

z/OS Systems Installation

This section describes the preparation for and installation of Adabas Parallel Services on z/OS systems.

Important:

Before proceeding with the installation, ensure that the prerequisite environment has been established. Review the earlier sections of this part of the documentation for detailed information.

The installation procedure outlined in this section corresponds to the jobs that SMA creates to install the product. If you do not use SMA, you can modify and use the sample JCL provided in section *Unload the Installation Libraries to Disk* to unload the libraries from the installation tape.

- Step 1: Unload the Installation Libraries to Disk
 - Step 2: APF-Authorize All Load Libraries
 - Step 3: Customize ADACOM
 - Step 4: Create a Startup Procedure for ADACOM
 - Step 5: Create a Startup Procedure for Each Cluster Nucleus
 - Step 6: Modify the z/OS Common Storage (CSA) Key
 - Step 7: Start Adabas Parallel Services
-

Step 1: Unload the Installation Libraries to Disk

Sample JCL is provided in the base Adabas source library. This JCL can be modified and used to unload the Adabas Parallel Services libraries from the installation tape.

The Adabas Parallel Services installation tape is a standard label tape. Refer to the *Software AG Product Delivery Report* that accompanies the tape for the volume serial number, density, media type, data set names, and data set sequence numbers.

The tape contains the installation data sets, a data set required by SMA, and one or more data sets containing maintenance fixes. Refer to *Zap Information* for information about applying fixes.

- Allocate DASD
- Copy the Contents of the Tape to Disk

Allocate DASD

The load library for Adabas Parallel Services requires one cylinder.

Copy the Contents of the Tape to Disk

Copy the data sets from the supplied installation medium to your disk before you perform the individual installation procedure for each component to be installed.

The way you copy the data sets depends on the installation method and the medium used:

- If you use System Maintenance Aid (SMA), refer to the copy job instructions provided in the *System Maintenance Aid* documentation.
- If you are not using SMA and want to copy the data sets from CD-ROM, refer to the README.TXT file on the CD-ROM.
- If you are not using SMA and want to copy the data sets from tape, follow the instructions in this section.

This section explains how to copy all data sets from tape to disk.

- Step 1: Copy Data Set COPY.JOB from Tape to Disk
- Step 2: Modify hilev.COPY.JOB on Your Disk
- Step 3: Submit COPY.JOB

Step 1: Copy Data Set COPY.JOB from Tape to Disk

- Modify the following sample job according to your requirements:

```
//SAGTAPE JOB SAG,CLASS=1,MSGCLASS=X
//* -----
//COPY EXEC PGM=IEBGENER
//SYSUT1 DD DSN=COPY.JOB,
// DISP=(OLD,PASS),
// UNIT=(CASS,,DEFER),
// VOL=(,RETAIN,SER=tape-volser),
// LABEL=(2,SL)
//SYSUT2 DD DSN=hilev.COPY.JOB,
// DISP=(NEW,CATLG,DELETE),
// UNIT=3390,VOL=SER=disk-volser,
// SPACE=(TRK,(1,1),RLSE),
// DCB=*.SYSUT1
//SYSPRINT DD SYSOUT=*
//SYSIN DD DUMMY
//
```

where:

tape-volser is the VOLSER of the tape, for example: T12345,

hilev is a valid high-level qualifier, and

disk-volser is the VOLSER of the disk.

- Execute the job to copy the data set COPY.JOB to your disk.

Step 2: Modify *hilev.COPY.JOB* on Your Disk

- Modify *hilev.COPY.JOB* according to your requirements:

Set EXPDT to a valid expiration date, for example, 99365.

Set HILEV to a valid high-level qualifier, for example, USERLIB.

Set LOCATION to a storage location, for example, STORCLAS=ABC or UNIT=3390,VOL=SER=USR123.

Step 3: Submit *COPY.JOB*

- Execute *hilev.COPY.JOB* to copy single, multiple, or all data sets to your disk.

Step 2: APF-Authorize All Load Libraries

Ensure that Adabas Parallel Services runs authorized.

To run authorized, the Adabas Parallel Services and Adabas version load libraries and all other load libraries in the STEPLIB concatenation must be APF-authorized.

- Ensure that all load libraries referenced in the STEPLIB concatenation for your Entire Net-Work, ADACOM, and Adabas cluster nuclei startup procedures are defined to the operating system as authorized libraries. If this is not done, ADACOM or the cluster nuclei will not initialize and may abnormally terminate, usually with an ABENDS047 or ABENDS306.

Either copy the base Adabas modules and the Adabas Parallel Services modules to an existing APF-authorized library or APF-authorize the Adabas Parallel Services library that was unloaded from the installation tape and the Adabas load library unloaded from the base Adabas installation tape.

