

# **eXtended Recovery System (XRS) for z/OS**

Installation and Operations Guide

Version 5.1.0

June 2016



improvIT Software Innovations GmbH  
Große Elbstraße 141 a  
22767 Hamburg  
Telephone: +49 (0)40 540 90 29 - 7  
Fax: +49 (0)40 540 90 29 - 9  
Email: [Contact@improvIT-Software-Innovations.de](mailto:Contact@improvIT-Software-Innovations.de)  
Web: [www.improvIT-Software-Innovations.de](http://www.improvIT-Software-Innovations.de)

This page intentionally left blank

# I. Content

I.	Content .....	1
II.	Listings .....	3
III.	Screens .....	4
IV.	Graphics.....	5
V.	Tables.....	6
1.	Introduction.....	7
1.1.	XRS Documentation .....	7
1.2.	eXtended Recovery System .....	7
2.	XRS Installation .....	8
2.1.	Software Prerequisites .....	8
2.2.	Other Prerequisites.....	8
2.3.	Runtime Datasets .....	9
2.4.	License Key .....	10
2.5.	XRS Core Options .....	11
3.	Getting started with XRS .....	12
3.1.	Defining XRS Runtime Datasets.....	12
3.2.	Recommendations.....	12
3.3.	Running XRS.....	13
3.3.1.	Loadlibs .....	13
3.3.2.	DSS Backup Jobs .....	13
3.3.3.	XRS Backup Return Codes .....	15
3.3.4.	DSS Errors during processing .....	15
3.4.	Backup Jobs.....	16
4.	General XRS Processing.....	17
4.1.	XRS Repository .....	17
4.2.	XRS Log File .....	17
4.3.	XRS Control Dataset (XCDS) .....	17
4.4.	DCollect.....	18
4.5.	Batch Utility XRSLLDMP .....	18
5.	XRS Dataset Processing .....	19
5.1.	XRS Dataset Maintenance Tasks.....	19
5.2.	XRS Logfile Processing.....	20
5.3.	XCDS Batch Housekeeping .....	21
5.3.1.	XCDS compression .....	21

**eXtended Recovery System (XRS) for z/OS**

- 5.3.2. Daily XCDS Housekeeping.....21
- 5.3.3. Periodic/Optional Housekeeping .....23
- 5.4. XCDS ISPF Maintenance Dialog .....24
- 5.5. XRS+ Data Collection.....26
- 6. Miscellaneous.....27
  - 6.1. DSS Reset Parameter Handling .....27
  - 6.2. Restrictions.....27
  - 6.3. Support for other (than DFDSS) backup methods .....27
- 7. Release Information.....28
  - 7.1. Changes in 5.1.0 .....28
  - 7.2. Changes in 4.9.0 .....28
  - 7.3. Migrating to XRS version 4.9.0 .....28
  - 7.4. Changes in 4.8.0 .....28
  - 7.5. Changes in 4.6.0 .....29
  - 7.6. Migrating to XRS version 4.6.0 .....29
- 8. Contact .....30
- 9. Index.....31

## II. Listings

Listing 1: Extracting delivery datasets.....	9
Listing 2: XRS Core Options in XRS\$\$OPT.....	11
Listing 3: Modified DSS backup JCL.....	14
Listing 4: Sample DCollect job.....	18
Listing 5: Filling the XRS Control Dataset.....	20

### III.Screens

Screen 1: XRS Control Dataset Maintenance.....	24
Screen 2: XRS Backup File List.....	25
Screen 3: Detailed backup information .....	25

## IV. Graphics

Graphic 1: XRS Repository .....	17
Graphic 2: XRS Dataset Maintenance Tasks and Data Flows .....	19

## V. Tables

Table 1: XRS backup return codes .....15



---

# 1. Introduction

Data Management is a vital activity in a modern data centre. The ever increasing amounts of data make it necessary to find easy solutions to manage the backup and recovery process in a IBM z/OS environment. XRS for z/OS and XRS+ simplify the handling of these demands as part of the data management process.

XRS for z/OS offers a powerful and comfortable method of collecting data from backups created with IBM DFSMSdss. The gathered information can be used to easily find and recover data using the XRS for z/OS ISPF interface.

---

## 1.1. XRS Documentation

The XRS documentation consists of 3 manuals:

- "XRS Vnnn: General Information"
- "XRS Vnnn: Installation and Operations Guide" (this manual)
- "XRS Vnnn: Users Guide"

This manual describes the XRS installation and processing.

---

## 1.2. eXtended Recovery System

The eXtended Recovery System (XRS) was developed to directly complement backup products like DSS and HSM. XRS supports backups created with DSS and HSM. It can be easily integrated into your backup and restore procedures and gives you an easy, fast and reliable way to manage your backups and recover your data using DSS or HSM.

XRS can also be integrated into your data centre contingency plans.

XRS encapsulates the DSS backup utility (physical and logical) and automatically captures detailed dataset information during the normal backup process. The collected data is stored in a repository and can then be used to easily obtain information on backups or to directly create jobs to recover individual datasets or whole volumes.

XRS also offers a direct interface to HSM, which can be used to display and easily recover datasets backed up with HSM.

A special feature within XRS is the ability to easily rebuild volumes from DSS backups. This allows volumes to be recovered even if there isn't an up-to-date full volume backup available.

Expired tape backups can automatically be removed from the XRS repository by using a Tape Management System (TMS) inventory extract. IBM DFSMSrmm (RMM) is directly supported. Other interfaces can easily be implemented.

.

