

ADVANCED JOB MANAGEMENT



AJM - Advanced Job Management

**Advanced Job Management
System Manual**

..... **T** .. Systems ..

Impressum:

©Copyright T-Systems International GmbH, Fasanenweg 9, 70771 Leinfelden-Echterdingen, Germany
All rights reserved.

Publisher

T-Systems International GmbH

Computing Services & Solutions (CSS)/System Products & Solutions (MSY-PS)

Responsible

T-Systems International GmbH

Computing Services & Solutions (CSS)/System Products & Solutions (MSY-PS)
Fasanenweg 9, 70771 Leinfelden-Echterdingen, Germany
cc.jobmgmt@t-systems.com
+49 89/1011-4685

Document information

Name of file

K:\Product_Manuals\DOC\AJM\Filial.\$ajmsyse\MASTER.\$AJMSYSE.doc

Version / Revision

Date

This edition relates to AJM/S version

19/01/2005 2:15 PM

AJM V3.10 PTF QZ03089

/

Word 5

Abstract:

This book is also available in Portable Document Format (Acrobat PDF) . It will be delivered with the SDM package.

Table of Contents

Impressum:	2
Document information	2
Table of Contents	3
Introduction	9
1 Scenario Job Management	11
2 AJM product family	13
3 System overview	15
3.1 Global / local AJM systems	15
3.2 Collecting Events ("Event routing")	16
3.3 Cross-system administration	16
3.4 Files in the AJM Complex	17
3.5 Use of the data space	18
External Systems	19
4 External systems	21
4.1 Characteristics	21
4.2 The AJM agent	21
4.3 Coupling to SAP/R3	22
4.4 Installation of the AJM agent	23
4.5 Operation of the AJM agent	24
4.5.1 Starting the agent	24
4.5.2 Checking the status	24
4.5.3 Ending the agent	24
4.5.4 Files in operation	24
4.5.5 Error analysis and remedy	26
4.5.6 Messages in ajmagt.log	28
Master-Administrationsfunktionen	31
5 Selection menu	33
6 Utilities	35
6.1 Restore deleted definitions	35
6.2 Cleanup definitions	35
6.3 Migration of definitions (e.g. JES3 nets)	37

6.4	Analysis of event files	37
6.5	Restoration of databases	37
6.6	Traces	38
6.7	Checking the installation	38
6.8	Merging AJM systems	38
7	System Information	41
7.1	Display of system information	41
7.2	Authorization check and logging	42
8	Parameterics	43
8.1	Overview of the parameter tables	43
8.2	The parameterization dialog	44
8.2.1	Parameterization during installation	44
8.2.2	Changing system parameters	61
8.2.2.1	Changing the AJM parameter module	61
8.2.2.2	Creating and initializing AJM files	62
8.2.2.3	Changing the AJMNames module	62
	Operation	63
9	Notes on operation	65
9.1	Start/Stop considerations	65
9.1.1	Start of the AJM system	65
9.1.2	Ending the AJM system	65
9.2	Database expansion	66
9.3	Expansion of the restart file	66
9.4	Expansion of the log files	66
9.5	Expansion of the Event files	67
9.6	Expansion of the AJM/P database	67
9.7	Database backup	67
9.8	Database restoration	68
9.9	Utility "AJMDBMRG"	70
9.10	Utility "AJMDBCLN"	73
9.11	Disaster strategies	77
9.11.1	Loss of individual job definitions	77

9.11.2	Loss of the database	78
9.11.3	Loss of the restart database	78
9.11.4	Loss of an event file	78
9.11.5	Loss of the AJM/P database	78
Interfaces		81
10 User exits in AJM		83
10.1	AJMSUX1: Submit pre-processing (1)	83
10.2	AJMSUX2: Submit pre-processing (2)	84
10.3	AJMLGUX1: Logging	84
10.4	AJMADUX1: Interface to the print management system	84
10.5	AJMADUX2: Interface to the SYSOUT management system	84
10.6	AJMADUX3: Interface to the job log system	85
11 System exits		87
11.1	SEM (System Extensions and Modifications)	87
12 Interfaces to automation tools		91
13 Data Security		93
13.1	Protection of Jobs / Nets / Started Tasks / External Systems	93
13.2	Protection of Other AJM Resources	95
13.3	Protection of External Triggers	96
13.4	Protection of JCL Libraries	97
13.5	Protection of Preparation Dialogs and Packages	97
13.6	Protection of AJM/P variables	97
13.7	Protection of AJM/P file definitions	98
13.8	Summary	98
14 Command / Batch / Program Interface		103
14.1	Introduction	103
14.2	Return codes	103
14.3	DD names	104
14.4	Examples	104

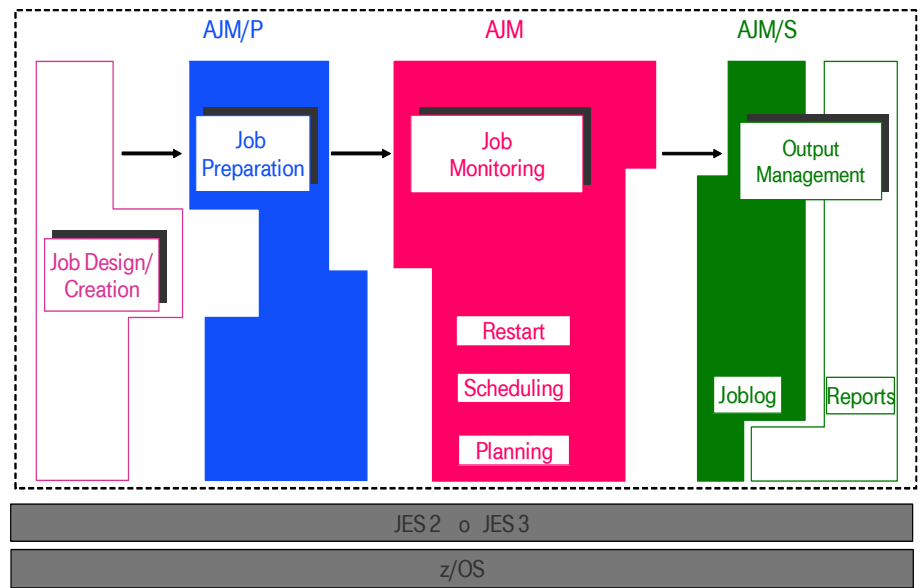
14.5	Syntax	105
14.6	Commands and their parameters	106
14.6.1	The parameters GROUP / NET / JOB	108
14.6.2	The ADDRESS command	108
14.6.3	The PROFILE command	108
14.6.4	The LRSUPD command	110
14.6.5	The LRSWHO command	110
14.6.6	The JCDLOCK command	110
14.6.7	The JCDUNL command	110
14.6.8	The JVRDEL command	110
14.6.9	The JVRSET command	110
14.6.10	The XTRSET command	111
14.6.11	The LOGREAD command	112
14.6.12	The LOGLIST command	113
14.7	AJM Definition File (AJMIN)	114
14.7.1	AJMIN Structure for Job Definitions	114
14.7.2	AJMIN Structure for Documentation	117
15	Batch/Command interface for job variables	119
15.1	SET - Creating or changing job variables	119
15.2	IF - Querying a job variable	121
15.3	GET - Reading a Job Variable	122
15.4	DEL - Deleting a job variable	123
15.5	Examples	123
16	Setting external triggers from programs	125
17	Program AJMTEST	127
Appendix		129
18	Appendix A. Notes on troubleshooting	131
18.1	SVC dump for problems in the AJM system	131
18.2	SYSDUMP for problems in the AJM administration	131
18.3	"DIA" line command	131
18.4	Program AJMUTEVT	132
18.5	Trace options in the AJM system	135
18.6	Special job status information	136
18.7	Checklists for common problems	138
18.7.1	Jobs in status "submitted"	138
18.7.2	Dataset trigger is not triggered	139

18.7.3	Communication error with an external system	140
19	Appendix B. Messages	141
20	Appendix C. Operator interface	143
20.1	Overview of the operator commands	143
21	Appendix D. Storage space requirements for the AJM files	145
22	Index	146

Introduction

1 Scenario Job Management

AJM Products



The entire spectrum of job development in a computer center is understood as the meaning of the expression job management. It therefore includes all components from the creation of a job on to the control/archiving of job runs or job output.

The above divisions assume the following distribution of functions:

- **Job Design / Creation**

As part of this section, the process for an application or a field of work, according to the process logistics, which results from the application is created. By using the variable parts in the JCL or the definition of the process it is possible to only have to make changes when the logic of the process changes.

- **Job Preparation**

Variable parts which are planned into the process during the job design are resolved by the values necessary for the next run during the job preparation. The variability refers to the replacement of parts of the JCL as well as to changes in the sequence of the process, provided this makes sense in terms of the definers. The replacement process usually takes place in the form of a dialog with the user since values which can be determined automatically can be used in the JCL at the job start.

- **Job Control**

At the beginning of the job monitoring phase there is a run-capable JCL for a run, however without the variable parts which are replaced during the job start. Job monitoring contains the components planning, scheduling and restart.

During the planning section criteria are determined according to which a specific process is to be started. Usually the process is started by a scheduling system. The job scheduled this way is monitored by the job management system, other jobs are started (or not) depending on the type of job end.

Job monitoring therefore contains the entire part from the planning of a job, its start, reactions depending on the result of a run, and any repetitions of the a run.

- **Output Management**

The final part of the job management scenario is output management. It is divided into two sections job log / sysout management and list administration.

- Joblog / Sysout management

Part of joblog / sysout management is the administration of logs created by the system as part the job run. The logs can be viewed online according to specific criteria and may be printed. The archiving of logs for reasons of updates is an important component.

- List administration

Many processes create lists with data related to applications. They can be printed or sometimes only viewed online. The data to be printed can be grouped according to various criteria so that the printout, for example, can be done specifically for the recipient. Several reports may be created from a list if necessary. Access to lists and reports can be by various criteria and the data may be archived for update purposes.

List administration therefore includes both the control of the printout as well as availability online, administration, and archiving the data.

When in the following job management is referred to, then the entire process is to be understood from the design of the process to the administration of its results.

2 AJM product family

The AJM product family includes functions for design, preparation, handling and control of all your batch processing. You can use the capabilities of AJM, especially the various schedule criteria to automate your restart/recovery possibilities. Such optimization is absolutely necessary due to the continually shrinking batch window.

The AJM products run 24 hours a day 7 days a week. Both the reorganization and the backup are done on the fly without any noticeable influence on availability.

All AJM products are client-capable, each customer only sees his own data. The easy-to-use, comfortable administration interface allows the monitoring of all your daily batch processes with little effort. Context sensitive help information makes work easier. The administration interface is available in both German and English.

AJM (= Advanced Job Management) is a client capable scheduler for the automation of batch processes which can run under different operating systems.

Batch processes in AJM can be:

- Jobs and started tasks under z/OS with JES2
- Jobs and started tasks under z/OS with JES3
- Background processes under SAP/R3
- Shell scripts under Unix
- Commands under Unix and Windows

You may control the start of a job through any combination of the following schedule criteria:

- Time criteria
 - date and time
 - Intervals (e.g. every hour, every second day, once a week)
 - Days of the week (e.g. Monday, Tuesday)
 - Types of day (working day, day of rest, holiday) based on individual calendars
 - Relative days (e.g. last working day of the week)
- Dependencies:
 - on a previous job (job in the same or a different net)
 - on another net
 - on the active state of another job or net
- Additional criteria:
 - Logical resources (controls the number of parallel jobs)
 - Contents of job variables (these may be set from any application)
 - Dataset trigger (reaction to the allocation of or a change to a file)
 - External trigger
 - Valid preparation with AJM/P

Additional features:

- Restart / Recovery
 - Reaction to termination codes (condition codes, system / user abends, special conditions as IPL)
 - Restart of the job or net, start of a recovery net
- Variable JCL:
 - Resolution of variables
 - IF / ENDIF / IMBED

AJM/P (AJM Job Preparation) is a client capable tool to prepare processes. You can create user-friendly dialogs using a set of dialog elements. The end user then calls up the dialog to prepare the preparation.

