD	Restore a disk from MTC created with DUMP-DISK or DUMP-PUBSET
RELOAD-PUBSET	RLP	Restore a complete pubset from MTC
SHOW-FDDRL-PARAMETERS	SFP	Display the current FDDRL parameters
SHOW-FDDRL-STATUS	SFS	Show running (foreign) FDDRL jobs
SHOW-FDDRL-TAPE- INFORMATION	SFT	Show the tape information of an FDDRL tape
SHOW-REQUESTS	SHR	Show running and completed FDDRL requests
START-JOBS	STJ	Start the FDDRL jobs

In addition, the following SDF standard statements are available during an FDDRL run:

SDF statement	Meaning
EXECUTE-SYSTEM-CMD	Execute a command during the program run
HELP-MSG-INFORMATION	Output a system message to SYSOUT
HOLD-PROGRAM	Interrupt a program to enter commands
MODIFY-SDF-OPTIONS	Activate or deactivate the user syntax file and modify the SDF settings
REMARK	Comment on statement sequences

RESET-INPUT-DEFAULTS	Reset the task-specific default values
RESTORE-SDF-INPUT	Show the last input
SHOW-INPUT-DEFAULTS	Show the task-specific default values
SHOW-INPUT-HISTORY	Output stored inputs to SYSOUT
SHOW-SDF-OPTIONS	Display the SDF settings
SHOW-STMT	Display statement syntax description
STEP	Define a restart point
WRITE-TEXT	Output a specific text to SYSOUT

The SDF standard statements are not described in this manual (with the exception of END). For their description, see the manual “SDF Dialog Interface” [4].

The MODIFY-FDDRL-PARAMETERS statement defines the parameters for an FDDRL run. The current parameter values are accepted when the CREATE-JOBS statement is executed. If you enter more than one MODIFY-FDDRL-PARAMETERS statement and specify operands more than once, only the last operand before the START-JOBS statement is valid.

The statements for saving and restoring are used to define the FDDRL functions.

The CREATE-JOBS statement creates an FDDRL job for each function (in the case of pubsets for each disk) and for each disk set. When the START-JOBS statement is called, the FDDRL jobs are executed as specified in the PROCESS-JOBS operand of the MODIFY-FDDRL-PARAMETERS statement:

- PROCESS-JOBS=*BY-MAINTASK
The FDDRL jobs are executed under their own task.
- PROCESS-JOBS=*BY-SUBTASKS
The FDDRL jobs are executed under FDDRL subtasks.

Once the FDDRL jobs are completed, you can define and start additional functions or terminate the program with END.

With the exception of CREATE-JOBS, START-JOBS and END, you can specify the statements in any order.

Only one job can be defined for a disk or for a disk set:

- If you specify the VSN of a disk in more than one statement, only the last statement entered is processed. A message is issued to inform the user of this.
- If, however, a disk is addressed first via a statement for a pubset (e.g. DUMP-PUBSET) and then via a statement for a disk (e.g. DUMP-DISK), some of the values are taken over from the DISK statement for the FDDRL job of the pubset disk of the same name. These values concern the specification of the output device and are as follows:
 - with the COPY-PUBSET and COPY-DISK statements:
OVERWRITE-DISK
 - with the DUMP-PUBSET and DUMP-DISK statements with specification of SAVE-ENTITY=*SINGLE-DISK:
TAPE, TAPE-DEVICE-TYPE

If a statement is found to contain a syntax or semantic error, the spin-off mechanism is triggered. This mechanism, however, is only effective in procedure or batch mode.

The spin-off mechanism is also triggered after START-JOBS if only one of the FDDRL jobs was terminated abnormally or aborted. Erroneous executions of the FDDRL jobs (also under the FDDRL subtasks) are thus spread out into the FDDRL statement level and, if the STEP statement is missing, into the command level also.

Description of the FDDRL statements

This section presents all FDDRL statements in alphabetical order.

Each statement description begins with a general explanation of the statement's function. It is followed by its format and a description of the operands with the associated values.

Statement formats

The statements are written in SDF format (for an explanation of the syntax elements, see the the "Commands" manual [9]). They are followed by the operand descriptions.

Operand descriptions

In the descriptions of the operands, not all operand values are listed again and explained in detail every time.

Where the meaning is clear from the description of the operand and the name of the operand value, an explanation has been omitted.

In most cases, the default value is not explicitly described. The operand value

*UNCHANGED is not explained in the operand descriptions. It means that the existing value of the operand is not changed.

Example from the FDDRL statement:

PROCESS-JOBS = *UNCHANGED / *BY-MAINTASK / *BY-SUBTASKS(...) The task for the FDDRL jobs is defined.

COPY-DISK Copy disk

This statement copies a disk to another disk of the same device type. If `OVERWRITE-DISK=*BY-OPERATOR` is specified, message FBD1251 is issued, asking for confirmation before the disk is overwritten.

The `ACTION` operand specifies whether the copy operation is for saving or restoration purposes.

COPY-DISK	Alias: CPD
DISK = <vsn 1..6>	
,OVERWRITE-DISK = <u>*SAME-VSN</u> / <vsn 1..6> / *UNIT(...) / *BY-OPERATOR	
*UNIT(...)	
UNIT = <alphanum-name 2..2> / <x-text 4..4>	
,ACTION = <u>*SAVE</u> / *RESTORE	

DISK = <vsn 1..6>

VSN of the disk to be copied.

FDDRL determines the device type automatically from the VSN. If this is not possible, for example because the disk is not attached, the disk with device type STDDISK is requested.

OVERWRITE-DISK =

Specifies the VSN of the output disk to be mounted. If copying is successful, the output disk is given the VSN of the input disk (original VSN).

OVERWRITE-DISK = *SAME-VSN

The output disk to be mounted has the same VSN as the input disk that was specified under the `DISK` operand.

OVERWRITE-DISK = <vsn 1..6>

The output disk to be mounted has the specified VSN.

OVERWRITE-DISK = *UNIT(...)

The output disk to be mounted is specified using its mnemonic device name (MN). The operator must confirm the disk assignment. The output disk can be used even if it is not initialized or has the wrong format. In this case it is automatically initialized and formatted in the required format (if this is possible with VOLIN).

UNIT = <alphanum-name 2..2> / <x-text 4..4>

Mnemonic device name (MN) of the output disk.

OVERWRITE-DISK = *BY-OPERATOR

The VSN of the output disk to be mounted is not known to the FDDRL user. The operator is responsible for assigning the correct VSN.

The operand value `*BY-OPERATOR` should not be used during normal operation, since an operator error may cause foreign disks to be overwritten inadvertently. `*BY-OPERATOR` is intended exclusively for cases where the VSN is not known. This situation can occur if a previous `COPY-DISK`, `COPY-PUBSET`, `RELOAD-DISK` or `RELOAD-PUBSET` run was not terminated correctly. In this case, the SVL still contains a dummy VSN which has been assigned by FDDRL (formed from the time of the FDDRL run) and which is designed to prevent BS2000 from working with this disk.

ACTION =

Specifies whether the copy operation is for saving (*SAVE) or restoring (*RESTORE) if the input disk has the same VSN as the output disk. Here it could happen that the operator mistakenly mixes up the two disks, in which case the copy direction would be wrong. The ACTION operand prevents this. FDDRL can use this operand to check the copy direction. Here the time stamps in the SVLs of the two disks are compared. If the disks are found to have been incorrectly assigned, the assignments for the input and output disks are rejected. Operation then continues with the mount request for the input disk.

ACTION = *SAVE

The disk contents are to be saved, i.e. the disk with the earlier time stamp is to be overwritten by the contents of the disk with the more recent time stamp. If the operator assigns an output disk with a more recent time stamp than that of the input disk, the assignment is rejected.

ACTION = *RESTORE

The disk contents are to be restored, i.e. the disk with the more recent time stamp is to be overwritten by the contents of the disk with the earlier time stamp. If the operator assigns an output disk with an earlier time stamp than that of the input disk, the assignment is rejected.

COPY-PUBSET Copy all disks of pubset

This statement copies all disks of a pubset to disks of the same device type. For each disk of the pubset to be copied, there must be one output disk. All disks of the pubset to be copied must be attached and available before CREATE-JOBS, otherwise the FDDRL function for this pubset will not be started.

The disks of the home pubset can be saved during ongoing operation by specifying PUBSET=*HOME.

The ACTION operand specifies whether the copy operation is for saving or restoration purposes.

COPY-PUBSET	Kurzname: CPP
PUBSET= <catid 1..4> / *HOME(...)	
*HOME(...)	
DEFAULT-PUBSET = *HOME / <catid 1..4>	
, TIME-LIMIT = 5 / <integer 1..120 <i>minutes</i> >	
, OVERWRITE-DISK = *SAME-VSN / *BY-OPERATOR / *UNIT(...) / list-poss(1000): <vsn 1..6>	
*UNIT(...)	
UNIT = list-poss(1000): <alphanum-name 2..2> / <x-text 4..4>	
, ACTION = *SAVE / *RESTORE	

PUBSET =

Determines the pubset which is to be copied.

PUBSET = <catid 1..4>

Catalog ID of the pubset to be copied.

FDDRL determines the device type automatically from the VSN of the PUBRES. If this is not possible, for example because the disk is not attached, the disk with device type STDDISK is requested.

PUBSET = *HOME(...)

The home pubset is to be copied during ongoing operation (online save).

DEFAULT-PUBSET = *HOME / <catid 1..4>

Determines the pubset on which the SYSOUT files of the subtasks and the FDDRL variable for recording the tape VSNs are stored.

The default setting is DEFAULT-PUBSET=*HOME: the SYSOUT log is disabled and the job variable is only supplied with a value at the end of the save.

TIME-LIMIT = 5 / <integer 1..120 minutes>

Defines a time limit for the online save. After the time limit has expired, the write lock for the disks of the home pubset is canceled. Any save run which has not yet been completed is continued, but in this case the save is inconsistent and unusable for restoration.

OVERWRITE-DISK =

Specifies the VSN of the output disk to be mounted. If copying is successful, the output disk is given the VSN of the input disk (original VSN).

OVERWRITE-DISK = *SAME-VSN

The output disks to be mounted have the same VSNs as the corresponding input disks.

OVERWRITE-DISK = *BY-OPERATOR

The VSNs of the output disks to be mounted are not known to the FDDRL user. The operator is responsible for assigning the correct VSN.

The operand value *BY-OPERATOR should not be used during normal operation, since an operator error may cause foreign disks to be overwritten inadvertently. *BY-OPERATOR is intended exclusively for cases where the VSN is not known. This situation can occur if a previous COPY-DISK, COPY-PUBSET, RELOAD-DISK or RELOAD-PUBSET run was not terminated correctly. In this case, the SVL still contains a dummy VSN which has been assigned by FDDRL (formed from the time of the FDDRL run) and which is designed to prevent BS2000 from working with this disk.

OVERWRITE-DISK = *UNIT(...)

The output disks to be mounted are specified using their mnemonic device names (MNs). The operator must confirm the disk assignment. The output disks can be used even if they are not initialized or have the wrong format. In this case they are automatically initialized and formatted in the required format (if this is possible with VOLIN).

UNIT = list-poss(1000): <alphanum-name 2..2> / <x-text 4..4>

Mnemonic device names (MNs) of the output disks.

OVERWRITE-DISK = list-poss(1000): <vsfn 1..6>

The output disks to be mounted have the specified volume serial numbers.

ACTION =

Specifies whether the copy operation is for saving (*SAVE) or restoring (*RESTORE) if the input disk has the same VSN as the output disk. Here it could happen that the operator mistakenly mixes up the two disks, in which case the copy direction would be wrong. The ACTION operand prevents this. FDDRL can use this operand to check the copy direction. Here the time stamps in the SVLs of the two disks are compared. If the disks are found to have been incorrectly assigned, the assignments for the input and output disks are rejected. Operation then continues with the mount request for the input disk.

ACTION = *SAVE

The disk contents are to be saved, i.e. the disk with the earlier time stamp is to be overwritten by the contents of the disk with the more recent time stamp. If the operator assigns an output disk with a more recent time stamp than that of the input disk, the assignment is rejected.

When an online save is performed for the home pubset (PUBSET=*HOME operand), only ACTION=*SAVE is possible.

ACTION = *RESTORE

The disk contents are to be restored, i.e. the disk with the more recent time stamp is to be overwritten by the contents of the disk with the earlier time stamp. If the operator assigns an output disk with an earlier time stamp than that of the input disk, the assignment is rejected.

CREATE-JOBS Create FDDRL jobs

Create FDDRL jobs

The CREATE-JOBS statement creates FDDRL jobs from the statements.

For a subset-related statement (DUMP-PUBSET, RELOAD-PUBSET, COPYPUBSET), FDDRL ascertains all the disks of the subset when a job is created and creates an FDDRL job for each disk or each disk set. Here each FDDRL job is supplied with the FDDRL parameters which are currently set.

In the event of a RELOAD-PUBSET or RELOAD-DISK from the disk set save entity, FDDRL requests a volume for saving purposes and, on the basis of the metadata recorded, ascertains the subset configuration or the single disks. When a job is created, FDDRL requests the output disk for each disk and checks the disk properties which are relevant for a successful restore.

In the case of the disk set save entity, an FDDRL job is also created for each disk set.

After job creation has been completed, FDDRL outputs a list of the jobs created (REQUESTED FDDRL JOBS) to SYSOUT or SYSLST. You can select the output target with the OUTPUT operand. You can check and, if required, correct the jobs on the basis of this list:

If you have entered a statement incorrectly, correct it and then call CREATE-JOBS again. When the job is created again, statements for disks which have an identical VSN and FDDRL function (copy, dump, reload) are eliminated. Statements which you have not corrected will be accepted unchanged and once more output in the list of created jobs.

When job creation ends with an error, spin-off is triggered in procedure and batch modes.

The FDDRL jobs created are started using the START-JOBS statement.

CREATE-JOBS	Alias: CRJ
OUTPUT = *SYSOUT / *SYSLST(...) / *NONE	
*SYSLST(...)	
SYSLST-NUMBER = *STD / <integer 1..99>	

OUTPUT =

Defines the output medium for the text output of reports and SHOW statements.

OUTPUT = *SYSOUT

Text output takes place to system file SYSOUT.

OUTPUT = *SYSLST(...)

Text output takes place to system file SYSLST.

SYSLST-NUMBER= *STD / <integer 1..99>

Defines whether the output is to take place to system file SYSLST or to a SYSLST file from the set SYSLST01 through SYSLST99. The default value is *STD, i.e. the output takes place to system file SYSLST.

OUTPUT = *NONE

No text output takes place.

DELETE-REQUESTS Delete completed FDDRRL requests

The DELETE-REQUESTS statement deletes requests from the request file. Only completed requests are deleted. By default all requests are deleted which are older than 40 days.

When a request is deleted, the associated report file in PDF format is also deleted.

DELETE-REQUESTS	Alias: DLR
<p>REQUEST-NAME = <u>*ALL</u> / <composed-name_1..10></p> <p>,REQUEST-DATE = <u>*INTERVAL</u>(...) / <u>*ANY</u> / <date>(…)</p> <p><u>*INTERVAL</u> (…)</p> <p> FROM = <u>*EARLIEST-DATE</u> / <date>(TIME = <u>00:00:00</u> / <time>)</p> <p> ,TO = <u>*STD</u> / <u>*TODAY</u> / <u>*YESTERDAY</u> / <integer_-99999..0 days> / <date>(TIME = <u>23:59:59</u> / <time>)</p> <p> <date>(…)</p> <p> TIME = <u>*ANY</u> / <time>)</p>	

REQUEST-NAME =

Selects the requests according to the request name.

REQUEST-NAME = *ALL

The request name is not a selection criterion.

REQUEST-NAME = <composed-name_1..10>

Deletes the requests with the specified name.

REQUEST-DATE =

The requests are selected in accordance with their request date.

REQUEST-DATE = *INTERVAL(…)

Only deletes requests whose request date lies within the specified period. Range specifications apply including the range limit values.

FROM =

Defines the lower limit of the period for the request date of the requests. Requests are deleted only if they were started on the specified date or later.

FROM = *EARLIEST-DATE

Defines that the period for the start time for the request date has no lower limit.

FROM = <date>(TIME = 00:00:00 / <time>)

Defines the specified date and, if required, also the specified time as the lower limit of the period for the request date of the requests.

TO =

Defines the upper limit of the period for the request date of the requests. Requests are deleted only if they were started on the specified date or earlier.

TO = *STD

The upper limit of the period for the request date is derived from the FDDRL parameter KEEP-REQUEST.

TO = *TODAY

Defines the date of the current day as the upper limit of the period for the request date of the requests.

TO = *YESTERDAY

Defines the date of the previous day as the upper limit of the period for the request date of the requests.

TO = <int -99999..0 days>

Defines the date which lies the specified number of days in the past as the upper limit of the period for the request date of the requests.

TO = <date>(TIME = 23:59:59 / <time>)

Defines the specified date and, if required, also the specified time as the upper limit of the period for the request date of the requests.

DUMP-DISK Save disk

This statement saves a disk onto MTC. You can then use this backup to restore the disk with the RELOAD-DISK statement if necessary.

DUMP-DISK	Alias: DMD
DISK = <vsn 1..6>(…) / list-poss(64): <vsn 1..6> <vsn 1..6>(…) SPECIAL-VSN = *NO / *YES ,TAPE = *ANY (…) / list-poss(255): <vsn 1..6> *ANY (…) LOCATION = *BY-PAR AMETER / *STD / <alphanum-name 1..8> ,TAPE-DEVICE-TYPE = *BY-VOL UME -CAT ALOG / <device> ,FILE-NAME = *STD / <filename 1..41 without-cat-user-gen-vers> ,RETENTION-PERIOD = *BY-PAR AMETER / <integer 0..32767 days>	

DISK = <vsn 1..6>(…)

Volume serial number of the disk to be copied. The disk is saved to a tape/tape set (single disk save entity).

FDDRL determines the device type automatically from the VSN. If this is not possible, for example because the disk is not attached, the disk with device type STDDISK is requested.

SPECIAL-VSN = *NO / *YES

In the case of disk mirroring with SHC-OSD (Storage Host Component), the additional disk units can be split and renamed during live operation. These are then assigned a special, temporarily modified VSN in order to ensure that they are not confused with the original disks. See also [chapter "Saving Clones \(SHC-OSD\)"](#).

The SPECIAL-VSN operand serves to assign disks and specifies whether the disk is an additional disk unit with a special VSN which was renamed in this way.

SPECIAL-VSN = *NO

The disk still has the original VSN.

SPECIAL-VSN = *YES

The disk has been assigned a special VSN as created by SHC-OSD with an unambiguous reference to the original VSN. The disk is saved as with the original VSN. It can be restored under the original VSN.

DISK = list-poss(64): <vsn 1..6>

All specified disks are combined to form a save entity and saved to a tape/tape set (disk set save entity).

TAPE = *ANY(…) / list-poss(255): <vsn 1..6>

Specifies the volumes to be used for the save. These volumes must have a valid label and their retention period must have already expired.

TAPE = *ANY(...)

The operator assigns any volumes of the specified type.
MARENUCP automatically assigns scratch tapes.

LOCATION =

The location specifies the set of devices, from which tape devices for SCRATCH requests are to be selected. Explicitly stating a location is only possible with the chargeable MAREN subsystem.

LOCATION = *BY-PARAMETER

The value set with MODIFY-FDDRL-PARAMETERS is to be used.

LOCATION = *STD

Device management is to determine the set of devices and select a suitable tape device from it. If MAREN is being used, device management obtains the information on the location from MAREN.

LOCATION = <alphanum-name 1..8>

Specifies the device set location from which tape devices for SCRATCH requests are to be selected. The /SHOW-DEVICE-DEPOT command can be used to display the allocations of locations to tape devices. If device management does not recognize the location specified, the device request is rejected.

Please refer to the additional information under [chapter "Defining a location"](#).

TAPE = list-poss(255): <vsn 1..6>

The designation and order of the volumes is determined by a list. In the mount process, the volumes specified here are requested. No reassignments to volumes with another VSN are permitted. In an emergency, the operator must use the INIT utility routine to initialize the missing volumes (see the "Utility Routines" manual [2]).

If the operator rejects volumes that have been specified incorrectly, the save run is terminated abnormally. If more volumes are required than are specified in the list, the remaining volumes are requested as with *ANY.

TAPE-DEVICE-TYPE = *BY-VOLUME-CATALOG / <device>

Data volume type of the MTC to be used for the save.

TAPE-DEVICE-TYPE = *BY-VOLUME-CATALOG

This specification is only possible when MAREN is being used.

MAREN determines the volume type itself. In the case of scratch tapes (TAPE=*ANY), this is the default volume type set for MAREN (MAREN parameter DEFAULT-DEVICE-TYPE).

TAPE-DEVICE-TYPE = <device>

Volume type of the MTC on the basis of which device management determines the type of device that will be required for tape processing. TAPE may not be specified.

FILE-NAME = *STD / <filename 1..41 without-cat-user-gen-vers>

Designates the file name in the MAREN volume catalog. The file name must begin with the string "FDDRL". If a job variable of the same name exists, it is supplied with the VSNs of the tapes used (see the [chapter "Job variable for backup volumes"](#)).

FILE-NAME = *STD

The file name is formed as follows:

Single disk: FDDRL.D.<vsn>

Disk set: FDDRL.D.<vsn>-1, where <vsn> is the VSN of the first disk specified and "1" designates the first disk set.

RETENTION-PERIOD =

Retention period of the volumes in days.

RETENTION-PERIOD = *BY-PARAMETER

The value set in the MODIFY-FDDRL-PARAMETERS statement is used.

RETENTION-PERIOD = 0

There is no retention period. The backup volumes can be overwritten on the same day, but not in the same save run.

RETENTION-PERIOD = <integer 1..32767 days >

The backup volumes are protected against overwriting until the specified number of days has elapsed.

DUMP-PUBSET Save all disks of pubset

This statement saves all disks of a pubset to MTC. PUBSET=*HOME enables you to save the home pubset online.

The disks of the pubset are saved either separately (SAVE-ENTITY=*SINGLE-DISK) or as multiple disks together (SAVE-ENTITY=*DISK-SET) to their own MTCs (tape sets).

In the case of SAVE-ENTITY=*SINGLE-DISK, the individual disks of the pubset are saved separately to their own MTC (tape sets). If a pubset is to be restored, every disk of this pubset must be restored with the RELOAD-DISK statement.

In the case of SAVE-ENTITY=*DISK-SET, the disks of the pubset are combined to form one or more disk sets. The disks of a disk set are saved together to their own MTC (tape sets). A pubset saved in this way can be restored using a RELOAD-PUBSET statement.

DUMP-PUBSET

Alias: **DMP**

PUBSET = <catid 1..4>(…) / *HOME(…)

<catid 1..4>(…)

| **SPECIAL-VSN = *NO / *YES**

***HOME(…)**

| **DEFAULT-PUBSET = *HOME / <catid 1..4>**

| **,TIME-LIMIT = 5 / <integer 1..120 minutes>**

,TAPE = *ANY (…) / list-poss(512): <vsfn 1..6>

***ANY (…)**

| **LOCATION = *BY-PAR AMETER / *STD / <alphanum-name 1..8>**

,TAPE-DEVICE-TYPE = *BY-VOL UME -CAT ALOG / <device>

,SAVE-ENTITY = *DISK -SET (…) / *SINGLE-DISK

***DISK -SET (…)**

| **NUMBER-OF-DISK-SETS = 1 / <integer 1..8> / *BY-TASK-LIMIT**

,RETENTION-PERIOD = *BY-PAR AMETER / <integer 0..32767 days>

PUBSET =

Determines the pubset which is to be copied.

PUBSET = <catid 1..4>(…)

Catalog ID of the pubset to be copied.

FDDRL determines the device type automatically from the VSN of the PUBRES. If this is not possible, for example because the disk is not attached, the disk with device type STDDISK is requested.

SPECIAL-VSN = *NO / *YES

In the case of disk mirroring with SHC-OSD (Storage Host Component), the additional disk units can be split and renamed during live operation. These are then assigned a special, temporarily modified VSN in order to ensure that they are not confused with the original disks. See also chapter "[Saving Clones \(SHC-OSD\)](#)"
Operand `NEW-VSN=*SPECIAL-VSN` .

The SPECIAL-VSN operand serves to assign disks and specifies whether the disks are additional disk units with special VSNs which was renamed in this way.

SPECIAL-VSN = *NO

The disks still have the original VSN.

SPECIAL-VSN = *YES

Die Platten haben eine spezielle, von SHC-OSD gebildete VSN mit eindeutigen Bezug zur Original-VSN:

They can also be restored under their original VSN.

PUBSET = *HOME(...)

The home pubset is to be copied during ongoing operation (online save).

DEFAULT-PUBSET = *HOME / <catid 1..4>

Determines the pubset on which the SYSOUT files of the subtasks and the FDDRL variable for recording the tape VSNs are stored.

The default setting is `DEFAULT-PUBSET=*HOME`: the SYSOUT log is disabled and the job variable is only supplied with a value at the end of the save.

TIME-LIMIT = 5 / <integer 1..120 minutes>

Defines a time limit for the online save. After the time limit has expired, the write lock for the disks of the home pubset is canceled. Any save run which has not yet been completed is continued, but in this case the save is inconsistent and unusable for restoration.

TAPE =

Specifies the volumes to be used for the save. These volumes must have a valid label and their retention period must have already expired.

TAPE = *ANY(...)

The operator assigns any volumes of the specified type.

MARENUCP automatically assigns scratch tapes.

LOCATION =

The location specifies the set of devices, from which MTC devices for SCRATCH requests are to be selected. Explicitly stating a location is only possible with the chargeable MAREN subsystem.

LOCATION = *BY-PARAMETER

The value set with `MODIFY-FDDRL-PARAMETERS` is to be used.

LOCATION = *STD

Device management is to determine the set of devices and select a suitable MTC device from it. If MAREN is being used, device management obtains the information on the location from MAREN.

LOCATION = <alphanum-name 1..8>

Specifies the device set location from which MTC devices for SCRATCH requests are to be selected. The `/SHOW-DEVICE-DEPOT` command can be used to display the allocations of locations to MTC devices. If device management does not recognize the location specified, the device request is rejected.

Please refer to the additional information under [chapter "Defining a location"](#).

TAPE = list-poss(512): <vsn 1..6>

The specified MTCs can be used for saving.

The first n MTCs in the list are permanently assigned to the n tapes sets as the first MTC. If a save requires further MTC for a tape set, the continuation MTCs are assigned to the tape sets in dynamic sequence.

TAPE-DEVICE-TYPE =

Defines the volume type of the MTCs to be used for the save.

TAPE-DEVICE-TYPE = *BY-VOLUME-CATALOG

This specification is only possible when MAREN is being used.

MAREN determines the volume type itself. In the case of scratch tapes (TAPE=*ANY), this is the default volume type set for MAREN (MAREN parameter DEFAULT-DEVICE-TYPE).

TAPE-DEVICE-TYPE = <device>

Volume type of the MTCs on the basis of which device management determines the type of device that will be required for tape processing. TAPE may not be specified.

SAVE-ENTITY = *DISK-SET(...)

Multiple disks are combined to form a save entity, this being referred to as a disk set. The number of disk sets is defined in the NUMBER-OF-DISK-SETS operand.

NUMBER-OF-DISK-SETS = 1 / <integer 1..8> / *BY-TASK-LIMIT

Determines the number of save entities to be created. This specification controls the utilization of the MTC capacity on the one hand and the parallelism of the save on the other:

The lower the number of disk sets created per pubset, the higher the number of disks which are assigned to the individual disk sets and, as a result, the better the utilization of the disk capacity.

The greater the number of disk sets created per pubset, the greater the degree of parallelism in the save.

The number of disk sets is automatically limited to the number of disks per pubset and to the value of the FDDRL parameter TASK-LIMIT. If PROCESS-JOBS=*BY-

MAINTASK is set, the number of disk sets is limited to 1.

One tape set is generated per save entity. You must ensure that not more than 255 MTC are required per tape set.

NUMBER-OF-DISK-SETS = *BY-TASK-LIMIT

The number of save entities to be formed is derived from the FDDRL parameter TASK-LIMIT.

SAVE-ENTITY = *SINGLE-DISK

The save entity is the single disk. Each disk of the pubset is saved to its own MTC/tape set. As many tape sets are created as there are disks per pubset.

RETENTION-PERIOD =

Retention period of the volumes in days.

RETENTION-PERIOD = *BY-PARAMETER

The value set in the MODIFY-FDDRL-PARAMETERS statement is used.

RETENTION-PERIOD = 0

There is no retention period. The backup volumes can be overwritten on the same day, but not in the same save run.

RETENTION-PERIOD = <integer 1..32767 days >

The backup volumes are protected against overwriting until the specified number of days has elapsed.