## 2. XRS Installation

---

### 2.1. Software Prerequisites

The following IBM software is required:

- z/OS
- DFSMSdss (other backup products can also be integrated)
- DFSMSHsm (optional)

All releases are supported.

The following **improvIT Software Innovations** software is also needed:

- *eXtended Common Services (XCS) v280 or higher.*  
XCS is the common runtime environment for all improvIT Software Innovations products. It does not need to be purchased separately.

---

### 2.2. Other Prerequisites

The TSO/ISPF terminals for XRS must be configured as at least a Model 3 with a minimum display size of 32\*80.

## 2.3. Runtime Datasets

XRS consists of two loadlibs (SXRSAPF, SXRSLOAD), one samplib (SXRSSAMP) and two ISPF libraries (SXRSMMSG, SXRSPANL).

XCS consists of two loadlibs (SXCSLOAD and SXCSAPF).

All files are delivered in TSO transmit format.

To install XRS just transfer the xmit files in binary mode to your host system (e.g. using FTP) to pre-allocated files with LRECL=80 and RECFM=FB. The files must be extracted using the TSO RECEIVE command. The target dataset names can be specified at this point. See the sample "\$XRS\$REC" in the SXRSSAMP.

The following JCL shows a sample extract Job:

```
//S1RECV EXEC PGM=IKJEFT01,DYNAMNBR=30
//SYSPRINT DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
  PROFILE NOPREFIX

  RECEIVE INDSN('?????????.XRS510.SXRSAPF.XMI')
          DSNAME('?????.XRS510.SXRSAPF')

  RECEIVE INDSN('?????????.XRS510.SXRSLOAD.XMI')
          DSNAME('?????.XRS510.SXRSLOAD')

  RECEIVE INDSN('?????????.XRS510.SXRSMMSG.XMI')
          DSNAME('?????.XRS510.SXRSMMSG')

  RECEIVE INDSN('?????????.XRS510.SXRSPANL.XMI')
          DSNAME('?????.XRS510.SXRSPANL')

  RECEIVE INDSN('?????????.XRS510.SXRSSAMP.XMI')
          DSNAME('?????.XRS510.SXRSSAMP')

  RECEIVE INDSN('?????????.XCS280.SXCSAPF.XMI')
          DSNAME('?????.XCS280.SXCSAPF')

  RECEIVE INDSN('?????????.XCS280.SXCSLOAD.XMI')
          DSNAME('?????.XCS280.SXCSLOAD')
```

### Listing 1: Extracting delivery datasets

It is recommended, that the XCS and XRS load libraries are defined in the Linklist concatenation. Alternatively the load libraries may be added to a STEPLIB concatenation. In this case all libraries must be authorised.

A sample LOADXR member is found in the samplib.

APF authorisation is required for the SXCSAPF library and the SXRSAPF library. SXCSLOAD must also be authorised if the libraries are used in STEBLIBs or under TSO/ISPF using LIBDEFs.

### eXtended Recovery System (XRS) for z/OS

The modules XCSAVUSR, XCSAVRAC and XCSAVOLS must be defined in the SYS1.PARMLIB member IKJTSO00 in section AUTHTSF. No other system modifications are needed.

---

## 2.4. License Key

To use XRS you will need a license key. This key is supplied by **improvIT Software Innovations**.

The key needs to be installed using the samplib member "ACTXRSLI". The sample job updates the license information in both XRS loadlibs.

If an invalid XRS license key is found during backup processing, then XRS will pass all commands directly to DSS. No information is collected and processed by XRS and the XRS/DSS step will terminate with a minimum return code of 4.

The XRS TSO/ISPF dialog will still function if the licence key has expired - although not with the normal performance. All actions are performed in "slow mode".

## 2.5. XRS Core Options

XRS allows runtime options and all information collected during backups to be centrally defined. The values are maintained using a simple assembler module (see samplib member "XRS\$\$OPT"). The sample job updates the options in both loadlibs. The resulting module is loaded and processed during runtime.

The following is an extract from XRS\$\$OPT.

```

* Insert your options name:
XRS$NAME DC      CL8'xxxxxxx'          OPTIONS NAME
*
XRS$PGMN DC      CL8'ADRDSU'          Name of IBM ADRDSU module
*
XRS$CMDS DC      X'F0'                DSS commands supported by XRS
XRS$CDUM EQU     X'80'                DUMP      command
XRS$CCPD EQU     X'40'                COPYDUMP  command
XRS$CCPY EQU     X'20'                COPY      command
XRS$CRES EQU     X'10'                RESTORE   command
*
* Change the following field if you want RACF infos for backup jobs:
XRS$RACF DC      X'00'                Do not log RACF infos
*XRS$RACF DC      X'80'                Log RACF infos
*
* Change the following field if you want SMF recording for XRS:
XRS$WSMF DC      X'00'                No SMF recording
*XRS$WSMF DC      X'xx'                Write SMF records with type xx
*
* Change the following field if you want to prevent the override of the DSN of the
* log datasets in JCL
XRS$LOCK DC      X'00'                Override allowed
*XRS$LOCK DC      X'80'                Override not allowed
*
* Supply LOG datasetnames (to be dynamically allocated):
XRS$LOG  DC      CL44' '              Name of your XRSDLOG dataset
XRS$LOGS DC      CL44' '              Name of your XRSDLOGS dataset
*
XRS$RES  DC      CL3'XRS'             RESET option handled by XRS
*XRS$RES DC      CL3'DSS'             RESET option handled by DSS
*
XRS$WTO  DC      CL3'MAX'             All WTOs are issued
*XRS$WTO DC      CL3'MIN'             Only important WTOs are issued
*

```

**Listing 2: XRS Core Options in XRS\$\$OPT**

The log file datasets may be overridden in the backup JCL. All loaded options and the overriding values used are displayed in the XRS job logs. The log dataset(s) must be preallocated as a sequential or GDG file (specified as X.Y.Z(0)). The job step will terminate with return code 12 if an override is attempted but not permitted (Note: an override with the same dataset name as specified in the options is allowed). See chapter 12.1 for further explanation of the RESET option. **Warning: Do not change the specified field lengths!**

---

## 3. Getting started with XRS

---

### 3.1. Defining XRS Runtime Datasets

Certain runtime files must be allocated before XRS can be used. Use the appropriate jobs in the Samplib (SXRSSAMP) to allocate these files.