AJM/P dialog element are

- Input of AJM/P variables
- Display of ISPF panels
- File creation with the ISPF File Tailoring Service
- Access to files using ISPF Edit
- Execution of REXXs

AJM/S (AJM Sysout Management) is a client capable tool to administer joblogs. The joblogs are read in from the JES spool and archived daily.

Using the administration interface you can select joblogs and then and then process them (display, print, copy). AJM/S also contains a powerful function to access to the Syslogs which allows you select them based on date / time as well as according to their contents.

AJM/J (AJM Job Variables) enables exchanging information between any application using values that are stored in job variables. You may use these job variables in AJM as schedule criteria or to adjust JCL.

AJM/R (AJM Remote) is an agent on non-mainframe platforms that schedules batch processes, supervises them and reports the current state to the scheduler AJM.

3 System overview

3.1 Global / local AJM systems

Within a JES complex (JES3 global / local or JES2 in the MAS complex) there is at least one AJM that

- contains the net/job definitions as well as all other definitions e.g. calendar or logical resources, in a database
- operates the scheduling, i.e. passes the jobs to be executed to JES
- conducts the communication with the administrator.

An AJM system with these functions is called *global AJM*.

If AJM/P is installed, then the global AJM system also contains the system functions of AJM/P. These run as subtasks in the address space of the global AJM system.

An AJM has to run on all systems in the JES complex for reporting to the global AJM when

- jobs or started tasks are started or ended
- IPL is executed
- job variables are changed
- files which act as dataset triggers are cataloged or updated
- external triggers are set
- a net is prepared in AJM/P
- AJM/S has information about the job.

An AJM system with these functions is called *local AJM*.

Each local AJM system writes the events listed above in an event file which is read cyclically by the global AJM. A requirement for the use of AJM in a JES complex is therefore a "shared DASD" environment.

Local AJM systems are also responsible for the start of started tasks (start command).

If the "JES complex" consists of a single system, only one AJM system that serves both global and local functions is sufficient.

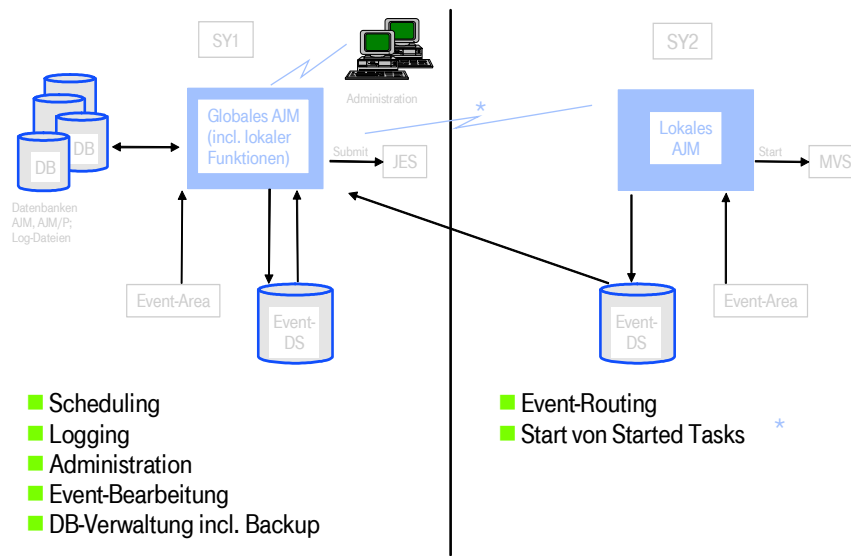
Within a JES complex usually *one* AJM system which serves both global and local functions is set up. On the other systems of the complex there is only an AJM system that serves the local functions.

The link up of a global AJM and the associated local AJM systems is called an *AJM complex*.

Up to four AJM complexes are possible for each JES complex; however, for general operation *one* AJM complex is adequate.

The following picture shows a global and a local AJM system:

Globales/lokales AJM



3.2 Collecting Events ("Event routing")

All events which occur on a computer and which are of importance for AJM (e.g. the start of a job which runs under AJM control) are covered by the local AJM systems.

Those events are reported by system exits that write a record for each event into a buffer in the ECSA. The data from this area is read out by the local AJM system and written into the event file.

The event buffer in the ECSA is allocated when AJM is started for the first time after an IPL. It is anchored using the name/token facility of the operating system. Events can only be routed when the event buffer has been initialized. Events which occur beforehand are lost.

When the local AJM is ended during running operations, the event buffer remains installed. When the system is restarted the events that have occurred in the meantime are brought into an event file as described above provided the buffer area is large enough.

3.3 Cross-system administration

The communication of the administrator with the global AJM uses VTAM. Appropriate commands can be used for cross-system access to the global AJM administration both via the online administration and batch / program interface.

The administration can take place from any computer which can create a VTAM connection via the network to a global AJM. In order to prevent misuse of this function, the authorization test always takes place on the global AJM that was addressed. If the administrator does not have a user ID there, any access is denied.

The following functions are **not** possible across systems if no "shared DASD" environment exists:

- The administration of job variables
- The access to the AJM log files
- The access to the JCL files (e.g. via "EDJ" or "BRJ" line commands)
- The parameterization of the AJM complex
- The execution of AJM/P preparation dialogs

3.4 Files in the AJM Complex

An AJM complex has the following files:

File	Function	DSORG	Number
<i>AJM database</i>	Contains all information on <ul style="list-style-type: none"> • Jobs • Logical resources • Event files 	LDS (VSAM)	1 per AJM complex
<i>AJM restart database</i>	Contains all information on <ul style="list-style-type: none"> • Calendar • JCL definitions • JCL of the last job run (restart JCL) • Files which were created on the last job run (provided "DS recording" is defined) 	KSDS (VSAM)	1 per AJM complex
<i>AJM/P database</i>	Contains all information on <ul style="list-style-type: none"> • AJM/P variables • Preparation dialogs • AJM/P link definitions 	KSDS (VSAM)	1 per AJM complex
<i>Event file</i>	Contains event records, i.e. information on <ul style="list-style-type: none"> • Job start / end of AJM controlled jobs • IPL • Job variables • Files which serve as dataset triggers 	DA	1 per system and AJM complex

<i>Log file</i>	Contains log records on system and user actions in AJM and AJM/P	PS	2 per AJM complex
<i>Backup file</i>	Contains the backup of the information from the AJM database and restart file (without Restart JCL and recorded files) as well as the backup of the AJM/P definitions	PS (GDG)	1 GDG per AJM complex GDG limit parameterizable
<i>Log archive dataset</i>	Contains the backup of the information from the AJM log files	PS (GDG)	1 GDG per AJM complex GDG limit parameterizable
<i>Job variable dataset</i>	Contains the job variables	KSDS (VSAM)	1 per JES complex

These datasets, with the exception of the job variable dataset, are created and initialized as part of the parameterization, either individually or as a complete set, (see also "Parameterics" on page 41) The job variable dataset must be created separately (see also AJM Installation Manual)

For reasons of performance and recovery, it is recommended that the AJM and AJM/P databases and the restart file not be created on the same disk as the event files. Furthermore, it is recommended to allocate the job variables separately from these files since a comparatively large number of accesses from the various address spaces takes place.

3.5 Use of the data space

The parameterization of the AJM system can be used to determine whether a data space should be set up when the global AJM system is started. This contains a copy of the AJM database which is re-created every full hour.

This copy is accessed when very large quantities of data are to be read. This applies to generic forecast requests in which statements are made on all jobs defined, e.g. to create a daily schedule. Since the AJM database is processed concurrently by many subtasks, such very time consuming access has a negative influence on the performance of the overall system. On the other hand, resources are required to create and maintain the data space contents. It is up to the AJM administrator to decide for or against the use of the data space.

If the parameterization indicates that the data space should not be created, then all accesses take place on the original database.

External Systems

4 External systems

4.1 Characteristics

External systems are those systems which have connections to other platforms and connections to other OS/390 systems which are outside the local JES2/JES3 complex. The goal is to be able to start and monitor jobs on these external systems. The scheduling of the jobs is performed centrally by AJM on the OS/390 host.

The following external systems are currently supported:

- SAP/R3 under UNIX (Sun Solaris 2.6)

The support of the following systems is planned:

- UNIX (execution of shell scripts)
- SAP/R3 under WINDOWS NT
- WINDOWS NT
- OS/390 Unix System Services
- non-local OS/390 systems

Generally the connection of external systems has the following characteristics:

- The scheduling is done centrally by AJM on the host.
- The control data ("JCL") is maintained centrally on the host.
- Job initialization and monitoring are performed by an AJM agent (UNIX, Windows NT, SAP/R3).
- NCI from T-Systems International GmbH is used for communication with the external systems.
 - TCP/IP for UNIX, SAP/R3, Windows NT (AJM agent required)
 - VTAM LU 6.2 for OS/390 systems

4.2 The AJM agent

An AJM agent is a simply structured assistant function which is installed on an external system (UNIX, Windows NT) and has the following tasks:

- Initiation of batch processes (e.g. SAP/R3 jobs, UNIX shell scripts) on the external system
- Status monitoring of the batch processes
- Reporting status changes (job start, job end, job error) to the AJM host

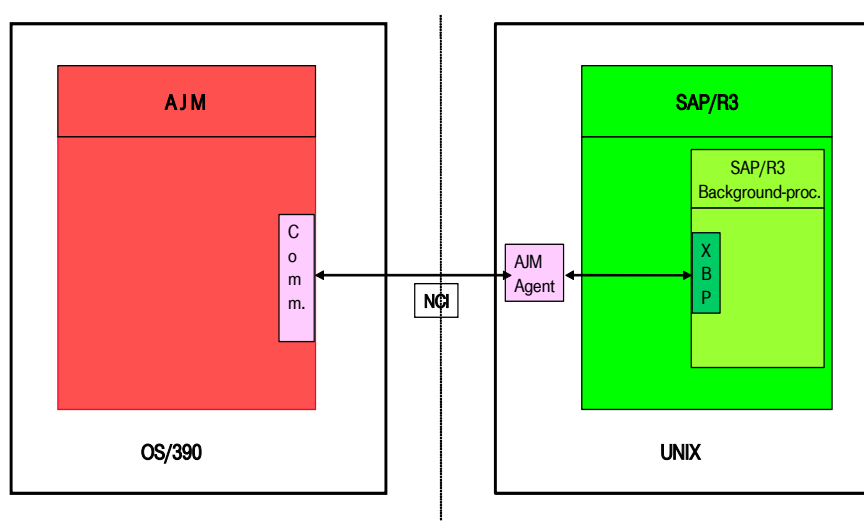
The AJM agent is currently available for the following platform:

- UNIX on Sun Solaris 2.6

Additional platforms are planned.

4.3 Coupling to SAP/R3

AJM 2.1: Kopplung zu SAP/R3



The following **technical requirements** are necessary for the control of SAP/R3 jobs (background processes) by AJM:

- SAP/R3 Release 4.5 (support for the XMI/XBP interface).
- Installation of an AJM agent on a UNIX system (starts and monitors the background processes).
- Runtime environment for NCI from T-Systems International GmbH on the UNIX system (for the communication between the AJM host and the AJM agent).