Step 3: Customize ADACOM

Make any needed additions and modifications to the ADACOM member.

Read *ADACOM Initialization Parameters* for more information about specifying values for ADACOM parameters.

Step 4: Create a Startup Procedure for ADACOM

An ADACOM initialization task is provided. This task must be active on the operating system image before any Adabas Parallel Services cluster nucleus is started. All load libraries in the STEPLIB concatenation of ADACOM must be APF-authorized.

ADACOM allocates the nucleus table for monitoring the active nuclei and the user table for monitoring users in the extended CSA (ECSA) above the 16MB line.

- The following is a sample job for running ADACOM:

```
//ADACOM PROC
//*-----*
//*      ADABAS PARALLEL SERVICES ADACOM Vv.r.s STARTUP      *
//*-----*
//ASMvrs EXEC PGM=ADARUN,REGION=4M,TIME=1440
//STEPLIB DD DISP=SHR,DSN=ADABAS.Vvrs.LOAD
//*
//COMPRINT DD SYSOUT=*
//DDPRINT DD SYSOUT=X
//DDCARD DD *
ADARUN PROGRAM=ADACOM
/*
//DDKARTE DD *
ADACOM SVC=sv1,DBID=dbid1,NU=150
ADACOM SVC=sv1,DBID=dbid2,NU=150,F=Y
ADACOM SVC=sv2,DBID=dbid1,NU=500
ADACOM SVC=sv2,DBID=dbid3,NU=500
/*
```

The COMPRINT DD statement must be specified when running ADACOM. It defines an output data set for all general messages printed by ADACOM. For each SVC/DBID set specified in the ADACOM task, two subtasks are attached and SYSOUT data sets are dynamically allocated to receive all messages specific to that combination. The DD-name of the SYSOUT data sets are "Pssdddd" and "Dssdddd", where *ss* represents the last two digits of the SVC number and *dddd* are the five digits of the DBID.

ADACOM can also be installed as a started task; no special considerations apply.

You may also want to add the ADACOM task's start command to member COMMND_{xx} of SYS1.PARMLIB; this enables the ADACOM task to begin automatically at IPL.

Step 5: Create a Startup Procedure for Each Cluster Nucleus

▶ Complete the following steps:

1. Customize a startup procedure to execute ADARUN.

For each Adabas cluster nucleus, customize the appropriate startup parameters and execute ADARUN from the Adabas load library.

2. Concatenate the Adabas Parallel Services load library ahead of the Adabas load library in the STEPLIB.
3. Allocate and format a Work data set for each nucleus.

All nuclei in an Adabas Parallel Services cluster share a common database resource; i.e., the same ASSO and DATA data sets. Each nucleus in the cluster must have its own Work data set; and all Work data sets within a cluster must have the same size and device type as defined in the general control block (GCB).

Use DISP=SHR on the DD card for the Work data set (DDWORKR1). During an offline or online restart/recovery, a nucleus may access the Work data sets belonging to other nuclei in the cluster.

4. Specify for each nucleus the ADARUN parameters CLUSTER, NUCID, CLUCACHETYPE, CLUCACHESIZE, CLULOCKSIZE, and CLUCACHEUNCHANGED.

Although each nucleus of an Adabas cluster shares the same database resource (DBID), each nucleus must have a unique NUCID value:

- a single (noncluster) nucleus: NUCID=0 (default)
- a cluster nucleus: NUCID=1-65000

Values for the CLUCACHESIZE and CLULOCKSIZE parameters are required for allocating the global data spaces. Read *Performance and Tuning* for sizing recommendations.

Use current values for all other ADARUN parameters, then reevaluate the values after monitoring the result. Ensure that each nucleus in the cluster is prepared to handle the entire workload for the common database, if necessary.

5. If protection logs are used, they must be dual or multiple logs and each nucleus must have its own. If one nucleus in the cluster runs with PLOGs, all nuclei in the cluster must run with PLOGs. The ADARUN PLOGRQ parameter must be the same for all nuclei (global parameter).

If user exit 2 or user exit 12 is supplied for one nucleus, the same user exit must be supplied for all nuclei in the cluster. User exit 12 must be used instead of user exit 2 if NCLOG/NPLOG is specified.

6. If command logs are used, each nucleus must have its own. If command logs are to be merged, they must be dual or multiple command logs and each nucleus in the cluster must have the same CLOG definition. To invoke automatic CLOG merging, CLOGMRG=YES must be specified in the ADARUN parameters or given as an operator or AOS/ADADBS command to any nucleus in the cluster.