- Allocate the primary XRS Log file (sample “ALLOCCLOG”). This file contains the DSS backup information records in an internal XRS format. The contained data must be regularly transferred to the XRS Control Dataset.  
Allocate one CYL per 1200 log records.
- Optionally allocate a secondary XRS Log file (sample “ALLOCDUP”). This file contains a copy of the information written to the primary XRS Log file.
- Allocate the primary XRS Control Dataset file (sample “ALLOCCDS”). This file is the central XRS repository containing all backup information records. It is used when recovering datasets or volumes.
- Optionally allocate a partitioned dataset which will contain the XRS generated recovery JCL (sample “ALLOCGEN”).
- If the installation uses RMM then allocate the files required for the inventory extract (sample “ALLOCRM”). This job is not required if the extract files are already available.

**Note: All sample jobs need to be adapted to the installation standards. The suggestions used in the samples to store and backup the XRS datasets can be changed as required.**

---

### 3.2. Recommendations

It is good practice to adhere to the following recommendations, when backup files are created on disk:

- Do not reuse the same datasets or dataset names for backup files
- Do not use the same OUTDD DDNAME for different DUMP operations (because DFSS overwrites this file)
- Use GDGs or place timestamps in the backup files names to prevent reuse
- Always catalog the backup datasets

---

## 3.3. Running XRS

### 3.3.1. Loadlibs

- To use XRS, the product loadlibs must be available. We recommend, that all XRS and XCS loadlibs are added to the Linklist. No further STEPLIBs are then required. If this is not the case, then the following rules apply:
  - All DSS backup jobs using XRS (PGM=XRSLMAIN) need SXRSAPF and SXCSAPF in the STEPLIB.
  - XRS Housekeeping jobs selectively require SXRSLOAD, SXRSAPF, SXCSLOAD and SXCSAPF in the STEPLIB.
  - The TSO/ISPF interface needs access to SXRSLOAD and SXCSLOAD.

### 3.3.2. DSS Backup Jobs

To use XRS you have to make some minor modifications to the DSS backup steps:

- Change the EXEC statement to “EXEC PGM=XRSLMAIN”
- Optionally add DD statement XRSDLOG for the primary XRS Log file. (\*)
- Optionally add DD statement XRSDLOGS for the secondary XRS Log file. (\*)

All specified runtime backup parameters are passed to DSS and the normal DSS output is available after processing.

No DSS control statement modifications are required. XRS will only process the DSS commands specified in the core options module. All other instructions are processed normally without XRS intervention. If the XRS licence has expired or is invalid, then all commands are passed directly to DSS for processing.

The following JCL shows a modified step:

## eXtended Recovery System (XRS) for z/OS

```
/* *****  
/*  BACKUP DATASETS AND GENERATE XRS LOG RECORDS  *  
/* *****  
//STEP01 EXEC PGM=XRSLMAIN  
//*STEPLIB DD DISP=SHR,DSN=????.XRS510.SXRSAPF.LINKAPF  
/*          DD DISP=SHR,DSN=????.XCS280.SXCSAPF.LINKAPF  
/* XRS LOG FILE  
/*XRSDLOG DD DSN=????.XRS999.XRSLOG01,DISP=SHR  
/*XRSDLOGS DD DSN=????.XRS999.XRSDUP01,DISP=SHR  
/*  
//SYSPRINT DD SYSOUT=*  
//OUTDASD DD DSN=<YOUR.DFDSS.OUTPUT.DATASET>,  
//          DISP=(,CATLG),UNIT=<TAPE>,LABEL=(1,SL)  
//SYSIN DD *  
DUMP DS(INCL(YOUR.APPL.**)) -  
      OUTDD(OUTDASD) ALLD(*) ALLEXCP -  
      COMPRESS TOL(ENQF)  
/*
```

### Listing 3: Modified DSS backup JCL

(\*) – Default values may be set in module XRSD\$OPT.



### 3.3.3. XRS Backup Return Codes

The following values are returned by program XRSLMAIN:

Return Code	Description
0	Processing successful
4	XRS license expired
16	Required DD statement missing
Nn	Other DSS runtime return codes

**Table 1: XRS backup return codes**

If DSS ends with a return code greater than 0, then the step will also terminate with this value, unless XRS encountered a higher return code during internal processing.

### 3.3.4. DSS Errors during processing

Under certain circumstances, DSS can produce errors during backup processing. These in turn can result in XRS also producing corresponding messages. The following situations can for example cause DSS and XRS to produce these errors during backup processing:

- A dataset selected for backup processing cannot be found by DSS. DSS builds a list of datasets to be processed prior to starting the real backup. If the backup process takes a long time and datasets selected by DSS for processing have in mean time been deleted or renamed, then the corresponding backup fails. No information is stored by XRS.
- XRS cannot locate dataset statistics during post processing. If the dataset was successfully backed up by DSS and the dataset was deleted or renamed prior to dump processing being completed, then XRS is unable to obtain dataset statistics. All available backup information is stored by XRS and the dataset can still be recovered. Some dataset information is however not available in the ISPF management dialog.
- General hardware or catalog errors.