END Terminate FDDRL

This statement terminates an FDDRL run.

If valid statements have been entered for FDDRL functions but have not been executed with the START-JOBS statement, message FBD1151 is output as a warning. In batch mode and procedure mode, FDDRL is terminated with this warning. In interactive mode, the user is asked via message FBD1152 to confirm that he really wants to terminate FDDRL.

END

This statement has no operands.

MODIFY-FDDRL-PARAMETERS Define execution parameters

This statement defines the execution parameters for an FDDRL run.

After /START-FDDRL all parameters are either initialized with their default value (see section "FDDRL parameter" in chapter [Installation of FDDRL](#)) or with the values in the FDDRL parameter file adjusted on a system-specific basis, if these exist. A value other than *UNCHANGED changes parameter values. The new parameters remain valid until they are changed again. The parameter values are passed on to the entered FDDRL jobs when the job is created (see chapter [CREATE-JOBS Create FDDRL jobs](#)).

The SHOW-FDDRL-PARAMETERS statement enables you to query the values currently set.

i The changes made apply only for the current FDDRL run and are **not** stored in the FDDRL parameter file. You must make system-specific adjustments in the FDDRL parameter file.

MODIFY-FDDRL-PARAMETERS

Alias: **MFP**

PROCESS-JOBS = *UNCHANGED / *BY-MAINTASK / *BY-SUBTASKS(...)

***BY-SUBTASKS(...)**

| **TASK-LIMIT = *UNCHANGED / *STD / <integer 1..16>**

| **,RUN-PRIORITY = *UNCHANGED / *STD / *OWN / <integer 128..255>**

| **,STATUS-OUTPUT-PERIOD = *UNCHANGED / *STD / <integer 0..60 minutes>**

| **,SECURE-WAIT-TIME = *UNCHANGED / *NO / <integer 0..604800>(...)**

| **<integer 0..604800>(...)**

| **DIMENSION = *SECONDS / *MINUTES**

| **,JOB-NAME = *UNCHANGED / *STD / *SAME / <name 1..8>**

,RETENTION-PERIOD = *UNCHANGED / *STD / <integer 0..32767 days>

,LOCATION = *UNCHANGED / *STD / <alphanumeric-name 1..8>

,TAPE-FORMAT = *UNCHANGED / *STD / *V17-FORMAT / *V16-FORMAT / *V15-FORMAT

,REPORT-FILE= *UNCHANGED / *STD / *NONE

,MONITORING= *UNCHANGED / *STD / *NO

,KEEP-REQUESTS= *UNCHANGED / *STD / <integer 1..32767 days> / *YES

PROCESS-JOBS = *UNCHANGED / *BY-MAINTASK / *BY-SUBTASKS(...)

The task for the FDDRL jobs is defined. Default: *BY-SUBTASKS

PROCESS-JOBS = *BY-MAINTASK

The FDDRL jobs are executed one after the other under the calling task (FDDRL main task).

PROCESS-JOBS = *BY-SUBTASKS(...)

Each FDDRL job is processed under a separate FDDRL subtask. The FDDRL subtasks are executed in the same job class as the FDDRL main task. If this job class does not permit batch jobs, the default batch job class of the user ID is used.

The degree of parallelism is determined by the TASK-LIMIT operand, taking due account of the job class limit.

TASK-LIMIT = *UNCHANGED / *STD / <integer 1..16>

Defines the number of FDDRL subtasks that can be active at the same time. Limiting the number of tasks can be useful in order to avoid too great a reduction in system performance or to occupy only as many tape devices as are available to FDDRL.

TASK-LIMIT = *STD

The default value (4 FDDRL subtasks) is set.

TASK-LIMIT = <integer 1..16>

Maximum number of FDDRL subtasks that can be active at the same time. The job class limit (CLASS-LIMIT) is always observed.

RUN-PRIORITY = *UNCHANGED / *STD / *OWN / <integer 128..255> Defines the priority of the FDDRL subtasks.

The run priority limit of the job class is always observed.

Default: *STD

RUN-PRIORITY = *STD

The default priority of the job class is assumed.

RUN-PRIORITY = *OWN

The FDDRL subtasks are given the same priority as the FDDRL main task.

RUN-PRIORITY = <integer 128..255>

The FDDRL subtasks are given the specified priority.

STATUS-OUTPUT-PERIOD = *UNCHANGED / *STD / <integer 0..60 minutes >

The status of the FDDRL subtasks is output when the specified period (in minutes) has expired. Output is to SYSOUT of the FDDRL main task.

Default: *STD

STATUS-OUTPUT-PERIOD = *STD

The default value (5 minutes) is selected for the status output period.

STATUS-OUTPUT-PERIOD = 0

The status of the FDDRL subtasks is not output.

STATUS-OUTPUT-PERIOD = <integer 1..60 minutes >

The status of the FDDRL subtasks is output when the specified time has expired.

SECURE-WAIT-TIME = *UNCHANGED / *NO / <integer 0..604800>(…)

Determines whether or how long the FDDRL subtasks should wait for successful device reservation before a tape device is reserved. The specified wait time controls the response to unsuccessful device reservation:

- In the case of *NO or a wait time < 180 seconds, further processing depends on the response to the console message FBD1256.
- In the case of a wait time >= 180 seconds, the subtask is aborted.

Default: *NO

SECURE-WAIT-TIME = *NO

The FDDRL subtasks are not to wait for successful device reservation.

SECURE-WAIT-TIME = <integer 0..604800>(…)

Period which the FDDRL subtasks are to wait for successful device reservation. The maximum wait time is limited to 7 days.

DIMENSION = *SECONDS / *MINUTES

Dimension of the specified period. The default is seconds.

JOB-NAME = *UNCHANGED / *STD / *SAME / <name 1..8>

JOB-NAME controls the job name for the subtask. The job name can, for example, be used by the MAREN location manager MARENLM to control the tape allocation.

JOB-NAME = *STD

The job name of the subtask is derived either from the VSN of the disk or from the name of the disk set. The VSN is preceded by the "\$" character. Illegal characters in the job name, such as a period ('.') or hyphen ('-'), are replaced by the '#' character.

JOB-NAME = *SAME

The subtasks get the same job name as the main task.

JOB-NAME = <name 1..8>

The subtasks get the specified job name.

RETENTION-PERIOD = *UNCHANGED / *STD / <integer 0..32767 days > Retention period of the backup volumes in days.

Default: 0 days, i.e. no retention period

RETENTION-PERIOD = *STD

The default value (0 days) is selected for the retention period.

RETENTION-PERIOD = 0

There is no retention period. The backup volumes can be overwritten on the same day, but not in the same save run.

RETENTION-PERIOD = <integer 1..32767 days >

The backup volumes are protected against overwriting until the specified number of days has elapsed.

LOCATION = *UNCHANGED / *STD / <alphanum-name 1..8>

The location specifies the set of devices, from which tape devices for SCRATCH requests are to be selected. Explicitly stating a location is only possible with the chargeable MAREN subsystem.

LOCATION = *STD

Device management is to determine the set of devices and select a suitable tape device from it. If MAREN is being used, device management obtains the information on the location from MAREN.

LOCATION = <alphanum-name 1..8>

Specifies the device set location from which tape devices for SCRATCH requests are to be selected. The /SHOW-DEVICE-DEPOT command can be used to display the allocations of locations to tape devices. If device management does not recognize the location specified, the device request is rejected.

Please refer to the additional information under [chapter "Defining a location"](#).

TAPE-FORMAT = *UNCHANGED / *STD / *V17-FORMAT / *V16-FORMAT / *V15-FORMAT

Determines the tape format and consequently the compatibility of the saves. The tape format set defines the maximum functionality which FDDRL may use for the save. Statements which require the functions of a higher FDDRL version are rejected. The tape format of the save determines the minimum FDDRL version required for restoration. Better performance is achieved with each higher tape format. The default value is *STD.

TAPE-FORMAT = *STD

The default value for the tape format is *V17-FORMAT for disk sets or *V16-FORMAT if FDDRL calculates a maximum multiplexing level of 2. The default value for the single disk save entity is *V15-FORMAT.

TAPE-FORMAT = *V17-FORMAT

The V17 tape format permits a multiplexing level of 4 for the disk set save entity. Otherwise it corresponds to the V16 tape format.

TAPE-FORMAT = *V16-FORMAT

V16 tape format is used for the disk set save entity. Large tape blocks are created. The block size is approx. 160 KB irrespective of the maximum I/O size of the individual disks. Multiple disks are saved to a series of tapes with dual multiplexing between disk and tape.

TAPE-FORMAT = *V15-FORMAT

Large tape blocks (64 - 256 KB) are created for saves in V15 tape format. The block size depends on the maximum I/O size of the disks and is defined for each disk type in such a way that a change of connection type is possible between saving and restoration.

REPORT-FILE = *UNCHANGED / *STD / *NONE Controls the report output.

REPORT-FILE = *STD

Generates a report file with the following default file name for each request:

```
§TSOS.S.FBD.<tsn>.<Auftragsdatum und -zeit>.<Auftragsname>.
```

The subtask reports are copied to the report file and then deleted. The name of the report file is displayed with the SHOW-REQUESTS statement.

REPORT-FILE = *NONE

Generates no report file. The subtask reports are not deleted. This corresponds to the behavior of FDDRL versions older than V19.0A.

MONITORING = *UNCHANGED / *STD / *NO

Controls the display of FDDRL requests in the SE Manager.

This entry only has a meaning when FDDRL is running on a Server Unit of an SE server. In other cases there is no SE Manager and thus no monitoring.

MONITORING = *STD

On a Server Unit of an SE server each change to the request status is reported to the SE Manager.

When a request is completed, a report in PDF format is also created from the report file. This file can be displayed in the SE Manager.

If FDDRL is not running on a Server Unit of an SE server, this entry has the same effect as MONITORING = *NO.

MONITORING = *NO

No monitoring is performed. No FDDRL events are sent to the SE Manager. No report file in PDF format is created. If FDDRL is running on a Server Unit of an SE server, no FDDRL requests for this Server Unit are displayed in the SE Manager.

KEEP-REQUESTS = *UNCHANGED / *STD / <integer 1..32767 days> / *YES

Controls automatic deletion of requests from the request file and deletion of report files in PDF format.

This delete operation is always performed when new requests are created or when the SHOW-FDDRL-REQUESTS statement is executed.

i This parameter can only be used meaningfully when its setting is not run-specific, but system-specific. You are therefore recommended to modify this parameter solely in the FDDRL parameter file.

KEEP-REQUESTS = *STD

FDDRL requests and the corresponding report files in PDF format are deleted automatically after 40 days.

KEEP-REQUESTS = <integer 0..32767 days>

FDDRL requests and the corresponding report files in PDF format are deleted automatically after the specified number of days has passed.

KEEP-REQUESTS = *YES

FDDRL requests and the corresponding report files in PDF format are never deleted automatically.

RELOAD-DISK Restore disk

This statement restores disks from backup volumes that were created via the DUMP-DISK or DUMP-PUBSET statement.

Single disk save entity

If disks or a pubset were saved using the single disk save entity, you must enter a separate RELOAD-DISK statement for each disk when restoration takes place. All saves created using an FDDRL version < V16.0 are involved here. In FDDRL V16.0 and higher, such saves are created by specifying precisely one VSN in the case of DUMP-DISK, and in the case of DUMP-PUBSET using SAVE-ENTITY=*SINGLE-DISK.

Disk set save entity

If multiple disks were saved using the disk set save entity (at least 2 VSNs specified in DUMP-DISK), the disks to be restored are also specified in a VSN list. A single one of these disks can also be restored. In all cases the SAVE-ENTITY=*DISK-SET operand must be specified.

If a pubset was saved using the disk set save entity (DUMP-PUBSET with SAVE-ENTITY=*DISK-SET specified), the RELOAD-PUBSET statement should be used for restoration.

When restoration takes place from the disk set save entity, the last tape of the tape set is requested during the job creation phase, and the metadata at the end of the tape is analyzed. Information on the disk set is then output, and the output disks are requested and checked.

Notes

- If OVERWRITE-DISK=*BY-OPERATOR is specified, message FBD1251 is issued, asking for confirmation before the disk is overwritten.
- If the output disks are specified using their mnemonic names (MNs; OVERWRITE-DISK=*UNIT(...) operand), they can also be used if they are not initialized or have the wrong format. In this case they are automatically initialized and formatted in the required format.
- If a tape is rejected in the case of RELOAD-DISK, important metadata on the FDDRL tape save is output. This data provides information on the characteristics of the saved disk. It helps you when mounting disks (with regard to device type, format, capacity), when restoring and when allocating MTCs for FDDRL saves (see [chapter "Tape information and information on disk backup"](#)).

RELOAD-DISK

Alias: RLD

DISK = <vsn 1..6> / list-poss(64): <vsn 1..6>**,TAPE = *BY-VOLUME-CATALOG(...) / *BY-OPERATOR / list-poss(255): <vsn 1..6>*****BY-VOLUME-CATALOG(...)**| **FILE-NAME = <filename 1..41 without-cat-user-gen-vers>**| **,VERSION = *LATEST / <integer -999..0>****,TAPE-DEVICE-TYPE = *BY-VOLUME-CATALOG / <device>****,SAVE-ENTITY = *SINGLE-DISK / *DISK-SET****,OVERWRITE-DISK = *SAME-VSN / *BY-OPERATOR / *UNIT(...) / list-poss(64): <vsn 1..6>*****UNIT(...)**| **UNIT = *SAME-UNIT / list-poss(64): <alphanum-name 2..2> / <x-text 4..4>****DISK =**

VSN of one or more input disks that were saved and are now to be restored. The device type of the disk is the same as that of the saved disk. It is determined automatically.

DISK = <vsn 1..6>

Specifies a single VSN: It is not necessary to specify anything in the SAVE-ENTITY operand for a disk of the single disk save entity (corresponds to the default). If a disk of the disk set save entity is involved, the SAVE-ENTITY= *DISK-SET operand must be specified.

DISK = list-poss(64): <vsn 1..6>

Specifies a list with multiple VSNs: This specification is only possible for disks of the disk set save entity. The VSNs can be specified in any order. Fewer disks can also be specified than for the save. The SAVE-ENTITY= *DISK-SET operand must be specified.

TAPE =

Specifies the volumes to be used for the restoration.

TAPE = *BY-VOLUME-CATALOG(...)

This specification is only possible when MAREN is being used.

The volumes are to be determined automatically by MAREN. The file name in the volume catalog and a version specification are required for this purpose.

FILE-NAME = <filename 1..41 without-cat-user-gen-vers>

Designates the file name of the save in the MAREN volume catalog. By default the file names for the save have the following format:

- FDDR.L.D.<vsn> for single disk
- FDDR.L.D.<vsn>-1 for disk set, where <vsn> designates the first VSN specified in the save and "1" the first disk set.

VERSION = *LATEST / <integer -999..0>

Designates a particular file version in the volume catalog and consequently a particular save version.

VERSION = *LATEST

Selects the latest save version, i.e. that created most recently.

VERSION = <integer -999..0>

Selects the specified save version, the specification being relative to the last save version created.

VERSION=0 corresponds to the last save version, VERSION=-1 to the last but one.

TAPE = *BY-OPERATOR

This specification is not permitted in robot operation.

The operator assigns the volumes when responding to the mount request.

TAPE = list-poss(255): <vsn 1..6>

The designation and order of the volumes is determined by a list. The volumes are requested in the specified order.

The first VSN must identify the first backup volume; otherwise the RELOAD-DISK run cannot be started. The first backup volume uniquely defines the save generation. FDDRL ensures that the continuation volumes are from the same save generation. If the continuation volumes are specified in the wrong order, FDDRL rejects each incorrect volume with message FBD2307 and requests the correct one.

In the mount process, the volumes specified here are requested. No reassignments to volumes with another VSN are permitted. If the operator rejects volumes which have been specified incorrectly, the restoration is terminated abnormally.

If not all the volumes required for restoration are specified, FDDRL uses the message FBD2300 to request the operator to enter the missing VSNs.

In the case of the disk set save entity you must ensure that the last VSN in the VSN list is requested twice. On the first occasion it is requested before the first VSN in order to read the metadata at the end of the last tape of the tape set.

TAPE-DEVICE-TYPE =

Specifies the volume type of the MTCs used for the restoration.

TAPE-DEVICE-TYPE = *BY-VOLUME-CATALOG

This specification is only possible when MAREN is being used.

MAREN determines the volume type itself. If the VSN is not specified (TAPE=*BY-OPERATOR), the default volume type set for MAREN is used here (MAREN parameter DEFAULT-DEVICE-TYPE).

TAPE-DEVICE-TYPE = <device>

Volume type of MTC on the basis of which device management is able to determine the type of device that will be required to process a tape. TAPE may not be specified.

SAVE-ENTITY = *SINGLE-DISK / *DISK-SET

Specifies whether the disk to be restored was saved as a single disk (single disk save entity) or in a disk set (disk set save entity). The default is the single disk save entity. Multiple VSNs can only be specified in the DISK operand in the case of disk sets. In this case SAVE-ENTITY=*DISK-SET must also be specified.

OVERWRITE-DISK =

Specifies the VSN of the output disk to be mounted. If restoration is successful, the output disk is given the VSN of the saved input disk (original VSN).

OVERWRITE-DISK = *SAME-VSN

The output disk to be mounted has the same VSN as the input disk that was specified under the DISK operand.

OVERWRITE-DISK = *BY-OPERATOR

The VSN of the output disk to be mounted is not known to the FDDRL user. The operator is responsible for assigning the correct VSN.

The operand value *BY-OPERATOR should not be used during normal operation, since an operator error may cause foreign disks to be overwritten inadvertently. *BY-OPERATOR is intended exclusively for cases where the VSN is not known. This situation can occur if a previous COPY-DISK, COPY-PUBSET, RELOAD-DISK or RELOAD-PUBSET run was not terminated correctly. In this case, the SVL still contains a dummy VSN which has been assigned by FDDRL (formed from the time of the FDDRL run) and which is designed to prevent BS2000 from working with this disk.

OVERWRITE-DISK = *UNIT(...)

The output disks to be mounted are specified using their mnemonic device names (MNs). The operator must confirm the disk assignment. The output disks can be used even if they are not initialized or have the wrong format. In this case they are automatically initialized and formatted in the required format (if this is possible with VOLIN).

UNIT = *SAME-UNIT

The output disks to be mounted have the same MNs as the saved original disks.

UNIT = list-poss(64): <alphanum-name 2..2> / <x-text 4..4>

Mnemonic device names (MNs) of the output disks.

When restoration takes place from a disk set, the MNs specified in a list are used in accordance with the backup sequence of the save.

OVERWRITE-DISK = list-poss(64): <vsn 1..6>

The output disks to be mounted have the specified volume serial numbers.

RELOAD-PUBSET Restore a pubset

This statement enables you to restore a complete pubset which was saved using the DUMP-PUBSET statement and SAVE-ENTITY=*DISK-SET.

During the job creation phase, the information about all disk sets in the pubset is read from the first volume of the first disk set and output to SYSOUT. The information about the pubset configuration and the main properties of the individual disks enables you to prepare the output disks required to restore the pubset.

Note

If OVERWRITE-DISK=*BY-OPERATOR is specified, message FBD1251 is issued, asking for confirmation before the disk is overwritten.

RELOAD-PUBSETAlias: **RLP****PUBSET** = <catid 1..4>**,TAPE** = ***BY-VOLUME-CATALOG(...)** / ***BY-OPERATOR** / ***GROUPED-BY-DISK-SET(...)** /

list-poss(255): <vsn 1..6>

BY-VOLUME-CATALOG(...)**| **FILE-NAME** = ***STD(...)** / <filename 1..41 without-cat-user-gen-vers>| ***STD(...)**| | **DATE** = <date with-compl>| | **,TIME** = <time>GROUPED-BY-DISK-SET(...)**| **SET-1-TAPE** = list-poss(255): <vsn 1..6>| **,SET-2-TAPE** = ***NONE** / list-poss(255): <vsn 1..6>| **,SET-3-TAPE** = ***NONE** / list-poss(255): <vsn 1..6>| **,SET-4-TAPE** = ***NONE** / list-poss(255): <vsn 1..6>| **,SET-5-TAPE** = ***NONE** / list-poss(255): <vsn 1..6>| **,SET-6-TAPE** = ***NONE** / list-poss(255): <vsn 1..6>| **,SET-7-TAPE** = ***NONE** / list-poss(255): <vsn 1..6>| **,SET-8-TAPE** = ***NONE** / list-poss(255): <vsn 1..6>**,TAPE-DEVICE-TYPE** = ***BY-VOLUME-CATALOG** / <device>**,OVERWRITE-DISK** = ***SAME-VSN** / ***BY-OPERATOR** / ***UNIT(...)** / list-poss(1000): <vsn 1..6>***UNIT(...)**| **UNIT** = ***SAME-UNIT** / list-poss(1000): <alphanum-name 2..2> / <x-text 4..4>**PUBSET** = <catid 1..4>

Catalog ID of the pubset to be restored.

TAPE =

Specifies the volumes to be used for the restoration.

TAPE = ***BY-VOLUME-CATALOG(...)***This specification is only possible when MAREN is being used.*

The volumes are to be determined automatically by MAREN. The file name in the volume catalog is required for this purpose.

FILE-NAME = *STD(...) / <filename 1..41 without-cat-user-gen-vers> Designates the file name of the disk set with the number 1 in the MAREN volume catalog.

FILE-NAME = *STD(...)

The following default file name was assigned for the first disk set in the save: FDDRL.P.<catid>.<date>.<time>-1

DATE = <date with-compl>

Date of the save (format yyyy-mm-dd).

TIME = <time>

Time of the save (format hh:mm). By default the seconds are 0.

FILE-NAME = <filename 1..41 without-cat-user-gen-vers>

File name which was assigned during the save. If the pubset was saved using multiple disk sets (NUMBER-OF-DISK-SETS > 1), the name of the first disk set must be specified here. The name must end with "1" so that FDDRL can form the file names of the further disk sets correctly.

TAPE = *BY-OPERATOR

This specification is not permitted in robot operation.

The operator assigns the volumes when responding to the mount request.

TAPE = *GROUPED-BY-DISK-SET(...)

If multiple disk sets are created in the save, the volumes are grouped into disk sets when they are restored. The order of the groups must match the order of the disk sets in the save.

SET-1-TAPE = list-poss(255): <vsn 1..6>

Volumes of the first disk set. The designation and order of the volumes is determined by a list. The volumes are requested in the specified order. For further details, see also [TAPE = list-poss\(255\):](#) .

SET-2-TAPE = *NONE / list-poss(255): <vsn 1..6>

Volumes of the second disk set if one was used. The default is *NONE, i.e. no further disk set exists.

For further details, see also [TAPE = list-poss\(255\):](#) .

If further disk sets were created when the pubset was saved, the volumes of these disk sets are specified accordingly in operands SET-3-TAPE through SET-8-TAPE.

TAPE = list-poss(255): <vsn 1..6>

This specification is only possible when only one disk set was created during the save (NUMBER-OF-DISK-SETS=1).

Volumes of the disk set. The designation and order of the volumes is determined by a list. The volumes are requested in the specified order.

The first VSN must identify the first backup volume; otherwise the RELOAD-PBSET run cannot be started. The first backup volume uniquely defines the save generation. FDDRL ensures that the continuation volumes are from the same save generation. If the continuation volumes are specified in the wrong order, FDDRL rejects each incorrect volume with message FBD2307 and requests the correct one.

In the mount process, the volumes specified here are requested. No reassignments to volumes with another VSN are permitted. If the operator rejects volumes which have been specified incorrectly, the restoration is terminated abnormally.

If not all the volumes required for restoration are specified, FDDRL uses the message FBD2300 to request the operator to enter the missing VSNs.

TAPE-DEVICE-TYPE =

Specifies the volume type of the MTCs used for the restoration.

TAPE-DEVICE-TYPE = *BY-VOLUME-CATALOG

This specification is only possible when MAREN is being used.

MAREN determines the volume type itself. If the VSN is not specified (TAPE=*BY-OPERATOR), the default volume type set for MAREN is used here (MAREN parameter DEFAULT-DEVICE-TYPE).

TAPE-DEVICE-TYPE = <device>

Volume type of MTC on the basis of which device management is able to determine the type of device that will be required to process a tape. TAPE may not be specified.

OVERWRITE-DISK =

Determines the VSNs of the output disks to be mounted. Following successful restoration, each output disk is assigned the VSN of the corresponding saved input disk (original VSN).

OVERWRITE-DISK = *SAME-VSN

The output disks to be mounted have the same VSNs as the corresponding input disks.

OVERWRITE-DISK = *BY-OPERATOR

The operator should specify the VSNs of the output disks to be mounted in the device assignment. The operator is responsible for assigning the correct VSN.

The operand value *BY-OPERATOR should not be used during normal operation, since an operator error may cause foreign disks to be overwritten inadvertently. *BY-OPERATOR is intended exclusively for cases where the VSN is not known. This situation can occur if a previous COPY-DISK, COPY-PUBSET, RELOAD-DISK or RELOAD-PUBSET run was not terminated correctly. In this case, the SVL still contains a dummy VSN which has been assigned by FDDRL (formed from the time of the FDDRL run) and which is designed to prevent BS2000 from working with this disk.

OVERWRITE-DISK = *UNIT(...)

The output disks to be mounted are specified using their mnemonic device names (MNs). The operator must confirm the disk assignment. The output disks can be used even if they are not initialized or have the wrong format. In this case they are automatically initialized and formatted in the required format (if this is possible with VOLIN).

UNIT = *SAME-UNIT

The output disks to be mounted have the same MNs as the saved original disks.

UNIT = list-poss(1000): <alphanum-name 2..2> / <x-text 4..4>

Mnemonic device names (MNs) of the output disks.

The MNs specified are used for the disks in accordance with the backup sequence:

- In an SF pubset the disks are saved in the alphabetical order of the VSNs.
- In an SM pubset first the disks of the control volume set and then those of the other volume sets are saved in the alphabetical order of the volume set IDs.

OVERWRITE-DISK = list-poss(1000): <vsn 1..6>

The output disks to be mounted have the specified volume serial numbers.

SHOW-FDDRL-PARAMETERS Show the current FDDRL parameters

This statement is executed immediately under the FDDRL main task and outputs the FDDRL parameters which are currently set to SYSOUT or SYSLST. You can select the output medium with the OUTPUT operand. The MODIFY-FDDRL-PARAMETERS statement enables you to change parameter values for the FDDRL run.

After FDDRL has started, the default values or, if they exist, the values in the FDDRL parameter file adjusted on a system-specific basis are displayed.

SHOW-FDDRL-PARAMETERS	Alias: SFP
OUTPUT = <u>*SYSOUT</u> / *SYSLST(...) / *NONE	
*SYSLST(...)	
SYSLST-NUMBER = *STD / <integer 1..99>	
,STRUCTURE-OUTPUT = <u>*SYSINF</u> / <structured-name 1..20>(…) / *NONE	
<structured-name 1..20>(…)	
WRITE-MODE = <u>*REPLACE</u> / *EXTEND	

OUTPUT =

Defines the output medium for the text output of reports and SHOW statements.

OUTPUT = *SYSOUT

Text output takes place to system file SYSOUT.

OUTPUT = *SYSLST(...)

Text output takes place to system file SYSLST.

SYSLST-NUMBER= *STD / <integer 1..99>

Defines whether the output is to take place to system file SYSLST or to a SYSLST file from the set SYSLST01 through SYSLST99. The default value is *STD, i.e. the output takes place to system file SYSLST.

OUTPUT = *NONE

No text output takes place.

STRUCTURE-OUTPUT =

Controls structured output of reports and SHOW statements to S variables.

STRUCTURE-OUTPUT = *SYSINF

Structured output is directed to the S variable stream SYSINF.

STRUCTURE-OUTPUT = <structured-name 1..20> (...)

Name of the S variable which is to be used for structured output.

WRITE-MODE =

Defines whether the output should replace or complement the current content of the S variable.

WRITE-MODE = *REPLACE

The current content of the S variable is replaced by the new output.

WRITE-MODE = *EXTEND

The new output extends the current content of the S variable.

STRUCTURE-OUTPUT = *NONE

No structured output takes place.