Notes on performance:

The installation of the AJM agent does not necessarily have to be on the same UNIX system on which the SAP/R3 system is installed, but this is recommended for reasons of performance.

Any number of SAP/R3 systems which are located on different UNIX computers can be operated using an AJM agent. The only requirement is that the SAP/R3 system can be reached by the AJM agent using TCP/IP.

If several SAP/R3 systems are operated which are located on different UNIX computers, then it is recommended that a separate AJM agent be installed on each for reasons of performance. This has the following advantages:

- Higher throughput through parallel processing

- Improved response times for status queries for the SAP/R3 system (faster since the query does not go via the network)
- Reduction of the risk of loss due to faults like
 - Problems with the TCP/IP communication between the AJM host and the AJM agent
 - Problems with the TCP/IP communication between the AJM agent and SAP/R3
 - Ending individual SAP/R3 systems or UNIX base systems, e.g. because of maintenance work or general system problems

4.4 Installation of the AJM agent

The components necessary for the installation of the AJM agent are supplied as a compressed archive file ("tar file"). Since this archive file is not a component of the basis delivery supplied with ISS OS/390, please consult the AJM team if necessary so that transfer to the desired UNIX carrier system can be initiated.

The installation of the AJM agent takes place in the following steps:

1. Creating a directory in which the AJM agent is to be installed (e.g. "/opt/ajm").
2. Unpack the archive file ("uncompress ajmagt.tar.Z")
3. Extract all files from the archive file ("tar vfx ajmagt.tar")

The following files are found in the archive file:

ajmagt	Executable program, contains the functional logic of the AJM agent
ajmagtt	Executable program, is used for communication with the AJM host via TCP/IP
runajmagt.sh	Model of a shell script to start the AJM agent
ajmagt.cfg	Configuration file, contains the control parameters for the AJM agent

4. Adjustment of the shell script to start the AJM agent

The following variables must be adjusted in the shell script "runajmagt.sh":

\$NCIHOME	Indicates the directory in which the NCI runtime environment is installed
\$AJMHOME	Indicates the directory in which the files which were unpacked in step 3 are found
\$WORKDIR	Indicates the directory in which the working files of the AJM agent are to be created. It is recommended that the same directory as under \$AJMHOME be used

5. Adjustment of the configuration file

In the configuration file "ajmagt.cfg" the parameter **IP-PORT** must be adjusted. This parameter determines which IP port is to be used for the communication with the AJM host. The default value entered is Port 9004.

4.5 Operation of the AJM agent

4.5.1 Starting the agent

The **start** of the AJM agent is done by executing the shell script "runajmagt.sh".

4.5.2 Checking the status

In order to check whether the AJM agent is active, please use the following command:

```
ps -ef | grep ajmagt
```

The command returns an overview of the UNIX processes of the AJM agent:

```
root 25063      1  0 07:21:30 ?        0:15 ajmagt -d -w /ajm ....
root 25064 25063  0 07:21:30 ?        0:01 ajmagt -d -w /ajm ....
root 26373 25064  0 13:49:37 ?        0:00 ajmagtt
```

The individual processes have the following functions:

The native process (PID 25063) is the actual control logic of the AJM agent. It carries out the following actions:

- Initiation of SAP/R3 background processes
- Status monitoring of the SAP/R3 background processes
- Sending status reports (job started, job ended) to the AJM host

The second process (PID 25064) is the communication manager of NCI, it is initiated by the native process at the start. The communication manager makes the TCP/IP connection (receive side) to the AJM host. It accepts all jobs which come from the AJM host.

The third process (PID 26373) is a TP (transactions process). It is started by the communication manager and carries out the actual interpretation of the jobs received and transmits them to the control logic of the AJM agent.

The TP is not constantly active. It is activated as needed, i.e. when data is sent by the AJM host, and ends itself when no new data is sent for 1 minute.

4.5.3 Ending the agent

Proceed as follows to end the AJM agent :

1. Determine the process numbers of the AJM agent (see above)
2. End the AJM agent with **KILL pid**, whereby pid is the process number of the native process (in the example 25063).
3. If the processes are not all ended after entering the KILL command, then the AJM agent must be ended by a **KILL -9 pid** for all active processes.

Note: The "hard" ending of the process with KILL -9 does not mean any danger for the consistency of the job files administered. The internal administration process is designed so that it can resume after an abort without any problems.

4.5.4 Files in operation

The AJM agent uses the working files described below in running operation. They are created automatically if they do not already exist or if they have been deleted.

ajmagt.tab Job file

This file is the central administration point for the job files. It is used by the AJM agent to control the SAP/R3 background processes which are to be started or monitored.

ajmagt.tab.out Temporary auxiliary file

The AJM agent uses this auxiliary file to reorganize the central job file at regular intervals. All job data is deleted in the reorganization which has been processed and successfully reported back to the AJM host.

ajmagt.rsq Resend queue

This file is used by the AJM agent when information is to be sent to the AJM host but the connection via TCP/IP cannot be made. All information is temporarily stored there until the connection can be restored and the information can be transmitted.

ajmagt.rsq.out Temporary auxiliary file

The AJM agent uses this auxiliary file to reorganize the resend queue at regular intervals. All entries are deleted in the reorganization which has been processed and successfully reported back to the AJM host.

ajmagt.log Log file of the AJM agent

All actions of the AJM agent and reports on errors which have occurred are logged in this file, for example:

- Error on creating a connection to the SAP/R3 system
- Error on initialization of SAP/R3 background processes
- Error in communication with the AJM host

Note: When the day changes the file is copied into an archive file with the name "ajmagt_log_ddd", the contents of the original file are deleted. The log file therefore only contains information from the current day. The time stamp "ddd" is formed from the running day of the year.

ajmnci.log Log file of the NCI communication manager

In this file all the actions of the communication manager are logged.

Note: The file is continued in running operation, no automatic reorganization takes place. It is therefore recommended that the file be deleted at regular intervals, if desired the file can be copied into an archive file beforehand. The AJM agent must be ended to carry out this action.

ajmagtt.log Log file of the TP program

All the actions of the TP started by the communication manager are logged in this file.

Note: When the day changes the file is copied into an archive file with the name "ajmagtt_log_ddd", the contents of the original file are deleted. The log file therefore only contains information from the current day. The time stamp "ddd" is formed from the running day of the year. If an error occurs when creating the archive file, then this will be

logged in the file "ajmagt.log".

dev_rfc Log file of the RFC interface of SAP/R3

This file is created by the RFC routines (remote function call) of the SAP/R3 interface. It contains the RFC-specific error messages.

Note: The file is continued in running operation, no automatic reorganization takes place. It is therefore recommended that the file be deleted at regular intervals, if desired the file can be copied into an archive file beforehand. The AJM agent does not have to be ended to carry out this action.

4.5.5 Error analysis and remedy

This chapter is intended to help you analyze and remedy errors which occur during operation. The following sources of information are available for your use:

In the environment of the AJM host

AJM log The call is made using Function 20 from the main menu or with the jump command "LOG". All errors which occur in the communication with the AJM agent are logged in the AJM log. Equally so, errors which occur in communication between the AJM agent and SAP/R3 are logged. All error messages in connection with external systems have the reporting code "AJMxCnnE".

Status display The status display for external systems can be viewed with a call using Function 9 in the main menu and entry of the line command "STA" (Status) in the list of the external systems. The display shows error messages when there is a problem at that time in the communication with the AJM agent in addition to some statistical data.

SYSLOG When communications problems occur between the AJM host and the AJM agent, NCI writes error messages into the SYSLOG.

In the environment of the AJM agent

Log files The individual function blocks of the AJM agent write both process and error messages into various log files. Which file can be used for analysis when an error occurs depends on the error in question (see below). Since the files are on a UNIX carrier system a LOGON on the UNIX system on which the AJM agent is installed is necessary. You must make sure that the necessary authorization level to read the files is set up for your user ID.

The following section is intended to assist you in the analysis and remedy of errors during operation:

Check whether there are connection problems using TCP/IP. You can use the program "NCIPING" for this. Call the program under ISPF Function 6 as follows:

```
====> nciping -a TCPIP -1 IP-addr. -2 IP-port#
```

Use the entries from the external system definitions as values for "IP-addr." and "IP- port#".

Check whether the communication manager was initialized by NCI during the start. After start of the AJM agent there must always be 2 processes with the name "ajmagt" active, where the process with the higher process number is the manager. If this process is not active, then please search the log file "ajmagt.ncilog" for possible error entries. Please consult the AJM team for additional error diagnosis.

Check whether the data transmitted cannot be entered into the job table of the agent. Check the file "ajmagt.log" to do this. If the job data cannot be entered, then the following error message repeats:

... AJMT00128 File lock: ajmagt.tab (lock/2) Already locked.

If the problem persists for a longer period of time, check to see whether the AJM agent is still working. If there is no problem, then the AJM agent writes the following message into the file "ajmagt.log" at regular intervals:

... AJMD00125 Waiting for work...

If this message does not appear, then do a restart of the AJM agent. If the problem cannot be solved by a restart, then please consult the AJM team.

Check whether there are connection problems with TCP/IP. Search the log file "ajmagt.log" for NCI error messages to do this (Message number "AJMD00116").

A possible error is, for example, that a Port# for TCP/IP communication was not entered in the AJM parameter module. In this case the AJM agent is not able to send messages on the job status to the AJM host.

If a job is in status "K-error(R)" a communications error which can be remedied has occurred, the AJM host cyclically (every minute) attempts to restore the connection (retry). There are two types of error situation which could lead to this:

1. The connection to the AJM agent cannot be made.
2. The AJM agent cannot make a connection to the SAP/R3 target system.

on 1.:

Check whether the AJM agent on the UNIX carrier system has been started. In this case, check whether the external system definition contains the correct values for the IP address and IP port of the AJM agent. Both cases lead to the same error status. Check the external system definitions with the line command "STA". You then receive the following error message:

```
NCI function      : NCIPGET
NCI returncode   : 16
NCI reasoncode   : 6004
NCI message      : NCI6004E SCLREQIP : TCP/IP connect() failed. Cannot connect
                  : to Hostname: 53.113.209.10, Servicename: 9005. TCP/IP Reason
                  : Code: 61.
```

If another error message appears, then please contact the AJM team.

If you do not have authorization to display the external system definitions, then please use the AJM log function and select the message category "System". You then see the following display in the case of an error:

```
SYS:  AJMXC17E - NCI ERROR:  FUNCTION=NCIPGET,RC=16,REASON=6004,XSD=.....
SYS:  AJMXC18E - NCI MESSAGE: NCI6004E SCLREQIP : TCP/IP connect() .....
SYS:  AJMXC18E - NCI MESSAGE: 209.10, Servicename: 9005. TCP/IP Reason ...
on 2.:
```

Check the log file of the AJM agent (ajmagt.log) and search for error messages which contain the text "RFC_ERROR". These messages provide information about what problem has occurred when making a connection to SAP/R3. The following is an example:

```
09/Oct/2000:07:07:57}AJMD00122 RFC_ERROR during{other failure}, message{LOCATION
CPIC (TCP/IP) on local host ERROR          connection to partner broken TIME
Sat Oct  7 12:13:15 2000 RELEASE }
```

Additional diagnosis information on RFC (remote function call) problems is found in the file "dev_rfc". It is created and described by the RFC routines of the SAP/R3.