## 3.4. Backup Jobs

Several sample backup jobs can be found in the XRS samplib:

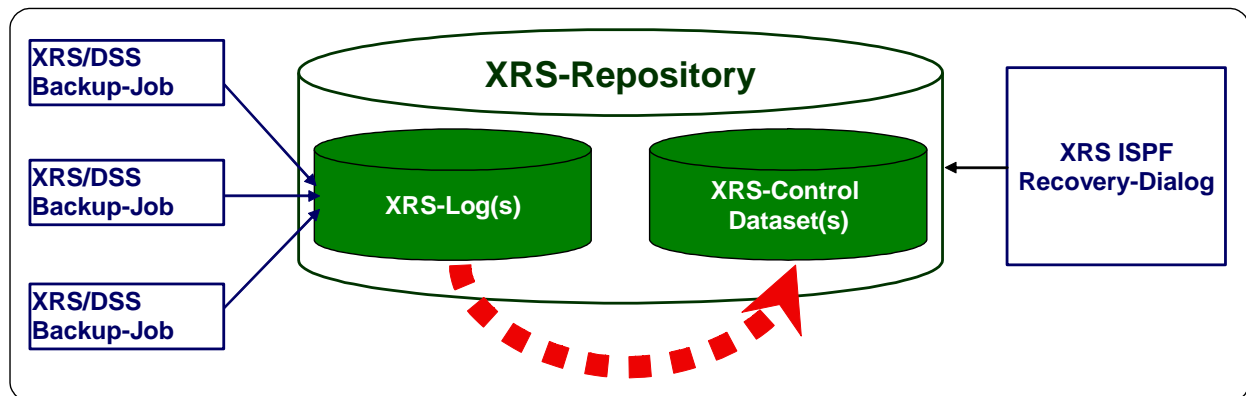
- BUP#DS01 - Create a dataset backup on DASD
- BUP#DS02 - Create a dataset backup on tape server
- BUP#DS03 - Create a dataset backup on tape
- BUP#LVOL - Create a logical volume backup on tape
- BUP#FVOL - Create a full volume dump on tape

These jobs must be modified to meet the installation standards.

## 4. General XRS Processing

### 4.1. XRS Repository

The XRS Repository is a virtual view of the XRS Log file and the XRS Control Dataset. Their usage is explained later. Backup jobs only see the XRS Log file whereas the XRS Recovery Dialog sees the Repository and optionally the Log file.



Graphic 1: XRS Repository

### 4.2. XRS Log File

The main XRS batch module XRSLMAIN is a wrapper program for DSS. Internally it calls program ADRDSSU and then saves information on the datasets being dumped. Job statistics and corresponding DSS backup dataset information (usually on tape) is also stored. All data is then written to the specified primary XRS Log file. The same information is also written to the secondary XRS Log file, if allocated. The XRS Log files can be shared between multiple backup jobs. File serialisation is managed internally by XRS.

### 4.3. XRS Control Dataset (XCDS)

The XRS Log files are processed by program XRSLREPT and the results are written to the XRS Control Dataset (XCDS). It is recommended that this should be performed on a daily basis. The XRS Control Dataset is used by the ISPF online application to generate restore jobs and supply information on available backups. See chapter 5.3 for further details.

## 4.4. DCollect

A sample job “DCOLLECT” is also located in the samplib. The information contained in the DCollect file is used by XRS when logically recovering volumes. XRS analyses the extracted volume and dataset DCOLLECT records and matches available backups from the XRS Control Dataset.

A DCollect should be run on a daily basis.

The following JCL is from sample job “DCOLLECT”.

```

//*****
//* Gather DCollect information for all volumes *
//*****
//STEP01 EXEC PGM=IDCAMS
//OUTDC DD DSN= <YOUR.DCOLLECT.DATASET>,
//      DISP=(,CATLG,DELETE),
//      SPACE=(CYL,(10,10),RLSE),
//      DCB=(RECFM=VB,LRECL=644,BLKSIZE=0)
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DCOLLECT -
  OUTFILE (OUTDC) -
  VOLUME (*)
/*

```

Listing 4: Sample DCollect job

## 4.5. Batch Utility XRSLLDMP

The XRS Repository is only filled when DFDSS Dumps are performed under XRS control. Using the batch utility XRSLLDMP, dumps created without XRS can be analysed and the available backup information optionally added to the XRS Repository.

To process existing DFDSS dumps, run the batch utility XRSLLDMP (see samplib member XRSLLDMP). Logical and/or full-volume DFDSS dumps can be processed. This utility lists the contents of each dump and optionally generates XRS Repository records for each dumped dataset.