Output information

The table below shows the fields output and their meaning. The default values of the FDDRL parameters are also specified:

Field name	Parameter name in the MODIFY-FDDRL-PARAMETERS statement	Preset value when FDDRL is called
PROCESS-JOBS	PROCESS-JOBS	*BY-SUBTASK(...)
TASK-LIM	TASK-LIMIT	4
RUN-PRIO	RUN-PRIORITY	*STD
STA-OUTPUT-PERIOD	STATUS-OUTPUT-PERIOD	5 minutes
SEC-WAIT-TIME	SECURE-WAIT-TIME	1 minute
JOB-NAME	JOB-NAME	*STD
RETENT-PERIOD	RETENTION-PERIOD	0 days
LOCATION	LOCATION	*STD
TAPE-FORM	TAPE-FORMAT	*STD
REPORT-FILE	REPORT-FILE	*STD
MONITORING	MONITORING	*NO
KEEP-REQUESTS	KEEP-REQUESTS	40 days

Table 2: Outputs of the SHOW-FDDRL-PARAMETERS statement with defaults

Typical output

/SHOW-FDDRL-PARAMETERS

```

FDDRL PARAMETERS                FDDRL V 21 .0A00        20 20 -08-11 08:32:52

PROCESS-JOBS      = *BY-SUBTASK (...)
TASK-LIM          = 4
RUN-PRIO          = *STD
STA-OUTPUT-PERIOD = 5 MINUTES
SEC-WAIT-TIME     = 1 MINUTE
JOB-NAME          = *STD
RETENT-PERIOD     = 0 DAYS
LOCATION            = *STD
TAPE-FORM         = *STD
REPORT-FILE       = *STD
MONITORING        = *NO
KEEP-REQUESTS     = 40 DAYS
  
```

Output to S variable

Output information	Name of the S variable	T	Contents
Output identification	var(*list).ID	S	FBDSFP
FDDRL version	var(*list).VERSION	S	V21.0A00
Current date	var(*list).DATE	S	<yyyy-mm-dd>
Current time	var(*list).TIME	S	<hh:mm:ss>
PROCESS-JOBS	var(*list).PROCESS-JOBS	S	*BY-SUBTASK *BY-MAINTASK
Subtask limit	var(*list).SUBTASK.TASK-LIM	S	<integer_1..16>
Subtask priority (128..255)	var(*list).SUBTASK.RUN-PRIO	S	<c-string 3..3>
Period of the status output	var(*list).SUBTASK.STA-OUTPUT-PERIOD	S	<0..60 minutes>
Secure wait time	var(*list).SUBTASK.SEC-WAIT-TIME	S	<0..604800 seconds>
Subtask job name	var(*list).JOB-NAME	S	*STD *SAME <name 1..8>
Retention period of the output tapes	var(*list).RETENT-PERIOD	S	<0..32767 days>
Location of the output tapes	var(*list).LOCATION	S	*STD <c-string 1..8>

Tape format	var(*list).TAPE-FORM	S	*STD *V17-FORMAT *V16-FORMAT *V15-FORMAT
Setting for the report file	var(*list).REPORT-FILE	S	*STD *NONE
Monitoring of the requests in the case of SE servers	var(*list).MONITORING	S	*NO *YES
Automatic deletion of the requests	var(*list).KEEP-REQUESTS	S	<c-string 1..5>

SHOW-FDDRL-STATUS Show running (foreign) FDDRL jobs

The SHOW-FDDRL-STATUS statement monitors foreign FDDRL jobs. These are jobs which were started in batch mode or by other users.

SHOW-FDDRL-STATUS	Alias: SFS
JOB-IDENTIFICATION = <u>*ALL</u> / *TSN(...) / *JOB-NAME(...)	
*TSN(...) TSN=<alphanum-name 1..4>	
*JOB-NAME(...)	
JOB-NAME = <name 1..8> / *NONE)	
,OUTPUT-PERIOD = 0 / <integer 0..3600 seconds>	
,OUTPUT = *SYSOUT / *SYSLST(...) / *NONE	
*SYSLST(...)	
SYSLST-NUMBER = *STD / <integer 1..99>	
,STRUCTURE-OUTPUT = *SYSINF / <structured-name 1..20>(…) / *NONE	
<structured-name 1..20>(…)	
WRITE-MODE = *REPLACE / *EXTEND	

JOB-IDENTIFICATION =

Selects the FDDRL main task.

JOB-IDENTIFICATION = *ALL

Supplies information about the jobs of all FDDRL main tasks.

JOB-IDENTIFICATION = *TSN(TSN=<alphanum-name 1..4>)

Supplies information only about the jobs of the FDDRL main task with the specified TSN.

JOB-IDENTIFICATION = *JOB-NAME(...)

Selects the FDDRL main task according to the request name.

JOB-NAME=<name 1..8> / *NONE)

Supplies information only about the jobs of the FDDRL main tasks with the specified request name.

JOB-NAME=<name 1..8> / *NONE)

Supplies information only about the jobs of the FDDRL main tasks without a request name.

OUTPUT-PERIOD =

Specifies whether and at what time intervals the text output requested using the OUTPUT operand is to be repeated.

The output to S variable requested with the STRUCTURE-OUTPUT operand can only take place once.

OUTPUT-PERIOD = 0

The output is performed once only and not repeated.

OUTPUT-PERIOD = <integer>

The output is repeated periodically after the specified number of seconds.

You can abort the output with //HOLD-PROGRAM (K2) and /INFORM-PROGRAM. If no further FDDRL jobs are found, the output is automatically aborted.

OUTPUT =

Defines the output medium for the text output of reports and SHOW statements.

OUTPUT = *SYSOUT

Text output takes place to system file SYSOUT.

OUTPUT = *SYSLST(...)

Text output takes place to system file SYSLST.

SYSLST-NUMBER= *STD / <integer 1..99>

Defines whether the output is to take place to system file SYSLST or to a SYSLST file from the set SYSLST01 through SYSLST99. The default value is *STD, i.e. the output takes place to system file SYSLST.

OUTPUT = *NONE

No text output takes place.

STRUCTURE-OUTPUT =

Controls structured output of reports and SHOW statements to S variables.

STRUCTURE-OUTPUT = *SYSINF

Structured output is directed to the S variable stream SYSINF.

STRUCTURE-OUTPUT = <structured-name 1..20> (...)

Name of the S variable which is to be used for structured output.

WRITE-MODE =

Defines whether the output should replace or complement the current content of the S variable.

WRITE-MODE = *REPLACE

The current content of the S variable is replaced by the new output.

WRITE-MODE = *EXTEND

The new output extends the current content of the S variable.

STRUCTURE-OUTPUT = *NONE

No structured output takes place.

Output information

The SHOW-FDDRL-STATUS statement outputs the following information:

Field name	Meaning
REQ-NAME	Request name
TSN	TSN of the main task or subtask
STATEMENT	FDDRL statement
OBJECT-NAME	Catalog ID or VSN or MN of the (first) disk

DISK/-SET	VSN of the disk of the foreign disk or name of the disk set
FORM	Formatting of the output disk.
JOB-STA	Job status (INACTIVE, ACTIVE, TERM, FORM)
PROCESS-HPS	Number of disk blocks already processed (HP: physical half-page, 2KB)
%	Processing progress in percent
TERM-MSG	FDDRL termination message in the event of errors

Table 3: Outputs of the SHOW-FDDRL-STATUS statement

Typical output

/SHOW-FDDRL-STATUS

```

FDDRL JOB STATUS                FDDRL V21.0A00      2020-06-24 09:23:38
REQ-NAME   TSN STATEMENT          OBJECT-NAME
  DISK/-SET TSN  FORM  JOB-STA  PROCESS-HPS   %  TERM-MSG
-----
DMP-SBZ3   0N4Y DUMP-PUBSET          SBZ3
          SBZ3.0           ACTIVE      106800  10%
          SBZ3.1           ACTIVE      106800  10%
          SBZ3.2           ACTIVE      106800  10%
          SBZ3-1   0N46   ACTIVE      320400

```

Output to S variable:

Output information	Name of the S variable	T	Contents
Output identification	var(*list).ID	S	FBDSFS
Request name	var(*list).REQ-NAME	S	<c-string 5..10>
Alias of the statement	var(*list).STMT-NAME	S	CPD CPP DMD DMP RLD RLP CMD
Object type	var(*list).OBJECT-TYPE	S	PUBSET DISK UNIT
Object name: <catid> <vsn> / (<vsn>,...) <mn> / (<mn>,...)	var(*list).OBJECT-NAME	S	<string 1..12>

Number of disks	var(*list).NUM-OF-DISKS	S	1..n
Number of disk sets	var(*list).NUM-OF-DISK-SETS	S	0..9
is the task sequence number of the main task	var(*list).TSN	S	<c-string 4..4>
Termination message (FBDnnnn) in the event of errors	var(*list).TERM-MSG	S	<cstring 7..7> "
VSN of the disk	var(*list).DISK(*list).DISK	S	<VSN_1..6>
MN of the input disk	var(*list).DISK(*list).INPUT-UNIT	S	<c-string 2..4>
MN of the output disk	var(*list).DISK(*list).OUTPUT-UNIT	S	<c-string 2..4>
TSN of the processing task (main task / subtask)	var(*list).DISK(*list).TSN	S	<c-string 4..4>
Job status	var(*list).DISK(*list).JOB-STA	S	INACTIVE ACTIVE TERM FORM
Save scope	var(*list).DISK(*list).SEL-HPS	S	<c-string 0..10>
Number of processed HPs	var(*list).DISK(*list).PROCESS-HPS	S	<c-string 0..10>
Processing progress [%]	var(*list).DISK(*list).PROCESS-PERCENT	S	<integer 0..100>
Termination message in the event of errors	var(*list).DISK(*list).TERM-MSG	S	<c-string 7..7> "
Disk set name	var(*list).DISK-SET(*list).DISK-SET	S	<c-string 3..6>
TSN of the processing task (main task / subtask)	var(*list).DISK-SET(*list).TSN	S	<tsn>
Job status	var(*list).DISK-SET(*list).JOB-STA	S	INACTIVE ACTIVE TERM
Number of processed HPs	var(*list).DISK-SET(*list).PROCESS-HPS	S	<c-string 0..10>
Termination message in the event of errors	var(*list).DISK-SET(*list).TERM-MSG	S	<c-string 7..7> "
TSN of the task for monitoring the online save	var(*list).OLSC.TSN	S	<c-string 4..4> "
Remaining time for the online save	var(*list).OLSC.REMAINING-TIME	S	<0..7200 seconds> "

Termination message in the event of errors

var(*list).OLSC.TERM-MSG

S

<c-string 7..7>
”

SHOW-FDDRL-TAPE-INFORMATION Show the tape information of an FDDRL tape

This statement enables you to display the tape information of an FDDRL tape.

This comprises information about the tape, the FDDRL run, the saved disk and its properties. This means that users can obtain information on the tape save and can mount the disks required for RELOAD-DISK (device type, format, capacity).

With DUMP-DISK detailed information is output explaining why the tape was rejected.

i The information is also displayed automatically when a tape was rejected in the FDDRL function RELOAD-DISK or DUMP-DISK.

The statement is executed directly after the entry under the FDDRL main task.

If the specified tape is a continuation tape, the search for the meta information can take a long time. In this case you can if necessary cancel execution with K2/BREAK.

SHOW-FDDRL-TAPE-INFORMATION	Alias: SFT
TAPE = <vsn 1..6>	
,TAPE-DEVICE-TYPE = <u>*BY-VOLUME-CATALOG</u> / <device>	
,OUTPUT = <u>*SYSOUT</u> / <u>*SYSLST(...)</u> / *NONE	
 *SYSLST(...)	
 SYSLST-NUMBER = <u>*STD</u> / <integer 1..99>	
,STRUCTURE-OUTPUT = <u>*SYSINF</u> / <structured-name 1..20>(…) / *NONE	
 <structured-name 1..20>(…)	
 WRITE-MODE = <u>*REPLACE</u> / <u>*EXTEND</u>	

TAPE = <vsn 1..6>

Specifies the MTC (VSN) from which the tape information is to be read.

TAPE-DEVICE-TYPE = *BY-VOLUME-CATALOG / <device>

Defines the volume type of the tape from which the tape information is to be read.

TAPE-DEVICE-TYPE = *BY-VOLUME-CATALOG

This specification is only possible when MAREN is being used.

MAREN determines the volume type itself. When TAPE=*BY-OPERATOR is specified, this is the default volume type set for MAREN (MAREN parameter DEFAULT-DEVICE-TYPE).

TAPE-DEVICE-TYPE = <device>

Volume type of the MTC on the basis of which device management determines the type of device that will be required for tape processing. TAPE may not be specified.

OUTPUT =

Defines the output medium for the text output of reports and SHOW statements.

OUTPUT = *SYSOUT

Text output takes place to system file SYSOUT.

OUTPUT = *SYSLST(...)

Text output takes place to system file SYSLST.

SYSLST-NUMBER= *STD / <integer 1..99>

Defines whether the output is to take place to system file SYSLST or to a SYSLST file from the set SYSLST01 through SYSLST99. The default value is *STD, i.e. the output takes place to system file SYSLST.

OUTPUT = *NONE

No text output takes place.

STRUCTURE-OUTPUT =

Controls structured output of reports and SHOW statements to S variables.

STRUCTURE-OUTPUT = *SYSINF

Structured output is directed to the S variable stream SYSINF.

STRUCTURE-OUTPUT = <structured-name 1..20> (...)

Name of the S variable which is to be used for structured output.

WRITE-MODE =

Defines whether the output should replace or complement the current content of the S variable.

WRITE-MODE = *REPLACE

The current content of the S variable is replaced by the new output.

WRITE-MODE = *EXTEND

The new output extends the current content of the S variable.

STRUCTURE-OUTPUT = *NONE

No structured output takes place.

Output information

The tables below shows the fields output and their meaning:

1. Tape data

TAPE VSN	Volume serial number (VSN) of the MTC
VOL-SEQ	Tape sequence number of a tape set
FIRST-VOL	Volume serial number of the first tape of a tape set
CRE-DATE	Creation date of the tape
EXPIR-DATE	Expiration date of the tape

USER-ID	User ID of the tape (with FDDRL, always TSOS)
F-NAME	Name of the tape file; provides information about its type (FDDRL, ARCHIVE, DMS)

2. FDDRL Data

FDDRL	FDDRL version of the save run
TAPE-FORM	FDDRL tape format (compatible from the relevant FDDRL version)
SAVE-DATE	Date and time of the FDDRL save (start time)
PUBSET	Catalog ID of the SF or SM pubset Only set for saves which were generated with DUMP-PUBSET.
DISK	VSN of the disk. Only supplied here in the case of single disk backup.
DISK-DATE	Time stamp of the pubset / single disk
SAVE-ENT	SAVE-ENTITY (*SINGLE-DISK, *DISK-SET)
DISK-SET	Name of the disk set of this tape
NUM-OF-DISK-SETS	Number of disk sets generated (with DUMP-PUBSET)

3. Disk properties

DISK	VSN of the disk
UNIT	Mnemonic device name (MN) of the disk
DEV-TYPE	Device type of the disk If the device type is no longer known, the TSOS type is output instead (e.g. A600)
FORM	Disk format (KEY, NK2, NK4),
A-U	Allocation unit (A-U: 6, 8, 64).
NUM-OF-HPS	Disk capacity (half-pages, last usable PHP number) or, in the case of old saves of CKD disks, the address of the alternative track area (format: CCHH and the indicator '*T'). With disks of the type D3435, the capacity is also output in the number of 512-byte blocks.
NUM-OF-512B	With disks of the type D3435, the capacity is output in the number of 512-byte sectors.
SAVED-HPS	Number of saved half-pages (2-KB units of the disk)

If it is available, the tape data is output for every tape. The FDDRL data and disk properties are only output in the case of an FDDRL backup tape. In other cases the text 'NO FDDRL TAPE' is output.

Typical outputs

*SAVE-ENTITY = *DISK-SET(...)*

```
FDDRL TAPE INFORMATION          FDDRL V21.0A00          2020-08-27 11:58:50
TAPE VSN      = QA3093      VOL-SEQ      = 0001          FIRST-VOL    = QA3093
  CRE-DATE    = 2020-08-27 EXPIR-DATE = 2020-08-27 USER-ID    = TSOS
  F-NAME      = FDDRL.P.B10S.2020-08-27.063900-1
FDDRL = V21.0A  TAPE-FORM = *V16-FORMAT SAVE-DATE = 2020-08-27 06:39:32
  PUBSET      = B10S      DISK      =          DISK-DATE = 2020-08-19 10:01:18
  SAVE-ENT    = *DISK-SET DISK-SET = B10S-1  NUM-OF-DISK-SETS = 3
DISK  UNIT DEV-TYPE FORM A-U NUM-OF-HPS NUM-OF-512B SAVED-HPS
-----
B10S.0 900E D3435  NK2  6      1966080      7864320      1345360
B10S.1 900F D3435  NK2  6      1966080      7864320      1330240
B10S.2 9068 D3435  NK2  6      1966080      7864320      1162880
B10S.3 9069 D3435  NK2  6      1966080      7864320      1171360
B10S.4 906A D3435  NK2  6      1966080      7864320      1119600
B10S.5 906B D3435  NK2  6      1966080      7864320      1123680
```

*SAVE-ENTITY = *SINGLE-DISK*

```
FDDRL TAPE INFORMATION          FDDRL V21.0A00          2020-08-27 11:59:25
TAPE VSN      = QA3096      VOL-SEQ      = 0001          FIRST-VOL    = QA3096
  CRE-DATE    = 2020-08-27 EXPIR-DATE = 2020-08-27 USER-ID    = TSOS
  F-NAME      = FDDRL.D.ABN4.0
FDDRL = V21.0A  TAPE-FORM = *V15-FORMAT SAVE-DATE = 2020-08-27 10:26:53
  PUBSET      =          DISK      = ABN4.0      DISK-DATE = 2020-08-20 08:46:23
  SAVE-ENT    = *SINGLE-DISK
DISK  UNIT DEV-TYPE FORM A-U NUM-OF-HPS NUM-OF-512B SAVED-HPS
-----
ABN4.0 591E D3435  NK2  6      1843200      7372800      113840
```

Output to S variable

Output information	Name of the S variable	T	Contents
Output identification	var(*list).ID	S	FBDSFT
VSN of the MTC	var(*list).VOL	S	<c-string 1..6>
Tape sequence number	var(*list).VOL-SEQ	S	<c-string 4>
1st volume of a tape set	var(*list).FIRST-VOL	S	<c-string 1..6>
Creation date	var(*list).CRE-DATE	S	<c-string 10>
Expiration date	var(*list).EXPIR-DATE	S	<c-string 10>
Length of the tape blocks	var(*list).BLOCK-LEN	S	<c-string 5..6>
Owner of the MTC (TSOS)	var(*list).USER-ID	S	<c-string 1..8>
Tape file name	var(*list).F-NAME	S	<c-string ..44>
FDDRL tape format or NONE if not an FDDRL tape	var(*list).TAPE-FORM	S	*V17-FORMAT *V16-FORMAT *V15-FORMAT NONE
Save date	var(*list).SAVE-DATE	S	<yyyy-mm-dd> "
Save time	var(*list).SAVE-TIME	S	<hh:mm:ss> "
Catid of the pubset	var(*list).PUBSET	S	<c-string 1..4> "
Save scope	var(*list).SEL-BLOCKS	S	*USED-BLOCKS *ALL-BLOCKS "
SAVE-ENTITY	var(*list).SAVE-ENTITY	S	*SINGLE-DISK *DISK-SET "
Name of the disk set	var(*list).DISK-SET	S	<c-string 3..8> "
Number of disk sets	var(*list).NUM-OF-DISK-SETS	S	<c-string 1..1> "
Number of disks	var(*list).NUM-OF-DISKS	S	<c-string 1..4> "
VSN of the disk	var(*list).DISK(*list).DISK	S	<c-string 1..6>

MN of the disk	var(*list).DISK(*list).UNIT	S	<c-string 2..4>
Device type of the disk	var(*list).DISK(*list).DEV-TYPE	S	<c-string 1.8>
Disk format	var(*list).DISK(*list).FORM	S	KEY NK2 NK4
Disk capacity in HPs	var(*list).DISK(*list).NUM-OF-HPS	S	<c-string 2..10>
Disk capacity in number of 512-byte sectors	var(*list).DISK(*list).NUM-OF-512B	S	<c-string 2..10> "
Saved HPs of the disk	var(*list).DISK(*list).SAVED-HPS	S	<c-string 2..10> "
Time stamp of the disk (date)	var(*list).DISK(*list).DATE	S	<yyyy-mm-dd>
Time stamp of the disk (time)	var(*list).DISK(*list).TIME	S	<hh:mm:ss> "

SHOW-REQUESTS Show running and completed FDDRL requests

The SHOW-REQUESTS statement displays the FDDRL requests which are running and those which have already been completed.

i When this statement is executed, old, completed FDDRL requests (by default older than 40 days) and the associated report files in PDF format are deleted automatically.

SHOW-REQUESTS

Kurzname: **SHR**

```
REQUEST-NAME = *ALL / <composed-name 5..10>
,REQUEST-DATE = *LAST-FDDRL-RUN / *ANY / *LAST-REQUEST-RUN / *TODAY / *YESTERDAY /
    <int -99999..0 days> / <date> / *INTERVAL(...)
*INTERVAL(...)
    | FROM = *EARLIEST-DATE / *TODAY / *YESTERDAY / <int -99999..0 days> / <date>(…)
    | <date>(…)
    | | TIME = 00:00:00 / <time>
    | ,TO = *TODAY / *YESTERDAY / <integer -99999..0 days> / <date>(…)
    | <date>(…)
    | | TIME = 23:59:59 / <time>
,STATE = *ANY / *ACCEPTED / *STARTED / *COMPLETED(...)
*COMPLETED(...)
    | SUB-STATE = *ANY / *COMPLETED-OK / *WITH-ERRORS
,SORT = *BY-DATE / *BY-NAME
,OUTPUT = *SYSOUT / *SYSLST(...) / *NONE
*SYSLST(...)
    | SYSLST-NUMBER = *STD / <integer 1..99>
,STRUCTURE-OUTPUT = *SYSINF / <structured-name 1..20>(…) / *NONE
<structured-name 1..20>(…)
    | WRITE-MODE = *REPLACE / *EXTEND
```

REQUEST-NAME =

Selects the requests according to their name.

REQUEST-NAME = *ALL

The request name is not a selection criterion.

REQUEST-NAME = <composed_name_5..10>

Information is output only for requests with the specified name.

REQUEST-DATE =

Selects the requests according to their start date.

REQUEST-DATE = *LAST-FDDRL-RUN

Information is output only for requests of the last FDDRL run (//START-JOBS).

REQUEST-DATE = *ANY

The start date is not a selection criterion.

REQUEST-DATE = *LAST-REQUEST-RUN

Information is output only for the most recent requests with the selected request name.

REQUEST-DATE = *TODAY

Information is output only for requests which were started on the current day.

REQUEST-DATE = *YESTERDAY

Information is output only for requests which were started on the previous day.

REQUEST-DATE = <int -99999..0 days >

Information is output only for requests whose start date lies the specified number of days in the past.

REQUEST-DATE = <date>

Information is output only for requests which were started on the specified date.

REQUEST-DATE = *INTERVAL(...)

Information is output only for requests whose start time lies in the specified period. Range specifications apply including the range limit values.

FROM =

Defines the lower limit of the period for the start time of the requests. Information is output only for requests which were started on the specified date or later.

FROM = *EARLIEST-DATE

Defines that the period for the start time of the requests has no lower limit.

FROM = *TODAY

Defines the date of the current date as the lower limit of the period for the start time of the requests.

FROM = *YESTERDAY

Defines the date of the previous day as the lower limit of the period for the start time of the requests.

FROM = <int -99999..0 days>

Defines the date which lies the specified number of days in the past as the lower limit of the period for the start time of the requests.

FROM = <date>(TIME = 00:00:00 / <time>)

Defines the specified date and, if required, the specified time as the lower limit of the period for the start time of the requests.

TO =

Defines the upper limit of the period for the request date of the requests. Information is output only for requests which were started on the specified date or earlier.

TO = *TODAY

Defines the date of the current date as the upper limit of the period for the start time of the requests.

TO = *YESTERDAY

Defines the date of the previous day as the upper limit of the period for the start time of the requests.

TO = <int -99999..0 days>

Defines the date which lies the specified number of days in the past as the upper limit of the period for the start time of the requests.

TO = <date>(TIME = 23:59:59 / <time>)

Defines the specified date and, if required, the specified time as the upper limit of the period for the start time of the requests.

STATE = *ANY / *ACCEPTED / *STARTED /

***COMPLETED (SUB-STATE= *ANY / *COMPLETED-OK / *WITH-ERRORS)** Selects the requests according to their request state and sub-state.

SORT =

Specifies the sort criteria for the output.

SORT = *BY-DATE

The requests are sorted according to the start date in descending order (most recent requests first).

SORT = *BY-NAME

The requests are sorted according to the request name in ascending order.

OUTPUT =

Defines the output medium for the text output of reports and SHOW statements.

OUTPUT = *SYSOUT

Text output takes place to system file SYSOUT.

OUTPUT = *SYSLST(...)

Text output takes place to system file SYSLST.

SYSLST-NUMBER= *STD / <integer 1..99>

Defines whether the output is to take place to system file SYSLST or to a SYSLST file from the set SYSLST01 through SYSLST99. The default value is *STD, i.e. the output takes place to system file SYSLST.

OUTPUT = *NONE

No text output takes place.

STRUCTURE-OUTPUT =

Controls structured output of reports and SHOW statements to S variables.

STRUCTURE-OUTPUT = *SYSINF

Structured output is directed to the S variable stream SYSINF.

STRUCTURE-OUTPUT = <structured-name 1..20> (...)

Name of the S variable which is to be used for structured output.

WRITE-MODE =

Defines whether the output should replace or complement the current content of the S variable.

WRITE-MODE = *REPLACE

The current content of the S variable is replaced by the new output.

WRITE-MODE = *EXTEND

The new output extends the current content of the S variable.

STRUCTURE-OUTPUT = *NONE

No structured output takes place.

Output information

REQ-NAME	Request name
REQ-DATE	Start date of the request
REQ-TIME	Start time of the request
TSN	TSN of the executing task (subtask or main task)
TERM-TIME	Termination time of the request
STA	Request status (ACCEPTED, STARTED, COMPLETED)
SUB-STA	Substatus of the request (WITH-ERROR / ")
REPORT-FILE	Name of the report file or *NONE
START-COUNT	Total number of accepted or started requests
COMPL-COUNT	Total number of completed requests
ERR-COUNT	Total number of requests completed with an error (SUB-STA=WITH-ERROR)

Typical output

```

FDDRL REQUESTS                FDDRL V21.0A00        2020-08-27 10:48:16
REQ-NAME  REQ-DATE  REQ-TIME  TSN   TERM-TIME  STA    SUB-STA
          REPORT-FILE
-----
DMD-ABN4.0 2020-08-27 10:26:53 59N6 10:27:08 COMPLETED
           :B10S:$TSOS.S.FBD.59N6.2020-08-27.102653.DMD-ABN4.0
RLP-B10S   2020-08-27 08:35:09 59MV 08:37:03 COMPLETED WITH-ERR
           :B10S:$TSOS.S.FBD.59MV.2020-08-27.083509.RLP-B10S
RLD#WK906E 2020-08-26 14:00:54 59PF 14:01:40 COMPLETED
           :B10S:$TSOS.S.FBD.59PF.2020-08-26.140054.RLD#WK906E
CPD-WK906C 2020-08-26 09:45:39 59MV 09:46:55 COMPLETED
           :B10S:$TSOS.S.FBD.59MV.2020-08-26.094539.CPD-WK906C
START-COUNT: 0      COMPL-COUNT: 4      ERR-COUNT: 1

```

Output to S variable

Output information	Name of the S variable	T	Contents
Output identification	var(*list).ID	S	FBDSHR
Request name	var(*list).REQ-NAME	S	<c-string 5..10>
Alias of the statement	var(*list).STMT-NAME	S	CPD CPP DMD DMP RLD RLP
Object type	var(*list).OBJECT-TYPE	S	PUBSET DISK UNIT
Object name	var(*list).OBJECT-NAME	S	<catid> <vsn> (<vsn>,...) <mn> (<mn>,...)
Start date	var(*list).REQ-DATE	S	<yyyy-mm-dd>
Start time	var(*list).REQ-TIME	S	<hh:mm:ss>
Termination date	var(*list).TERM-DATE	S	<yyyy-mm-dd>
Termination time	var(*list).TERM-TIME	S	<hh:mm:ss>
is the task sequence number of the main task	var(*list).TSN	S	<c-string 1..4>
Status	var(*list).STA	S	ACCEPTED STARTED COMPLETED
Substatus	var(*list).SUB-STA	S	WITH-ERRORS "
Name of the report file	var(*list).REPORT-FILE	S	<c-string_43..54> *NONE

START-JOBS Start FDDRL jobs

The START-JOBS statement starts the generated FDDRL jobs.