4.5.6 Messages in ajmagt.log

So that the messages in the log file can be described better, here are the messages for a correct job :

```
{11/Dec/2000:14:55:43}AJMD00122 Table of all Handles before a new open
{11/Dec/2000:14:55:43}AJMD00122 Entry 00 :  Handle: 00000000
.....AJMD00122 Entry 3E :  Handle: 00000000
{11/Dec/2000:14:55:43}AJMD00122 Entry 3F :  Handle: 00000000
```

The existing connections are displayed when you log on to another SAP system so that you have a better overview of the current connections of the agent (here is the table in abbreviated form).

```
{11/Dec/2000:14:55:43}AJMD00122 RfcOpen - user{AJMUSER} language{DE}
hostname{11.22.333.44} sysNr{00} GW{11.22.333.44} GW service{sapgw00}
{11/Dec/2000:14:55:43}AJMD00122 RfcOpen - RFC_HANDLE: 00000001
{11/Dec/2000:14:55:43}AJMD00122 XMI_LOGON - RFC_HANDLE{00000001} ExternalCompany{T-
Systems}
ExternalProduct{T-Systems-ajm} Interface{xbp} Version{1.0}
{11/Dec/2000:14:55:43}AJMD00122 XMI_LOGON - RfcRc{0}, Returntype{
ReturnNumber{000},
SessionId{354A710B5AE33A34000}
```

By calling the function RfcOpen, the connection with the appropriate SAP system is initiated. After successfully calling it a logical connection number (handle) is returned. All subsequent function calls use this handle. The XMI logon is the actual logon to the SAP system with user ID and password.

```
{11/Dec/2000:14:55:43}AJMD00122 Table of all Handles after open and XMI-Logon
{11/Dec/2000:14:55:43}AJMD00122 Entry 00 : E1Z110AJMUSER
11.22.333.4411.22.333.44sapgw00
Handle: 00000001
{11/Dec/2000:14:55:43}AJMD00122 Entry 01 : Handle: 00000000
.....AJMD00122 Entry 3E : Handle: 00000000
{11/Dec/2000:14:55:43}AJMD00122 Entry 3F : Handle: 00000000
After completing the logon all connections are listed again. If a job is now to be
started on a "connected" SAP system, then no buildup of the connection is necessary.

{11/Dec/2000:14:55:43}AJMD00122 XBP_JOB_OPEN - RFC_HANDLE{00000001}
ExternalUser{AJMADM}
ExternalJobname{SAP_SAPNET_SAPJOB}
{11/Dec/2000:14:55:44}AJMD00122 XBP_JOB_OPEN - RfcRc{0}, Returntype{ },
ReturnNumber{000},
JobCount{14554301}
{11/Dec/2000:14:55:44}AJMD00122 XBP_JOB_ADD_ABAP_STEP - RFC_HANDLE{00000001}
ABAP{ABAPPROG}
Variante{WAIT10} ExternalUser{AJMADM} JobCount{14554301} JobName{SAP_SAPNET_SAPJOB}
Language{DE} JobUser{AJMUSER}
{11/Dec/2000:14:55:44}AJMD00122 XBP_JOB_ADD_ABAP_STEP - Destin Printimm Release
Copies Priarcmode Showpasswd Sapbanner
{11/Dec/2000:14:55:44}AJMD00122 XBP_JOB_ADD_ABAP_STEP - Bannerpage Expiration
Printrecip Numlines Numcolumns PrintPrm
{11/Dec/2000:14:55:44}AJMD00122 XBP_JOB_ADD_ABAP_STEP - Sapobject Info ArchPrm
{11/Dec/2000:14:55:44}AJMD00122 XBP_JOB_ADD_ABAP_STEP - RfcRc{0}, Returntype{ },
ReturnNumber{000}, StepNumber{1}
{11/Dec/2000:14:55:44}AJMD00122 XBP_JOB_CLOSE - RFC_HANDLE{00000001}
ExternalUser{AJMADM}
JobCount{14554301} JobName{SAP_SAPNET_SAPJOB}
{11/Dec/2000:14:55:44}AJMD00122 XBP_JOB_CLOSE - RfcRc{0}, Returntype{ },
ReturnNumber{000}
```

A new job is opened using the XBP interface when a new SAP job is started. All parameters are filled and then the definition is completed. A job ID is assigned by the SAP system on opening with which the job is clearly identified.

```
{11/Dec/2000:14:55:44}AJMD00122 XBP_JOB_START_IMMEDIATELY - RFC_HANDLE{00000001}
ExternalUser{AJMADM} JobCount{14554301} JobName{SAP_SAPNET_SAPJOB}
{11/Dec/2000:14:55:44}AJMD00122 XBP_JOB_START_IMMEDIATELY - RfcRc{0}, Returntype{ },
ReturnNumber{000}
{11/Dec/2000:14:55:44}AJMD00110 Sent Msg to Adress: 99.888.777.66, Port: 9876, TP:
AJMXCRCV
The job defined previously is now released for processing with Start Immediately.
This successful submit is then reported to AJM.

{11/Dec/2000:14:55:52}AJMD00122 XBP_JOB_STATUS_GET - RfcRc{0} JobCount{14554301}
JobName{SAP_SAPNET_SAPJOB} Returntype{ } ReturnNumber{000} JobState{F}
{11/Dec/2000:14:55:53}AJMD00110 Sent Msg to Adress: 99.888.777.66, Port: 9876, TP:
AJMXCRCV
```

Since the SAP system does not have any interfaces (exits) which report the changes in the job status, the AJM agent queries cyclically (every 5 seconds). The receipt of JobState "F" means that the corresponding job ended correctly. This information is then reported to AJM.

```
{11/Dec/2000:14:55:03}AJMD00117 No resend file available, name=ajmagt.rsq
```

This message is only to inform you that at this time no resend file has been created. If this file is required by an agent, then it is automatically created.

```
{11/Dec/2000:14:41:37}AJMD00122 RFC_ERROR during{other failure}, message{LOCATION  
SAP-Gateway on host host01 ERROR      hostname '11.22.999.99' unknown TIME  
Mon Dec 11 14:41:37 2000 RELEASE 45B COMPONENT NI (network interface)  
VERSION  
33 RC          -2 MODULE      niuxi.c LINE      370 DETAIL      NiPHos}
```

The incorrect IP address of SAP system is acknowledged with an Rfc error. The message contains the incorrect address in single quotes.

```
{11/Dec/2000:14:42:52}AJMD00122 RFC_ERROR during{other failure}, message{LOCATION  
CPIC (TCP/IP) on local host ERROR      hostname '11.22.999.99' unknown TIME  
Mon Dec 11 14:42:52 2000 RELEASE 46B COMPONENT NI (network interface) VERSION  
34 RC          -2 MODULE      niuxi.c LINE      405 DETAIL      NiPH}
```

This message is very similar to the one for the incorrect entry of the host name. However, the 'local host' is indicated for the IP address.

```
{11/Dec/2000:14:44:17}AJMD00122 RfcOpen - user{AJMXXXX} language{DE}  
hostname{11.22.333.44}  
sysNr{00} GW{11.22.333.44} GW service{sapgw00}  
{11/Dec/2000:14:44:17}AJMD00122 RfcOpen - RFC_HANDLE: 00000005  
{11/Dec/2000:14:44:17}AJMD00122 XMI_LOGON - RFC_HANDLE{00000005} ExternalCompany{T-  
Systems}  
ExternalProduct{T-Systems-ajm} Interface{xbp} Version{1.0}  
{11/Dec/2000:14:44:17}AJMD00122 XMI_LOGON - RfcRc{3}, Return Type, Return Number,  
SessionId{AJMD00110 Sent M}  
{11/Dec/2000:14:44:17} AJMD00122 RFC_ERROR during{system exception raised}, message  
{Keine Berechtigung zur Anmeldung am Zielsystem (Fehlercode 1).}
```

After an attempted logon a return code 3 is displayed when the user ID or the password is not allowed.

Master-Administrationsfunktionen

5 Selection menu

After calling the AJM administration interface you see the following menu as master :

```
AJM ----- Main Menu (Master) -----  
CMD===>  
  
Action: ___  0. Session Options  
              N. News  
  
              1. Net/Job Processing  
              2. Mass Update  
              3. Job Variables  
              4. Logical Resources  
              5. Dataset Trigger  
              6. JCL Dataset Definition  
              7. Calendar  
              8. Forecast  
              9. External systems  
  
             10. AJM/P Preparation dialogs  
             11. AJM/P Variables  
             12. AJM/P Datasets  
  
             20. Log View  
             21. Group-specific Statistics  
             22. Utilities  
  
             30. System Information  
             31. Parameters
```

Points 0 to 21 are described in detail in the user handbook. There you will find the utilities available to all users, as well (item 22). The additional functions in the master menu are:

22 Utilities

Additional programs for system maintenance and system control.
More detailed information on this can be found in "Utilities"above.

30 System Information

Display of important AJM system information, for example the maintenance level, state of the AJM database, overview of the number of jobs/nets which are defined under AJM.
More detailed information on this can be found in "System Information"on page 35.

31 Parameters

Adjustment of the AJM system to the specific conditions in your computer center.
More detailed information on this can be found in "Parameterics"on page 41.

6 Utilities

The following tools are available in the menu item "Utilities" with the master authorization (at least "READ" on "#GLOBAL"):

```
AJM ----- Utilities -----  
CMD==> SCROLL==> PAGE  
Selection: __ 1. Restore deleted definitions  
              2. Cleanup definitions  
              3. Migrate definitions (e.g. JES3 nets)  
  
              10. Analyze event datasets  
              11. Restore databases  
              12. Traces  
              13. Check installation parameters  
              14. Merge AJM systems
```

6.1 Restore deleted definitions

This function is described in the user handbook.

6.2 Cleanup definitions

Using the utility described below, you can get an overview of the definitions which may need cleaning up. These include, for example, jobs that have not run for some times, documentation texts or AJM/P dialogs for which the corresponding net definitions no longer exist. Optionally, such definitions can also be deleted directly.

When you choose selection 2 in the Utilities menu the following panel is displayed:

```
AJM ----- Utility AJMDBCIN -----  
CMD==> SCROLL==> PAGE  
Cleanup      : Jobs: Y      Documentation texts: N  
              : AJM/P Dialogs: N  
              : "Old JCL": N      Recorded datasets: N  
Selection    : Group: * _____  
              : Job not run since: _____ days (Cleanup of jobs)  
              : Net doesn't exist: N  
              : Age: _____ days ("Old JCL" and datasets)  
Function     : List: Y      Delete: N      Output DS: N  
Output DS   : _____  
Text        : _____
```

The following entries are possible:

Cleanup Type of definitions that are to be cleaned up

Here you should enter whether you want to clean up job definitions, documentation texts, AJM/P dialogs, "old JCL" records and / or records about the files of the last job run.

Selection The following selection criteria are available:

Group Name of a group

The group name may only be abbreviated with "*" when neither documentation texts nor AJM/P dialogs are selected. A check of the group name is **not** performed for the clean up of "old JCL" and file records. All other types of definitions are only selected when they belong to the group given.

Job not run since Entry stating how long ago the last start of the job must be.

Only those job definitions are selected which fulfill this condition.

Net doesn't exist Only those documentation texts or AJM/P dialogs are selected for which no associated AJM net or job definition exists.

Age Only those "old JCL" records which have reached this age given.

Function Type of processing

Here you decide what is to happen to definitions which fulfill the selection criteria.

List All definitions which fulfill the criteria given are listed. The list is written into the file allocated in AJMPRINT.

Delete All definitions which fulfill the criteria given are deleted.

Output DS For each definition which fulfills the criteria given a record is written into the output file allocated under AJMOUT. The structure of this record is determined in the field "Text".

