

z/VSE Systems Installation

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

The installation procedure outlined in this section corresponds to the jobs that SMA creates to install the product.

- Step 1: Unload the Installation Libraries to Disk
 - Step 2: Install the Adabas Parallel Services Library
 - Step 3: Customize the ADACOM Procedure
 - Step 4: Create a Startup Procedure for ADACOM
 - Step 5: Create a Startup Procedure for Each Cluster Nucleus
 - Step 6: Start Adabas Parallel Services
-

Step 1: Unload the Installation Libraries to Disk

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.

- Copy the Contents of the Tape to Disk

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 the data sets .LIBJ, .LIBR and .LICS from tape to disk. All other data sets can be installed directly from the tape.

- Step 1: Copy Data Set COPYTAPE.JOB to Disk
- Step 2: Modify COPYTAPE.JOB on Your Disk
- Step 3: Submit COPYTAPE.JOB

Step 1: Copy Data Set COPYTAPE.JOB to Disk

- Modify the following sample job according to your requirements:

```
* $$ JOB JNM=LIBRCAT,CLASS=0,
* $$ DISP=D,LDEST=(*,UID),SYSID=1
* $$ LST CLASS=A,DISP=D
// JOB LIBRCAT
* *****
*     STORE COPYTAPE.JOB IN LIBRARY
* *****
// ASSGN SYS004,nnn
// MTC REW,SYS004
// MTC FSF,SYS004,4
ASSGN SYSIPT,SYS004
// TLBL IJSYSIN,'COPYTAPE.JOB'
// EXEC LIBR,PARM='MSHP; ACC S=lib.sublib'
/*
// MTC REW,SYS004
ASSGN SYSIPT,FEC
/*
/&
* $$ EOJ
```

where:

nnn is the tape address, and

lib.sublib is the library and sublibrary in which the data set COPYTAPE .JOB is to be stored.

- Execute the job to copy the data set COPYTAPE .JOB to disk.

COPYTAPE .JOB contains the JCL required to copy the data sets .LIBJ, .LIBR and .LICS from tape to disk.

Step 2: Modify COPYTAPE.JOB on Your Disk

- Modify COPYTAPE .JOB according to your requirements and set the disk space parameters as appropriate.

Step 3: Submit COPYTAPE.JOB

- Execute COPYTAPE .JOB to copy the data sets .LIBJ, .LIBR and .LICS to your disk.

Step 2: Install the Adabas Parallel Services Library

- Define the Sublibrary
- Restore the Adabas Parallel Services Sublibrary

Define the Sublibrary

Adabas Parallel Services users must define an additional sublibrary in the Adabas library for the Adabas Parallel Services components.

A sample job to accomplish this is as follows:

```
// JOB ASMDEF DEFINE NON-VSAM SUBLIB
// OPTION LOG
// DLBL SAGLIB,'ADABAS.Vvrs.LIBRARY',2099/365,SD
// EXTENT SYS010
// ASSGN SYS010,DISK,VOL=vvvvvv,SHR
// EXEC LIBR
DEFINE S=SAGLIB.ASMvrs REUSE=AUTO R=Y
LD L=SAGLIB OUTPUT=STATUS
/*
/ &
```

where:

SYS010	is the logical unit for the Adabas library
vvvvvv	is the volume for the Adabas library
vrs	is the Adabas version/revision/system maintenance (SM) level

Restore the Adabas Parallel Services Sublibrary

A sample job to restore the Adabas Parallel Services components is as follows:

```
// JOB ASMRST RESTORE NON-VSAM
// OPTION LOG
// ASSGN SYS006, cuu
// PAUSE MOUNT ADABAS INSTALL TAPE cuu
// MTC REW, SYS006
// MTC FSF, SYS006, tt
// DLBL SAGLIB,'ADABAS.Vvrs.LIBRARY'
// EXTENT SYS010
// ASSGN SYS010,DISK,VOL=vvvvvv,SHR
// EXEC LIBR
RESTORE SUB=SAGLIB.ASMvrs -
TAPE=SYS006 LIST=Y R=Y
LD SUB=SAGLIB.ASMvrs OUTPUT=NORMAL
/*
// MTC REW, SYS006
/ &
```

where:

SAGLIB	is the Adabas library name
SYS010	is the logical unit for the Adabas library
SYS006	is the Adabas installation tape
<i>cuu</i>	is the physical unit address of the tape drive
<i>tt</i>	is the number of tape marks to space forward (see the <i>Software AG Product Delivery Report</i>)
<i>vvvvvv</i>	is the volume for the Adabas library
<i>vrs</i>	is the Adabas version/revision/system maintenance (SM) level

Step 3: Customize the ADACOM Procedure

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.

ADACOM allocates the nucleus table for monitoring the active nuclei and the user table for monitoring users in the SVA above the 16MB line.

The following is a sample job for running ADACOM:

```
// JOB ADACOM
// *-----*
// *      ADABAS PARALLEL SERVICES ADACOM Vv.r.s STARTUP      *
// *-----*
// LIBDEF PHASE,SEARCH=(SAGLIB.ASMvrs,SAGLIB.ADAvrs)
// EXEC ADARUN,SIZE=ADARUN
ADARUN PROG=ADACOM
/*
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
/*
/&
```

Note:

ADACOM must run in a dynamic partition.

Step 5: Create a Startup Procedure for Each Cluster Nucleus

 To 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.

Note:

Each Adabas cluster nucleus must run in a dynamic partition.

2. Concatenate the Adabas Parallel Services load library ahead of the Adabas load library in the LIBDEF PHASE SEARCH statement.
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).

4. Specify for each nucleus the ADARUN parameters CLUSTER, NUCID, 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*.

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.
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.

The following sample JCL executes the Adabas ADARUN program to implement session parameters for an Adabas Parallel Services cluster nucleus.

Note:

The Adabas Parallel Services library is concatenated ahead of the Adabas library in the LIBDEF PHASE SEARCH statement.

```

// JOB ADANUC
//*-----
/** 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
/**-----
// LIBDEF PHASE,SEARCH=(SAGLIB.ASMvrs,SAGLIB.ADAvrs)
// DLBL ASSOR1,'EXAMPL.DBYYY.ASSOR1'
// EXTENT SYS020
// DLBL DATAR1,'EXAMPL.DBYYY.DATAR1'
// EXTENT SYS021
// DLBL WORKR1,'EXAMPL.DBYYY.WORKR1'
// EXTENT SYS022
// DLBL PLOGR1,'EXAMPL.DBYYY.PLOGR1'
// EXTENT SYS027
// DLBL PLOGR2,'EXAMPL.DBYYY.PLOGR2'
// EXTENT SYS028
// ASSGN SYS009,PRINTER
// EXEC ADARUN,SIZE=ADARUN
ADARUN PROG=ADANUC
ADARUN CLUSTER=LOCAL
ADARUN MODE=MULTI
ADARUN SVC=SSS <--- INSERT YOUR SVC NUMBER
ADARUN DBID=YYY <--- INSERT YOUR DATABASE ID
ADARUN NUCID=NNNNN <--- INSERT YOUR NUCLEUS ID
ADARUN CLUCACHESIZE=XXXXXXXX <--- INSERT YOUR CACHE SIZE
ADARUN CLULOCKSIZE=XXXXXXXX <--- 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: 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*

.