If FDDRL jobs have not been created explicitly for all the statements entered with CREATE-JOBS, the FDDRL jobs are created implicitly before they are started (see the description of "[CREATE-JOBS Create FDDRL jobs](#)"). If errors occur during this implicit job generation, all the jobs which have been created without error are started anyway.

Depending on the value of the PROCESS-JOBS parameter in the MODIFY-FDDRL-PARAMETERS statement, they are either executed under their own task (FDDRL main task) or under FDDRL subtasks. If the FDDRL jobs are executed under FDDRL subtasks, you must note the following:

- The FDDRL subtasks are started with the parameters that were set with via the MODIFY-FDDRL-PARAMETERS statement (TASK-LIMIT, RUN-PRIORITY).
- While the FDDRL subtasks are running, the status of the FDDRL jobs is periodically output. The output period depends on the value of the STATUS-OUTPUT-PERIOD parameter in the MODIFY-FDDRL-PARAMETERS statement.
- If volumes in the MAREN catalog are protected by passwords, the passwords must be added to the password list with the /ADD-PASSWORD command before the START-JOBS statement (see the "Utility Routines" manual [2]). The password list is copied into the FDDRL subtasks.

When the FDDRL jobs have terminated, an execution log is output (see "[Results log of the FDDRL jobs](#)"). Only after this log is output can you enter further FDDRL statements or END.

If at least 1 FDDRL job was terminated abnormally, message FBD1156 is output and in procedure mode the spin-off mechanism is activated. If additional FDDRL statements are to be executed in procedure mode, the START-JOBS statement must be followed by a STEP statement.

Until the START-JOBS statement you can enter statements for up to 5000 FDDRL jobs.

START-JOBS	Alias: STJ
OUTPUT = <u>*SYSOUT</u> / *SYSLST(...) / *NONE *SYSLST(...) SYSLST-NUMBER = *STD / <integer 1..99> ,STRUCTURE-OUTPUT = <u>*SYSINF</u> / <structured-name 1..20>(…) / *NONE <structured-name 1..20>(…) WRITE-MODE = <u>*REPLACE</u> / *EXTEND	

OUTPUT =

Defines the output medium for the text output of reports and SHOW statements.

OUTPUT = *SYSOUT

Text output takes place to system file SYSOUT.

OUTPUT = *SYSLST(...)

Text output takes place to system file SYSLST.

SYSLST-NUMBER= *STD / <integer 1..99>

Defines whether the output is to take place to system file SYSLST or to a SYSLST file from the set SYSLST01 through SYSLST99. The default value is *STD, i.e. the output takes place to system file SYSLST.

OUTPUT = *NONE

No text output takes place.

STRUCTURE-OUTPUT =

Controls structured output of reports and SHOW statements to S variables.

STRUCTURE-OUTPUT = *SYSINF

Structured output is directed to the S variable stream SYSINF.

STRUCTURE-OUTPUT = <structured-name 1..20> (...)

Name of the S variable which is to be used for structured output.

WRITE-MODE =

Defines whether the output should replace or complement the current content of the S variable.

WRITE-MODE = *REPLACE

The current content of the S variable is replaced by the new output.

WRITE-MODE = *EXTEND

The new output extends the current content of the S variable.

STRUCTURE-OUTPUT = *NONE

No structured output takes place.

Duplicating FDDRL backup volumes with PERCON

Backup MTCs generated by means of FDDRL can be logically copied with the aid of the software product PERCON (see also the “PERCON” manual [11]). Only users with the TSOS privilege can copy FDDRL tapes.

Any PERCON version can be used. When you copy FDDRL files, you can also reset the expiration date to protect the volume from being overwritten.

The internal structure of the backup volumes is not affected. The result of a subsequent restoration run is the same regardless of whether the original backup MTCs or those generated by PERCON are specified in the RELOAD-DISK statement.

Assigning a TASKLIB

FDDRL backup MTCs include certain user labels (UHL, UTL). The special processing routines NDRLEIN and NDRLEOUT are provided in the FDDRL module library (logical name SYSLNK, see also [chapter "Scope of supply of FDDRL"](#)) to enable the user labels to be transferred correctly to the output volumes when the backup MTCs are copied with the aid of PERCON. This library must be defined as TASKLIB before PERCON is called:

```
/SET-TASKLIB LIBRARY=SYSLNK.FDDRL.210
```

i If it is certain that only one input volume and one output volume have to be assigned, TASKLIB assignment is not necessary. In this case, the backup MTCs can be copied without processing routines. However, it is always safer to duplicate the backup MTCs with a TASKLIB assignment (as shown in the example below).

Name of the output file

FDDRL saves a disk to a series of tapes. It produces a tape file whose name begins with the string “FDDRL“. A check is made for this string when the contents of a disk are restored. For this reason the name of the output file specified for copying with PERCON must begin with this string.

See also the description of PERCON in the “PERCON” manual [11].

Example

Converting an FDDRL save from MTC (TAPE-C2) to MTC (TAPE-C4)

```
/ SET-TASKLIB -                               -- (1)
/   LIBRARY=SYSLNK.FDDRL.210

/ IMPORT-FILE -                               -- (2)
/   SUPPORT=*TAPE                             |
/   (FILE-NAME=FDDRL.D.DISK01, -             |
/   VOLUME=(TAP001,TAP002,TAP003), -         |
/   DEVICE-TYPE=TAPE-C2)                     |
/ ADD-FILE-LINK -                             |
/   LINK-NAME=PCIN, -                         |
/   FILE-NAME=FDDRL.D.DISK01, -              |
/   ACCESS-METHOD=*BTAM                    |
                                           -- (2)
```

```

/ CREATE-FILE -                                     --+( 3 )
/   FILE-NAME=PCOUT, -                             |
/   SUPPORT=*TAPE -                               |
/     (VOLUME=(CAS001,CAS002), -                 |
/     DEVICE-TYPE=TAPE-C4) )                     |
/ ADD-FILE-LINK -                                  |
/   LINK-NAME=PCOUT, -                             |
/   FILE-NAME=FDDRL.D.DISK01.COPY, -            |
/   ,RETENTION-PERIOD=365                         --+( 3 )

/ START-PERCON                                     --( 4 )

//ASSIGN-INPUT-FILE -                             --+( 5 )
//   FILE=*TAPE-FILE(NAME=*STD, -                |
//   LABEL-EXIT=*MODULE -                       |
//   (NDRLEIN,CONTROLLED-LABEL=(HDR,EOV,END)) ) |
//ASSIGN-OUTPUT-FILE -                           |
//   FILE=*TAPE-FILE(NAME=*STD, -                |
//   FILE-ATTRIBUTES=*INPUT-FILE, -             |
//   LABEL-EXIT=*MODULE -                       --+( 5 )
//   (NAME=NDRLEOUT,CONTROLLED-LABEL=(HDR,EOV,END)) )
//END                                             --( 6 )

```

- (1) The object module library (OML) for dynamic loading is assigned as TASKLIB.
- (2) The input file is defined.
- (3) The output file is defined. The output MTCs are to be protected against overwriting for 365 days.
- (4) PERCON is called via START-PERCON.
- (5) PERCON statements are used to assign the input and output files.
- (6) END initiates data transfer.

Notes

- Once the copying has been successfully completed, the volume entries in MAREN can be adjusted (see [chapter "Adjusting volume characteristics of PERCON copies"](#)).
- PERCON copies of FDDRL save entities which used more than one input or output volume per disk or disk set can only be reloaded later if the label exits were activated in the PERCON run (see example).
- The version of the TASKLIB used should be higher than or equal to the FDDRL version used in DUMP-DISK or DUMP-PUBSET.
- FDDRL backup MTCs which have been copied in this way can also be reloaded if the number or type of output MTCs differs from the number or type of input MTCs or if the MTCs contain compressed backup data.
- The name of the output file must begin with "FDDRL.", otherwise the tapes will not be recognized as FDDRL tapes in the case of RELOAD function.
- In the copy of a pubset save with multiple disk sets, a uniform name with correctly assigned disk set numbers must be created as a suffix for all the tape sets (see statement, ["RELOAD-PUBSET Restore a pubset"](#)).

Administering the volumes with MAREN

If you are using the software product “MAREN“, MAREN administers all FDDRL MTCs.

If the software product MAREN is not available, the functionality of FDDRL is restricted:

- The tape cannot be determined automatically. `TAPE=*BY-VOLUME-CATALOG` can consequently not be specified.
- The volume type cannot be determined automatically. `TAPE-DEVICE-TYPE=*BY-VOLUME-CATALOG` can consequently not be specified.

If the saves were created using MAREN, restoration of the backup volumes is also possible without MAREN in the event of a disaster.

Table of Contents

- [MAREN catalog \(FDDRL V21.0A 2021-06, #84\)](#)
- [MAREN tape set \(FDDRL V21.0A 2021-06, #85\)](#)
- [Creating an overview of the FDDRL backup volumes \(FDDRL V21.0A 2021-06, #86\)](#)
- [Subsequently modifying the expiration date of a backup volume \(FDDRL V21.0A 2021-06, #87\)](#)
- [Adjusting volume characteristics of earlier FDDRL versions \(FDDRL V21.0A 2021-06, #88\)](#)
- [Adjusting volume characteristics of PERCON copies \(FDDRL V21.0A 2021-06, #89\)](#)
- [Defining a location \(FDDRL V21.0A 2021-06, #90\)](#)

MAREN catalog

MAREN administers all MTCs in the MAREN catalog (see the “MAREN” manual [10]). As soon the statements are entered, MAREN checks the specified volumes. The volumes, which are specified via their VSNs, must be entered in the MAREN catalog. If volumes are to be used which are not in this catalog, the `FOREIGN-TAPE-CHECK` parameter in the `//MODIFY-MAREN-PARAMETERS` statement must be set to `*NO` by means of the MARENADM administration program.

All the specified volumes of an FDDRL function must be either cataloged or not cataloged. If both cataloged and non-cataloged volumes are specified together in a list of VSNs, the first unknown volume will be rejected.

If the volume type is specified directly, the specification must match the entry in the MAREN catalog in the case of input volumes (RELOAD function). In the case of output volumes, the values need not be identical (e.g. T-C4 instead of T-C2), which means that the volume type may be switched.

If volumes in the MAREN catalog are protected by passwords, the passwords must be added to the password list with the `/ADD-PASSWORD` command before the statement with the VSN specification, but at the latest before the `//START-JOBS` statement (see the “Commands” manual [9]).

When MAREN is used, in particular the following archive entries for the volumes are updated in the MAREN catalog:

- CREATION-CATALOG-ID (catalog ID of the home pubset)
- CREATION-USER-ID
- CREATION-DATE
- CREATION-TIME
- CREATION-JOB-NAME
- OPEN-MODE
- RESERVATION-DATE
- USER-ACCESS
- PASSWORD(=*NONE)
- VOLUME-SEQUENCE (sequence of the volumes in the tape set)
- FIRST-VOLUME (VSN of the first volume in the tape set)
- FILE-NAME (see [chapter "MAREN tape set"](#))
- CLOSE-INDICATOR (with valid backup volumes CLOSED)
- LAST-CLOSE-DATE
- LAST-CLOSE-TIME
- EXPIRATION-DATE (calculated from the FDDRL RETENTION-PERIOD)

MAREN tape set

If more than one volume is required for a save entity, the volumes are combined to form a so-called MAREN tape set. The volumes of a tape set are identified by the following characteristics:

- They all have the same file name (FILE-NAME).
- For all volumes, the FIRST-VOLUME field contains the VSN of the first volume in the tape set.
- The sequence of the volumes in the tape set is based on ascending VOLUME-SEQUENCE (0001 – 0255) and ascending CREATION-DATE and CREATION-TIME.

At the end of a save, all volumes of the tape set are marked as valid with the CLOSE-INDICATOR=CLOSED attribute. Here the number of tape blocks that are written are transferred to MAREN as well. The first volume of a tape set is marked as valid only after the save has been completed successfully. The first volume of a tape set consequently indicates the successful creation of all volumes of a tape set.

MAREN file name during save

By default, FDDRL forms the file name (FILE-NAME) in the MAREN catalog as follows:

- For the single disk save entity
 - **FDDRL.D.<vsn_1..6>** in the case of //DUMP-DISK
 - **FDDRL.P.<catid_1..4>.<vsn_6..6>** in the case of //DUMP-PUBSET
- For the disk set save entity
 - **FDDRL.D.<vsn_1..6>-1** in the case of //DUMP-DISK
 - **FDDRL.P.<catid_1..4>.<date>.<time>-<disk-set-no>** in the case of //DUMP-PUBSET

with the following meanings:

FDDRL.P	Prefix for the backup of a pubset disk (in the case of DUMP-PUBSET)
FDDRL.D	Prefix for the backup of single disks
<catid_1..4>	Pubset-ID, 1- bis 4-stellig (Katalogkennung)
<vsn_6..6>	VSN of a pubset disk, 6 characters
<vsn_1..6>	VSN of an individual disk, 1-6 characters
<date>	Date of the save (format yyyy-mm-dd)
<time>	Time of the save (format hhmm00)
<disk-set-no>	Number of the disk set (depends on NUMBER-OF-DISK-SETS)

Free tape allocation facility (MARENUCP) and MAREN file name (FILE-NAME)

If the TAPE=*ANY operand is specified for a save, MARENUCP reserves free volumes and assigns these. The volumes are then assigned the previously defined file names in the MAREN catalog entries.

MAREN file name during restoration

When disks are restored, the tapes required can be determined automatically using MAREN. To permit this, the user specifies the MAREN file name.

In the case of RELOAD-DISK, the save version can also be selected. In the case of RELOAD-PUBSET, the save version is selected implicitly by means of the save time (DATE and TIME operands) contained in the file name.

If the user does not specify a MAREN file name for restoration, FDDRL creates a file name. This has no relation to MAREN or the user labels, but is merely used for information purposes (in the message NKVT013 and in the job log). FDDRL creates the file name both during restoration and saving (for information on name creation and the meaning of the name parts, see [MAREN filename during save](#)), the following exceptions applying here:

- **FDDRL.P.<catid_1..4>-1** in the case of //RELOAD-PUBSET during job creation
- **FDDRL.D.<vs_n_1..6>** in the case of //RELOAD-DISK with SAVE-ENTITY=*DISK-SET during job creation
- **FDDRL.D.<vs_n_6..6>** in the case of //RELOAD-DISK with SAVE-ENTITY=*SINGLE-DISK (of a pubset save)

MAREN test mode

In order to support the MAREN test mode for FDDRL, the job name of the FDDRL main task must have the name "MARENTST". This job name is then passed on to the subtasks.

Creating an overview of the FDDRL backup volumes

MAREN enables you to create an overview of the FDDRL saves that have been made. Therefore the `//SHOW-VOLUME-ATTRIBUTES` statement is provided.

Examples

1. All backup versions of the individual disk DISK01 are to be output, sorted according to version and volume sequence:

```
//SHOW-VOLUME-ATTRIBUTES FILE-NAME=FDDRL.D.DISK01(VERSION=*ALL, -  
// TYPE-OF-VOLUMES=*VALID(CLOSE-CHECK=*YES))
```

2. The last version of the PUBA01 disk of a pubset save of single disks is to be output:

```
//SHOW-VOLUME-ATTRIBUTES FILE-NAME=FDDRL.P.A.PUBA01(VERSION=*LATEST, -  
// TYPE-OF-VOLUMES=*VALID(CLOSE-CHECK=*YES))
```

3. All volumes of the backups of the SM pubset MX are to be output:

```
//SHOW-VOLUME-ATTRIBUTES FILE-NAME=FDDRL.P.MX.*
```

4. All volumes of the disk set save of the FDL pubset which was performed on 6 March 2020 are to be output:

```
//SHOW-VOLUME-ATTRIBUTES FILE-NAME=FDDRL.P.FDL.2020-03-06.
```

Subsequently modifying the expiration date of a backup volume

MAREN allows you to modify the expiration date of a backup volume at a later stage.

Example

All volumes of a save run are to be protected against overwriting for 31 days as of today. This involves changing the expiration date:

```
//MODIFY-TAPE-SET-ATTRIBUTES -  
//  SELECT=*BY-ATTRIBUTES(FILE-NAME=FDDRL.D.WORK01,VERSION=*ALL), -  
//  PROTECTION=*PARAMETERS(EXPIRATION-DATE=31)
```

Adjusting volume characteristics of earlier FDDRL versions

In the volume catalog, you can adjust entries of backup volumes that were created with an earlier FDDRL version (FDDRL < V12.0A) to the new format.

Example

The disk WORK01 was saved to tapes TAPE01 and TAPE02 on 14.10.1993 with FDDRL V11.2. The following entries were generated in the volume catalog:

```
VOLUME=TAPE01, FILE-NAME=FDDRL.DUMP.WORK01.TAPE.01,
                VOL-SEQ=0001, FIRST-VOL=TAPE01
VOLUME=TAPE02, FILE-NAME=FDDRL.DUMP.WORK01.TAPE.02,
                VOL-SEQ=0001, FIRST-VOL=TAPE02
```

In the volume entries, the file name and the attributes which identify a MAREN tape set are modified:

```
//MODIFY-VOLUME-ATTRIBUTES -
//  VOLUME=TAPE01 -
//    (FILE-NAME=FDDRL.D.WORK01, -
//      VOLUME-SEQUENCE=1, -
//      FIRST-VOLUME=TAPE01, -
//      CREATION-DATE=1993-10-14, -
//      CREATION-TIME=*NONE, -
//      CLOSE-INDICATOR=*CLOSE)
```

and

```
//MODIFY-VOLUME-ATTRIBUTES -
//  VOLUME=TAPE02 -
//    (FILE-NAME=FDDRL.D.WORK01, -
//      VOLUME-SEQUENCE=2, -
//      FIRST-VOLUME=TAPE01, -
//      CREATION-DATE=1993-10-14, -
//      CREATION-TIME=*NONE, -
//      CLOSE-INDICATOR=*CLOSE)
```

! WARNING!

If the tape set is to be correct, the creation date and creation time of the continuation tapes must not be in descending chronological order.

If tapes from earlier FDDRL versions are used with RELOAD-DISK, then CLOSE-INDICATOR=*CLOSE must be set.

Adjusting volume characteristics of PERCON copies

If you create PERCON copies from FDDRL backups, you must adjust the creation date for these volumes so that the version of the tape set, which is determined from this date, is correctly positioned in the overview.

Example

The disk DISK01 was saved at 2005-10-20 on the tapes TAP001, TAP002 and TAP003. The file name FDDRL.D.DISK01 was generated in the volume catalog.

This backup is now copied to the MTCs CAS001 and CAS002 with PERCON. The current date is being set for CREATION-DATE. In addition, the file name in the volume catalog has changed to FDDRL.D.DISK01.COPY.

The following statements correct the creation date and produce the original name:

```
//MODIFY-VOLUME-ATTRIBUTES -  
//  VOLUME=CAS001 -  
//    (FILE-NAME=FDDRL.D.DISK01, -  
//      CREATION-DATE=2005-10-20)  
//MODIFY-VOLUME-ATTRIBUTES -  
//  VOLUME=CAS002 -  
//    (FILE-NAME=FDDRL.D.DISK01, -  
//      CREATION-DATE=2005-10-20)
```

This ensures that MAREN can also determine the tape set in the case of PERCON copies of FDDRL saves. In addition, the tape set is correctly positioned in the volume overview, which is sorted according to ascending save versions.

Defining a location

You can use the `LOCATION` operand of the `DUMP-DISK` and `DUMP-PUBSET` statements to specify the set of devices, from which tape devices for `SCRATCH` requests are to be selected.

You can also use the `MODIFY-FDDRL-PARAMETERS` statement to set a default value for this operand.

When doing so, you should note the following:

- If the Maren administrator has used a MAREN exit to assign the user tapes from a different location, this specification will be ignored.
- If the Maren administrator has used the MAREN location manager to assign the user tapes from a different location, the system specific MAREN parameter `OVERRULE-LOC` decides how to proceed further:
 - `OVERRULE-LOC=REJECT`: the location specified by the user will be used (differing locations lead to a rejected reservation)
 - `OVERRULE-LOC=YES`: the location specified in the MARENLM will be used

Sample applications

This chapter presents examples of typical applications. User input is shown in bold print.

The various messages which FDDRL outputs after the `/START-FDDRL` command and up to the prompt with message FBD1400 have been omitted in the examples below. `/START-FDDRL` is therefore followed immediately by the first statement.

Table of Contents

- [Copying a private disk for backup](#)
- [Copying a private disk for restoration](#)
- [Saving an SM pubset to MTC](#)
- [Restoring an SM pubset from MTC](#)
- [Saving private disks](#)
- [Restoring a private disk from a disk set save](#)
- [Saving the home pubset from the mirror disks](#)
- [Restoring a pubset on new disks](#)
- [Saving the home pubset online](#)

Copying a private disk for backup

Backup copies of private disks are created regularly using the COPY-DISK statement with the default value ACTION=*SAVE.

Log of the FDDRL main task

```
/START-FDDRL _____ (1)
//MODIFY-FDDRL-PARAMETERS PROCESS-JOBS=*BY-SUBTASK(
                        STATUS-OUTPUT-PERIOD=1) _____ (2)
//COPY-DISK DISK=PRIV01, OVERWRITE-DISK=*SAME _____ (3)
//CREATE-JOBS _____ (4)
%(MSG) ? % NKVD013 MOUNT DISK 'PRIV01' ON DEVICE '906C'; '(REASON :
VSN_NOT_UNIQUE)'; (USE='SPECIAL',SYS-ALLOC='EXCL',INPUT DISK FOR FDDRL COPY). '(ETX
= YES;MN; N = NO )'?
%(RPL) . _____ (5)
%(MSG) % % NKVD017 MOUNTING OF DISK 'PRIV01' ON DEVICE '906C' FOR TASK '59SN'
COMPLETED
REQUESTED FDDRL JOBS          FDDRL V21.0A00          2020-08-27 13:06:51 (6)
STATEMENT                      REQ-NAME
DISK      DEV-TYPE I-DEV >O-DEV O-DISK FORM          DISK-DATE
DISK-SET TAPE-DEV LOCATION (TAPES)
-----
COPY-DISK          PRIV01          CPD-PRIV01
PRIV01    D3435      906C > 906D PRIV01          2020-08-27 13:04:41
-----
//START-JOBS _____ (7)
% FBD1260 FDDRL SUBTASK '59SP' FOR DISK/DISK-SET 'PRIV01' STARTED _____ (8)
ACTUAL FDDRL JOB STATUS          FDDRL V21.0A00          2020-08-27 13:08:25 (9)
STATEMENT
DISK/-SET  TSN FORM JOB-STA      SEL-HPS PROCESS-HPS    %
-----
COPY-DISK          PRIV01
PRIV01    59SP      ACTIVE      1572864      1297440      82%
SUMMARY: INACTIVE = 0      ACTIVE = 1      TERMINATED = 0      TOTAL = 1
-----
% FBD1262 FDDRL SUBTASK '59SP' FOR DISK/DISK-SET 'PRIV01' TERMINATED _____ (10)
-----
FINAL FDDRL JOB PROTOCOL          FDDRL V21.0A00          2020-08-27 13:08:40 - (11)
STATEMENT                      REQ-NAME  REQ-DATE  REQ-TIME
DISK  DEV-TYPE  IN  >  OUT          DISK-DATE  TSN      HPS      TERM-MSG
D-SET F-NAME          TSN      HPS      TERM-MSG
```

TAPE-DEV (TAPES)

```
-----  
COPY-DISK          PRIV01          CPD-PRIV01 2020-08-27 13:07:25  
PRIV01 D3435      906C > 906D   2020-08-27 13:04:40 59SP   1572864  
-----
```

```
//END _____ (12)  
% FBD1402 UTILITY ROUTINE 'FDDRL' TERMINATED  
%(MSG) % % ESM0216 FUNCTION 'DELETE' STARTED FOR SUBSYSTEM 'FDDRL /V21.0'
```

- (1) The FDDRL program is started.
- (2) The FDDRL parameters are set:
 - processing by subtask
 - output of status log after 1 minute in each case.
- (3) The PRIV01 disk is copied (saved). Because the disk has been copied before, the output disk has the same name as the input disk.
- (4) The FDDRL job is generated.
- (5) The operator assigns the input disk.
- (6) The FDDRL job is displayed by means of the output "REQUESTED FDDRL JOBS".
- (7) The job is started.
- (8) The FDDRL main task reports that an FDDRL subtask has been started.
- (9) After 1 minute the status log is output. 90% of the disk areas to be saved have been processed up to this point.
- (10) The FDDRL main task reports that an FDDRL subtask has been terminated.
- (11) The final log of the FDDRL jobs is output. The PRIV01 disk on device 906C has been copied to device 906D.
- (12) The FDDRL program is terminated.

Report file \$TSOS.S.FBD.59SN.2020-08-27.130725.CPD-PRIV01

```
REQUEST REPORT FILE :BIOS:$TSOS.S.FBD.59SN.2020-08-27.130725.CPD-PRIV01  
++++++ REQUEST COMPLETED OK ++++++  
-----  
FINAL FDDRL JOB PROTOCOL          FDDRL V21.0A00          2020-08-27 13:08:40
```

```

STATEMENT                               REQ-NAME  REQ-DATE  REQ-TIME
DISK  DEV-TYPE  IN  >  OUT          DISK-DATE  TSN      HPS    TERM-MSG
D-SET  F-NAME                                TSN      HPS    TERM-MSG
      TAPE-DEV (TAPES)
-----
COPY-DISK          PRIV01          CPD-PRIV01 2020-08-27 13:07:25
PRIV01 D3435      906C > 906D 2020-08-27 13:04:40 59SP    1572864
-----
FDDRL PARAMETERS          FDDRL V21.0A00          2020-08-27 13:08:40
PROCESS-JOBS      = *BY-SUBTASK (...)
TASK-LIM          = 4
RUN-PRIO          = *STD
STA-OUTPUT-PERIOD = 1 MINUTE
SEC-WAIT-TIME    = 1 MINUTE
JOB-NAME         = *STD
RETENT-PERIOD    = 0 DAYS
LOCATION           = *STD
TAPE-FORM        = *UNDEF
REPORT-FILE      = *STD
MONITORING       = *NO
KEEP-REQUESTS    = 40 DAYS
=====
SYSOUT PROTOCOL OF SUBTASK 59SP :B10S:$TSOS.S.FBD.59SP.2020-08-27.59SN.PRIV01
NAME      TSN  TID    UNP / Q# SPOOLIN          LOGON
$PRIV01  59SP 00430047  00/000 2020-08-27.1307 2020-08-27.1307

% FBD1205 FDDRL FUNCTION 'COPY(*S)' FOR DISK/DISK-SET 'PRIV01' STARTED ----- (13)
% FBD1403 1572864 HPS OF DISK 'PRIV01' HAVE BEEN PROCESSED
% FBD1206 FDDRL FUNCTION 'COPY(*S)' TERMINATED ----- (14)
(MSG) % % EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 0.1854 SEC, USER ID:
TSOS, TASK ID: 00430047 ----- (15)
% EXC0419 /LOGOFF AT 1308 ON 2020-08-27 FOR TSN '59SP'
% EXC0421 CPU TIME USED: 0.1854

```

- (13) Start message of the FDDRL processing.
- (14) End message of the FDDRL processing.
- (15) The FDDRL subtask is terminated.

Copying a private disk for restoration

The original disk is restored from the backup copy of the private disk from the example in [chapter "Copying a private disk for backup"](#) with the COPY-DISK statement and the operand ACTION=*RESTORE.