**Note: Not all XRS information fields are filled by this utility, as the necessary information is no longer available (e.g. the original dump job data). However all information required to restore datasets using XRS is available.**

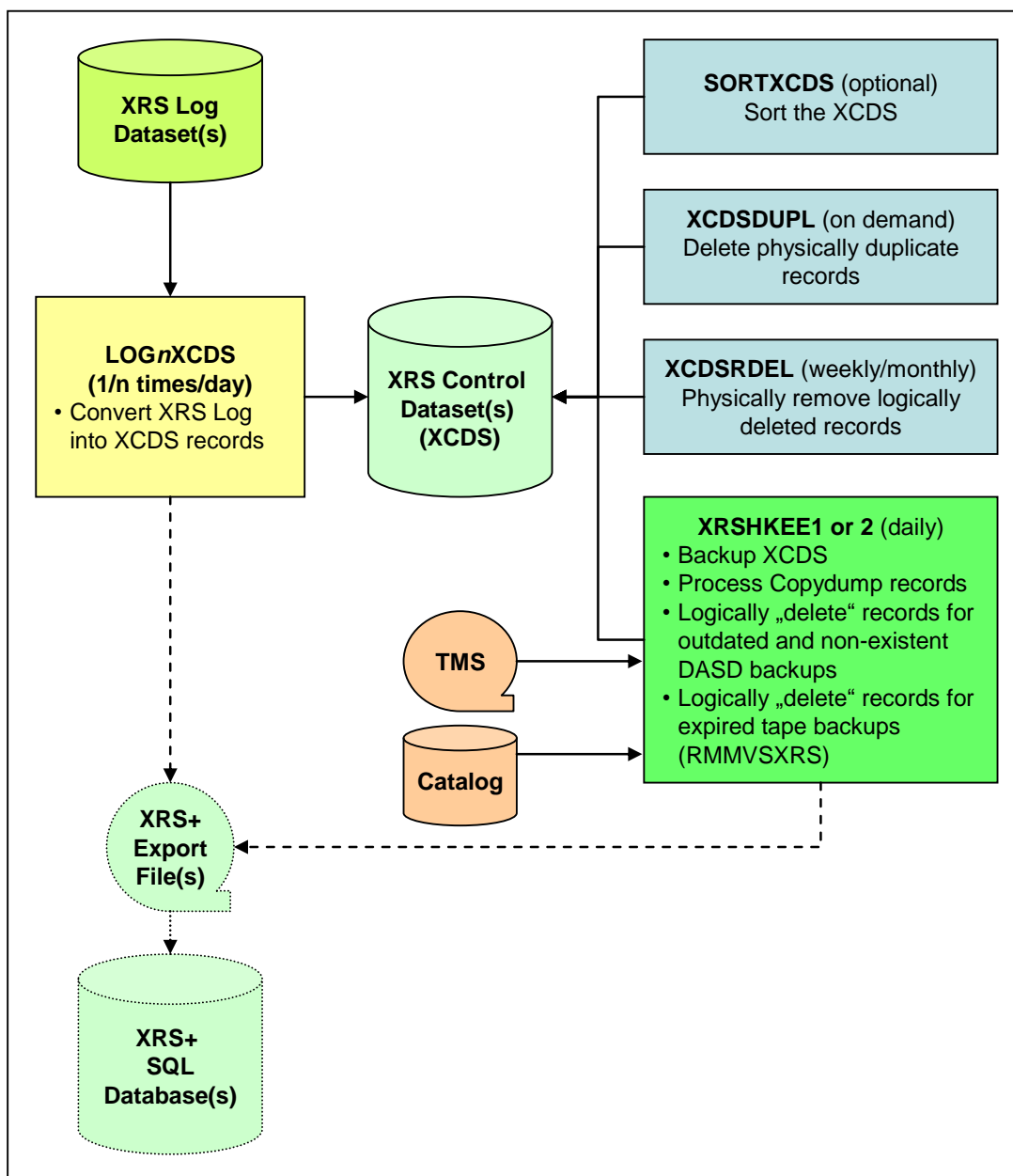
## 5. XRS Dataset Processing

### 5.1. XRS Dataset Maintenance Tasks

There are three types of processing:

- Regular/Daily - LOGnXCDS
- Daily - XRSHKEE1 or 2
- Periodic/Optional.

The relationships and data flows are illustrated in the following graphic. All housekeeping jobs can be found in the samplib.



Graphic 2: XRS Dataset Maintenance Tasks and Data Flows

## 5.2. XRS Logfile Processing

The following JCL is an extract from sample job "LOGnXCDS". It demonstrates how dataset backup records are moved from the XRS Log file to the XRS Control Dataset. Optionally a backup copy of the moved data in the XRS Log file may be created by adding the DD name XRSDLOGO. The XRS Log file is initialised after processing (if PARM='RESET' is specified).

```
//S1MRG      EXEC PGM=XRSLREPT, PARM='RESET'
//*STEPLIB  DD DISP=SHR,DSN=????.XRS510.SXRSAPF.LINKAPF
//*          DD DISP=SHR,DSN=????.XCS280.SXCSAPF.LINKAPF
//XRSDLOGI  DD DISP=SHR,DSN=????.XRS888.XRSLOG01
//*XRSDLOGO DD DISP=SHR,DSN=????.XRS888.BUPXLOG(+1),
//*          DISP=(,CATLG,DELETE),SPACE=(CYL,(5,5),RLSE)
//XRSDXCDS  DD DISP=SHR,DSN=????.XRS888.XCDS
//* Optional XRSDEXPT DD for import in XRS+:
//XRSDEXPT  DD DISP=SHR,DSN=????.XRS888.XCDSEXPT
//XRSD#ERR  DD DUMMY
```

### Listing 5: Filling the XRS Control Dataset

Only data for successfully completed DSS backups is written to the XRS Control Dataset. If DSS found errors while backing up datasets, then the corresponding records will only be written to the DD name XRSD#ERR and the return code will be set to 4. Otherwise the return code is 0.

All access to the XRS Log files is internally serialised.

