

# Initialization

This section provides information about initialization in an Adabas cluster environment.

This chapter covers the following topics:

- Sequence
  - ADACOM Process
  - Adabas Cluster Nucleus Process
  - PPT Processing
- 

## Sequence

Due to the interdependence among Entire Net-Work, ADACOM, and Adabas cluster nuclei, these programs have certain co-occurrence requirements.

- Whether currently running or not, ADACOM must have been executed in the local operating system image when there are cluster users but no cluster nuclei to set up the environment.
- Entire Net-Work can start before or after ADACOM or a nucleus is active.

Note that although ADACOM and several Adabas cluster nuclei could run without Entire Net-Work within a single operating system image, there could then be no Adabas Cluster Services programs for that database in any other operating system image.

- A nucleus will not start if another nucleus with the same NUCID (whether local or remote) is already active.

## ADACOM Process

The ADACOM initialization process is recorded in *PLInnn* messages.

All *PLInnn* messages are printed to the console. *PLI001-049* messages are specific to a particular *SVC/DBID* set and are written to the *SYSOUT* data set dynamically allocated for that set; starting with *PLI050*, the messages apply to ADACOM in general and are written to the *COMPRINT* data set.

A new Adabas Cluster Services control block is acquired if none yet exists.

### Note:

Prior to *PLI050*, the system default ADARUN parameter values for *MODE*, *DBID*, *DEVICE*, *SVC*, and *AMODE* are displayed. These are not the values for the current ADACOM.

```
PLI050 00161  INITIALIZING ADACOM
PLI002 00161  INITIALIZING DBID=dbid SVC=svc
                ACQUIRING NEW PLXCB
                PLXCB IS LOCATED AT address
PLI063 00161  PROCESSING: ADACOM SVC=svc,DBID=dbid,NU=users
                INITIALIZATION COMPLETE
```

## Adabas Cluster Nucleus Process

Each Adabas cluster nucleus serializes during initialization by means of an enqueue. This is done because each nucleus puts information into the CSA nucleus table.

```
PLX050 00161 ADACLU INIT  DBID=00161  NUCID=00132
PLX006 00161 PLXCB LOCATED AT 0C893000
ADAN03 00161 INITIALIZING NUCID=132  INTNUCID=2
ADAX20 00161 XCF TRANSPORT INITIALIZATION COMPLETE
ADAX28 00161 IXCJOIN XCFTI      RET 00000004 RSN 00000004
ADAX28 00161 IXCQUERY XCFTI     RET 00000000 RSN 00000000
PLX059 00161 INITIALIZATION OF ADACLU COMPLETE
ADAN03 00161 ADABAS COMING UP
ADAX31 00161 OPENING WORK DATASET FOR NUCID=132
```

Access to the coupling facility lock structure is established and acknowledged by messages issued by the operating system:

```
IXL014I IXLCONN REQUEST FOR STRUCTURE ADA_LOCK11
WAS SUCCESSFUL.  JOBNAME: USADFM7 ASID: 00C0
CONNECTOR NAME: DB00006P00002N02 CFNAME: DCF1
IXL015I STRUCTURE ALLOCATION INFORMATION FOR
STRUCTURE ADA_LOCK11, CONNECTOR NAME DB00006P00002N02
  CFNAME      ALLOCATION STATUS/FAILURE REASON
-----
  DCF1        STRUCTURE ALLOCATED
  DCF2        PREFERRED CF ALREADY SELECTED
```

Adabas Cluster Services follows the operating system messages with lock structure statistics:

```
ADAX70 00161 CONNECTED TO LOCK STRUCTURE ADA_LOCK11
ADAX70 00161 NUMBER OF LOCK ENTRIES                32,768
ADAX70 00161 MAX NUMBER OF RECORD ELEMENTS        11,184
```

Access to the coupling facility cache structure is established and acknowledged by messages issued by the operating system:

```
IXL014I IXLCONN REQUEST FOR STRUCTURE ADA_CACHE11
WAS SUCCESSFUL.  JOBNAME: USADFM7 ASID: 00C0
CONNECTOR NAME: DB00006P00002N02 CFNAME: DCF1
IXL015I STRUCTURE ALLOCATION INFORMATION FOR
STRUCTURE ADA_CACHE11, CONNECTOR NAME DB00006P00002N02
  CFNAME      ALLOCATION STATUS/FAILURE REASON
-----
  DCF1        STRUCTURE ALLOCATED
  DCF2        PREFERRED CF ALREADY SELECTED
```

Adabas Cluster Services follows the operating system messages with cache structure statistics:

```
ADAX57 00161 CONNECTED TO CACHE STRUCTURE ADA_CACHE11
ADAX57 00161 DIRECTORY ELEMENTS                    11,490
ADAX57 00161 DATA ELEMENTS                        2,872
ADAX57 00161 DATA ELEMENT SIZE                    1,024
ADAN19 00161 BUFFER FLUSH IS A S Y N C H R O N O U S
ADAN01 00161 A D A B A S Vv.r.s IS ACTIVE
ADAN01 00161 MODE = MULTI
ADAN01 00161 RUNNING WITHOUT RECOVERY LOG
```

## PPT Processing

The parallel participant table (PPT), which exists for both cluster and noncluster nuclei, is used to determine if any PLOGs still need to be copied from previous sessions. If the PPT indicates that PLOGs remain to be copied, the PLOG data sets are read and, if necessary, the user exit (user exit 2 or user exit 12) is invoked.

### First Sysplex Cluster Nucleus Starts

The Adabas sysplex cluster nucleus that is the first to initialize checks all the PLOG entries from the previous session for all nuclei and marks any that are "still being written" to completed status. In this way, the user exit (user exit 2 or user exit 12) need not be called each time a cluster nucleus autostarts. The first nucleus then calls the user exit but waits only if the PLOGs that need to be copied are for its own NUCID.

### Subsequent Sysplex Cluster Nucleus Starts

A subsequent cluster nucleus checks only its own PLOGs and invokes user exit 2 or user exit 12 if the PLOGs are still not copied/merged. It waits if the user exit instructs it to. If there is no user exit 2 or user exit 12, it overwrites the PLOGs.

### Noncluster Nucleus Starts

A noncluster nucleus checks whether the previous session was a sysplex cluster session and has a pending autorestart. If so, the noncluster nucleus is not allowed to start.

If PLOGs from a previous sysplex cluster session remain to be copied, ADARES invokes the merge or the PLCOPY as required. A noncluster nucleus always uses block 1 of the PPT and can only overwrite it when PLOGs from previous sessions have been processed to completion.

A user exit 2 or user exit 12 controls the copy/merge process. If there is no user exit 2 or user exit 12, the PLOG and PPT entry are overwritten.

### Different PLOG Detected

If PLOGRQ=FORCE is specified and an uncopied PLOG is detected that does not match that specified in the last session, a parameter error occurs. If the PLOG has been copied, the PPT entry is overwritten and the nucleus starts.