Log of the FDDRL main task

```
/START-FDDRL _____ (1)
//MODIFY-FDDRL-PARAMETERS PROCESS-JOBS=*BY-SUBTASK(
                        STATUS-OUTPUT-PERIOD=1) _____ (2)
//COPY-DISK DISK=PRIV01, OVERWRITE-DISK=*SAME-VSN, ACTION=*RESTORE _____ (3)
//CREATE-JOBS _____ (4)
%(MSG) ? % NKVD013 MOUNT DISK 'PRIV01' ON DEVICE '906C'; '(REASON :
VSN_NOT_UNIQUE)'; (USE='SPECIAL',SYS-ALLOC='EXCL',INPUT DISK FOR FDDRL COPY). '(ETX
= YES; MN; N = NO )'?
%(RPL) . 906D _____ (5)
%(MSG) % % NKVD017 MOUNTING OF DISK 'PRIV01' ON DEVICE '906D' FOR TASK '59SQ'
COMPLETED _____ (6)
REQUESTED FDDRL JOBS          FDDRL V21.0A00          2020-08-27 13:14:14
STATEMENT                      REQ-NAME
DISK      DEV-TYPE I-DEV >O-DEV O-DISK FORM          DISK-DATE
DISK-SET TAPE-DEV LOCATION (TAPES)
-----
COPY-DISK          PRIV01          CPD-PRIV01
PRIV01  D3435      906D > 906C PRIV01          2020-08-27 13:04:40
-----
//START-JOBS _____ (7)
% FBD1260 FDDRL SUBTASK '59SR' FOR DISK/DISK-SET 'PRIV01' STARTED _____ (8)
ACTUAL FDDRL JOB STATUS          FDDRL V21.0A00          2020-08-27 13:15:51 (9)
STATEMENT
DISK/-SET TSN FORM JOB-STA          SEL-HPS PROCESS-HPS%
-----
COPY-DISK          PRIV01
PRIV01  59SR      ACTIVE          1572864          1280320 81%
SUMMARY: INACTIVE = 0          ACTIVE = 1          TERMINATED = 0          TOTAL = 1
-----
% FBD1262 FDDRL SUBTASK '59SR' FOR DISK/DISK-SET 'PRIV01' TERMINATED _____ (10)
-----
FINAL FDDRL JOB PROTOCOL          FDDRL V21.0A00          2020-08-27 13:16:11 (11)
STATEMENT                      REQ-NAME  REQ-DATE  REQ-TIME
DISK  DEV-TYPE  IN  >  OUT          DISK-DATE  TSN      HPS      TERM-MSG
D-SET  F-NAME          TSN      HPS      TERM-MSG
TAPE-DEV (TAPES)
-----
COPY-DISK          PRIV01          CPD-PRIV01 2020-08-27 13:14:51
PRIV01 D3435      906D > 906C 2020-08-27 13:04:41 59SR  1572864
```

//END ----- (12)

% FBD1402 UTILITY ROUTINE 'FDDRL' TERMINATED

%(MSG) % % ESM0216 FUNCTION 'DELETE' STARTED FOR SUBSYSTEM 'FDDRL /V21.0'

- (1) The FDDRL program is started.
- (2) The FDDRL parameters are set:
 - processing by subtask
 - output of status log after 1 minute in each case.
- (3) The PRIV01 disk is copied (restored). As the original disk still exists, it has the same name as the input disk.
- (4) The FDDRL job is generated.
- (5) The operator assigns the input disk.
- (6) The FDDRL job is displayed by means of the output "REQUESTED FDDRL JOBS".
- (7) The job is started.
- (8) The FDDRL main task reports that an FDDRL subtask has been started.
- (9) After 1 minute the status log is output. 95% of the disk areas to be restored have been processed up to this point.
- (10) The FDDRL main task reports that an FDDRL subtask has been terminated.
- (11) The final log of the FDDRL jobs is output. The PRIV01 disk on device 906D is copied to device 906C.
- (12) The FDDRL program is terminated.

Saving an SM pubset to MTC

An SM pubset is saved to magnetic tape cartridge with the DUMP-PUBSET statement.

The SM pubset consists not only of the control volume set, but also of two further volume sets with a total of eight disks. The use of the "disk set" save entity means that all disks of the pubset are saved on a tape set. Only one MTC device and one MTC are required here to do this (3 MTCs would be needed with the single disk save entity).

In multiplexing mode the blocks to be saved are written from four disks alternately and in parallel to one MTC (see [chapter "Disk set"](#)).

Log of the FDDRL main task

```
/START-FDDRL _____ (1)
```

```
//DUMP-PUBSET PUBSET=SMP, SAVE-ENTITY=*DISK-SET(NUMBER-OF-DISK-SETS=1) _____ (2)
```

```
//CREATE-JOBS _____ (3)
```

```
%(MSG) % % DMS035B IMPORT PUBSET TASK WITH TSN 'XBHB' FOR PUBSET WITH  
PUBSET ID 'SMP' HAS BEEN CREATED AND STARTED
```

```
%(MSG) % % DMS035D EXPORT-PUBSET TASK WITH TSN 'XBHC' FOR PUBSET WITH  
PUBSET ID 'SMP' HAS BEEN CREATED AND STARTED
```

```
REQUESTED FDDRL JOBS FDDRL V21.0A00 2020-08-27 13:21:37 (4)  
STATEMENT REQ-NAME
```

```
DISK DEV-TYPE I-DEV >O-DEV O-DISK FORM DISK-DATE  
DISK-SET TAPE-DEV LOCATION (TAPES)
```

```
-----  
DUMP-PUBSET SMP DMP-SMP  
PC.000 D3490-30 9F28 2020-08-26 12:14:04  
PV.000 D3490-30 9F29 2020-08-26 12:14:04  
PV.001 D3490-30 9F2A 2020-08-26 12:14:04  
PV.002 D3490-40 9F30 2020-08-26 12:14:04  
PV.003 D3435 FA48 2020-08-26 12:14:04  
P1.000 D3490-30 9F2B 2020-08-26 12:14:04  
P1.001 D3490-40 9F31 2020-08-26 12:14:04  
P1.002 D3435 FA49 2020-08-26 12:14:04  
SMP-1 TAPE-C4  
-----
```

```
//START-JOBS _____ (5)
```

```
% FBD1260 FDDRL SUBTASK '59ST' FOR DISK/DISK-SET 'SMP-1' STARTED
```

```
% FBD1262 FDDRL SUBTASK '59ST' FOR DISK/DISK-SET 'SMP-1' TERMINATED _____ (6)
```

```
FINAL FDDRL JOB PROTOCOL FDDRL V21.0A00 2020-08-27 13:23:19 (7)  
STATEMENT REQ-NAME REQ-DATE REQ-TIME
```

```
DISK DEV-TYPE IN > OUT DISK-DATE TSN HPS TERM-MSG  
D-SET F-NAME TSN HPS TERM-MSG  
TAPE-DEV (TAPES)
```

```
-----  
DUMP-PUBSET SMP DMP-SMP 2020-08-27 13:22:14  
PC.000 D3490-30 9F28 >SMP-1 2020-08-26 12:14:04 70240  
PV.000 D3490-30 9F29 >SMP-1 2020-08-26 12:14:04 16480  
PV.001 D3490-30 9F2A >SMP-1 2020-08-26 12:14:04 640  
PV.002 D3490-40 9F30 >SMP-1 2020-08-26 12:14:04 80  
-----
```

```
PV.003 D3435    FA48 >SMP-1  2020-08-26 12:14:04          80
P1.000 D3490-30 9F2B >SMP-1  2020-08-26 12:14:04        16480
P1.001 D3490-40 9F31 >SMP-1  2020-08-26 12:14:04       184320
P1.002 D3435    FA49 >SMP-1  2020-08-26 12:14:04          80
SMP-1  FDDRL.P.SMP.2020-08-27.132200-1      59ST    288400
        TAPE-C4  (QA3103)
```

//END _____ (8)

```
% FBD1402 UTILITY ROUTINE 'FDDRL' TERMINATED
%(MSG) % % ESM0216 FUNCTION 'DELETE' STARTED FOR SUBSYSTEM 'FDDRL /V21.0'
```

- (1) The FDDRL program is started.
- (2) The SM pubset SMP1 is saved. The disk set save entity is used.
- (3) The FDDRL jobs are created. In order to determine the disks of the SM pubset, a so-called mini-import with a subsequent export is executed.
- (4) The generated jobs are displayed by means of "REQUESTED FDDRL JOBS". One job exists for each disk and the disk set SMP1-1.
- (5) The jobs are started. The FDDRL main task reports that an FDDRL subtask has been terminated.
- (6) The FDDRL main task reports that an FDDRL subtask has been terminated.
- (7) The final log of the FDDRL jobs is output. The pubset was saved to MTC QA3103. The MAREN file name of the backup volume is FDDRL.P.SMP.2014-08-27.132200-1.
- (8) The FDDRL program is terminated.

Report file \$TSOS.S.FBD.59SS.2020-08-27.132214.DMP-SMP

REQUEST REPORT FILE :B10S:\$TSOS.S.FBD.59SS.2020-08-27.132214.DMP-SMP
 ++++++ REQUEST COMPLETED OK ++++++

```
-----
FINAL FDDRL JOB PROTOCOL          FDDRL V21.0A00          2020-08-27 13:23:19
STATEMENT                          REQ-NAME  REQ-DATE  REQ-TIME
DISK  DEV-TYPE  IN  >  OUT          DISK-DATE  TSN      HPS    TERM-MSG
D-SET  F-NAME                               TSN      HPS    TERM-MSG
      TAPE-DEV (TAPES)
-----
```

```
DUMP-PUBSET          SMP          DMP-SMP          2020-08-27 13:22:14
PC.000 D3490-30 9F28 >SMP-1 2020-08-26 12:14:04          70240
PV.000 D3490-30 9F29 >SMP-1 2020-08-26 12:14:04          16480
PV.001 D3490-30 9F2A >SMP-1 2020-08-26 12:14:04           640
PV.002 D3490-40 9F30 >SMP-1 2020-08-26 12:14:04           80
PV.003 D3435    FA48 >SMP-1 2020-08-26 12:14:04           80
P1.000 D3490-30 9F2B >SMP-1 2020-08-26 12:14:04          16480
P1.001 D3490-40 9F31 >SMP-1 2020-08-26 12:14:04         184320
P1.002 D3435    FA49 >SMP-1 2020-08-26 12:14:04           80
SMP-1  FDDRL.P.SMP.2020-08-27.132200-1          59ST    288400
      TAPE-C4 (QA3103)
-----
```

```
FDDRL PARAMETERS          FDDRL V21.0A00          2020-08-27 13:23:19
PROCESS-JOBS              = *BY-SUBTASK (...)
TASK-LIM                  = 4
RUN-PRIO                  = *STD
STA-OUTPUT-PERIOD        = 5 MINUTES
SEC-WAIT-TIME            = 1 MINUTE
JOB-NAME                  = *STD
RETENT-PERIOD            = 0 DAYS
LOCATION                   = *STD
TAPE-FORM                 = *V17-FORMAT
REPORT-FILE              = *STD
MONITORING                = *NO
KEEP-REQUESTS            = 40 DAYS
=====
```

```
SYSOUT PROTOCOL OF SUBTASK 59ST :B10S:$TSOS.S.FBD.59ST.2020-08-27.59SS.SMP-1
NAME      TSN  TID      UNP / Q# SPOOLIN      LOGON
SMP#1     59ST 003F0040  00/000 2020-08-27.1322 2020-08-27.1322
(MSG) ?   %   MARUP44 MOUNT VOLUME 'QA3103' ON DEVICE 'FQ' FOR TASK '59ST'
FROM 'CSVTA500'? REPLY (Y=YES; N=NO)?
(RPL) . Y
(MSG) %   %   NKVT017 MOUNTING OF TAPE 'QA3103' ON DEVICE 'FQ' FOR TASK '59ST'
COMPLETED ----- (9)
(MSG) %   %   FBD1271 FDDRL JOB FOR DISK 'PC.000' ON DEVICE '9F28' STARTED
----- (10)
%   FBD1205 FDDRL FUNCTION 'DUMP' FOR DISK/DISK-SET 'SMP-1' STARTED ----- (11)
(MSG) %   %   FBD2201 TAPE 'QA3103' ON DEVICE 'FQ' AS REEL '001' FOR DUMP OF
DISK/DISK-SET 'SMP-1' USED
(MSG) %   %   FBD1271 FDDRL JOB FOR DISK 'PV.000' ON DEVICE '9F29' STARTED
----- (12)
(MSG) %   %   FBD1271 FDDRL JOB FOR DISK 'PV.001' ON DEVICE '9F2A' STARTED
(MSG) %   %   FBD1271 FDDRL JOB FOR DISK 'PV.002' ON DEVICE '9F30' STARTED
%   FBD1403 640 HPS OF DISK 'PV.001' HAVE BEEN PROCESSED ----- (13)
```

```

(MSG) % % FBD1271 FDDRL JOB FOR DISK 'PV.003' ON DEVICE 'FA48' STARTED
% FBD1403 80 HPS OF DISK 'PV.003' HAVE BEEN PROCESSED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'P1.000' ON DEVICE '9F2B' STARTED
% FBD1403 16480 HPS OF DISK 'PV.000' HAVE BEEN PROCESSED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'P1.001' ON DEVICE '9F31' STARTED
% FBD1403 16480 HPS OF DISK 'P1.000' HAVE BEEN PROCESSED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'P1.002' ON DEVICE 'FA49' STARTED
% FBD1403 80 HPS OF DISK 'P1.002' HAVE BEEN PROCESSED
% FBD1403 80 HPS OF DISK 'PV.002' HAVE BEEN PROCESSED
% FBD1403 70240 HPS OF DISK 'PC.000' HAVE BEEN PROCESSED _____ (14)
% FBD1403 184320 HPS OF DISK 'P1.001' HAVE BEEN PROCESSED
% FBD1206 FDDRL FUNCTION 'DUMP' TERMINATED _____ (15)
(MSG) % % FBD2208 TAPE 'QA3103' ON DEVICE 'FQ' AS REEL '001' FOR DUMP OF
DISK/DISK-SET 'SMP-1' COMPLETE
(MSG) % % EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 0.9346 SEC, USER ID:
TSOS, TASK ID: 003F0040 _____ (16)
% EXC0419 /LOGOFF AT 1323 ON 2020-08-27 FOR TSN '59ST'
% EXC0421 CPU TIME USED: 0.9346

```

- (9) The first backup volume with the VSN QA3103 is mounted and being processed.
- (10) Disk PC.000 is saved.
- (11) Start message of the FDDRL processing.
- (12) Further disks are saved (simultaneously).
- (13) The backup of disk PV.001 has been completed.
- (14) The backup of disk PC.000 (with which the backup was started) has been completed.
- (15) End messages for FDDRL processing.
- (16) The FDDRL subtask is terminated.

Restoring an SM pubset from MTC

The SM pubset that was saved in the example in [chapter "Saving an SM pubset to MTC"](#) is restored from MTC with the RELOAD-PUBSET statement. The RELOAD-PUBSET statement requires the disk set save entity on which multiple or all disks of a pubset have been saved to a tape set. During job creation, the FDDRL main task determines the pubset configuration from the first backup volume.

Log of the FDDRL main task

```
/START-FDDRL _____ (1)
                //EXEC-SYS-CMD (CREATE-JV FDDRL.D.WK906C-1, SUPPRESS-ERRORS=*JV-
EXIST)_____ (2) //DUMP-DISK DISK=(WK906C,WK906D,WK906E),
                TAPE=(QA3900,QA3901,QA3902),TAPE-DEVICE-TYPE=TAPE-C4 _____ (3)
//CREATE-JOBS _____ (4)
REQUESTED FDDRL JOBS          FDDRL V21.0A00      2020-08-27 13:42:06
STATEMENT                      REQ-NAME
DISK      DEV-TYPE I-DEV >O-DEV O-DISK FORM      DISK-DATE
DISK-SET TAPE-DEV LOCATION (TAPES)
-----
DUMP-DISK          (WK906C,...) DMD#WK906C
WK906C  D3435      906C          0000-00-00 00:00:00
WK906D  D3435      906D          0000-00-00 00:00:00
WK906E  D3435      906E          0000-00-00 00:00:00
WK906C-1 TAPE-C4   CSVTA500 (QA3900,QA3901,QA3902)
-----
//START-JOBS _____ (5)
% FBD1260 FDDRL SUBTASK '59SY' FOR DISK/DISK-SET 'WK906C-1' STARTED
% FBD1262 FDDRL SUBTASK '59SY' FOR DISK/DISK-SET 'WK906C-1' TERMINATED
-----
FINAL FDDRL JOB PROTOCOL      FDDRL V21.0A00      2020-08-27 13:43:02 (6)
STATEMENT                      REQ-NAME  REQ-DATE  REQ-TIME
DISK  DEV-TYPE  IN  >  OUT      DISK-DATE  TSN      HPS      TERM-MSG
D-SET F-NAME                                TSN      HPS      TERM-MSG
      TAPE-DEV (TAPES)
-----
DUMP-DISK          (WK906C,...) DMD#WK906C 2020-08-27 13:42:51
WK906C D3435      906C >      0000-00-00 00:00:00      1120
WK906D D3435      906D >      0000-00-00 00:00:00      1120
WK906E D3435      906E >      0000-00-00 00:00:00      1120
      FDDRL.D.WK906C-1          59SY      3360
      TAPE-C4 (QA3900)
-----
//END _____ (7)
% FBD1402 UTILITY ROUTINE 'FDDRL' TERMINATED
%(MSG) % % ESM0216 FUNCTION 'DELETE' STARTED FOR SUBSYSTEM 'FDDRL /V21.0'
```

- (1) The FDDRL program is started.
- (2) Pubset SMP1 is restored using //RELOAD-PUBSET. The save entity used is disk set. The associated MTCs are determined by means of the MAREN file name specified (see item (7) in the [chapter "Saving an SM pubset to MTC"](#)).
- (3) The FDDRL jobs are created. In order to determine the disks of the pubset, the first backup volume is mounted.
- (4) Information about the backup volumes is output in the information block "FDDRL TAPE INFORMATION":
The pubset SMP1 was saved on the MTC.
- (5) Information about the save scope and the properties of the individual disks of the pubset is output in this information block. This information enable systems support to configure and provide the disks with the required properties (disk type, format, capacity).
- (6) The generated jobs are displayed in the information block "REQUESTED FDDRL JOBS". One job exists for each disk and the disk set SMP1-1. All output disks are checked at this point. The MN (O-DEV) and the VSN of the output disks to be mounted are known. In the event of faulty checked output disks, processing of the CREATE-JOBS statement would be terminated with an error.
- (7) The jobs are started. The FDDRL main task reports that an FDDRL subtask has been started.
- (8) The FDDRL main task reports that an FDDRL subtask has been terminated. The final log of the FDDRL jobs is output: The pubset has been restored completely from the backup volumes OS0433 and OS0434. The same time stamp (TIME-STAMP) for all disks guarantees consistent restoration of all disks.
- (9) The FDDRL program is terminated.

Report file \$TSOS.S.FBD.59SU.2020-08-27.133219.RLP-SMP

REQUEST REPORT FILE :B10S:\$TSOS.S.FBD.59SX.2020-08-27.134251.DMD#WK906C

++++++ REQUEST COMPLETED OK ++++++

```
-----
FINAL FDDRL JOB PROTOCOL          FDDRL V21.0A00          2020-08-27 13:43:02
STATEMENT                          REQ-NAME  REQ-DATE  REQ-TIME
DISK  DEV-TYPE  IN  >  OUT          DISK-DATE  TSN      HPS    TERM-MSG
D-SET  F-NAME
      TAPE-DEV (TAPES)
-----
```

```
DUMP-DISK          (WK906C,...) DMD#WK906C 2020-08-27 13:42:51
WK906C D3435      906C >      0000-00-00 00:00:00          1120
WK906D D3435      906D >      0000-00-00 00:00:00          1120
WK906E D3435      906E >      0000-00-00 00:00:00          1120
      FDDRL.D.WK906C-1          59SY      3360
      TAPE-C4 (QA3900)
-----
```

```
FDDRL PARAMETERS          FDDRL V21.0A00          2020-08-27 13:43:02
PROCESS-JOBS      = *BY-SUBTASK (...)
TASK-LIM          = 4
RUN-PRIO          = *STD
STA-OUTPUT-PERIOD = 5 MINUTES
SEC-WAIT-TIME    = 1 MINUTE
JOB-NAME         = *STD
RETENT-PERIOD    = 0 DAYS
LOCATION          = CSVTA500
TAPE-FORM       = *V17-FORMAT
REPORT-FILE     = *STD
MONITORING      = *NO
KEEP-REQUESTS   = 40 DAYS
=====
```

SYSOUT PROTOCOL OF SUBTASK 59SY :B10S:\$TSOS.S.FBD.59SY.2020-08-27.59SX.WK906C-1

```
NAME      TSN  TID      UNP / Q# SPOOLIN          LOGON
WK906C#1  59SY 002F00EE  00/000 2020-08-27.1342 2020-08-27.1342
(MSG) ? % NKVT013 MOUNT TAPE 'QA3900' ON DEVICE 'FU';
(USE='SPECIAL',WR='ON',TYPE='TAPE-C4',FDDRL.D.WK906C-1 TAPE 001). '(ETX =
YES; MN; N = NO )'?
(MSG) ? FU (TM)
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3900' ON DEVICE 'FU' FOR TASK '59SY'
COMPLETED _____ (9)
_____ (10)
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'WK906C' ON DEVICE '906C' STARTED
% FBD1205 FDDRL FUNCTION 'DUMP' FOR DISK/DISK-SET 'WK906C-1' STARTED _____ (11)
(MSG) % % FBD2201 TAPE 'QA3900' ON DEVICE 'FU' AS REEL '001' FOR DUMP OF
DISK/DISK-SET 'WK906C-1' USED _____ (12)
_____ (13)
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'WK906D' ON DEVICE '906D' STARTED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'WK906E' ON DEVICE '906E' STARTED
% FBD1403 1120 HPS OF DISK 'WK906C' HAVE BEEN PROCESSED
% FBD1403 1120 HPS OF DISK 'WK906D' HAVE BEEN PROCESSED
% FBD1403 1120 HPS OF DISK 'WK906E' HAVE BEEN PROCESSED
% FBD1206 FDDRL FUNCTION 'DUMP' TERMINATED
(MSG) % % FBD2208 TAPE 'QA3900' ON DEVICE 'FU' AS REEL '001' FOR DUMP OF
DISK/DISK-SET 'WK906C-1' COMPLETE _____ (14)
```

```
(MSG) % % EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 0.2025 SEC, USER ID:  
TSOS, TASK ID: 002F00EE  
% EXC0419 /LOGOFF AT 1342 ON 2020-08-27 FOR TSN '59SY'  
% EXC0421 CPU TIME USED: 0.2025
```

- (10) The first backup volume with the VSN QA3103 was mounted.
- (11) Disk PC.000 is restored.
- (12) Start message of the FDDRL processing.
- (13) The second backup volume is processed.
- (14) Further disks are restored (simultaneously).
- (15) Restoration of disk PV.001 is terminated.
- (16) End messages for FDDRL processing.
- (17) The FDDRL subtask is terminated.

Saving private disks

Three private disks are saved together on a tape set using the disk set save entity. The save is performed on one MTC device and two MTCs. During the save a job variable is supplied with the names (VSNs) of the backup volumes used.

Log of the FDDRL main task

```
/START-FDDRL _____ (1)
//EXEC-SYS-CMD (CREATE-JV FDDRL.D.WK906C-1, SUPPRESS-ERRORS=*JV-EXIST)_____ (2)
//DUMP-DISK DISK=(WK906C,WK906D,WK906E),
              TAPE=(QA3900,QA3901,QA3902),TAPE-DEVICE-TYPE=TAPE-C4 _____ (3)
//CREATE-JOBS _____ (4)
REQUESTED FDDRL JOBS          FDDRL V21.0A00      2020-08-27 13:42:06
STATEMENT                      REQ-NAME
DISK      DEV-TYPE I-DEV >O-DEV O-DISK FORM      DISK-DATE
DISK-SET TAPE-DEV LOCATION (TAPES)
-----
DUMP-DISK          (WK906C,...) DMD#WK906C
WK906C   D3435    906C          0000-00-00 00:00:00
WK906D   D3435    906D          0000-00-00 00:00:00
WK906E   D3435    906E          0000-00-00 00:00:00
WK906C-1 TAPE-C4  CSVTA500 (QA3900,QA3901,QA3902)
-----
//START-JOBS _____ (5)
% FBD1260 FDDRL SUBTASK '59SY' FOR DISK/DISK-SET 'WK906C-1' STARTED
% FBD1262 FDDRL SUBTASK '59SY' FOR DISK/DISK-SET 'WK906C-1' TERMINATED
-----
FINAL FDDRL JOB PROTOCOL          FDDRL V21.0A00      2020-08-27 13:43:02 (6)
STATEMENT                      REQ-NAME  REQ-DATE  REQ-TIME
DISK  DEV-TYPE  IN  >  OUT          DISK-DATE  TSN      HPS      TERM-MSG
D-SET F-NAME                                TSN      HPS      TERM-MSG
      TAPE-DEV (TAPES)
-----
DUMP-DISK          (WK906C,...) DMD#WK906C 2020-08-27 13:42:51
WK906C D3435    906C >          0000-00-00 00:00:00          1120
WK906D D3435    906D >          0000-00-00 00:00:00          1120
WK906E D3435    906E >          0000-00-00 00:00:00          1120
      FDDRL.D.WK906C-1          59SY      3360
      TAPE-C4 (QA3900)
-----
//END _____ (7)
% FBD1402 UTILITY ROUTINE 'FDDRL' TERMINATED
%(MSG) % % ESM0216 FUNCTION 'DELETE' STARTED FOR SUBSYSTEM 'FDDRL /V21.0'
```

- (1) The FDDRL program is started.
- (2) Job variable FDDRL.D.WK906C-1 is created, which has the function of accommodating the VSNs of the backup volumes used.
- (3) The three private disks WK906C, WK906D and WK906E are to be saved on a tape set using the disk set save entity. The three specified MTCs QA3900, QA3901 and QA3902 with the volume type TAPE-C4 are to be used for the save.
- (4) The FDDRL jobs are created. The generated jobs are displayed in the information block "REQUESTED FDDRL JOBS": One job exists for each disk and for the disk set. The disk set name WK906C-1 was formed from the VSN of the first disk supplemented by "-1".
- (5) The jobs are started. The FDDRL main task reports that an FDDRL subtask has been started.
- (6) The final log of the FDDRL jobs is output. The disks have been saved only to MTC QA3900. MTCs QA3901 and QA3902 remain unused.
- (7) The FDDRL program is terminated.
- (8) The content of job variable FDDRL.D.WK906C-1 is displayed. This contains only the VSN of the MTCs actually used.

Report file \$TSOS.S.FBD.59SX.2020-08-27.134251.DMD#WK906C

```

REQUEST REPORT FILE :B10S:$TSOS.S.FBD.59SX.2020-08-27.134251.DMD#WK906C
++++++ REQUEST COMPLETED OK ++++++
-----
FINAL FDDRL JOB PROTOCOL          FDDRL V21.0A00          2020-08-27 13:43:02
STATEMENT                          REQ-NAME  REQ-DATE  REQ-TIME
DISK  DEV-TYPE  IN  >  OUT          DISK-DATE  TSN      HPS    TERM-MSG
D-SET  F-NAME                    TSN      HPS    TERM-MSG
      TAPE-DEV (TAPES)
-----
DUMP-DISK          (WK906C,...) DMD#WK906C 2020-08-27 13:42:51
WK906C D3435      906C >      0000-00-00 00:00:00          1120
WK906D D3435      906D >      0000-00-00 00:00:00          1120
WK906E D3435      906E >      0000-00-00 00:00:00          1120
      FDDRL.D.WK906C-1          59SY      3360
      TAPE-C4 (QA3900)
-----
FDDRL PARAMETERS          FDDRL V21.0A00          2020-08-27 13:43:02
PROCESS-JOBS      = *BY-SUBTASK (...)
TASK-LIM          = 4
RUN-PRIO          = *STD
STA-OUTPUT-PERIOD = 5 MINUTES
SEC-WAIT-TIME    = 1 MINUTE
JOB-NAME         = *STD
RETENT-PERIOD    = 0 DAYS

```

```
LOCATION          = CSVTA500
TAPE-FORM       = *V17-FORMAT
REPORT-FILE     = *STD
MONITORING      = *NO
KEEP-REQUESTS  = 40 DAYS
```

```
=====
SYSOUT PROTOCOL OF SUBTASK 59SY :B10S:$TSOS.S.FBD.59SY.2020-08-27.59SX.WK906C-1
```

```
NAME      TSN  TID      UNP / Q# SPOOLIN      LOGON
WK906C#1 59SY 002F00EE  00/000 2020-08-27.1342 2020-08-27.1342
(MSG) ? % NKVT013 MOUNT TAPE 'QA3900' ON DEVICE 'FU';
(USE='SPECIAL',WR='ON',TYPE='TAPE-C4',FDDRL.D.WK906C-1 TAPE 001). '(ETX =
YES; MN; N = NO )'?
(MSG) ? FU (TM)
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3900' ON DEVICE 'FU' FOR TASK '59SY'
COMPLETED _____ (9)
_____ (10)
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'WK906C' ON DEVICE '906C' STARTED
% FBD1205 FDDRL FUNCTION 'DUMP' FOR DISK/DISK-SET 'WK906C-1' STARTED _____ (11)
(MSG) % % FBD2201 TAPE 'QA3900' ON DEVICE 'FU' AS REEL '001' FOR DUMP OF
DISK/DISK-SET 'WK906C-1' USED _____ (12)
_____ (13)
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'WK906D' ON DEVICE '906D' STARTED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'WK906E' ON DEVICE '906E' STARTED
% FBD1403 1120 HPS OF DISK 'WK906C' HAVE BEEN PROCESSED
% FBD1403 1120 HPS OF DISK 'WK906D' HAVE BEEN PROCESSED
% FBD1403 1120 HPS OF DISK 'WK906E' HAVE BEEN PROCESSED
% FBD1206 FDDRL FUNCTION 'DUMP' TERMINATED
(MSG) % % FBD2208 TAPE 'QA3900' ON DEVICE 'FU' AS REEL '001' FOR DUMP OF
DISK/DISK-SET 'WK906C-1' COMPLETE _____ (14)
(MSG) % % EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 0.2025 SEC, USER ID:
TSOS, TASK ID: 002F00EE
% EXC0419 /LOGOFF AT 1342 ON 2020-08-27 FOR TSN '59SY'
% EXC0421 CPU TIME USED: 0.2025
```

- (9) The first backup volume with the VSN QA3900 was mounted.
- (10) Disk WK906C is saved.
- (11) Start message of the FDDRL processing.
- (12) The second backup volume is processed.
- (13) Further disks are saved.
- (14) The backup volume was processed. The FDDRL subtask is terminated.