The sample job "LOG1XCDS" is compatible with JES2 and JES3 allocation handling. A second sample "LOG2XCDS" is only for JES2 environments. In this case the XRS Log dataset is renamed and reallocated prior to processing the backup records. This approach is only for existing XRS installations prior to version 3.0.0. Generally the processing in sample "LOG1XCDS" should be used.

If you use the optional secondary XRS Log file, then we recommend that you use the sample job "LOG3BACK" to backup and reset this file. This job should be performed after the data in the primary XRS Log file has been loaded into the XRS Control dataset.

**Note: The XRS Log file records are normally deleted after processing. See the corresponding sample jobs on how to save this information if required.**

---

## 5.3. XCDS Batch Housekeeping

Certain backup information contained in the XRS Control Dataset may be outdated. This is generally caused by different housekeeping periods in other subsystems (e.g. the TMS<sup>1</sup>). As a result some recoveries may fail, if for example the tape was freed by the TMS and XRS has not yet performed housekeeping.

It is recommended that XRS housekeeping is performed daily after all backups and TMS maintenance have been performed.

### 5.3.1. XCDS compression

Starting with XRS version 450 the XCDS may be compressed to save DASD space. The (de)compression is done with XRS utility XRSLCODE and is transparent to all XRS programs and ISPF interfaces. Compression means the removal of all blanks in the XCDS records. The XCDS only needs to be decompressed during housekeeping (see next topic).

Samplib member XCDSCODE contains an example job to compress or decompress a XCDS.

### 5.3.2. Daily XCDS Housekeeping

The recommended daily housekeeping jobs can be found in the samplib members

- XRSHKEE1 - if the XCDS is not compressed or
- XRSHKEE2 - if the XCDS is compressed

It is also possible to run the individual housekeeping tasks in separate jobs.

All XRS sort offsets are coded via SORT symbols (defined in samplib member XCDSSYMS).

**Note1: Daily housekeeping must not be run parallel to LOGnXCDS!**

**Note2: Do not use sample job XRSHKEE1 if your XCDS is compressed. If you do then the result will be an empty XCDS!**

The individual processing steps are now described in detail.

---

<sup>1</sup> Tape Management System

### **5.3.2.1. XCDS backup**

The first step performs a backup of the XCDS using the new (V450) XCDS backup program XRSLBKUP. The output size is approximately only 30 percent of the original XCDS size.

Samplib members XCDSBKUP and XCDSREST demonstrate dumping and restoring the XCDS.

### **5.3.2.2. Decompress the XCDS (only in XRSHKEE2)**

Decompress the XCDS prior to further processing.

### **5.3.2.3. Process XRS log file records**

This process analyses and reformats XRS log file records and creates XCDS records.

Only data for successfully completed DSS backups is written to the XRS Control Dataset. If DSS found errors while backing up datasets, then the corresponding records will only be written to the DDName XRSD#ERR, a message written to the Joblog and the return code will be set to 4. Otherwise the return code is 0.

### **5.3.2.4. Copydump record processing**

Handles the results of DSS Copydump processing. For each dump dataset that was copied using Copydump, new backup records are created for all source dataset records within the dump that were copied.

Only successfully completed DSS Copydumps are processed by XRS. If DSS found errors while processing the Copydump, then a message written to the Joblog.

### **5.3.2.5. Tidy up outdated backup records / Catalog Compare**

If you use the same dataset name for different backups on disk, then only the last performed backup is valid. This step eliminates all the XCDS information for older disk backups with this dataset name. The XRS Control Dataset is also compared with the catalog. This process removes all deleted disk backup datasets from the XRS Control Dataset. Backups stored on tape are bypassed.

### **5.3.2.6. TMS Compare**

It is recommended that the contents of the XRS Control Dataset are regularly compared with an inventory report from the installed tape management system (e.g. RMM). This removes all deleted or expired tape backup datasets from the XRS Control Dataset. Backups stored on disk are bypassed.

Other tape management systems can easily be integrated into this procedure. In this case active DSS backup datasets and their volumes as well as expired volsers must be extracted from the implemented TMS. Please contact improvIT Software Innovations for further information.

### **5.3.2.7. Compress the XCDS (only in XRSHKEE2)**

Compress the XCDS.



## 5.3.3. Periodic/Optional Housekeeping

### 5.3.3.1. Removing deleted records

No XRS Control Dataset records are physically deleted during normal batch or online processing. Instead the records are logically deleted using an internal flag and timestamp. This procedure allows records to be manually undeleted in an emergency.

From time to time you should physically delete these records. Use sample job "XCDSRDEL". In this sample, the deleted records are stored in a separate GDG file.

### 5.3.3.2. Sorting

This process is generally not required unless the XRS file access method "direct" is being used. In this case the XRS Control Dataset should be sorted regularly using the sample job "SORTXCDS". After sorting the newest backup records are at the beginning of the file. This can improve performance.

### 5.3.3.3. Duplicate Records

Duplicate records can occur in the XRS Control Dataset, if program XRSLREPT is run more than once using the same XRS Log file. This problem can be resolved using the sample job "XCDSDUPL". All duplicate records are then physically removed from the XRS Control Dataset.

## 5.4. XCDS ISPF Maintenance Dialog

The dialog option “M” from the XRS primary option menu allows backup information to be explicitly deleted from the XRS Control Dataset (XCDS).

Generally this function **should not** need to be used. The XRS Control Dataset should be maintained using the XRS batch utilities (see samples chapter 5.3).