Output DS The name of the output dataset can be entered here. Otherwise the output is done to SYSOUT.

Text Layout of the output records

If output into a file was selected, then you can determine here how the records are to be structured. The variables group (%1), net (%2), and job (%3) can be placed anywhere in the text which may be freely defined.

Example for the clean up of job definitions:

"DEL GROUP(%1) NET(%2) JOB(%3)" creates a file which can be used as the input for the JAM batch interface.

When you have made all entries, then press Return. This generates a job which is then displayed. You may start this job with "SUB" or leave the generated JCL with PF3.

6.3 Migration of definitions (e.g. JES3 nets)

This function is described in the user handbook.

6.4 Analysis of event files

When you choose selection 10 in the Utilities menu you are taken to the utility AJMUTEVT. A detailed description is found in "Program AJMUTEVT" on page 132.

6.5 Restoration of databases

With the aid of the utilities described below, AJM and AJM/P databases can be restored after a system error. AJM creates these backup files at regular intervals. They get all their definitions from AJM and AJM/P.

Before restoration the files must be created, for example using the parameterization dialog (see "Creating and initializing AJM files" on page 62).

When you choose selection 11 in the Utilities menu the following panel is displayed:

```
AJM ----- Utility AJMDBRLD -----  
CMD===> SCROLL==> PAGE  
  
Backup Dataset : HLQ.AJM.DATABASE.BACKUP.G0123V00_____  
  
AJM Database   : HLQ.AJM.DATABASE_____  
Restart Dataset: HLQ.AJM.RESTART_____  
AJM/P Database : HLQ.AJM.PREP.DATABASE_____  
  
Options       : Replace definitions: Y (y/n)
```

The following entries are possible:

- Backup Dataset** Name of the backup dataset which is to be used for the restoration of the databases
- AJM Database** Enter the name of the AJM database here if it is to be restored. This file must have already been created and initialized using the parameterization dialog.
- Restart Dataset** Enter the name of the AJM restart dataset if it is to be restored.
- AJM/P Database** Enter the name of the AJM/P database here if it is to be restored.
- Options** The following options can be entered:

Replace definitions

If you enter "Y" here then existing definitions in the restart file or in the AJM/P database are overwritten.
If you enter "N" then existing definitions in the restart file or in the AJM/P database are not overwritten.

When you have made all entries, then press Enter. This generates a job which is then displayed. You may start this job with "SUB" or leave the generated JCL with PF3.
Detailed information on the utility "AJMDBRLD" can be found under "Database restoration" on page 68.

6.6 Traces

This function is not yet available.

6.7 Checking the installation

When you choose selection 13 in the Utilities menu the CLIST AJMDFBAS is called. It checks the SEM environment of your installation.
Additional information can be found in the AJM Installation Manual.

6.8 Merging AJM systems

With the aid of the utility "AJMDBMRG" you can restore definitions which you accidentally deleted. This function is described in the user handbook.
In addition, you can use this tool to merge two AJM systems.

When you choose selection 14 in the Utilities menu the following panel is displayed:

```

AJM ----- Utility AJMDBMRG -----
CMD==>                                     SCROLL==> PAGE

Group          : * _____ Net: * _____ Job: * _____
Log. Resource  : * _____ Calendar: * _____ Ext. Sys: * _____
AJM/P Variable : * _____ Package: * _____

Definitions AJM : Net/Job: Y Documentation: N Ext. System: N
                  : Log. Res: N JCL-Def: N Calendar: N
                  : Dialog: N Variable: N Link-Def: N

Options        : Lock job          : Y
                  : Add job to net: N

AJM Database   : HLQ.AJM.DATABASE _____
Backup Dataset : HLQ.AJM.DATABASE.BACKUP.G0123V00 _____
    
```

The description of the upper fields is found in the user handbook.

Additional field:

**AJM
database** Name of the AJM database from which the job definitions and logical resources are to be brought into the current AJM system ("Merge").
Please remember that all other types of definitions can only be taken from the backup file.

When you have made all entries, then press Enter. This generates a job which is then displayed. You may start this job with "SUB" or leave the generated JCL with PF3.

7 System Information

You may query the AJM system information to get an overview of

- Which PTF version the active AJM system has
- When it was started
- How the database is allocated
- How many definitions exist
- How many jobs have run.

7.1 Display of system information

When you call the system information (item 30 in the main menu) the following panel is displayed:

```
AJM ----- System Information -----  
CMD====>                                     SCROLL====> CSR  
  
                FMID :          PAJM210  
                RMID :          QZ00nnn  
  
                Start date :      28.02.02  
                Start time :      15:43  
  
Database size      :          1.003.520 Bytes  
Free space         :          188.494 Bytes  
Fragmented space  :          30.296 Bytes  
  
No. of defined groups      :          43  
No. of defined nets       :          398  
No. of defined jobs       :          2.428  
No. of started jobs      :          290  
No. of terminated jobs    :          290  
No. of restart jobs      :          0
```

The fields displayed have the following meanings:

- FMID** Name of the SMP/E function
- RMID** Current PTF version of the AJM system
- Start date** Date on which the AJM system was started
- Start time** Time when the AJM system was started
- Database size** Current size of the AJM database

This value is not the physical size of the database, but provides the current size of the window with which the AJM system works. This window is dynamically expanded as necessary (see also "Database expansion" on page 66).

- Free space** Current free space in the window for the database (see above)

If this value undershoots a specific threshold value, then an automatic compression of the database takes place.

Fragmented space	Current fragmented space in the window of the database (see above) This fragmentation is the result of deleting or changing definitions. After compression of the database the fragmentation is again zero.
No. of defined groups	Number of groups which are defined in this AJM system
No. of defined nets	Number of nets which are defined in this AJM system
No. of defined jobs	Number of jobs which are defined in this AJM system
No. of started jobs	Number of jobs which have been started since the last start of the AJM system (see above)
No. of terminated jobs	Number of jobs which have been ended since the last start of AJM (see above)
No. of restart jobs	Number of jobs which have been started as part of a restart since the last start of the AJM system (see above)

7.2 Authorization check and logging

The following RACF profile is used for the authorization check for the query of the system information:

```
#GLOBAL
```

The authorization necessary there is READ.

No logging of this action takes place.

8 Parameterics

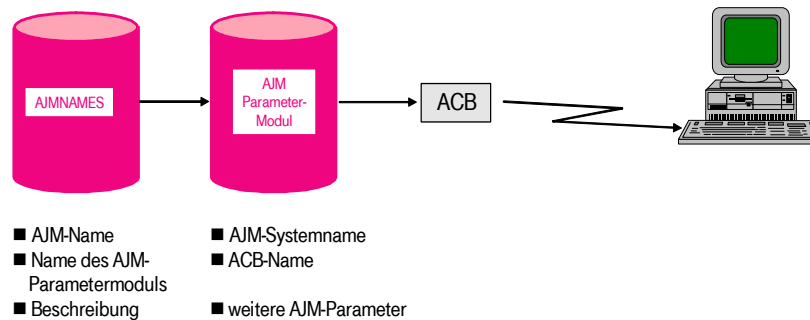
8.1 Overview of the parameter tables

The definitions necessary for the operation of an AJM system are kept in two types of parameter tables. They are to be created by the system administrator during the installation so that they can be adjusted to the specific circumstances of the computer center.

The parameter tables are created in the form of load modules. They are loaded dynamically by the AJM system at startup and by the user using batch / TSO calls and should therefore be kept in a generally available load module library. The creation of the tables is done using the menu system.

Which tables must be created and what dependencies there are between them can be seen in the following diagram:

AJM-Parameterstruktur



AJMNames Table of the global AJM systems

The individual global AJM systems are assigned the logical names in this table, which the user uses to address the AJM system desired. An entry consists of the logical name of the AJM system and the name of the AJM parameter module which describes the system assigned, as well as a comment field with a brief description of the system .

If a specific system is selected by the user using the logical name, then the associated AJM parameter module is determined and then loaded. The parameters necessary for the communication are then read from this module.

The AJMNames table is currently only used by the batch / command interface (see

"AJM" parameter in the "ADDRESS" command).

**AJM
parameter
module**

Contains all definitions which are necessary for the AJM system itself and for the administration interface. In detail, these are:

- Entries on the AJM complex
- Entries on the AJM files (all files which are required for the operation of AJM are dynamically allocated when run)
- Control parameters for scheduling, event processing, logging, etc.
- Parameter for batch / TSO communication (ACB name, timeout)

8.2 The parameterization dialog

In order to execute the parameterization dialog, the AJM master authorization is required, i.e. the user must have at least READ authorization in the general resource class \$AJM in the profile "#GLOBAL" (see the AJM Installation Manual).

8.2.1 Parameterization during installation

The Clist **AJMINST** is called during the parameterization of the system. Then the AJM main menu appears:

```
AJM ----- Main Menu (Master) -----  
CMD==>  
  
Action: __ 0. Session Options  
           N. News  
  
           1. Net/Job Processing  
           2. Mass Update  
           3. Job Variables  
           4. Logical Resources  
           5. Dataset Trigger  
           6. JCL Dataset Definition  
           7. Calendar  
           8. Forecast  
           9. External systems  
  
           10. AJM/P Preparation dialogs  
           11. AJM/P Variables  
           12. AJM/P Datasets  
  
           20. Log View  
           21. Group-specific Statistics  
           22. Utilities  
  
           30. System Information  
           31. Parameters
```

You move into the parameterization dialog with Function 31.

The parameterization dialog is only available in English, the administration language set under selection 0 "Session options" has no influence on the language of the parameterization dialog.

Then the following input panel appears:

```
----- AJM - Creation of a compound parameter table -----  
COMMAND ==>  
  
      INPUT LIBRARY ==> ISSDC.Vvv.AJM.SAJMLOAD _____  
      INPUT MEMBER  ==> AJMMDPRM  
  
      OUTPUT LIBRARY ==> SYS5.LINKLIB _____  
      OUTPUT MEMBER  ==> _____  
  
This dialog function will assist you in creating a new parameter table or  
viewing an existing one.  
  
Specify INPUT LIBRARY and INPUT MEMBER to obtain the definitions of an  
existing table as input for the new one.  
  
OUTPUT LIBRARY and OUTPUT NAME are used to define the targets for a  
subsequent following compile and link.  
  
To view the definitions of an existing parameter module, leave OUTPUT MEMBER  
MEMBER blank and press ENTER after each display.
```

In this first step in the dialog you decide what the AJM parameter module is to be called and in which load library it is to be stored.

INPUT LIBRARY Name of the input file for the parameter module

The name of the input file is to be entered where the parameter module is located which is to serve as a model.

INPUT MEMBER Name of the parameter module in the input file

The member name of the parameter module is to be entered which is to serve as a model.

Note: When the dialog is called for the first time, "ISSDC.Vvv.AJM.SAJMLOAD" is input as standard for the file and "AJMMDPRM" as standard for the member. This module is supplied with the installation and is preset with standard values.

OUTPUT LIBRARY Name of the output file for the parameter module

The name of the output file in which the new parameter module is to be saved should be entered here.

OUTPUT MEMBER Name of the parameter module in the output file

The member name of the parameter module which is to be created in the following dialog is to be entered here.

If no member name is entered, then the dialog only takes place in BROWSE mode, i.e. only the definition of the input member name is displayed. The dialog steps for the creation of a new module or the steps to create and initialize the AJM files are not offered.

Note: To make changes to the parameters effective, AJM must be ended and restarted. A refresh function does not currently exist.