Restoring a private disk from a disk set save

One of the saved disks from the disk set save performed in [chapter "Saving private disks"](#) is to be restored. During the save operation the VSNs of the backup volumes used were written to a job variable. This job variable is now used to determine the backup volumes. Restoration is executed under the main task.

Log of the FDDRL main task

```
/SHOW-JV FDDRL.D.WK906C-1
QA3900 _____ (1)
/START-FDDRL _____ (2)
//MODIFY-FDDRL-PARAMETERS PROCESS-JOBS=*BY-MAINTASK _____ (3)
//RELOAD-DISK DISK=WK906E, TAPE=&(FDDRL.D.WK906C-1),
          TAPE-DEVICE-TYPE=TAPE-C4, SAVE-ENTITY=*DISK-SET _____ (4)
//CREATE-JOBS _____ (5)
%(MSG) ? % NKVT013 MOUNT TAPE 'QA3900' ON DEVICE 'FT';
(USE='SPECIAL',WR='OFF',TYPE='TAPE-C4',FDDRL.D.WK906E TAPE *LAST). '(ETX =
YES; MN; N = NO )'?
%(MSG) ? FT (TM)
%(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3900' ON DEVICE 'FT' FOR TASK '59SZ'
COMPLETED
FDDRL TAPE INFORMATION          FDDRL V21.0A00          2020-08-27 13:48:05
TAPE VSN      = QA3900          VOL-SEQ      = 0001          FIRST-VOL    = QA3900
  CRE-DATE    = 2020-08-27 EXPIR-DATE = 2020-08-27 USER-ID     = TSOS
  F-NAME      = FDDRL.D.WK906C-1
FDDRL = V21.0A TAPE-FORM = *V17-FORMAT SAVE-DATE = 2020-08-27 13:42:51
  PUBSET      =          DISK =          DISK-DATE =
  SAVE-ENT    = *DISK-SET DISK-SET = WK906C-1 NUM-OF-DISK-SETS = 1
DISK  UNIT DEV-TYPE FORM A-U NUM-OF-HPS NUM-OF-512B SAVED-HPS _____ (6)
-----
WK906C 906C D3435 KEY 6 1572864 7864320 1120
WK906D 906D D3435 KEY 6 1572864 7864320 1120
WK906E 906E D3435 KEY 6 1572864 7864320 1120
REQUESTED FDDRL JOBS          FDDRL V21.0A00          2020-08-27 13:48:32 (7)
STATEMENT                      REQ-NAME
  DISK      DEV-TYPE I-DEV >O-DEV O-DISK FORM          DISK-DATE
  DISK-SET TAPE-DEV LOCATION (TAPES)
-----
RELOAD-DISK          (WK906E)          RLD#WK906E
  WK906E D3435 906E > 906E WK906E          0000-00-00 00:00:00
  WK906C-1 TAPE-C4 CSVTA500 (QA3900)
-----
//START-JOBS _____ (8)
%(MSG) % % FBD4015 PREPARE TAPE 'QA3900' FOR FDDRL RUN (ACCESS='INPUT',
DISK/DISK-SET='WK906C-1')
%(MSG) ? % NKVT013 MOUNT TAPE 'QA3900' ON DEVICE 'FU';
(USE='SPECIAL',WR='OFF',TYPE='TAPE-C4',FDDRL.D.WK906C-1 TAPE 001). '(ETX =
YES; MN; N = NO )'? _____ (9)
%(MSG) ? FU (TM)
%(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3900' ON DEVICE 'FU' FOR TASK '59PF'
COMPLETED
% FBD2354 TAPE 'QA3900' ON DEVICE 'FU' IS SAVE TAPE OF DISK/DISK-SET
'WK906C-1'FROM 2020-08-26 13:42:49. ACCEPT TAPE? REPLY (Y=YES; N=NO)?
Y _____ (10)
%(MSG) % % FBD1271 FDDRL JOB FOR DISK 'WK906E' ON DEVICE '906E' STARTED
% FBD1205 FDDRL FUNCTION 'RELOAD' FOR DISK/DISK-SET 'WK906C-1' STARTED
%(MSG) % % FBD2303 TAPE 'QA3900' ON DEVICE 'FP' AS REEL '001' FOR RELOAD OF
```

```

DISK/DISK-SET 'WK906C-1' USED
% FBD1403 1120 HPS OF DISK 'WK906E' HAVE BEEN PROCESSED
%(MSG) % % FBD2304 TAPE 'QA3900' ON DEVICE 'FP' AS REEL '001' FOR RELOAD OF
DISK/DISK-SET 'WK906C-1' RELOADED
% FBD1206 FDDRL FUNCTION 'RELOAD' TERMINATED

```

```

-----
FINAL FDDRL JOB PROTOCOL          FDDRL V21.0A00          2020-08-27 13:49:05 (11)
STATEMENT          REQ-NAME  REQ-DATE  REQ-TIME
DISK  DEV-TYPE  IN  >  OUT          DISK-DATE  TSN      HPS  TERM-MSG
D-SET  F-NAME          TSN      HPS  TERM-MSG
      TAPE-DEV (TAPES)
-----
RELOAD-DISK          (WK906E)          RLD#WK906E 2020-08-27 13:48:56
WK906E D3435          > 906E 0000-00-00 00:00:00          1120
      FDDRL.D.WK906C-1          59SZ          1120
      TAPE-C4 (QA3900)
-----

```

```

//END ----- (12)
% FBD1402 UTILITY ROUTINE 'FDDRL' TERMINATED
%(MSG) % % ESM0216 FUNCTION 'DELETE' STARTED FOR SUBSYSTEM 'FDDRL /V21.0'

```

- (1) The content of job variable FDDRL.D.WK906C-1 is displayed. It still contains the VSN of the tape cartridge used for saving.
- (2) The FDDRL program is started.
- (3) Restoration is to be executed under the main task.
- (4) Of the three disks that were saved together, only disk WK906E is to be restored. Disk set is specified as the save entity using the SAVE-ENTITY operand.
- (5) The FDDRL jobs are created. The last backup volume is mounted and the "FDDRL TAPE INFORMATION" is output.
- (6) This information block shows the disks saved on the tape set and their properties.
- (7) After the output disks have been checked, the generated jobs are displayed in the information block "REQUESTED FDDRL JOBS": One job exists for the requested disk and the disk set.
- (8) The jobs are started. The FDDRL main task reports that an FDDRL subtask has been started.
- (9) FDDRL requests the backup volume.
- (10) The user confirms that the tape cartridge is the correct one. Restoration of the disk begins.
- (11) When restoration has been completed, the final log "FINAL FDDRL JOB PROTOCOL" is output: The disk was restored to the device with the MN 6BD9. Tape cartridge OS0201 was used for restoration.
- (12) The FDDRL program is terminated.

Saving the home pubset from the mirror disks

The mirror disks (clone) of home pubset B10K (8-disk pubset) are split during ongoing operation using an SHC-OSD command and then saved using FDDRL (DUMP-PUBSET statement).

```

/ACTIVATE-CLONE UNIT=*BY-PUBSET(PUBSET=B10K, NEW-PUBSET=*SPECIAL-VSN,
      HOLD-IO=*UNTIL-ACTIVATED), WAIT=*UNTIL-SYNCHRONIZATION ----- (1)
% NDE1073 ACTIVATE-CLONE PROCESSED FOR PAIR 3420      ,3428
% NDE1073 ACTIVATE-CLONE PROCESSED FOR PAIR 3421      ,3429
% NDE1073 ACTIVATE-CLONE PROCESSED FOR PAIR 3422      ,342A
% NDE1073 ACTIVATE-CLONE PROCESSED FOR PAIR 3423      ,342B
% NDE1073 ACTIVATE-CLONE PROCESSED FOR PAIR 3424      ,342C
% NDE1073 ACTIVATE-CLONE PROCESSED FOR PAIR 3425      ,342D
% NDE1073 ACTIVATE-CLONE PROCESSED FOR PAIR 3426      ,342E
% NDE1073 ACTIVATE-CLONE PROCESSED FOR PAIR 3427      ,342F
/START-FDDRL ----- (2)
//MOD-FDDRL-PARAM PROC-JOBS=*BY-SUBTASKS(STATUS-OUTPUT-PERIOD=5) ----- (3)
//DUMP-PUBSET PUBSET=B10K(SPECIAL-VSN=*YES), SAVE-ENTITY=*DISK-SET(1)----- (4)
% FBD1158 THE PUBSET 'B10K' IS IMPORTED. PROCESSING CONTINUES
//CREATE-JOBS ----- (5)
REQUESTED FDDRL JOBS          FDDRL V21.0A00          2020-08-28 08:27:22
STATEMENT                     REQ-NAME
DISK    DEV-TYPE I-DEV >O-DEV O-DISK FORM          DISK-DATE
DISK-SET TAPE-DEV LOCATION (TAPES)
-----
DUMP-PUBSET          B10K          DMP-B10K
B10K:0  D3435        3428          2020-08-28 07:37:05
B10K:1  D3435        3429          2020-08-28 07:37:05
B10K:2  D3435        342A          2020-08-28 07:37:05
B10K:3  D3435        342B          2020-08-28 07:37:05
B10K:4  D3435        342C          2020-08-28 07:37:05
B10K:5  D3435        342D          2020-08-28 07:37:05
B10K:6  D3435        342E          2020-08-28 07:37:05
B10K:7  D3435        342F          2020-08-28 07:37:05
B10K-1  TAPE-C4
-----
//START-JOBS ----- (6)
% FBD1260 FDDRL SUBTASK '0AGC' FOR DISK/DISK-SET 'B10K-1' STARTED ----- (7)
ACTUAL FDDRL JOB STATUS          FDDRL V21.0A00          2020-08-28 08:32:47
STATEMENT
DISK/-SET  TSN FORM JOB-STA      SEL-HPS PROCESS-HPS      %
-----
DUMP-PUBSET          B10K
B10K:3  0AGC          ACTIVE          1843200      1843200 100%
B10K:4  0AGC          ACTIVE          1843200      638160  34%
B10K:5  0AGC          ACTIVE          1843200      638160  34%
B10K:6  0AGC          ACTIVE          1843200      638160  34%
B10K:7  INACTIVE
B10K-1  0AGC          ACTIVE          9287280
SUMMARY: INACTIVE = 1      ACTIVE = 5      TERMINATED = 3      TOTAL = 9
-----
% FBD1262 FDDRL SUBTASK '0AGC' FOR DISK/DISK-SET 'B10K-1' TERMINATED
-----
FINAL FDDRL JOB PROTOCOL          FDDRL V21.0A00          2020-08-28 08:36:38 (8)
STATEMENT                     REQ-NAME  REQ-DATE  REQ-TIME
DISK  DEV-TYPE  IN  >  OUT          DISK-DATE  TSN      HPS      TERM-MSG
D-SET  F-NAME          TSN      HPS      TERM-MSG

```

TAPE-DEV (TAPES)

```

-----
DUMP-PUBSET      B10K      DMP-B10K      2020-08-28 08:27:47
  B10K.0 D3435    3428 >B10K-1  2020-08-28 07:37:05      1843200
  B10K.1 D3435    3429 >B10K-1  2020-08-28 07:37:05      1843200
  B10K.2 D3435    342A >B10K-1  2020-08-28 07:37:05      1843200
  B10K.3 D3435    342B >B10K-1  2020-08-28 07:37:05      1843200
  B10K.4 D3435    342C >B10K-1  2020-08-28 07:37:05      1843200
  B10K.5 D3435    342D >B10K-1  2020-08-28 07:37:05      1843200
  B10K.6 D3435    342E >B10K-1  2020-08-28 07:37:05      1843200
  B10K.7 D3435    342F >B10K-1  2020-08-28 07:37:05      1843200
  B10K-1 FDDRL.P.B10K.2020-08-28.082700-1      0AGC 14745600
        TAPE-C4 (QA3122,QA3123,QA3124,QA3125,QA3126,QA3127,QA3128
        ,QA3129,QA3130)
-----

```

//SHOW-FDDRL-TAPE-INFORMATION TAPE=QA3122 _____ (9)

```

FDDRL TAPE INFORMATION      FDDRL V21.0A00      2020-08-28 08:37:32

```

```

TAPE VSN      = QA3122      VOL-SEQ      = 0001      FIRST-VOL      = QA3122
  CRE-DATE    = 2020-08-28  EXPIR-DATE    = 2020-08-28  USER-ID      = TSOS
  F-NAME      = FDDRL.P.B10K.2020-08-28.082700-1

```

```

FDDRL = V21.0A      TAPE-FORM = *V17-FORMAT  SAVE-DATE    = 2020-08-28 08:27:47
  PUBSET      = B10K      DISK      =          DISK-DATE    = 2020-08-28 07:37:05
  SAVE-ENT    = *DISK-SET  DISK-SET    = B10K-1      NUM-OF-DISK-SETS = 1
DISK  UNIT  DEV-TYPE  FORM  A-U  NUM-OF-HPS  NUM-OF-512B  SAVED-HPS
-----

```

```

B10K.0 3428 D3435    NK2    6      1843200    7372800    1843200
B10K.1 3429 D3435    NK2    6      1843200    7372800
B10K.2 342A D3435    NK2    6      1843200    7372800
B10K.3 342B D3435    NK2    6      1843200    7372800
B10K.4 342C D3435    NK2    6      1843200    7372800
B10K.5 342D D3435    NK2    6      1843200    7372800
B10K.6 342E D3435    NK2    6      1843200    7372800
B10K.7 342F D3435    NK2    6      1843200    7372800

```

//END _____ (10)

```

% FBD1402 UTILITY ROUTINE 'FDDRL' TERMINATED
%(MSG) % % ESM0216 FUNCTION 'DELETE' STARTED FOR SUBSYSTEM 'FDDRL /V21.0'

```

- (1) The mirror disks are split using an SHC-OSD command. The I/Os are stopped briefly so that a consistent copy of the pubset disks can be created.
- (2) FDDRL is started.
- (3) The FDDRL parameters are set. The status log is displayed every 5 minutes.
- (4) Pubset B10K is saved from the split mirror disks with special notation. All disks are saved to tape as a disk set.
- (5) The FDDRL jobs are generated and displayed in the *REQUESTED FDDRL JOBS* list.
- (6) The FDDRL jobs are started.
- (7) The status of the jobs is displayed every 5 minutes. The disks are simultaneously backed up at multiplexing level 4. The processing progress is consequently the same for all 4 disks. The disks are saved completely because splitting the mirror disks means that no consistent F5 label exists.
- (8) The pubset is saved. The disks processed and the tape used are output in the final log.

(9) The tape information of the FDDRL tape is read and displayed.

(10) The FDDRL run is terminated.

Report file \$TSOS.S.FBD.0AF3.2020-08-28.082747.DMP-B10K

REQUEST REPORT FILE :B10K:\$TSOS.S.FBD.0AF3.2020-08-28.082747.DMP-B10K

++++++ REQUEST COMPLETED OK ++++++

```
-----  
FINAL FDDRL JOB PROTOCOL          FDDRL V21.0A00          2020-08-28 08:36:38  
STATEMENT                          REQ-NAME   REQ-DATE   REQ-TIME  
DISK  DEV-TYPE  IN  >  OUT          DISK-DATE   TSN      HPS    TERM-MSG  
D-SET  F-NAME                               TSN      HPS    TERM-MSG  
      TAPE-DEV (TAPES)  
-----
```

```
DUMP-PUBSET          B10K          DMP-B10K    2020-08-28 08:27:47  
B10K.0 D3435        3428 >B10K-1 2020-08-28 07:37:05          1843200  
B10K.1 D3435        3429 >B10K-1 2020-08-28 07:37:05          1843200  
B10K.2 D3435        342A >B10K-1 2020-08-28 07:37:05          1843200  
B10K.3 D3435        342B >B10K-1 2020-08-28 07:37:05          1843200  
B10K.4 D3435        342C >B10K-1 2020-08-28 07:37:05          1843200  
B10K.5 D3435        342D >B10K-1 2020-08-28 07:37:05          1843200  
B10K.6 D3435        342E >B10K-1 2020-08-28 07:37:05          1843200  
B10K.7 D3435        342F >B10K-1 2020-08-28 07:37:05          1843200  
B10K-1 FDDRL.P.B10K.2020-08-28.082700-1          OAGC 14745600  
      TAPE-C4 (QA3122,QA3123,QA3124,QA3125,QA3126,QA3127,QA3128  
              ,QA3129,QA3130)  
-----
```

```
-----  
FDDRL PARAMETERS          FDDRL V21.0A00          2020-08-28 08:36:38  
PROCESS-JOBS              = *BY-SUBTASK (...)  
TASK-LIM                  = 4  
RUN-PRIO                   = *STD  
STA-OUTPUT-PERIOD         = 5 MINUTES  
SEC-WAIT-TIME             = 1 MINUTE  
JOB-NAME                   = *STD  
RETENT-PERIOD             = 0 DAYS  
LOCATION                    = *STD  
TAPE-FORM                  = *V17-FORMAT  
REPORT-FILE                = *STD  
MONITORING                 = *NO  
KEEP-REQUESTS             = 40 DAYS  
=====
```

SYSOOT PROTOCOL OF SUBTASK OAGC :B10K:\$TSOS.S.FBD.0AGC.2020-08-28.0AF3.B10K-1

```
NAME      TSN  TID    UNP / Q# SPOOLIN      LOGON  
B10K#1    OAGC 000300BA 00/000 2020-08-28.0827 2020-08-28.0827  
(MSG) ? % MARUP44 MOUNT VOLUME 'QA3122' ON DEVICE 'FU' FOR TASK 'OAGC'  
FROM 'CSVTA500'? REPLY (Y=YES; N=NO)?  
(RPL) . Y  
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3122' ON DEVICE 'FU' FOR TASK 'OAGC' COMPLETED  
% FBD1230 ALL PAM PAGES OF DISK 'B10K:0' ON '3428' ARE TO BE SAVED  
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10K:0' ON DEVICE '3428' STARTED  
% FBD1205 FDDRL FUNCTION 'DUMP' FOR DISK/DISK-SET 'B10K-1' STARTED  
(MSG) % % FBD2201 TAPE 'QA3122' ON DEVICE 'FU' AS REEL '001' FOR DUMP OF DISK/DISK-SET  
'B10K-1' USED  
% FBD1230 ALL PAM PAGES OF DISK 'B10K:1' ON '3429' ARE TO BE SAVED
```

```

(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10K:1' ON DEVICE '3429' STARTED
% FBD1230 ALL PAM PAGES OF DISK 'B10K:2' ON '342A' ARE TO BE SAVED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10K:2' ON DEVICE '342A' STARTED
% FBD1230 ALL PAM PAGES OF DISK 'B10K:3' ON '342B' ARE TO BE SAVED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10K:3' ON DEVICE '342B' STARTED
(MSG) % % FBD2208 TAPE 'QA3122' ON DEVICE 'FU' AS REEL '001' FOR DUMP OF DISK/DISK-SET
'B10K-1' COMPLETE
(MSG) ? % MARUP44 MOUNT VOLUME 'QA3123' ON DEVICE 'FU' FOR TASK '0AGC' FROM 'CSVTA500'?
REPLY (Y=YES; N=NO)?
(RPL) . Y
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3123' ON DEVICE 'FU' FOR TASK '0AGC' COMPLETED
(MSG) % % FBD2201 TAPE 'QA3123' ON DEVICE 'FU' AS REEL '002' FOR DUMP OF DISK/DISK-SET
'B10K-1' USED
(MSG) % % FBD2208 TAPE 'QA3123' ON DEVICE 'FU' AS REEL '002' FOR DUMP OF DISK/DISK-SET
'B10K-1' COMPLETE
(MSG) ? % MARUP44 MOUNT VOLUME 'QA3124' ON DEVICE 'FU' FOR TASK '0AGC' FROM 'CSVTA500'?
REPLY (Y=YES; N=NO)?
(RPL) . Y
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3124' ON DEVICE 'FU' FOR TASK '0AGC' COMPLETED
(MSG) % % FBD2201 TAPE 'QA3124' ON DEVICE 'FU' AS REEL '003' FOR DUMP OF DISK/DISK-SET
'B10K-1' USED
(MSG) % % FBD2208 TAPE 'QA3124' ON DEVICE 'FU' AS REEL '003' FOR DUMP OF DISK/DISK-SET
'B10K-1' COMPLETE
(MSG) ? % MARUP44 MOUNT VOLUME 'QA3125' ON DEVICE 'FU' FOR TASK '0AGC' FROM 'CSVTA500'?
REPLY (Y=YES; N=NO)?
(RPL) . Y
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3125' ON DEVICE 'FU' FOR TASK '0AGC' COMPLETED
(MSG) % % FBD2201 TAPE 'QA3125' ON DEVICE 'FU' AS REEL '004' FOR DUMP OF DISK/DISK-SET
'B10K-1' USED
% FBD1403 1843200 HPS OF DISK 'B10K:0' HAVE BEEN PROCESSED
% FBD1230 ALL PAM PAGES OF DISK 'B10K:4' ON '342C' ARE TO BE SAVED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10K:4' ON DEVICE '342C' STARTED
% FBD1403 1843200 HPS OF DISK 'B10K:1' HAVE BEEN PROCESSED
% FBD1230 ALL PAM PAGES OF DISK 'B10K:5' ON '342D' ARE TO BE SAVED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10K:5' ON DEVICE '342D' STARTED
% FBD1403 1843200 HPS OF DISK 'B10K:2' HAVE BEEN PROCESSED
% FBD1230 ALL PAM PAGES OF DISK 'B10K:6' ON '342E' ARE TO BE SAVED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10K:6' ON DEVICE '342E' STARTED
(MSG) % % FBD2208 TAPE 'QA3125' ON DEVICE 'FU' AS REEL '004' FOR DUMP OF DISK/DISK-SET
'B10K-1' COMPLETE
(MSG) ? % MARUP44 MOUNT VOLUME 'QA3126' ON DEVICE 'FU' FOR TASK '0AGC' FROM 'CSVTA500'?
REPLY (Y=YES; N=NO)?
(RPL) . Y
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3126' ON DEVICE 'FU' FOR TASK '0AGC' COMPLETED
(MSG) % % FBD2201 TAPE 'QA3126' ON DEVICE 'FU' AS REEL '005' FOR DUMP OF DISK/DISK-SET
'B10K-1' USED
(MSG) % % FBD2208 TAPE 'QA3126' ON DEVICE 'FU' AS REEL '005' FOR DUMP OF DISK/DISK-SET
'B10K-1' COMPLETE
(MSG) ? % MARUP44 MOUNT VOLUME 'QA3127' ON DEVICE 'FU' FOR TASK '0AGC'
FROM 'CSVTA500'? REPLY (Y=YES; N=NO)?
(RPL) . Y
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3127' ON DEVICE 'FU' FOR TASK '0AGC' COMPLETED
(MSG) % % FBD2201 TAPE 'QA3127' ON DEVICE 'FU' AS REEL '006' FOR DUMP OF DISK/DISK-SET
'B10K-1' USED
(MSG) % % FBD2208 TAPE 'QA3127' ON DEVICE 'FU' AS REEL '006' FOR DUMP OF DISK/DISK-SET
'B10K-1' COMPLETE
(MSG) ? % MARUP44 MOUNT VOLUME 'QA3128' ON DEVICE 'FU' FOR TASK '0AGC' FROM 'CSVTA500'?
REPLY (Y=YES; N=NO)?
(RPL) . Y

```

```
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3128' ON DEVICE 'FU' FOR TASK 'OAGC' COMPLETED
(MSG) % % FBD2201 TAPE 'QA3128' ON DEVICE 'FU' AS REEL '007' FOR DUMP OF DISK/DISK-SET
'B10K-1' USED
(MSG) % % FBD2208 TAPE 'QA3128' ON DEVICE 'FU' AS REEL '007' FOR DUMP OF DISK/DISK-SET
'B10K-1' COMPLETE
(MSG) ? % MARUP44 MOUNT VOLUME 'QA3129' ON DEVICE 'FU' FOR TASK 'OAGC' FROM 'CSVTA500'?
REPLY (Y=YES; N=NO)?
(RPL) . Y
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3129' ON DEVICE 'FU' FOR TASK 'OAGC' COMPLETED
(MSG) % % FBD2201 TAPE 'QA3129' ON DEVICE 'FU' AS REEL '008' FOR DUMP OF DISK/DISK-SET
'B10K-1' USED
% FBD1403 1843200 HPS OF DISK 'B10K:4' HAVE BEEN PROCESSED
% FBD1230 ALL PAM PAGES OF DISK 'B10K:7' ON '342F' ARE TO BE SAVED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10K:7' ON DEVICE '342F' STARTED
% FBD1403 1843200 HPS OF DISK 'B10K:5' HAVE BEEN PROCESSED
% FBD1403 1843200 HPS OF DISK 'B10K:6' HAVE BEEN PROCESSED
% FBD1403 1843200 HPS OF DISK 'B10K:3' HAVE BEEN PROCESSED
(MSG) % % FBD2208 TAPE 'QA3129' ON DEVICE 'FU' AS REEL '008' FOR DUMP OF DISK/DISK-SET
'B10K-1' COMPLETE
(MSG) ? % MARUP44 MOUNT VOLUME 'QA3130' ON DEVICE 'FU' FOR TASK 'OAGC' FROM 'CSVTA500'?
REPLY (Y=YES; N=NO)?
(RPL) . Y
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3130' ON DEVICE 'FU' FOR TASK 'OAGC' COMPLETED
(MSG) % % FBD2201 TAPE 'QA3130' ON DEVICE 'FU' AS REEL '009' FOR DUMP OF DISK/DISK-SET
'B10K-1' USED
% FBD1403 1843200 HPS OF DISK 'B10K:7' HAVE BEEN PROCESSED
% FBD1206 FDDRL FUNCTION 'DUMP' TERMINATED
(MSG) % % FBD2208 TAPE 'QA3130' ON DEVICE 'FU' AS REEL '009' FOR DUMP OF DISK/DISK-SET
'B10K-1' COMPLETE
(MSG) % % EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 5.6475 SEC, USER ID: TSOS, TASK ID:
000300BA
% EXC0419 /LOGOFF AT 0836 ON 2020-08-28 FOR TSN 'OAGC'
% EXC0421 CPU TIME USED: 5.6475
```

Restoring a pubset on new disks

Pubset B10K is taken from a backup (see [chapter "Saving the home pubset from the mirror disks"](#)) and restored on unused disks.

```
//START-FDDRL _____ (1)
```

```
//MOD-FDDRL-PARAM PROC-JOBS=*BY-SUBTASKS(STATUS-OUTPUT-PERIOD=5) _____ (2)
```

```
//RELOAD-PUBSET PUBSET=B10K,TAPE=*BY-VOLUME-CATALOG(  
    FILE-NAME=FDDRL.P.B10K.2020-08-28.082700-1),  
    OVERWRITE-DISK=*UNIT(  
        UNIT=(3428,3429,342A,342B,342C,342D,342E,342F)) _____ (3)
```

```
//CREATE-JOBS _____ (4)
```

```
FDDRL TAPE INFORMATION          FDDRL V21.0A00          2020-08-28 08:48:31  
TAPE VSN      = QA3122      VOL-SEQ      = 0001      FIRST-VOL    = QA3122  
  CRE-DATE    = 2020-08-28  EXPIR-DATE    = 2020-08-28  USER-ID     = TSOS  
  F-NAME      = FDDRL.P.B10K.2020-08-28.082700-1  
FDDRL = V21.0A  TAPE-FORM = *V17-FORMAT  SAVE-DATE   = 2020-08-28 08:27:47  
  PUBSET      = B10K      DISK      =          DISK-DATE   = 2020-08-28 07:37:05  
  SAVE-ENT    = *DISK-SET  DISK-SET  = B10K-1  NUM-OF-DISK-SETS = 1  
DISK  UNIT DEV-TYPE FORM A-U NUM-OF-HPS NUM-OF-512B SAVED-HPS  
-----  
B10K.0 3428 D3435  NK2  6      1843200      7372800      1843200  
B10K.1 3429 D3435  NK2  6      1843200      7372800  
B10K.2 342A D3435  NK2  6      1843200      7372800  
B10K.3 342B D3435  NK2  6      1843200      7372800  
B10K.4 342C D3435  NK2  6      1843200      7372800  
B10K.5 342D D3435  NK2  6      1843200      7372800  
B10K.6 342E D3435  NK2  6      1843200      7372800  
B10K.7 342F D3435  NK2  6      1843200      7372800
```

(1) FDDRL is started.