The following panel is shown when backup records need to be explicitly removed from the XRS Control Dataset. Enter the name of a **DSS backup file** (not the backed up dataset name) and corresponding first backup volume. The file name can, for example, be found in the backup job.

```

XRS Vnnn - Control Dataset (XCDS) Maintenance
Command ==>

Userid      - GZU
Prefix      - **NONE**
XRS Sysid   - ADCD

Delete backup datasets matching
Backup dataset name:  XRSBUP.XRS999.BUP1#GZU_____
Located on primary Volume:  IMP001 _

System default XCDS file:  INST.XRS999.XCDSM

Press ENTER to process request or END to terminate
    
```

Screen 1: XRS Control Dataset Maintenance

This dialog shows all files contained in the requested DSS backup dataset. Detailed backup information may be displayed by entering an “I” in front of the required dataset.

```

XRS Vnnn - Control Dataset (XCDS) Maintenanc Row 1 to 9 of 23
Command ==> _
                                                    Userid   - GZU
                                                    Prefix   - **NONE**
                                                    XRS Sysid - ADCD

A job will be generated to delete the following backup records

I  Dataset Name
Backup: Date      Time      Jobname  Stepname  Volume
                DDName   SysId   CC
-  INST.XCSMAINT.CNTL
    20120425  11353143  GZU1    DUMP01   IMP002
    OUTDASD1  ADCD    0
-  INST.XCSMAINT.EXEC
    20120425  11353143  GZU1    DUMP01   IMP002
    OUTDASD1  ADCD    0
-  INST.XCSMAINT.SAMPLIB
    20120425  11353143  GZU1    DUMP01   IMP002
    OUTDASD1  ADCD    0
-  INST.XCS000.ASM.MACLIB
    20120425  11353143  GZU1    DUMP01   IMP002
    OUTDASD1  ADCD    0
-  INST.XCS000.INCLUDE
    20120425  11353143  GZU1    DUMP01   IMP002
    OUTDASD1  ADCD    0
-  INST.XCS000.XMITIP
    20120425  11353143  GZU1    DUMP01   IMP002
    OUTDASD1  ADCD    0
-  INST.XCS000.XMITP.EXEC
    20120425  11353143  GZU1    DUMP01   IMP002
    OUTDASD1  ADCD    0
-  INST.XCS260.ASM
    20120425  11353143  GZU1    DUMP01   IMP002
    OUTDASD1  ADCD    0
-  INST.XCS260.ASM.MACLIB
    20120425  11353143  GZU1    DUMP01   IMP002
    OUTDASD1  ADCD    0
    
```

Screen 2: XRS Backup File List

If an "I" is entered in front of one of the displayed dataset names, then all available backup information is shown.

```

XRS Vnnn - Detailed Dataset Backup Information
Command ==> _
                                                    Userid   - GZU
                                                    Prefix   - **NONE**
                                                    XRS Sysid - ADCD

Source Dataset Statistics
Name:          INST.XCSMAINT.CNTL
Volume:        IMP002      Creation Date: 12051      Change Flag: Y
Organisation:  PO          Size:                    0.01 MB
Record Format:  FB          Record Length: 80        Blocksize: 27920
SMS Dataclass: SMS Mgmtclass: SMS Storcl.: SCINST

Backup Job Execution Statistics
Name:          GZU1        Job Number:  JOB01291     Step:          DUMP01
System:        ADCD        OS Release:   SP7.1.2           RACF Userid:  GZU
Date:          20120425   Time:         113531     RACF Attr.:   SO

Backup Dataset Statistics
Name:          XRSBUP.XRS999.BUP1#GZU
DDName:        OUTDASD1   Backup Date:  12116     Expiration:    00000
Organisation:  PS          Record Format:  U        Backup Tool:   DSS Log.
Record Length: 0           Blocksize:     27998     File Seq No:  0
Vol Seq No:    0           Volume Count:
Volumes 1 to 8: IMP001

DFDSS Statistics
Condition Code: 0          Message:
    
```

Screen 3: Detailed backup information

Upon exiting the Backup File List dialog, the JCL needed to logically delete the matching records from the XRS Control Dataset is then temporarily generated and displayed.

## 5.5. XRS+ Data Collection

If you are using XRS+, then the DDname “XRSDEXPT” needs to be added to various steps within the XRS housekeeping process. XRSDEXPT can be allocated using the sample job “ALLOCEXP”. The export dataset contains formatted synchronisation records which can be loaded into the XRS+ database.

Please see the following samples for housekeeping changes:

- LOG1XCDS / LOG2XCDS
- RMMVSXRS
- XCSUNLD
- XRSHKEEP

An initial XCDS unload may be performed using sample job “XCDSUNLD”.

**Note: Only allocate and add DDName XRSDEXPT, if you wish to collect SMF records. Do not allocate XRSDEXPT to DUMMY, as this will incur a performance overhead.**

---

## 6. Miscellaneous

---

### 6.1. DSS Reset Parameter Handling

Normally when the DSS RESET control statement is used to reset the change bit, DSS performs an enqueue on all datasets to be backed up, during the whole backup operation. As a result the datasets cannot be used by other tasks during backup processing.

There is a new (V460) XRS option: if specified, XRS removes the RESET control statement, prior to calling DSS. After backup processing has completed, XRS internally resets the dataset change bit for all backed up files.

This only applies when the RESET parameter was specified by the user.

---

### 6.2. Restrictions

Up to 250 tape volumes may be written for a single DSS backup dataset. This limit has historical reasons and should no longer be required nowadays. As a result XRS only logs the first 20 volumes of any DSS backup dataset.

XRS does not support the “LVOL” parameter of the DSS Copydump function. If used, then XRS will treat the Copydump command as a full copy of the original dump file.