After entering the values described above, the following panel is displayed:

```

----- AJM - SPECIFICATION OF GLOBAL PARAMETERS -----
COMMAND ==>

AJM-ID                ==> AJM1          (AJM1/2/3 or AJM4)
SYSTEM-NAME           ==> SY1          (of global)
EVENT ROUTING FUNCTION ==> Y           (Y or N)
SCL VTAM APPLICATION-ID ==> AP??AJMG TCP/IP PORT-# ==> 1234_
SCL TIMEOUT (SHORT/LONG) ==> 060 / 300 (seconds)
NAME OF DATA BASE    ==> ????????.AJM.DATABASE
NAME OF BACKUP DATA BASE ==> ????????.AJM.DATABASE.BACKUP
NAME OF RESTART FILE  ==> ????????.AJM.RESTART
NAME OF 1ST LOGGING FILE ==> ????????.AJM.LOGFILE1
NAME OF 2ND LOGGING FILE ==> ????????.AJM.LOGFILE2
NAME OF LOGFILE ARCHIVE DS ==> ????????.AJM.LOGFILE.ARCHIVE
SCHEDULE CYCLE TIME   ==> 005          (seconds)
DB RELEASE TIME       ==> 000          (seconds)
DB SAVE CYCLE TIME    ==> 120          (seconds)
DB BACKUP CYCLE TIME  ==> 024          (hours)
DB COPY IN DATASPACE  ==> N           (Y or N)
EVENT AREA SIZE OF ECSA ==> 50          (kilobytes)
NAME OF MESSAGE MODULE ==> AJMMSGEN

INSTALLATION WITH AJM/P ==> Y           (Y OR N)

```

The meanings of the individual parameters are as follows:

AJM-ID Name of the AJM system

It is possible to run up to four global AJM systems in a complex in parallel. The expressions global and local AJM system are described in detail in Chapter "Global / local AJM systems" on page 15.

Only the entries AJM1/2/3/4 are possible here.

SYSTEM-NAME	<p>System name of the computer on which the global AJM system is to run</p> <p>The name is the same as the "SYSNAME=" parameter in the "IEASYS" member. AJM determines the system name of its own computer from the CVT on system startup and compares it with this entry. If it agrees, then AJM operates in the global mode, if it does not agree then AJM operates in local mode.</p> <p>This standard way of working can be overwritten using a suitable "PARM" entry on system startup (see also AJM Installation Manual).</p>
EVENT-ROUTING	<p>Entry whether this AJM system should carry out "event routing"</p> <p>This parameter is only interpreted when AJM is started as a global system.</p> <p>It determines whether the global AJM system should also assume the functions of the local AJM system.</p> <p>If only a global AJM system is defined within a system complex, then this values should be set to "Y".</p> <p>The functions "global" and "local" are described in Chapter "Global / local AJM systems" on page 15.</p>
SCL VTAM APPLID	<p>Name of the VTAM application (ACB name) for this AJM system</p> <p>The VTAM-ACB name which is used by the global AJM system for communication must be entered here.</p>
TCP/IP PORT-#	<p>TCP/IP Port number which is to be opened for the TCP/IP communication with external systems. Additional information on external systems is found in "External systems" on page 19. If this field is left blank, then no interface for TCP/IP communication is initialized.</p>
SCL TIMEOUT	<p>Value for the SCL timeout detection</p> <p>Here you can enter after how many seconds NCI should detect a timeout situation. After expiry of this time SCL interrupts the requestor, i.e. the administrator receives a message and can decide whether he wishes to continue the dialog or end it.</p> <p>You may enter two values: "SCL TIMEOUT (LONG)" is only used for complex queries which ought to take longer (e.g. plausibility checks with the PLS command).</p>
NAME OF DATA BASE	<p>Name of the AJM database (cf. Chapter "Files in the AJM Complex" on page 17)</p>
NAME OF BACKUP DB	<p>Name of the GDG base for database backup datasets (see Chapter "Files in the AJM Complex" on page 17)</p> <p>AJM backs up the AJM database cyclically in generation files. The backup takes place during running operation. The cycle in which this backup takes place and how many generations are to be maintained is given in the following dialog. Which files are included in the backup is described in Chapter "Database backup" on page 67.</p>
NAME OF	<p>Name of the restart database (see Chapter "Files in the AJM Complex" on page 17)</p>

RESTART FILE

NAME OF 1ST LOGGING FILE Name of the first log file

NAME OF 2ND LOGGING FILE Name of the second log file

Note: AJM uses two files to write log records, they are preformatted on initialization for a fixed number of blocks.

If a file is full, then the system switches to the second. This is displayed by the message "AJMLG07I". The full log file is automatically archived.

NAME OF LOGFILE ARCHIVE DS Name of the GDG base for the archive datasets for the log files

The archive copies of the AJM log files are automatically generated when the change is made from one log file to the other or when they are manually changed (operator command "F ajm,LOGARCH=xxx"). Enter the name of the generation files here. How many generations are maintained is given in the following dialog.

SCHEDULE CYCLE Check interval of the scheduler

Enter the interval in which the scheduler should check whether the conditions for the submit for a job (date, time, events) are fulfilled. The value is to be given in seconds.

The value should be adjusted to suit the circumstances. Please remember that a short interval and the associated high degree of currency also means a high CPU demand.

A value of about 5 seconds is sensible.

DB RELEASE TIME Release interval of the scheduler

You can enter here after which interval the scheduler should interrupt its activity and release the AJM database so that other tasks (e.g. administration) may access it. If a zero is entered, then the AJM database will only be released when the scheduler has checked all jobs.

DB SAVE CYCLE Backup interval for the database

AJM generally works with a copy of the AJM database in the virtual storage. Logical changes to the databases are generally executed in the virtual storage.

This parameter determines the interval at which the changes are written to the hard disk. The access method used here ("DIV") optimizes the backup processes by not writing the entire database back onto the disk, but only those pages (4K blocks) which have been changed since the last save cycle.

DB BACKUP CYCLE Backup interval of the database

Enter the interval at which a copy of the AJM database and the significant information for operations from the restart database (calendar, JCL file definitions) are to be written into a backup file (GDS, see above). This value is given in hours.

The backup takes place during running operation without any noticeable

impairment of the schedule or dialog functions.

The backup file is the basis for the restoration of the database after a logical or physical defect.

DB COPY IN DATASPACE

Keep a copy of the database in a dataspace

If you enter "Y" AJM maintains a copy of the AJM database in a dataspace and updates it every full hour. This copy is used when large quantities of data are requested, e.g. as part of a forecast for all defined jobs.

If you enter "N", the access always takes place on the AJM database itself. A copy is not maintained. This means that the time necessary for the creation of the copy and indexes is not necessary.

EVENT AREA SIZE Size of the event buffer in the ECSA

All events relevant for AJM (e.g. job start, job end, change of job variables) are entered in the buffer by the system exits. From there the information is fetched cyclically by the local AJM system and written into an event file. More information on this process is available in Chapter "Collecting Events ("Event routing")" on page 16.

The event buffer is written using the "wrap around" process, the writing and fetching of the data is done asynchronously. If the buffer runs full because it is too small or the fetch interval is too large, this data is lost. The data loss is reported on the console (message **AJM0001W - AJMEHRSM: EVENT AREA FULL ...**)

The default value of 50K is normally adequate.

NAME OF MESSAGE MODULE

Selection of the message module for global and local AJM systems

Here you determine in what language AJM messages are written to the console.

Currently the following modules are available:

AJMMSGEN English

AJMMSGGE German

Note: This entry has no influence on the language in the administration dialog. The user can individually choose the language there.

INSTALLATION WITH AJM/P

Activation of AJM/P

Enter "Y" when AJM/P is to be activated in addition to AJM.

If you enter "N" then only the AJM functions are available in both the AJM system and in the AJM administration interface.

After entering the parameter and pressing the Enter key the following panel is displayed:

```
----- AJM - SPECIFICATION OF EVENTS TO BE LOGGED -----  
COMMAND ==>  
  
UPDATE  JOB ENTRY           ==> Y  (Y OR N)  
LOCK    JOB ENTRY           ==> Y  (Y OR N)  
UNLOCK  JOB ENTRY           ==> Y  (Y OR N)  
RESTART JOB ENTRY           ==> Y  (Y OR N)  
BYPASS  JOB ENTRY           ==> Y  (Y OR N)  
START   JOB ENTRY           ==> Y  (Y OR N)  
DELETE  JOB ENTRY           ==> Y  (Y OR N)  
CLEAR   JOB ENTRY           ==> Y  (Y OR N)  
MASS UPDATE OF JOB ENTRIES ==> Y  (Y OR N)  
DATASETS OF LAST RUN DELETED ==> Y  (Y OR N)
```

In this and the following panel you determine which actions and events are to be logged in the log files.

It is recommended that you use the preset values so that you have as complete as possible an overview of the events in AJM should you need it.

The individual fields have the following meanings:

UPDATE JOB ENTRY	Log record for each creation and change of a job definition
LOCK JOB ENTRY	Log record for each lock of a job ("LCK" command)
UNLOCK JOB ENTRY	Log record for each unlock of a job ("UNL" command)
RESTART JOB ENTRY	Log record for each manual restart of a job ("RMO" command with subsequent "REX")
BYPASS JOB ENTRY	Log record for each manual BYPASS of a job ("BYP" command)
START JOB ENTRY	Log record for each manual start of a job ("STR" / "RUN" / "EXE" command)
DELETE JOB ENTRY	Log record for each deletion of a job definition
CLEAR JOB ENTRY	Log record for each manual clean up of a job ("CLR" command)
MASS UPDATE OF JOB ENTRIES	Log record for each Job which was modified using "Mass Update"
DATASETS OF LAST RUN DELETED	Log record when newly created files were manually deleted in the last job run ("RES" command)

The following panel has the structure:

```

----- AJM - SPECIFICATION OF EVENTS TO BE LOGGED -----
COMMAND ==>

CHANGE  LRS ENTRY          ==> Y  (Y or N) LRS = LOGICAL RESOURCE
DELETE  LRS ENTRY          ==> Y  (Y OR N)

JOB STARTS BY AJM          ==> Y  (Y or N)
TERMINATION OF JOBS        ==> Y  (Y or N)
NET STARTS BY AJM          ==> Y  (Y or N)
NET CLEARED BY AJM         ==> Y  (Y or N)

JCD UPDATED                 ==> Y  (Y or N)
JCD DELETED                 ==> Y  (Y or N)
CALENDAR UPDATED           ==> Y  (Y or N)
CALENDAR DELETED           ==> Y  (Y or N)
DATASET TRIGGER ARRIVED    ==> N  (Y or N)
JOB-VARIABLE CHANGED       ==> N  (Y or N)
VALIDITY OF JOB PREPAR.    ==> Y  (Y or N)

GENERAL MESSAGES           ==> Y  (Y or N)

```

The individual fields have the following meanings:

- CHANGE LRS ENTRY** Log record for each creation and change of a logical resource
- DELETE LRS ENTRY** Log record for each deletion of a logical resource
- JOB STARTS BY AJM** Log record for each automatic start of a job
- TERMINATION OF JOBS** Log record for each end of a job
- NET STARTS BY AJM** Log record for each start of a net
- NET CLEARED BY AJM** Log record for each end of a net
- JCD UPDATED** Log record for each creation and change of a JCL file definition
- JCD DELETED** Log record for each deletion JCL file definition
- CALENDAR UPDATED** Log record for each creation and change of a calendar
- CALENDAR DELETED** Log record for each deletion of a calendar
- DATASET TRIGGER ARRIVED** Log record for each "dataset trigger" event
- JOB-VARIABLE CHANGED** Log record for each "job variable" event
Currently no log records of this type are written - regardless of what you enter.
Use the trace options for job variables for trace purposes.
- VALIDITY OF JOB PREPAR.** Log record for each change in the validity of a job preparation