- (2) The FDDRL parameters are set. The status log is to be displayed every 5 minutes.
- (3) Pubset B10K is to be restored. The unit specification for the output disks causes the disks to be formatted if necessary.
- (4) The FDDRL jobs are created. The tape's meta information is read when this happens.

Extract from the console log for CREATE-JOBS

The console log displays that the input tape has been mounted and that the output disks have been mounted. The operator confirms that the output disks have been mounted (NKVD013).

```

<T ?0AF3-000.084831 % NKVT013 MOUNT TAPE 'QA3122' ON DEVICE 'FT'; (USE='SPECIAL',
WR='OFF',TYPE='TAPE-C4',FDDRL.P.B10K.2020-08-28.082700-1 TAPE 001). '(ETX = YES; MN; N = NO
) ' ?
<T %ROB1-000.084831 % ROB4008 MB-KASSETTE MONTIERT (DEV=FT / TSN=0AF3 / VSN=QA3122)
0AF3 R TM-000.084833.FT (TM)
<T % UCO-000.084833 % NBR0792 QUESTION '000' FROM '0AF3' ANSWERED BY 'TM'. REPLY: 'FT
(TM)'. MSG-ID: 'NKVT013'
<T % TM-000.084833 % NKVT010 TAPE 'QA3122' MOUNTED ON DEVICE 'FT'
<T %0AF3-000.084833 % NKVT017 MOUNTING OF TAPE 'QA3122' ON DEVICE 'FT' FOR TASK '0AF3'
COMPLETED
<D ?0AF3-000.084834 % NKVD013 MOUNT DISK '*UNKNO' ON DEVICE '3428'; (USE='SPECIAL',SYS-
ALLOC='EXCL',OUTPUT DISK FOR FDDRL RELOAD OF 'B10K.0'). '(ETX = YES; N = NO)' ?
AF3 R(C0)-000.084903.
<D % UCO-000.084903 % NBR0792 QUESTION '000' FROM '0AF3' ANSWERED BY '(C0)'. REPLY: ' '.
MSG-ID: 'NKVD013'
<D %0AF3-000.084903 % NKVD017 MOUNTING OF DISK 'F-3428' ON DEVICE '3428' FOR TASK '0AF3'
COMPLETED
<D ?0AF3-000.084904 % NKVD013 MOUNT DISK '*UNKNO' ON DEVICE '3429'; (USE='SPECIAL',SYS-
ALLOC='EXCL',OUTPUT DISK FOR FDDRL RELOAD OF 'B10K.1'). '(ETX = YES; N = NO)' ?
AF3 R(C0)-000.084908.
<D % UCO-000.084908 % NBR0792 QUESTION '000' FROM '0AF3' ANSWERED BY '(C0)'. REPLY: ' '.
MSG-ID: 'NKVD013'
<D %0AF3-000.084908 % NKVD017 MOUNTING OF DISK 'F-3429' ON DEVICE '3429' FOR TASK '0AF3'
COMPLETED
<D ?0AF3-000.084909 % NKVD013 MOUNT DISK '*UNKNO' ON DEVICE '342A'; (USE='SPECIAL',SYS-
ALLOC='EXCL',OUTPUT DISK FOR FDDRL RELOAD OF 'B10K.2'). '(ETX = YES; N = NO)' ?
AF3 R(C0)-000.084915.
<D % UCO-000.084915 % NBR0792 QUESTION '000' FROM '0AF3' ANSWERED BY '(C0)'. REPLY: ' '.
MSG-ID: 'NKVD013'
<D %0AF3-000.084915 % NKVD017 MOUNTING OF DISK 'F-342A' ON DEVICE '342A' FOR TASK '0AF3'
COMPLETED
2020-08-28
<D ?0AF3-000.084916 % NKVD013 MOUNT DISK '*UNKNO' ON DEVICE '342B'; (USE='SPECIAL',SYS-
ALLOC='EXCL',OUTPUT DISK FOR FDDRL RELOAD OF 'B10K.3'). '(ETX = YES; N = NO)' ?
AF3 R(C0)-000.084919.
:
:
<T % TM-000.084936 % NKVT097 UNLOADING OF TAPE 'QA3122' FROM DEVICE 'FT' INITIATED
<T %ROB1-000.084956 % ROB4004 MBK-LAUFWERK ENTLADEN (DEV=FT / VSN=QA3122)
:
:

```

**** End of the console log extract****

```

REQUESTED FDDRL JOBS          FDDRL V21.0A00          2020-08-28 08:49:36  (5)
STATEMENT                     REQ-NAME
DISK      DEV-TYPE I-DEV >O-DEV O-DISK FORM      DISK-DATE
DISK-SET TAPE-DEV LOCATION (TAPES)

```

```

-----
RELOAD-PUBSET      B10K          RLP-B10K
B10K.0  D3435      3428 > 3428          YES 2020-08-28 07:37:05
B10K.1  D3435      3429 > 3429          YES 2020-08-28 07:37:05
B10K.2  D3435      342A > 342A          YES 2020-08-28 07:37:05
B10K.3  D3435      342B > 342B          YES 2020-08-28 07:37:05
B10K.4  D3435      342C > 342C          YES 2020-08-28 07:37:05
B10K.5  D3435      342D > 342D          YES 2020-08-28 07:37:05
B10K.6  D3435      342E > 342E          YES 2020-08-28 07:37:05
B10K.7  D3435      342F > 342F          YES 2020-08-28 07:37:05
B10K-1  TAPE-C4    CSVTA500 (QA3122,QA3123,QA3124,QA3125,QA3126,...)
-----

```

//START-JOBS

(6)

```

% JMS0066 JOB '$B10K#0' ACCEPTED ON 20-08-28 AT 08:50, TSN = 0AGF
% JMS0066 JOB '$B10K#1' ACCEPTED ON 20-08-28 AT 08:50, TSN = 0AGG
% JMS0066 JOB '$B10K#2' ACCEPTED ON 20-08-28 AT 08:50, TSN = 0AGH
% JMS0066 JOB '$B10K#3' ACCEPTED ON 20-08-28 AT 08:50, TSN = 0AGI
% JMS0066 JOB '$B10K#4' ACCEPTED ON 20-08-28 AT 08:50, TSN = 0AGJ
% JMS0066 JOB '$B10K#5' ACCEPTED ON 20-08-28 AT 08:50, TSN = 0AGK
% JMS0066 JOB '$B10K#6' ACCEPTED ON 20-08-28 AT 08:50, TSN = 0AGL
% JMS0066 JOB '$B10K#7' ACCEPTED ON 20-08-28 AT 08:50, TSN = 0AGM
% FBD1260 FDDRL SUBTASK '0AGN' FOR DISK/DISK-SET 'B10K-1' STARTED

```

(5) The FDDRL jobs generated are displayed. The input and output devices involved here are displayed. YES for FORM indicates that the output disks are formatted before they are used.

(6) The FDDRL jobs are started. Initially a batch job for disk formatting is started for each output disk.

SYSOUT log for VOLIN batch task 0AGM

The SYSOUT log shows the progress of formatting for a disk.

/LOGON LOG=

?% JMS0155 BATCH JOB STARTED AT 08:50 ON 2020-08-28 FOR TSN 0AGM, CALLED BY 'TSN 0AF3'. SYSCMD FILE ':B10K:\$TSOS.S.E.0AF3.2020-08-28.08.50.59'

/ADD-PASSWORD PASSWORD=P

/CALL-PROC NAME=:B10K:\$TSOS.S.PROC.0AF3.2020-08-28.08.50.59

,PROCEDURE-PARAMETERS=(VOLUME=B10K.7,UNIT='342F',DEVICE=D3435

,OVERWRITE-DISK=*ANY,FORMAT='NK(2K(6))',FMT=NO),LOGGING=*YES

```
1 1 /SET-PROC-OPT DATA-ESCAPE-CHAR=STD
3 1 /BEGIN-PARAMETER-DECL
4 1 /DECL-PARAMETER VOLUME (INIT-VALUE=*PROMPT) "<vsn>"
5 1 /DECL-PARAMETER UNIT (INIT-VALUE=*PROMPT) "<unit>"
6 1 /DECL-PARAMETER DEVICE (INIT-VALUE='*ANY') "<device>"
7 1 /DECL-PARAMETER OVERWRITE-DISK(INIT-VALUE='*ANY') "**ANY/<vsn>"
8 1 /DECL-PARAMETER FORMAT (INIT-VALUE='UNCHANGED') "K/NK(2K)/NK(4K)"
9 1 /DECL-PARAMETER FMT (INIT-VALUE='NO') "NO/YES"
10 1 /DECL-PARAMETER F1SIZE (INIT-VALUE='3') "<integer>"
11 1 /END-PARAMETER-DECL
31 1 /REMARK PROZEDURPARAMETER:
32 1 /REMARK VOLUME = B10K.7
33 1 /REMARK UNIT = 342F
34 1 /REMARK DEVICE = D3435
35 1 /REMARK OVERWRITE-DISK = *ANY
36 1 /REMARK FORMAT = NK(2K(6))
37 1 /REMARK FMT = NO
38 1 /MODIFY-JOB-OPTIONS OPERATOR-INTERACTION=*YES
39 1 /ASSIGN-SYSDTA TO=*SYSCMD
40 1 /IF (DEVICE = '*ANY')
42 1 /END-IF
43 1 /IF (OVERWRITE-DISK= '*ANY')
44 1 /OVERWRITE-DISK='SCRATCH'
45 1 /END-IF
47 1 /START-VOLIN
```

?% BLS0523 ELEMENT 'NVLTU', VERSION '210', TYPE 'L' FROM LIBRARY ':B10K:\$TSOS.SYSLNK.VOLIN.210' IN PROCESS

?% BLS0990 PROCESSING OF REP FILE ':B10K:\$TSOS.SYSREP.VOLIN.210' STARTED

?% BLS0980 REP COMMENT: '*%VOLIN 21.0A00 VER: 1 (04/06/20)

VOLIN V21.0A00 #00.01'

?% BLS0980 REP COMMENT: '*% RMS Lader wurde neu installiert'

?% BLS0980 REP COMMENT: '*% VOLIN V21.0A00 #00.01 (04/06/20)'

?% BLS0524 LLM 'NVLTU', VERSION '21.0A00' OF '2020-06-04 16:15:14' LOADED

?% BLS0551 COPYRIGHT (C) FUJITSU TECHNOLOGY SOLUTIONS 2020. ALL RIGHTS RESERVED

(MSG) % % BLS0519 PROGRAM 'NVLTU' LOADED

```
?% NVL0000 VOLIN VERSION V21.0A00 READY
?% NVL0002 ENTER VSN AND DEVICE TYPE AND / OR UNIT, OR TERMINATE VOLIN WITH 'END'
  VSN=B10K.7, DEVICE=D3435, CHECK=SCRATCH, UNIT=342F
?% NVL0003 ENTER ADDITIONAL INITIALIZATION PARAMETERS OR TERMINATE INPUT WITH
'EOT=YES'
  FORMAT=NK(2K(6)) , FMT=NO, F1SIZE=3, EOT=Y
?% NVL0031 INITIALIZATION STARTED FOR VOLUME 'B10K.7' ON UNIT '342F' IN FORMAT
'NK2(A-U=6)'
?% NVL0017 INITIALIZATION OF VOLUME 'B10K.7' ON UNIT '342F' COMPLETED. VOLUME
FORMAT: 'NK2(A-U=6)'
?% NVL0002 ENTER VSN AND DEVICE TYPE AND / OR UNIT, OR TERMINATE VOLIN WITH 'END'
  (MSG) % % NVL0042 VOLIN ACQUIRES DISK DEVICE AND DISK PACK FOR INITIALIZATION
OR REPAIR OF DISK 'B10K.7'
  (MSG) % % NVL0010 DISK 'B10K.7': STANDARD VOLUME LABEL INVALID
  (MSG) ? % NVL0032 OVERWRITE NON-STANDARD DISK ' ' ON DEVICE '342F' IN NEW
FORMAT 'NK2(A-U=6)'? REPLY (Y=YES; N=NO)? (RPL) . Y
  (MSG) % % NVL0031 INITIALIZATION STARTED FOR VOLUME 'B10K.7' ON UNIT '342F' IN
FORMAT 'NK2(A-U=6)'
  (MSG) % % NVL0017 INITIALIZATION OF VOLUME 'B10K.7' ON UNIT '342F' COMPLETED.
VOLUME FORMAT: 'NK2(A-U=6)'
  END
      51 1 /ASSIGN-SYSDTA TO=*PRIMARY
      52 1 /EXIT-PROC ERROR=*YES "to keep S.OUT file"
?% CMD0205 ERROR IN PRECEDING COMMAND OR PROGRAM AND PROCEDURE STEP TERMINATION:
COMMANDS WILL BE IGNORED UNTIL /SET-JOB-STEP OR /EXIT-JOB OR /LOGOFF IS RECOGNIZED
  /SET-JOB-STEP
  /EXIT-JOB SYSTEM-OUTPUT=*ALL
  (MSG) % % EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 0.0638 SEC, USER ID: TSOS,
TASK ID: 000300F4
?% EXC0419 /LOGOFF AT 0851 ON 2020-08-28 FOR TSN '0AGM'
?% EXC0421 CPU TIME USED: 0.0638
```

**** *End of the SYSOUT log* ****

ACTUAL FDDRL JOB STATUS FDDRL V21.0A00 2020-08-28 08:55:54 - (7)

STATEMENT

DISK/-SET	TSN	FORM	JOB-STA	SEL-HPS	PROCESS-HPS	%

RELOAD-PUBSET	B10K					
B10K.4	0AGN		ACTIVE	1843200	293280	15%
B10K.5	0AGN		ACTIVE	1843200	293200	15%
B10K.6	0AGN		ACTIVE	1843200	293200	15%
B10K.7		0AGM FORM		1843200		
B10K-1	0AGN		ACTIVE		8252480	
SUMMARY: INACTIVE = 1 ACTIVE = 3 TERMINATED = 4 TOTAL = 9						

% FBD1262 FDDRL SUBTASK '0AGN' FOR DISK/DISK-SET 'B10K-1' TERMINATED

----- (8)

FINAL FDDRL JOB PROTOCOL FDDRL V21.0A00 2020-08-28 08:59:27

STATEMENT

DISK	DEV-TYPE	IN	>	OUT	DISK-DATE	TSN	HPS	TERM-MSG
D-SET	F-NAME					TSN	HPS	TERM-MSG
TAPE-DEV (TAPES)								

RELOAD-PUBSET	B10K		RLP-B10K		2020-08-28 08:50:54			
B10K.0	D3435	>	3428	2020-08-28 07:37:04	1843200			
B10K.1	D3435	>	3429	2020-08-28 07:37:04	1843200			
B10K.2	D3435	>	342A	2020-08-28 07:37:04	1843200			
B10K.3	D3435	>	342B	2020-08-28 07:37:04	1843200			
B10K.4	D3435	>	342C	2020-08-28 07:37:04	1843200			
B10K.5	D3435	>	342D	2020-08-28 07:37:04	1843200			
B10K.6	D3435	>	342E	2020-08-28 07:37:04	1843200			
B10K.7	D3435	>	342F	2020-08-28 07:37:04	1843200			
B10K-1	FDDRL.P.B10K.2020-08-28.082700-1				0AGN	14745600		
TAPE-C4 (QA3122,QA3123,QA3124,QA3125,QA3126,QA3127,QA3128,QA3129,QA3130)								

//END

% FBD1402 UTILITY ROUTINE 'FDDRL' TERMINATED

%(MSG) % % ESM0216 FUNCTION 'DELETE' STARTED FOR SUBSYSTEM 'FDDRL /V21.0'

-
- (7) The status log shows that restoration of the disks has begun. The already terminated 3 FDDRL jobs are no more displayed.
 - (8) The final log shows that the pubset has been restored successfully to disks from 3428 until 342F . During restoration the disks' TIME-STAMP was reduced by one second. This enables the disks of the restored pubsets to be distinguished from the original disks.


```

1' )
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10K.1' ON DEVICE '3429' STARTED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10K.2' ON DEVICE '342A' STARTED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10K.3' ON DEVICE '342B' STARTED
(MSG) % % FBD2304 TAPE 'QA3122' ON DEVICE 'FV' AS REEL '001' FOR RELOAD OF DISK/DISK-SET
'B10K-1' RELOADED
(MSG) ? % NKVT013 MOUNT TAPE 'QA3123' ON DEVICE 'FV'; (USE='SPECIAL',WR='OFF',TYPE='TAPE-C4',
FDDRL.P.B10K.2020-08-28.082700-1 TAPE 002). '(ETX = YES; MN; N = NO )'?
(MSG) ? FV (TM)
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3123' ON DEVICE 'FV' FOR TASK '0AGN' COMPLETED
(MSG) % % FBD2303 TAPE 'QA3123' ON DEVICE 'FV' AS REEL '002' FOR RELOAD OF DISK/DISK-SET
'B10K-1' USED
(MSG) % % FBD4015 PREPARE TAPE 'QA3124' FOR FDDRL RUN (ACCESS='INPUT', DISK/DISK-SET='B10K-
1' )
(MSG) % % FBD2304 TAPE 'QA3123' ON DEVICE 'FV' AS REEL '002' FOR RELOAD OF DISK/DISK-SET
'B10K-1' RELOADED
(MSG) ? % NKVT013 MOUNT TAPE 'QA3124' ON DEVICE 'FV'; (USE='SPECIAL',WR='OFF',TYPE='TAPE-C4',
FDDRL.P.B10K.2020-08-28.082700-1 TAPE 003). '(ETX = YES; MN; N = NO )'?
(MSG) ? FV (TM)
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3124' ON DEVICE 'FV' FOR TASK '0AGN' COMPLETED
(MSG) % % FBD2303 TAPE 'QA3124' ON DEVICE 'FV' AS REEL '003' FOR RELOAD OF DISK/DISK-SET
'B10K-1' USED
(MSG) % % FBD4015 PREPARE TAPE 'QA3125' FOR FDDRL RUN (ACCESS='INPUT', DISK/DISK-SET='B10K-
1' )
(MSG) % % FBD2304 TAPE 'QA3124' ON DEVICE 'FV' AS REEL '003' FOR RELOAD OF DISK/DISK-SET
'B10K-1' RELOADED
(MSG) ? % NKVT013 MOUNT TAPE 'QA3125' ON DEVICE 'FV'; (USE='SPECIAL',WR='OFF',TYPE='TAPE-C4',
FDDRL.P.B10K.2020-08-28.082700-1 TAPE 004). '(ETX = YES; MN; N = NO )'?
(MSG) ? FV (TM)
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3125' ON DEVICE 'FV' FOR TASK '0AGN' COMPLETED
(MSG) % % FBD2303 TAPE 'QA3125' ON DEVICE 'FV' AS REEL '004' FOR RELOAD OF DISK/DISK-SET
'B10K-1' USED
(MSG) % % FBD4015 PREPARE TAPE 'QA3126' FOR FDDRL RUN (ACCESS='INPUT', DISK/DISK-SET='B10K-
1' )
% FBD1403 1843200 HPS OF DISK 'B10K.0' HAVE BEEN PROCESSED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10K.4' ON DEVICE '342C' STARTED
% FBD1403 1843200 HPS OF DISK 'B10K.1' HAVE BEEN PROCESSED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10K.5' ON DEVICE '342D' STARTED
% FBD1403 1843200 HPS OF DISK 'B10K.2' HAVE BEEN PROCESSED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10K.6' ON DEVICE '342E' STARTED
(MSG) % % FBD2304 TAPE 'QA3125' ON DEVICE 'FV' AS REEL '004' FOR RELOAD OF DISK/DISK-SET
'B10K-1' RELOADED
(MSG) ? % NKVT013 MOUNT TAPE 'QA3126' ON DEVICE 'FV'; (USE='SPECIAL',WR='OFF',TYPE='TAPE-C4',
FDDRL.P.B10K.2020-08-28.082700-1 TAPE 005). '(ETX = YES; MN; N = NO )'?
(MSG) ? FV (TM)
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3126' ON DEVICE 'FV' FOR TASK '0AGN' COMPLETED
(MSG) % % FBD2303 TAPE 'QA3126' ON DEVICE 'FV' AS REEL '005' FOR RELOAD OF DISK/DISK-SET
'B10K-1' USED
(MSG) % % FBD4015 PREPARE TAPE 'QA3127' FOR FDDRL RUN (ACCESS='INPUT', DISK/DISK-SET='B10K-
1' )
(MSG) % % FBD2304 TAPE 'QA3126' ON DEVICE 'FV' AS REEL '005' FOR RELOAD OF DISK/DISK-SET
'B10K-1' RELOADED
(MSG) ? % NKVT013 MOUNT TAPE 'QA3127' ON DEVICE 'FV'; (USE='SPECIAL',WR='OFF',TYPE='TAPE-C4',
FDDRL.P.B10K.2020-08-28.082700-1 TAPE 006). '(ETX = YES; MN; N = NO )'?
(MSG) ? FV (TM)
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3127' ON DEVICE 'FV' FOR TASK '0AGN' COMPLETED
(MSG) % % FBD2303 TAPE 'QA3127' ON DEVICE 'FV' AS REEL '006' FOR RELOAD OF DISK/DISK-SET
'B10K-1' USED
(MSG) % % FBD4015 PREPARE TAPE 'QA3128' FOR FDDRL RUN (ACCESS='INPUT', DISK/DISK-SET='B10K-

```

```
1')
(MSG) % % FBD2304 TAPE 'QA3127' ON DEVICE 'FV' AS REEL '006' FOR RELOAD OF DISK/DISK-SET
'B10K-1' RELOADED
(MSG) ? % NKVT013 MOUNT TAPE 'QA3128' ON DEVICE 'FV'; (USE='SPECIAL',WR='OFF',TYPE='TAPE-C4',
FDDRL.P.B10K.2020-08-28.082700-1 TAPE 007). '(ETX = YES; MN; N = NO )'?
(MSG) ? FV (TM)
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3128' ON DEVICE 'FV' FOR TASK '0AGN' COMPLETED
(MSG) % % FBD2303 TAPE 'QA3128' ON DEVICE 'FV' AS REEL '007' FOR RELOAD OF DISK/DISK-SET
'B10K-1' USED
(MSG) % % FBD4015 PREPARE TAPE 'QA3129' FOR FDDRL RUN (ACCESS='INPUT', DISK/DISK-SET='B10K-
1')
(MSG) % % FBD2304 TAPE 'QA3128' ON DEVICE 'FV' AS REEL '007' FOR RELOAD OF DISK/DISK-SET
'B10K-1' RELOADED
(MSG) ? % NKVT013 MOUNT TAPE 'QA3129' ON DEVICE 'FV'; (USE='SPECIAL',WR='OFF',TYPE='TAPE-C4',
FDDRL.P.B10K.2020-08-28.082700-1 TAPE 008). '(ETX = YES; MN; N = NO )'?
(MSG) ? FV (TM)
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3129' ON DEVICE 'FV' FOR TASK '0AGN' COMPLETED
(MSG) % % FBD2303 TAPE 'QA3129' ON DEVICE 'FV' AS REEL '008' FOR RELOAD OF DISK/DISK-SET
'B10K-1' USED
(MSG) % % FBD4015 PREPARE TAPE 'QA3130' FOR FDDRL RUN (ACCESS='INPUT', DISK/DISK-SET='B10K-
1')
% FBD1403 1843200 HPS OF DISK 'B10K.4' HAVE BEEN PROCESSED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10K.7' ON DEVICE '342F' STARTED
% FBD1403 1843200 HPS OF DISK 'B10K.5' HAVE BEEN PROCESSED
% FBD1403 1843200 HPS OF DISK 'B10K.6' HAVE BEEN PROCESSED
% FBD1403 1843200 HPS OF DISK 'B10K.3' HAVE BEEN PROCESSED
(MSG) % % FBD2304 TAPE 'QA3129' ON DEVICE 'FV' AS REEL '008' FOR RELOAD OF DISK/DISK-SET
'B10K-1' RELOADED
(MSG) ? % NKVT013 MOUNT TAPE 'QA3130' ON DEVICE 'FV'; (USE='SPECIAL',WR='OFF',TYPE='TAPE-C4',
FDDRL.P.B10K.2020-08-28.082700-1 TAPE 009). '(ETX = YES; MN; N = NO )'?
(MSG) ? FV (TM)
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3130' ON DEVICE 'FV' FOR TASK '0AGN' COMPLETED
(MSG) % % FBD2303 TAPE 'QA3130' ON DEVICE 'FV' AS REEL '009' FOR RELOAD OF DISK/DISK-SET
'B10K-1' USED
% FBD1403 1843200 HPS OF DISK 'B10K.7' HAVE BEEN PROCESSED
% FBD1206 FDDRL FUNCTION 'RELOAD' TERMINATED
(MSG) % % FBD2304 TAPE 'QA3130' ON DEVICE 'FV' AS REEL '009' FOR RELOAD OF DISK/DISK-SET
'B10K-1' RELOADED
(MSG) % % EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 5.5912 SEC, USER ID: TSOS, TASK ID:
000300DD
% EXC0419 /LOGOFF AT 0859 ON 2020-08-28 FOR TSN '0AGN'
% EXC0421 CPU TIME USED: 5.5912
```

Saving the home pubset online

Home pubset B10S is saved online.

Log of the FDDRL main task

```
/START-FDDRL _____ (1)
//MOD-FDDRL-PARAM PROC-JOBS=*BY-SUBTASKS _____ (2)
//DUMP-PUBSET PUBSET=*HOME(DEFAULT-PUBSET=ABN2,TIME-LIMIT=10),
      SAVE-ENTITY=*DISK-SET(NUMBER-OF-DISK-SETS=3) _____ (3)
//CREATE-JOBS _____ (4)
REQUESTED FDDRL JOBS          FDDRL V21.0A00          2020-08-27 06:38:51
STATEMENT                      REQ-NAME
DISK      DEV-TYPE I-DEV >O-DEV O-DISK FORM          DISK-DATE
DISK-SET TAPE-DEV LOCATION (TAPES)  -----
----- DUMP-
PUBSET      B10S          DMP-B10S      B10S.
0   D3435      900E          2020-08-19 10:01:18
      B10S.1      D3435      900F          2020-08-19 10:01:18
      B10S.2      D3435      9068          2020-08-19 10:01:18
      B10S.3      D3435      9069          2020-08-19 10:01:18
      B10S.4      D3435      906A          2020-08-19 10:01:18
      B10S.5      D3435      906B          2020-08-19 10:01:18
      B10S-1      TAPE-C4
      B10S-2      TAPE-C4
      B10S-3      TAPE-C4
-----
//START-JOBS _____ (5)
% FBD1260 FDDRL SUBTASK '59P7' FOR DISK/DISK-SET 'B10S-1' STARTED
% FBD1260 FDDRL SUBTASK '59P8' FOR DISK/DISK-SET 'B10S-2' STARTED
% FBD1260 FDDRL SUBTASK '59P9' FOR DISK/DISK-SET 'B10S-3' STARTED
% FBD1281 WRITE LOCK FOR PUBSET 'B10S' IS ACTIVATED _____ (6)
%(MSG) % % FBD1281 WRITE LOCK FOR PUBSET 'B10S' IS ACTIVATED
% FBD1262 FDDRL SUBTASK '59P7' FOR DISK/DISK-SET 'B10S-1' TERMINATED
% FBD1262 FDDRL SUBTASK '59P8' FOR DISK/DISK-SET 'B10S-2' TERMINATED
% FBD1262 FDDRL SUBTASK '59P9' FOR DISK/DISK-SET 'B10S-3' TERMINATED
% FBD1285 WRITE LOCK FOR DISK/PUBSET 'B10S' IS REMOVED _____ (7)
%(MSG) % % FBD1285 WRITE LOCK FOR DISK/PUBSET 'B10S' IS REMOVED
-----
FINAL FDDRL JOB PROTOCOL          FDDRL V21.0A00          2020-08-27 06:42:30 (8)

STATEMENT                      REQ-NAME  REQ-DATE  REQ-TIME
DISK  DEV-TYPE  IN  >  OUT          DISK-DATE  TSN      HPS    TERM-MSG
D-SET  F-NAME                      TSN      HPS    TERM-MSG
      TAPE-DEV (TAPES)
-----
DUMP-PUBSET          B10S          DMP-B10S  2020-08-27 06:39:32
```

B10S.0	D3435	900E	>B10S-1	2020-08-19	10:01:18	1345360
B10S.1	D3435	900F	>B10S-1	2020-08-19	10:01:18	1330240
B10S.2	D3435	9068	>B10S-2	2020-08-19	10:01:18	1162880
B10S.3	D3435	9069	>B10S-2	2020-08-19	10:01:18	1171360
B10S.4	D3435	906A	>B10S-3	2020-08-19	10:01:18	1119600
B10S.5	D3435	906B	>B10S-3	2020-08-19	10:01:18	1123680
B10S-1	FDDRL.P.B10S.2020-08-27.063900-1					59P7 2675600
	TAPE-C4 (QA3093)					
B10S-2	FDDRL.P.B10S.2020-08-27.063900-2					59P8 2334240
	TAPE-C4 (QA3094)					
B10S-3	FDDRL.P.B10S.2020-08-27.063900-3					59P9 2243280
	TAPE-C4 (QA3095)					

//END

% FBD1402 UTILITY ROUTINE 'FDDRL' TERMINATED

%(MSG) % % ESM0216 FUNCTION 'DELETE' STARTED FOR SUBSYSTEM 'FDDRL /V21.0'

- (1) FDDRL is started.
- (2) The FDDRL parameters are set.
- (3) Home pubset BURX is saved online. The save is backup onto 3 disk sets (the pubset comprises 6 disks). The write lock for the home pubset is to be canceled after at most 10 minutes.
- (4) The FDDRL job is created and displayed by means of the output "REQUESTED FDDRL JOBS".
- (5) The job is started.
- (6) After all subtasks (one subtask per disk set) for the save run have been started, message FBD1281 shows that a write lock is set for the home pubset starting now.
- (7) The subtasks of the online save have been completed. The subsequent message FBD1285 shows that the write lock for the home set has been canceled.
- (8) The final log shows that home pubset B10S has been saved successfully online. The disks of the pubset, their assignment to the disk sets and the disk set backup with the volumes used are listed.