---

### 6.3. Support for other (than DFDSS) backup methods

Starting with XRS 4.8.0 other backup methods will be supported by XRS and integrated in the XRS processing. The first supported method is IDCAMS-REPRO. This is usually used for the backup of VSAM datasets. The XRS support consists of an additional XRS step (executing the new batch program XRSLGLOG) directly after the IDCAMS-REPRO step. The batch program analyzes the source and the backup dataset of the IDCAMS-REPRO step and generates one XRS log record from these informations. The dataset display and recovery panels in XRS handle these new log records. So the VSAM cluster may be restored from the XRS “Dataset Recovery Panel”.

See Samplib member IDCREPRO for an example of this new XRS extension.

It is intended to support more backup methods in the future (e.g. IEBGENER/SORT).

---

## 7. Release Information

---

### 7.1. Changes in 5.1.0

- Multiple XCDS support (up to 9)
- Full volume dumps HSM support
- Additional optional JCL parm for generated jobcard

---

### 7.2. Changes in 4.9.0

- Improved HSM support. This includes:
  - A new option for checking if (newer) HSM backups are available for a dataset
  - A template may now be selected in Option 5 (Recover a dataset using HSM backups)
- New option 6 (Recover all datasets from a backup)
- The dataset mask in the “Dataset Recovery Panel” is now more flexible: a string search like DSN=\*ABC\* is allowed
- The XRS documentation consists now of 3 manuals:
  - "XRS Vnnn: General Information"
  - "XRS Vnnn: Installation and Operations Guide"
  - "XRS Vnnn: Users Guide"

---

### 7.3. Migrating to XRS version 4.9.0

- Always complete housekeeping before activating the new XRS software release. The XRS Log file must be empty.
- No specific migration steps are required.

---

### 7.4. Changes in 4.8.0

- The SMS attributes of a dataset are collected and displayed (this applies to logical dumps only).
- The mask for dataset names in the “Dataset Recovery Panel” is extended (\*, \*\* and % are possible). This behaviour is “ISPF 3.4”-like.
- The REGION parameter for the generated restore job may now be specified in the options 1.1

- New additional LLQ for the dataset name of the restored dataset.
- Support for other backup methods like IDCAMS-REPRO.
- The source dataset size is now displayed for full-volume backups.

---

## 7.5. Changes in 4.6.0

- New batch program XRSLLDMP - lists DFDSS dump datasets and optionally constructs XRS repository records for the backed up datasets (for logical or full-volume dumps). See chapter 4.5 for details.
- New XRS\$\$OPT runtime options. See chapter 2.5 for details.
  - WTO messages – defines how many messages are produced by XRS
  - DFDSS-RESET handling – controls whether XRS or DFDSS performs a reset
- XRS now supports large block sizes for DFDSS dump datasets on tape (up to 65520)

---

## 7.6. Migrating to XRS version 4.6.0

- Always complete housekeeping before activating the new XRS software release. The XRS Log file must be empty.
- No specific migration steps are required.

## 8. Contact

For further information regarding the eXtended Recovery System please contact:

**improvIT Software Innovations GmbH**

Große Elbstraße 141 a

D-22767 Hamburg

Germany

Telephone: +49 (0)40 540 90 29 - 7

Fax: +49 (0)40 540 90 29 - 9

Email: [Contact@improvIT-Software-Innovations.de](mailto:Contact@improvIT-Software-Innovations.de)

Web: [www.improvIT-Software-Innovations.de](http://www.improvIT-Software-Innovations.de)



## 9. Index

Assembler.....	11
Authorised.....	9
Copydump .....	22, 27
DASD.....	16, 21
Dcollect.....	18
Ddname .....	12, 22, 26
DFSMS	
DFSMSdss.....	7, 8
DFSMSHsm.....	8
DSS.....	7, 10, 12, 13, 14, 15, 17, 20, 22, 24, 27
Dump .....	12, 15, 16, 18, 22, 27, 29
Export .....	26
Extract.....	7, 9, 11, 12, 20
Ftp.....	9
Housekeeping.....	13, 19, 21, 23, 26, 28, 29
HSM.....	7
IBM .....	7, 8
Install .....	9
Installation.....	8, 12, 16
ISPF.....	7, 8, 9, 10, 13, 15, 17, 21, 24
Key.....	10
Licence .....	10, 13
Linklist.....	9, 13
Loadlib .....	9, 10, 11, 13
Log.....	11, 12, 13, 17, 20, 22, 23, 28, 29
Logical .....	7, 10, 16, 18, 29
Migrating.....	29
Options .....	11, 13, 29
Parameter .....	27
Performance .....	10, 23, 26
Physical .....	7
Receive.....	9
Recommendation.....	12
Reset .....	20, 27, 29
Restriction.....	27
Return Code .....	10, 11, 15, 20, 22
RMM .....	7, 12, 22
Runtime .....	8, 9, 11, 12, 13, 15, 29
Sample.....	9, 10, 11, 12, 16, 18, 20, 21, 23, 26
Samplib.....	9, 10, 11, 12, 16, 18, 19, 21, 22
SMF .....	26
Statistics .....	15, 17
Steplib.....	9, 13
Tape.....	7, 16, 17, 21, 22, 27, 29
TMS .....	7, 21, 22
TSO/ISPF .....	8, 9, 10, 13
XCDS.....	17, 21, 22, 24, 26

**eXtended Recovery System (XRS) for z/OS**

XCS .....	8, 9, 13
Xmit.....	9, 10
XRS\$\$OPT .....	11, 14, 29
XRS+ .....	7, 26
XRSDEXPT .....	26
z/OS.....	7, 8