If you enter "Y" then the setting or resetting of the validity of a job preparation is logged in the AJM log.
Note: Using the "STA" command the current state of the job preparation can be queried.
- GENERAL MESSAGES** Writing certain AJM system messages into the AJM log in addition to output on the system console

If you entered that AJM/P is to be installed in the first panel, then the following panel is displayed:

```

----- AJM/P - DATA BASE DEFINITION AND EVENTS TO BE LOGGED -----
COMMAND ==>

Definitions for AJM/P Database:

NAME OF PREPARATION DATA BASE ==> ????????.AJM.PREP.DATABASE

CYCLE FOR AJM/P FULL BACKUP ==> 02 (1 TO 99)
(With every n'th AJM-backup a full backup of the AJM/P database
will be taken)

Specification of events to be logged:

UPD/DEL PREP-DLG DEFINITION ==> Y (Y OR N)
UPD/DEL PACKAGE DEFINITION ==> Y (Y OR N)
UPD/DEL LINKDEF ==> Y (Y OR N)
UPD/DEL AJM/P-VARIABLE ==> Y (Y OR N)
UPD/DEL NET DOCUMENTATION ==> Y (Y OR N)
UPD/DEL JOB DOCUMENTATION ==> Y (Y OR N)

```

The meanings of the individual parameters are as follows:

- | | |
|------------------------------------|--|
| NAME OF PREPARATION DB | Enter the name of the AJM/P database |
| CYCLE FOR AJM/P FULL BACKUP | Enter how often the AJM/P database is to be backed up with the AJM database |
| | The AJM/P database contains, in contrast to the AJM database, very large quantities of data (e.g. variable definitions, dialog definitions) which change relatively infrequently. The backup of this data is therefore not as critical as the backup of the AJM database (scheduling, events). Therefore, and because it is a considerable quantity of data, it does not make sense to back up the AJM/P data every time the AJM database is backed up. However, the documentation of jobs and nets is always backed up with the AJM data. This ensures that all data on jobs /nets is of the same date. |
| | Using this parameter you can enter in which cycle a full backup should take place. The first backup after a restart of the AJM system is always a full backup. |
| | Note: Please check whether the number of backup generations is adequate so that at least one full backup is kept (data from AJM and AJM/P). |
| UPD/DEL PREP-DLG DEFINITION | Log record for each creation, change, and deletion of a preparation dialog definition |
| UPD/DEL PACKAGE DEFINITION | Log record for each creation, change, and deletion of a packet definition |
| UPD/DEL LINKDEF | Log record for each creation, change, and deletion of an AJM/P file definition |
| UPD/DEL AJM/P-Variable | Log record for each creation, change, and deletion of an AJM/P-variable |
| UPD/DEL NET DOCUMENTATION | Log record for each creation, change, and deletion of net documentation |

UPD/DEL JOB DOCUMENTATION Log record for each creation, change, and deletion of job documentation

After the parameterization of the global part of AJM has been completed, the definitions for the local AJM systems must be performed for each system.

Such a definition must take place for **each** system in the complex. Which local definition applies to the current system is detected later based on the system name.
Additional information on the global and local functions in AJM are found under "Global / local AJM systems" on page 15.

The definitions for the individual computers are performed on the following panel:

```
----- AJM - SPECIFICATION OF LOCAL PARAMETERS SETS -----  
COMMAND ===>  
  
SYSTEM-NAME          ===> SY1_____  
EVENT CYCLE TIME     ===> 005_____ (SECONDS)  
  
VTAM APPLID FOR LOCAL SYSTEM ===> AP??AJML  
  
EVENT FILE FOR AJM SYSTEM 1 ===> ???????.AJM.EVENT.SY1_____  
EVENT FILE FOR AJM SYSTEM 2 ===> _____  
EVENT FILE FOR AJM SYSTEM 3 ===> _____  
EVENT FILE FOR AJM SYSTEM 4 ===> _____  
  
MORE LOCAL DEFINITIONS ?   ===> Y          (Y OR N)
```

The individual fields have the following meanings:

SYSTEM-NAME System name of the computer on which the local AJM system is to run.

The name is that of the "SYSNAME=" parameter in the "IEASYS" member. AJM determines the system name of its own computer on startup from the CVT and compares this with the entries. If a suitable entry is found, then it works with the values entered there.
If no suitable entry is found, then the error message "AJMBF08E" ("No local definition found") appears and the started task is terminated.

EVENT CYCLE TIME Event cycle

The events collected in a buffer in the ECSA are read by the local AJM system and written into event files. The event cycle time indicates at what interval this takes place. The value is given in seconds.

Note: The minimum waiting time for "refresh" of the display in the administration is calculated based on the event cycle entry: at least one event cycle is waited out between two requests to the AJM system.

VTAM APPLID FOR LOCAL SYSTEM Name of the VTAM application for communication with the local AJM system

This ACB name is used for communication between the global AJM system and the local system in question (present once per computer).
If an AJM system has both local and global functions, then **the same ACB name** must be entered here as in the definition of the global parameters (field "SCL VTAM APPLICATION-ID").

- EVENT FILE FOR AJM SYSTEM 1** Name of the event file for the first AJM complex (global system AJM1)
- EVENT FILE FOR AJM SYSTEM 2** Name of the event file for the second AJM complex (global system AJM2)
- EVENT FILE FOR AJM SYSTEM 3** Name of the event file for the third AJM complex (global system AJM3)
- EVENT FILE FOR AJM SYSTEM 4** Name of the event file for the fourth AJM complex (global system AJM4)

MORE LOCAL DEFINITIONS Definition of additional local systems

If you enter **"Y"** here, then this panel is displayed again and the definitions for the next local AJM system can be entered.
Once all systems in the complex have been parameterized you must enter **"N"**.

As soon as you enter an "N" for **MORE LOCAL DEFINITIONS** the following panel is displayed:

```
----- AJM - DEFINE YOUR OWN MESSAGE MODULES -----  
COMMAND ==>  
  
The current version of AJM includes two message modules to support german  
and english languages for administration functions.  
If you want to extend the language definition table, type in the language  
code and the name of the appropriate language.  
  
LANGUAGE      LANGUAGE  
CODE            NAME  
  
EN            ENGLISH  
GE            DEUTSCH  
  
_____  
_____  
_____  
_____  
_____  
_____  
_____  
_____
```

AJM offers "National Language Support" (NLS), i.e. it is possible to set the administration interface to the appropriate national language.

The concept is based on a collection of language modules which contain the language-dependent parts (texts and constants). German and English are part of the standard delivery. If you require an expansion for other languages, then please contact the AJM development team.

You may activate additional language versions in this panel if any are present.

A language identifier and a comment field are entered in which the name of the language can be entered. These entries are then offered in the administration interface (selection 0, "Session options") for selection.

After all entries have been made, press the Enter key to generate the job.
This job is displayed in the edit mode:

```

EDIT ---- HLQ.SPFTEMP1.CNTL ----- COLUMNS 001 072
COMMAND ==> SCROLL ==> HALF
          ENTER SUB(MIT) TO START ASSEMBLY OR CANCEL/PF3 TO TERMINATE
000001 //jobname  JOB acctnt,
000002 //          CLASS=Q,MSGCLASS=V
000003 //ASM      EXEC  PGM=ASMA90,REGION=2048K,
000004 //          PARM='NODECK,OBJECT,USING(NOWARN) '
000005 //SYSLIB   DD     DSN=SYS1.MACLIB,DISP=SHR
000006 //          DD     DSN=SYS4.MACLIB,DISP=SHR
000007 //SYSUT1   DD     DSN=&SYSUT1,UNIT=VIO,SPACE=(1700,(600,100.))
000008 //SYSUT2   DD     DSN=&SYSUT2,UNIT=VIO,SPACE=(1700,(300,50.))
000009 //SYSUT3   DD     DSN=&SYSUT3,UNIT=VIO,SPACE=(1700,(300,50.))
000010 //SYSPRINT DD     SYSOUT=*,DCB=BLKSIZE=1089
000011 //SYSPUNCH DD     DUMMY
000012 //SYSIN   DD     *
000013          TITLE 'AJM COMPOUND PARAMETER TABLE'
000014 AJM1PARM CSECT
000015 AJM1PARM AMODE 31
000016 AJM1PARM RMODE ANY
000017 *-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
000018 *-*
000019 *-* PROGRAM      : any
000020 *-* PTF-LEVEL    : PAJM210
000021 *-* ATTRIBUTES  : reenterable
000022 *-* AMODE       : 31
000023 *-* RMODE       : any
000024 *-* This program is not executable. It is merely
000025 *-* a table module which contains basic
000026 *-* information used by the AJM system.

```

Please check the entries in the job card and submit the job, the parameter module is then generated.
The job must end with completion code 0 for all steps.

Now that the AJM parameter module has been generated, the files for the AJM system can be created and initialized.
Press the PF3 key to do this, the following panel then appears:

```

----- AJM DATASET ALLOCATION AND INITIALIZATION -----
COMMAND ==>

SELECTION ==> C      C or S

Enter C if you want to allocate a complete set of AJM-datasets or
S to select single datasets.

For S(elective) allocation, specify only the dataset names for those
datasets you want to allocate.

Press PF3 to suppress dataset initialization.

```

In the above panel you choose whether all ("C") or only some files are to be created and initialized. If you do not wish this, then you may exit the dialog using PF3.

During installation you naturally choose "C", selective creation is used, for example, after the loss of a file.

The following panel is displayed:

```

AJM-DATASET-DEFINITION-----
COMMAND ==>

ESTIMATED NUMBER OF JOBS    ==>
NAME OF AJM DATA BASE      ==> ????????.AJM.DATABASE
VOLSER                       ==> SMSVOL
NAME OF BACKUP DATA BASE   ==> ????????.AJM.DATABASE.BACKUP
GDG-LIMIT                    ==>
NAME OF RESTART FILE        ==> ????????.AJM.RESTART
VOLSER                       ==> SMSVOL

LOGFILE1 ==> ????????.AJM.LOGFILE1          VOLSER ==> SMSVOL
LOGFILE2 ==> ????????.AJM.LOGFILE2          VOLSER ==> SMSVOL

NAME OF LOGFILE ARCHIVE DS ==> ????????.AJM.LOGFILE.ARCHIVE
GDG-LIMIT                    ==>

DFSMS IS INSTALLED AND ACTIVE.
IF THE DATASET IS DFSMS MANAGED, THERE IS NO NEED TO SPECIFY ANY VOLSER.

```

The fields have the following meanings:

- ESTIMATED NUMBER OF JOBS** Number of jobs to be defined in AJM
Here you enter the maximum number of jobs that are to be defined in AJM. This value serves as the basis for calculating the size of the AJM database, restart database, log and event files. A table for size calculation is found in the appendix under Appendix D. "Storage space requirements for the AJM files" on page 143.
- NAME OF DATA BASE** Dataset name of the AJM database
The cluster for the linear dataset is allocated under this name. The data part has the same name with the additional qualifier "DATA".
- VOLSER** VOL=SER entry for the AJM database
The AJM database is created on this volume AJM database. If DFSMS is active, then no explicit volume entry is necessary.
- NAME OF BACKUP DATA BASE** The base GDG for the backup file is allocated under this name.
- GDG-LIMIT** Number of generations for the backup file
- NAME OF RESTART FILE** Dataset name of the restart database
The cluster for the KSDS file is allocated under this name. The data part has the same name with the additional qualifier "DATA", the index part has the additional qualifier

	"INDEX".
VOLSER	VOL=SER entry for the restart file
	The restart file is created on this volume. If DFSMS is active, then no explicit volume entry is necessary.
LOGFILE1	Dataset name of the first log file
	The die DA file is allocated under this name.
VOLSER	VOL=SER entry for the first log file
	The first log file is created on this volume. If DFSMS is active, then no explicit volume entry is necessary.
LOGFILE2	Dataset name of the second log file
	The die DA file is allocated under this name.
VOLSER	VOL=SER entry for the second log file
	The second log file is created on this volume. If DFSMS is active, then no explicit volume entry is necessary.
NAME OF LOGFILE ARCHIVE DS	The base GDG for the archive of the log files is allocated under this name.
GDG-LIMIT	Number of generations for the log archive file

When DFSMS is installed, a message appears in the lower part of the panel (see above). The VOLSER entries are then initialized to "SMSVOL" and do not need to be explicitly defined.