7. The following sample JCL (job ASMNUC in the MVSJOBS data set) executes the Adabas ADARUN program to implement session parameters for an Adabas Parallel Services cluster nucleus.

Note that the Adabas Parallel Services library is concatenated ahead of the Adabas library in the STEPLIB.

```
//ADANUC      JOB      MSGCLASS=X,TIME=1440
//*-----
//*   THIS IS A STARTUP JOB FOR A PARALLEL SERVICES NUCLEUS
//*
//*   THE ADARUN PARMS HAVE TO BE CUSTOMIZED
//*   DEPENDING ON THE USER'S ENVIRONMENT
//*   DETAILS ARE PROVIDED IN THE OPERATIONS MANUAL
//*-----
//NUC        EXEC    PGM=ADARUN,REGION=10M
//STEPLIB    DD      DISP=SHR,DSN=ASM.Vvrs.LOAD          <=== ASM LOAD
//          DD      DISP=SHR,DSN=ADABAS.Vvrs.LOAD       <=== ADABAS LOAD
//DDASSOR1   DD      DISP=SHR,DSN=EXAMPL.DByyy.ASSOR1   <=== ASSO
//DDDATAR1   DD      DISP=SHR,DSN=EXAMPL.DByyy.DATAR1   <=== DATA
//DDWORKR1   DD      DISP=SHR,DSN=EXAMPL.DByyy.WORKR1   <=== WORK
//DDPLOGR1   DD      DISP=SHR,DSN=EXAMPL.DByyy.PLOGR1   <=== PLOG1
//DDPLOGR2   DD      DISP=SHR,DSN=EXAMPL.DByyy.PLOGR2   <=== PLOG2
//DDDRUCK    DD      SYSOUT=X
//DDPRINT    DD      SYSOUT=X
//MPMDUMP    DD      SYSOUT=X
//DDCARD     DD      *
ADARUN PROG=ADANUC
```

```

ADARUN CLUSTER=LOCAL
ADARUN MODE=MULTI
ADARUN SVC=num          <--- INSERT YOUR SVC NUMBER
ADARUN DBID=yyy        <--- INSERT YOUR DATABASE ID
ADARUN NUCID=nnnnn     <--- INSERT YOUR NUCLEUS ID
ADARUN CLUCACHESIZE=size <--- INSERT YOUR CACHE SIZE
ADARUN CLULOCKSIZE=size <--- INSERT YOUR LOCK SIZE
ADARUN DEVICE=3390
ADARUN CT=60
ADARUN OPENRQ=NO       ---> DEFAULT = YES
ADARUN PLOGRQ=NO       ---> DEFAULT = YES
ADARUN LBP=900000
ADARUN LFIOP=300000
ADARUN LCP=10000
ADARUN LFP=12000
ADARUN LWP=350000
ADARUN LI=10000
ADARUN LS=20000
ADARUN LU=65535
ADARUN LP=1500
ADARUN NAB=16
ADARUN NISNHQ=1000     ---> FOR BATCH NATURAL INPL
ADARUN NT=8
ADARUN TT=600
ADARUN TNAA=600
ADARUN TNAE=600
ADARUN TLSCMD=300
ADARUN DUALPLS=6750    ---> 50 CYL
ADARUN DUALPLD=3390
ADARUN LOGGING=NO
/*

```

Step 6: Modify the z/OS Common Storage (CSA) Key

If necessary, modify the z/OS Common Storage (CSA) key.

With z/OS version 1.9, the default setting of the ALLOWUSERKEYCSA parameter has been changed to NO, prohibiting the allocation of CSA storage in a non-privileged storage key. In past releases of Adabas Parallel Services, its data structures in common storage (PLXCB and subordinate control blocks) were stored in user key storage. Due to the change of the ALLOWUSERKEYCSA default, you must either change the ALLOWUSERKEYCSA setting in z/OS 1.9 to YES or accept the ALLOWUSERKEYCSA default setting in z/OS 1.9 (NO). If you accept the default, you will need to modify your Adabas Parallel Services installation and the Adabas Parallel Services and ADACOM startup procedures to use a system key.

For more information, read *Modifying the z/OS Common Storage (CSA) Key*

Step 7: Start Adabas Parallel Services

 To start Adabas Parallel Services, complete the following steps:

1. Start the ADACOM initialization task on the operating system image that is hosting the Adabas Parallel Services cluster environment.

2. Start the Adabas Parallel Services cluster nuclei in any order.

The Adabas Parallel Services cluster is now ready to process user requests.

Rules for subsequent starts of Adabas Parallel Services are described in *Restart/Recovery Processing*

.