Report file :ABN2:\$TSOS.S.FBD.59MV.2020-08-27.063932.DMP-B10S

REQUEST REPORT FILE :ABN2:\$TSOS.S.FBD.59MV.2020-08-27.063932.DMP-B10S

++++++ REQUEST COMPLETED OK ++++++

```

-----
FINAL FDDRL JOB PROTOCOL          FDDRL V21.0A00          2020-08-27 06:42:30
STATEMENT                          REQ-NAME  REQ-DATE  REQ-TIME
DISK  DEV-TYPE  IN  >  OUT          DISK-DATE  TSN      HPS      TERM-MSG
D-SET  F-NAME                                TSN      HPS      TERM-MSG
      TAPE-DEV (TAPES)
-----

```

```

-----
DUMP-PUBSET          B10S          DMP-B10S          2020-08-27 06:39:32
B10S.0 D3435        900E >B10S-1 2020-08-19 10:01:18          1345360
B10S.1 D3435        900F >B10S-1 2020-08-19 10:01:18          1330240
B10S.2 D3435        9068 >B10S-2 2020-08-19 10:01:18          1162880
B10S.3 D3435        9069 >B10S-2 2020-08-19 10:01:18          1171360
B10S.4 D3435        906A >B10S-3 2020-08-19 10:01:18          1119600
B10S.5 D3435        906B >B10S-3 2020-08-19 10:01:18          1123680
B10S-1 FDDRL.P.B10S.2020-08-27.063900-1          59P7  2675600
      TAPE-C4 (QA3093)
B10S-2 FDDRL.P.B10S.2020-08-27.063900-2          59P8  2334240
      TAPE-C4 (QA3094)
B10S-3 FDDRL.P.B10S.2020-08-27.063900-3          59P9  2243280
      TAPE-C4 (QA3095)
-----

```

```

-----
FDDRL PARAMETERS          FDDRL V21.0A00          2020-08-27 06:42:30
PROCESS-JOBS              = *BY-SUBTASK (...)
TASK-LIM                  = 4
RUN-PRIO                  = *STD
STA-OUTPUT-PERIOD        = 5 MINUTES
SEC-WAIT-TIME             = 1 MINUTE
JOB-NAME                  = *STD
RETENT-PERIOD            = 0 DAYS
LOCATION                   = *STD
TAPE-FORM                 = *V16-FORMAT
REPORT-FILE              = *STD
MONITORING                = *NO
KEEP-REQUESTS            = 40 DAYS
SEL-BLOCKS               = *USED-BLOCKS
UNLOAD-TAPE              = *YES
FORM-DISK                 = *STD
BLOCK-LIM                = *NONE
DEF-PUBSET               = ABN2
TIME-LIM                 = 10 MINUTES
-----

```

SYSOUT PROTOCOL OF SUBTASK 59P7 :ABN2:\$TSOS.S.FBD.59P7.2020-08-27.59MV.B10S-1

```

NAME      TSN  TID      UNP / Q#  SPOOLIN      LOGON
B10S#1    59P7 0032004A  00/000  2020-08-27.0639 2020-08-27.0639
(MSG) ?   %   MARUP44 MOUNT VOLUME 'QA3093' ON DEVICE 'FP' FOR TASK '59P7' FROM 'CSVTA500'?
REPLY (Y=YES; N=NO)? ----- (9)
(RPL) . Y
(MSG) %   %   NKVT017 MOUNTING OF TAPE 'QA3093' ON DEVICE 'FP' FOR TASK '59P7' COMPLETED
(MSG) %   %   FBD1271 FDDRL JOB FOR DISK 'B10S.0' ON DEVICE '900E' STARTED
%   FBD1205 FDDRL FUNCTION 'DUMP' FOR DISK/DISK-SET 'B10S-1' STARTED ----- (10)
%   FBD2201 TAPE 'QA3093' ON DEVICE 'FP' AS REEL '001' FOR DUMP OF DISK/DISK-SET 'B10S-1'

```

USED ----- (11)

FDDRL TAPE INFORMATION FDDRL V21.0A00 2020-08-27 06:39:32

TAPE VSN = QA3093 VOL-SEQ = 0001 FIRST-VOL = QA3093
CRE-DATE = 2020-08-27 EXPIR-DATE = 2020-08-27 USER-ID = TSOS
BLOCK-LEN = 165888 REC-FORM = U
F-NAME = FDDRL.P.B10S.2020-08-27.063900-1

FDDRL = V21.0A TAPE-FORM = *V16-FORMAT SAVE-DATE = 2020-08-27 06:39:32
PUBSET = B10S DISK = DISK-DATE = 2020-08-19 10:01:18
SAVE-ENT = *DISK-SET DISK-SET = B10S-1 NUM-OF-DISK-SETS = 3
SW-COMP = *NO SEL-BLOCKS = *USED-BLOCKS

DISK UNIT DEV-TYPE FORM A-U NUM-OF-HPS NUM-OF-512B SAVED-HPS

B10S.0 900E D3435 NK2 6 1966080 7864320
B10S.1 900F D3435 NK2 6 1966080 7864320
B10S.2 9068 D3435 NK2 6 1966080 7864320
B10S.3 9069 D3435 NK2 6 1966080 7864320
B10S.4 906A D3435 NK2 6 1966080 7864320
B10S.5 906B D3435 NK2 6 1966080 7864320

(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10S.1' ON DEVICE '900F' STARTED ----- (12)

% FBD1403 1330240 HPS OF DISK 'B10S.1' HAVE BEEN PROCESSED
% FBD1000 'MN,NOWAIT,WAIT:' '900F' '16426' '202' ''
% FBD1403 1345360 HPS OF DISK 'B10S.0' HAVE BEEN PROCESSED
% FBD1000 'MN,NOWAIT,WAIT:' '900E' '16550' '267' ''
% FBD2208 TAPE 'QA3093' ON DEVICE 'FP' AS REEL '001' FOR DUMP OF DISK/DISK-SET 'B10S-1'

COMPLETE ----- (13)

% FBD1000 'MN,NOWAIT,WAIT:' 'FP' '454' '32991' ''
% FBD1206 FDDRL FUNCTION 'DUMP' TERMINATED

(MSG) % % EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 1.4228 SEC, USER ID: TSOS, TASK ID: 0032004A

% EXC0419 /LOGOFF AT 0642 ON 2020-08-27 FOR TSN '59P7'
% EXC0421 CPU TIME USED: 1.4228

=====

SYSOUT PROTOCOL OF SUBTASK 59P8 :ABN2:\$TSOS.S.FBD.59P8.2020-08-27.59MV.B10S-2

NAME TSN TID UNP / Q# SPOOLIN LOGON
B10S#2 59P8 0036008C 00/000 2020-08-27.0639 2020-08-27.0639

(MSG) ? % MARUP44 MOUNT VOLUME 'QA3094' ON DEVICE 'FT' FOR TASK '59P8' FROM 'CSVTA500'?

REPLY (Y=YES; N=NO)? ----- (14)

(RPL) . Y

(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3094' ON DEVICE 'FT' FOR TASK '59P8' COMPLETED

(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10S.2' ON DEVICE '9068' STARTED

% FBD1205 FDDRL FUNCTION 'DUMP' FOR DISK/DISK-SET 'B10S-2' STARTED

% FBD2201 TAPE 'QA3094' ON DEVICE 'FT' AS REEL '001' FOR DUMP OF DISK/DISK-SET 'B10S-2'

USED ----- (15)

FDDRL TAPE INFORMATION FDDRL V21.0A00 2020-08-27 06:39:32

TAPE VSN = QA3094 VOL-SEQ = 0001 FIRST-VOL = QA3094
CRE-DATE = 2020-08-27 EXPIR-DATE = 2020-08-27 USER-ID = TSOS
BLOCK-LEN = 165888 REC-FORM = U
F-NAME = FDDRL.P.B10S.2020-08-27.063900-2

FDDRL = V21.0A TAPE-FORM = *V16-FORMAT SAVE-DATE = 2020-08-27 06:39:32
PUBSET = B10S DISK = DISK-DATE = 2020-08-19 10:01:18
SAVE-ENT = *DISK-SET DISK-SET = B10S-2 NUM-OF-DISK-SETS = 3
SW-COMP = *NO SEL-BLOCKS = *USED-BLOCKS

DISK UNIT DEV-TYPE FORM A-U NUM-OF-HPS NUM-OF-512B SAVED-HPS

B10S.0 900E D3435 NK2 6 1966080 7864320

```

B10S.1 900F D3435   NK2   6   1966080   7864320
B10S.2 9068 D3435   NK2   6   1966080   7864320
B10S.3 9069 D3435   NK2   6   1966080   7864320
B10S.4 906A D3435   NK2   6   1966080   7864320
B10S.5 906B D3435   NK2   6   1966080   7864320
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10S.3' ON DEVICE '9069' STARTED
% FBD1403 1162880 HPS OF DISK 'B10S.2' HAVE BEEN PROCESSED
% FBD1000 'MN,NOWAIT,WAIT:' '9068' '14442' '94' ''
% FBD1403 1171360 HPS OF DISK 'B10S.3' HAVE BEEN PROCESSED ----- (16)
% FBD1000 'MN,NOWAIT,WAIT:' '9069' '14542' '100' ''
% FBD2208 TAPE 'QA3094' ON DEVICE 'FT' AS REEL '001' FOR DUMP OF DISK/DISK-SET 'B10S-
2'COMPLETE ----- (17)
% FBD1000 'MN,NOWAIT,WAIT:' 'FT' '169' '29009' ''
% FBD1206 FDDRL FUNCTION 'DUMP' TERMINATED
(MSG) % % EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 1.2976 SEC, USER ID: TSOS, TASK ID:
0036008C
% EXC0419 /LOGOFF AT 0642 ON 2020-08-27 FOR TSN '59P8'
% EXC0421 CPU TIME USED: 1.2976

```

SYSOUT PROTOCOL OF SUBTASK 59P9 :ABN2:\$TSOS.S.FBD.59P9.2020-08-27.59MV.B10S-3

```

NAME      TSN  TID      UNP / Q#  SPOOLIN      LOGON
B10S#3    59P9 000500B2  00/000 2020-08-27.0639 2020-08-27.0639
(MSG) ? % MARUP44 MOUNT VOLUME 'QA3095' ON DEVICE 'FR' FOR TASK '59P9' FROM 'CSVTA500'?
REPLY (Y=YES; N=NO)?

```

```

(RPL) . Y
(MSG) % % NKVT017 MOUNTING OF TAPE 'QA3095' ON DEVICE 'FR' FOR TASK '59P9' COMPLETED
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10S.4' ON DEVICE '906A' STARTED
% FBD1205 FDDRL FUNCTION 'DUMP' FOR DISK/DISK-SET 'B10S-3' STARTED
% FBD2201 TAPE 'QA3095' ON DEVICE 'FR' AS REEL '001' FOR DUMP OF DISK/DISK-SET 'B10S-3'
USED

```

```

FDDRL TAPE INFORMATION          FDDRL V21.0A00          2020-08-27 06:39:32
TAPE VSN      = QA3095          VOL-SEQ      = 0001          FIRST-VOL    = QA3095
  CRE-DATE    = 2020-08-27      EXPIR-DATE   = 2020-08-27      USER-ID      = TSOS
  BLOCK-LEN   = 165888          REC-FORM     = U
  F-NAME      = FDDRL.P.B10S.2020-08-27.063900-3
FDDRL = V21.0A  TAPE-FORM = *V16-FORMAT SAVE-DATE = 2020-08-27 06:39:32
  PUBSET      = B10S  DISK      =          DISK-DATE   = 2020-08-19 10:01:18
  SAVE-ENT    = *DISK-SET DISK-SET = B10S-3  NUM-OF-DISK-SETS = 3
  SW-COMP     = *NO    SEL-BLOCKS = *USED-BLOCKS
DISK  UNIT  DEV-TYPE  FORM  A-U  NUM-OF-HPS  NUM-OF-512B  SAVED-HPS
-----

```

```

B10S.0 900E D3435   NK2   6   1966080   7864320
B10S.1 900F D3435   NK2   6   1966080   7864320
B10S.2 9068 D3435   NK2   6   1966080   7864320
B10S.3 9069 D3435   NK2   6   1966080   7864320
B10S.4 906A D3435   NK2   6   1966080   7864320
B10S.5 906B D3435   NK2   6   1966080   7864320
(MSG) % % FBD1271 FDDRL JOB FOR DISK 'B10S.5' ON DEVICE '906B' STARTED
% FBD1403 1119600 HPS OF DISK 'B10S.4' HAVE BEEN PROCESSED
% FBD1000 'MN,NOWAIT,WAIT:' '906A' '13762' '233' ''
% FBD1403 1123680 HPS OF DISK 'B10S.5' HAVE BEEN PROCESSED
% FBD1000 'MN,NOWAIT,WAIT:' '906B' '13865' '181' ''
% FBD2208 TAPE 'QA3095' ON DEVICE 'FR' AS REEL '001' FOR DUMP OF DISK/DISK-SET 'B10S-3'
COMPLETE
% FBD1000 'MN,NOWAIT,WAIT:' 'FR' '308' '27733' ''
% FBD1206 FDDRL FUNCTION 'DUMP' TERMINATED
(MSG) % % EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 1.2305 SEC, USER ID: TSOS, TASK ID:
000500B2

```

```
% EXC0419 /LOGOFF AT 0642 ON 2020-08-27 FOR TSN '59P9'  
% EXC0421 CPU TIME USED: 1.2305
```

- (9) Mount message for the first backup volume (QA3093).
- (10) The backup of disk set B10S-1 has been started.
- (11) The backup to the first volume begins. The first volume has been mounted.
- (12) The backup begins after message FBD1271 has been issued for each subtask.
- (13) The backup to the first volume has been completed.
- (14) Mount message for the next backup volume (QA3094).
- (15) The backup to the second volume begins.
The backup then extends over volumes three and four.
- (16) Disks B10S.3 and B10S.4 have been backed up.
- (17) The backup to the fourth volume has been completed. As all disks of the disk set have now been backed up, the subtask terminates.

Extract from the CONSOLE log

```
:
<J %59P7-000.063932 % JMS0154 'TSOS' LOGGED ON FOR 'SUB'. JOB NAME 'B10S#1'. CALLER 'TSN
59MV'. TID 0032004A — (1)
<J %59P8-000.063934 % JMS0154 'TSOS' LOGGED ON FOR 'SUB'. JOB NAME 'B10S#2'. CALLER 'TSN
59MV'. TID 0036008C — (2)
<J %59P9-000.063936 % JMS0154 'TSOS' LOGGED ON FOR 'SUB'. JOB NAME 'B10S#3'. CALLER 'TSN
59MV'. TID 000500B2 — (3)
<J %59QA-000.063938 % JMS0154 'TSOS' LOGGED ON FOR 'SUB'. JOB NAME '$FBDOLSC'. CALLER 'TSN
59MV'. TID 004100E1 — (4)
:
:
<* %59P7-000.063937 % FBD1271 FDDRL JOB FOR DISK 'B10S.0' ON DEVICE '900E' STARTED
<* %59P7-000.063938 % FBD1271 FDDRL JOB FOR DISK 'B10S.1' ON DEVICE '900F' STARTED
<* %59P8-000.063939 % FBD1271 FDDRL JOB FOR DISK 'B10S.2' ON DEVICE '9068' STARTED
<* %59P8-000.063940 % FBD1271 FDDRL JOB FOR DISK 'B10S.3' ON DEVICE '9069' STARTED
<* %59P9-000.063942 % FBD1271 FDDRL JOB FOR DISK 'B10S.4' ON DEVICE '906A' STARTED
<* %59P9-000.063943 % FBD1271 FDDRL JOB FOR DISK 'B10S.5' ON DEVICE '906B' STARTED
<* %59MV-000.063944 % FBD1281 WRITE LOCK FOR PUBSET 'B10S' IS ACTIVATED — (5)
:
:
<J %59P9-000.064208 % EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 1.2305 SEC, USER ID: TSOS,
TASK ID: 000500B2 — (6)
<J %59P8-000.064212 % EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 1.2976 SEC, USER ID: TSOS,
TASK ID: 0036008C — (7)
<J %59P7-000.064220 % EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 1.4228 SEC, USER ID: TSOS,
TASK ID: 0032004A — (8)
<J %59QA-000.064228 % EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 0.0073 SEC, USER ID: TSOS,
TASK ID: 004100E1 — (9)
  <* %59MV-000.064230 % FBD1285 WRITE LOCK FOR DISK/PUBSET 'B10S' IS REMOVED
  <R %59MV-000.064319 % ESM0216 FUNCTION 'DELETE' STARTED FOR SUBSYSTEM 'FDDRL /V21.0'
  <R %DSSM-000.064319 % ESM0220 FUNCTION 'DELETE' FOR SUBSYSTEM 'FDDRL/V21.0' COMPLETELY
PROCESSED
```

**** End of the CONSOLE log extract ****

- (1) Subtask 59P7 for backing up disk set B10S-1 starts.
- (2) Subtask 59P8 for backing up disk set B10S-2 starts.
- (3) Subtask 59P9 for backing up disk set B10S-3 starts.
- (4) Monitoring task 59QA with the job name \$FBDOLSC starts.
- (5) Now that all three subtasks are ready (device assignment and first backup volume mounted), the monitoring task sets the write lock for the home pubset and the backup begins.
- (6) The backup of disk set 3 has been completed and subtask 59P9 consequently terminates.
- (7) Subtask 59P8 for backing up disk set 2 has been completed.
- (8) Subtask 59P7 for backing up disk set 1 has been completed.
- (9) Monitoring task 59QA with the job name \$FBDOLSC has been completed.

Glossary

A term in *italic* font means that it is explained somewhere else in the glossary.

additional mirror unit

Additional mirror disk which can be split without impairing ongoing input/output operation.

allocation unit

Smallest possible unit of storage space available to the system when creating a file.

control volume set

Labeled volume set of an SM pubset; it contains the administration information for the whole pubset.

data backup

Periodical creation of copies of the data set so that it can be *restored* in the event of data loss due to a hardware error or inadvertent deletion.

data compression

Used to reduce the number of *volumes* required for backup. FDDRL recognizes two types of data compression: *hardware compression*, which is performed by MTC devices, and *software compression*.

disk format

FDDRL supports disks with the formats KEY, NK2 and NK4. KEY disks can only be copied or restored to KEY disks, NK2 disks only to NK2 disks, and NK4 disks only to NK4 disk

disk set

Save entity of FDDRL: When saving, FDDRL combines multiple disks to form a disk set and saves this to a tape set.

F5 label

On magnetic disks the format 5 labels define the assigned and unassigned PAM blocks.

FDDRL job

For each FDDRL function statement, one FDDRL job is defined per individual or pubset disk. A further FDDRL job is defined for each disk set. Each FDDRL job can be executed under either the calling task (*FDDRL maintask*) or a separate task (*FDDRL subtask*).

FDDRL request

An FDDRL request is created for each FDDRL function statement. A request has the status accepted, started or completed.

FDDRL maintask

The task which starts the program FDDRL by command /START-FDDRL is called FDDRL maintask.

FDDRL subtask

FDDRL jobs can be processed by a subtask created by FDDRL.

file expiration date

see *retention period*

hardware compression

This is only possible on MTC devices which are appropriately equipped. Hardware compression is performed automatically on these devices if the corresponding volume type (e.g. TAPE-C4) is used. The data is thereby compressed by the controller of the MTC device and the blocking is optimized. The capacity of the cartridges is thus better utilized.

input disk

Disk which is to be copied or saved by FDDRL.

MAREN

Software product for the management of magnetic tape cartridges in computer centers.

MAREN catalog

Central database of *MAREN* containing all information on the *volumes*. The MAREN catalog consists of the *volume catalog* with the archive entries and the MAREN parameter set, the secondary index files and possibly a logging file.

MAREN tape set

FDDRL is used to save a save entity (single disk or disk set) to one or more tapes. In MAREN, the tapes of a disk or disk set save are combined logically to form a tape set.

multi-mirror pair

A multi-mirror pair consists of a normal unit and an additional mirror unit.

online backup

Backup of an imported pubset.

output disk

Disk to which the *input disk* is to be copied or *restored*.

Parallel Access Volume (PAV)

Multiple I/O jobs can be executed simultaneously on a logical volume. A logical PAV is represented by a basic unit and up to three alias units.

PERCON

Software product that can be used to duplicate FDDRL saves.

pubset

Public volume set; a set of related public disks which, for example, have a common catalog. FDDRL can only process the disks of a pubset if they have been exported and are therefore exclusively available. (Exception: online backup of the HOME pubset).

A pubset is identified by a catalog ID comprising 1 to 4 characters. See also *SF pubset* and *SM pubset*.

restore

Restoration of a disk's contents in the event of data loss due to a hardware error or inadvertent deletion. To do this FDDRL uses backup copies which were created at an earlier stage.

retention period

Period during which data cannot be modified or deleted. The *physical* retention period prevents backup files and volumes from being overwritten during this time. The *logical* retention period, which is defined by the file expiration date, prevents the modification or deletion of files.

save

Generally, writing (copying) data to a backup file.

save entity

FDDRL either saves single disks (*single disk*) or combines multiple disks to form a save entity (*disk set*).

save, logical

Data is read from one or more *volumes* and is written contiguously to one or more volumes.

save, physical

All data of a *volume*, including the volume labels, is written block by block to a second volume. This volume is identical to the original volume in content and structure.

SF pubset (single-feature pubset)

An SF pubset consists of a number of public disk storage units with a common catalog; it is defined by its catalog ID. An SF pubset is different from an *SM pubset*.

SHC-OSD

The Storage Host Component subsystem is the BS2000 host component for disk storage systems.

single disk

Save entity of FDDRL: When saving, FDDRL uses a separate tape set for a single disk or for each individual disk of a pubset.

SM pubset (system-managed pubset)

An SM pubset is a combination of several *volume sets* to form an addressable unit. An SM pubset is different from an *SF pubset*.

software compression

Before being written to backup volume, the data is compressed according to fixed algorithms and then written to the volume in this compressed form. After compression, the actual amount of data that has to

be transferred is smaller, so the save run may require fewer volumes. As development of tape cartridge devices progresses, software compression is becoming increasingly less important compared to *hardware compression* and is consequently no longer offered in FDDRL V16.0A and higher.

tape

In the FDDRL manual, *tape* is used as a generic term for all real, virtual, or emulated magnetic tapes when no further distinction need be made. *Volume* and *MTC* are commonly used synonyms. Each tape must have its own unique identification (*archive number*, synonym: VSN).

tape set

see MAREN type set

volume

Storage medium from the perspective of BS2000. Every volume is used by BS2000. Each volume is managed by BS2000 under a unique VSN (Volume Sequence Number). The term *tape* or *MTB* is typically used in the FDDRL manual.

volume catalog

File of the *MAREN catalog*. The volume catalog consists of the global MAREN parameter set and archive entries containing information on every *volume* that is administered in *MAREN*.

volume set

This comprises one or more disks (volumes) with the same attributes with respect to performance behavior, allocation unit, format and availability. Several volume sets make up an *SM subset*.

Abbreviations

2K	2 Kilobyte = 2,048 Byte
4K	4 Kilobyte = 4,096 Byte
A-U	Allocation Unit
bpi	bits per inch
catid	Catalog Identifier
CPU	Central Processing Unit
DMS	Data Management System
DRV	Dual Recording by Volume (subsystem)
DSSM	Dynamic Subsystem Management
I/O	Input/output
EOF	End of File Label
EOT	End of Tape
EOV	End of Volume Label
FBA	Fixed Block Architecture
FBD...	Fast Backup of Disk (FDDRL message class ID)
FDDRL	Fast Disk Dump and Reload
HDR	File Header Label
HPS	Half Pages, 2-KB units of the disk
HSMS	Hierarchical Storage Management System (utility)
IMON	Installation Monitor
INIT	Utility Routine for tape initialization
INOP	Inoperable (device status)
K	Key (disk format)
MAREN	Utility Routine for tape processing
MIP	Message Improvement Processing (system component)
MN	Mnemotechnical device name
MRSCAT	MRS Catalog
MT	Magnetic Tape
MTC	Magnetic Tape Cartridge
NDM	Nucleus Device Management

NK	None-Key (disk format)
NK2	None-Key 2K (disk format)
NK4	None-Key 4K (disk format)
OLSC	Online Save Control
PBT	Physical Half Pages per Track
PERCON	Utility Routine for data conversion
PTAM	Privileged Tape Access Method
PPAM	Privileged Primary Access Method
SCSI	Small Computer System Interface
SDF	System Dialog Facility
SF pubset	Single-Feature pubset (pubset type)
SHC-OSD	Storage Host Component (subsystem)
SM pubset	System-Managed pubset (pubset type)
SRV	Single Recording by Volume
SVC	Supervisor Call
SVL	Standard Volume Label
TLS	Tape Library System
TM	Tapemark
TPR	Task Privileged (privileged system status)
TSN	Task Sequence Number
TSOS	Privileged user ID or privilege
TU	Task Unprivileged
UHL	User File Header Label
UTL	User Trailer Label
VAV	Enhanced recording method (for tape cartridges) - German abbreviation
VOL	Volume Header Label
VOLIN	Utility Routine for formatting disks
VSN	Volume Serial Number
VTOC	Volume Table of Contents
XHCS	Extended Host Code Support

Related publications

You will find the manuals on the internet at <http://bs2manuals.ts.fujitsu.com>. You can order printed versions of manuals which are displayed with the order number.

- [1] **ARCHIVE** (BS2000/OSD)
User Guide
- [2] **BS2000 OS DX**
Utility Routines
User Guide
- [3] **DRV** (BS2000)
Dual Recording by Volume
User Guide
- [4] **SDF** (BS2000)
SDF Dialog Interface
User Guide
- [5] **BS2000 OS DX**
Introduction to System Administration
User Guide
- [6] **HSMS** (BS2000)
Hierarchical Storage Management System
Volume 1: Functions, Management and Installation
User Guide
- [7] **IMON** (BS2000)
Installation Monitor
User Guide
- [8] **JV** (BS2000)
Job Variables
User Guide
- [9] **BS2000 OS DX**
Commands
User Guide
- [10] **MAREN** (BS2000)
Tape Management in BS2000
Basics and MAREN Application
- [11] **PERCON** (BS2000)
User Guide
- [12] **SDF-P** (BS2000)
Programming in the Command Language
User Guide

-
- [13] **BS2000 OS DX**
System Installation
User Guide
 - [14] **SHC-OSD (BS2000)**
Storage Management for BS2000
User Guide
 - [15] **SPACEOPT (BS2000)**
Disk Optimization and Reorganization
User Guide