An overview of the files and their function in AJM can be found under "Files in the AJM Complex" on page 17.

The files entered are created and initialized when you press Enter. The course of this process is logged using messages in the lower portion of the screen.

In the following panel the event files are created and initialized. It has the following structure:

```

AJM-DATASET-DEFINITION-----
COMMAND ==>

NAME OF EVENT FILE      ==> ???????.AJM.EVENT.SY1
VOLSER                  ==> SMSVOL
NAME OF EVENT FILE      ==>
VOLSER                  ==> SMSVOL
NAME OF EVENT FILE      ==>
VOLSER                  ==> SMSVOL
NAME OF EVENT FILE      ==>
VOLSER                  ==> SMSVOL
NAME OF EVENT FILE      ==>
VOLSER                  ==> SMSVOL

DFSMS IS INSTALLED AND ACTIVE.
IF THE DATASET IS DFSMS MANAGED, THERE IS NO NEED TO SPECIFY ANY VOLSER.

```

The fields have the following meanings:

NAME OF EVENT FILE Dataset name of the Event file

The DA file is allocated under this name.

VOLSER VOL=SER entry for the event file

The corresponding event file is created in this volume. If DFSMS is active, then no explicit volume entry is necessary.

After all event files have been allocated, the message **PRESS END-KEY TO TERMINATE OR ENTER FOR ADDITIONAL DEFINITIONS** appears. Additional event files can be created or you may leave this panel with the PF3 key.

If an installation with AJM/P is carried out, then the AJM/P database is created in the following panel:

```
AJM/P-DATASET-DEFINITION-----  
COMMAND ==>  
  
NAME OF AJM/P-DATABASE    ==> ????????.AJM.PREP.DATABASE  
VOLSER                     ==> SMSVOL  
  
DFSMS IS INSTALLED AND ACTIVE.  
IF THE DATASET IS DFSMS MANAGED, THERE IS NO NEED TO SPECIFY ANY VOLSER.
```

NAME OF DATA BASE Dataset name of the AJM/P database

The cluster for the KSDS file is allocated under this name. The data part has the same name with the additional qualifier "DATA", the index part has the additional qualifier "INDEX".

VOLSER VOL=SER entry for the AJM/P database

The AJM/P database is created on this volume. If DFSMS is active, then no explicit volume entry is necessary.

When DFSMS is installed, a message appears in the lower part of the panel (see above). The files entered are created and initialized when you press Enter. The course of this process is logged using messages in the lower portion of the screen.

After you press the PF3 key the log of the allocation and initialization is displayed:

```

BROWSE -- AJM-DATASET-PROTOCOL ----- LINE 00000000 COL 001 080
COMMAND ==> SCROLL ==> PAGE
ENTER SAVE TO STORE THE PROTOCOL IN A SEQUENTIAL DATASET.
DEFINITION OF >?????????.AJM.DATABASE< IN PROGRESS.
IDCAMS SYSTEM SERVICES TIME: 12:50:

  DEFINE CLUSTER (NAME ('?????????.AJM.DATABASE')
                  VOLUME (SMSVOL)
                  RECORDS (00000125,00000062 )
                  SHAREOPTIONS (1 3)
                  LINEAR)
IDC0508I DATA ALLOCATION STATUS FOR VOLUME VOL107 IS 0
IDC0512I NAME GENERATED- (D) ?????.AJM.DATABASE.DATA
IDC0181I STORAGECLASS USED IS STANDARD
IDC0181I MANAGEMENTCLASS USED IS STANDARD
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0

IDC0002I IDCAMS PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 0
DEFINITION OF >?????????.AJM.DATABASE< ENDED SUCCESSFUL.
  WARNING : DATASET >?????????.AJM.DATABASE< - WAS ALLOCATED ON VOLSER VOL107

....

INITIALIZATION OF >?????????.AJM.EVENT.SY1< IN PROGRESS.
INITIALIZATION OF >?????????.AJM.EVENT.SY1< ENDED SUCCESSFUL.
  WARNING : DATASET >?????????.AJM.EVENT.SY1< - WAS ALLOCATED ON VOLSER VOL109

```

When you enter the command "SAVE", the log is saved to a file. The message **PROTOCOL WAS SAVED TO DATASET >filename<** is output after leaving the log display with PF3. The log was saved under the name "filename".

If you do not wish to backup the log, then just press PF3.

You now move to the dialog for the creation of the "AJM NAMES" module.

First a check is made to see if the module already exists. In this case its contents are displayed.

If no module exists, then the error message **"CSV003I REQUESTED MODULE AJM NAMES NOT FOUND"** is output and a panel with the following model entries is displayed:

```

AJM ----- List of AJM Systems ----- ROW 1 TO 1 OF 1
CMD ==> SCROLL==> PAGE

  Name of LOADLIB: xxxxxx_____

+-- Commands: R(epeat) I(nsert) D(elete)
!
V  AJM Name  Parameter Module  Description
-----
  ?????.AJM?PARM  PLEASE CHANGE THIS ENTRY_____
-----

```

The fields have the following meanings:

Name of LOADLIB	Name of the output file Enter the load module library in which the AJM NAMES module which is generated is to be placed. The field is preset with the file name you entered at the beginning of the dialog as "OUTPUT LIBRARY" for the AJM parameter module.
AJM Name	Name of the AJM complex The end user can address the global AJM using this name (see also "The ADDRESS command" on page 108). All AJM systems which are to be addressed using the batch / command interface must be defined here.
Parameter module	Name of the AJM parameter module The name of the desired parameter module belonging to the global AJM system must be entered here.
Description	Description of the AJM complex Any descriptive text may be entered here.

The following line commands are available to accept AJM system:

- D Delete line
- I Insert line
- R Repeat line

After all entries have been made, the job is generated after you press the PF3 key. This job is displayed in the edit mode:

```

EDIT ---- HLQ.SPFTEMP1.CNTL ----- COLUMNS 001 072
COMMAND ==> SCROLL ==> HALF
***** ***** TOP OF DATA *****
000001 //jobname JOB acctn,
000002 // CLASS=Q,MSGCLASS=V
000003 //ASM EXEC PGM=ASMA90,REGION=2048K,
000004 // PARM='NODECK,OBJECT,USING(NOWARN) '
000005 //SYSLIB DD DSN=SYS1.MACLIB,DISP=SHR
000006 // DD DSN=SYS4.MACLIB,DISP=SHR
000007 //SYSUT1 DD DSN=&SYSUT1,UNIT=VIO,SPACE=(1700,(600,100.))
000008 //SYSUT2 DD DSN=&SYSUT2,UNIT=VIO,SPACE=(1700,(300,50.))
000009 //SYSUT3 DD DSN=&SYSUT3,UNIT=VIO,SPACE=(1700,(300,50.))
000010 //SYSPRINT DD SYSOUT=V,DCB=BLKSIZE=1089
000011 //SYSPUNCH DD DUMMY
000012 //SYSIN DD *
000013 TITLE 'AJM NAMES TABLE '
000014 AJMNames CSECT
000015 AJMNames AMODE 31
000016 AJMNames RMODE ANY
000017 *-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
000018 *-* *-* *-* *-* *-* *-* *-* *-* *-* *-* *-* *-* *-* *-* *-* *-* *-* *-* *-*
000019 *-* PROGRAM : any *-*
000020 *-* VERSION : 00 *-*
000021 *-* ATTRIBUTES : reenterable *-*
000022 *-* AMODE : 31 *-*
000023 *-* RMODE : any *-*
000024 *-* This program is not executable. It is merely *-*
000025 *-* a table module which contains information *-*
000026 *-* about the AJM systems. *-*

```

Please check the entries in the job card, adjust the name of the MACLIB (DD name SYSLIB) and submit the job. Now the AJMNames module is generated. The job must end with completion code 0 for all steps.

You end the parameterization dialog with PF3.

8.2.2 Changing system parameters

If you wish to change system parameters, call the parameterization dialog (Function 31 in the Main menu).

As described in the previous chapter, the dialog is divided into three steps which are executed one after the other:

- Creation of the AJM parameter module
- Creation and initialization of the AJM files
- Creation of the AJMNames module

If you only wish to perform one function, then please proceed as follows:

8.2.2.1 Changing the AJM parameter module

If you wish to change an existing module, then enter the module name in the entry panel as the input member and the output member at the same time. If an existing module is to serve as a model for a module to be created, then enter this new module name as the output member. Change the AJM parameters according to your requirements and submit the generated job.

You then go to the entry panel to create the AJM files. Because no files are to be created, exit this dialog with PF3.

Then the panel for the creation of the AJMNames module is displayed. If no changes need to be made there, exit the dialog with PF3.

Note: AJM must be restarted for the changes to take effect, if necessary perform a **LLA REFRESH** beforehand.

8.2.2.2 Creating and initializing AJM files

To get to this step of the dialog you must first go through the AJM parameterization dialog. If no changes in parameters are desired, then exit the generated job with PF3.

You come to the entry panel for the AJM files. Enter an "S" in the field **SELECTION**. This entry means that only files whose names in the following panel are explicitly given are created and initialized.

Delete the names of the files which you do not want to create in the following panels.

After finishing this dialog step you come to the panel for the creation of the AJMNames module. If you do not wish to make any changes there, then exit the dialog with PF3.

8.2.2.3 Changing the AJMNames module

To get to this step of the dialog you must first go through the AJM parameterization dialog. If you leave the field **OUTPUT MEMBER** blank in the first dialog then the dialog operates in BROWSE mode, i.e. you only see the AJM parameters displayed.

Page through the individual parameter panel by repeatedly pressing the Enter key until you come to the panel to change the AJMNames module.

Then change the parameter to suit your needs and then submit the generated job.

A **LLA REFRESH** may be necessary to make the changes effective immediately.
A restart of AJM is not necessary.

Operation

9 Notes on operation

9.1 Start/Stop considerations

The following is a brief description of how AJM should be integrated into the computer operation. The procedure before or after an IPL is especially discussed.

9.1.1 Start of the AJM system

The message "AJMBF15I" is output at the start of the AJM system (for global AJM systems) or the message "AJMBF14I" (for local AJM systems). It contains the AJM ID (AJM1-4) and the name of the computer. This message indicates that the AJM system is ready with all its functions. If AJM/P is installed, then the message "AJPBF01I" shows that all AJM/P system functions are ready.

As described under "Collecting Events ("Event routing")" on page 16, the environment for the recording of an event is only set up after the first start of the AJM system after an IPL. Events which occur beforehand are lost.

As soon as an IPL is executed on a computer and the initiator address space is started, jobs may run which are either submitted by AJM and not started before IPL or were submitted by a global AJM on another system. In order to make sure that AJM detects all events (job start / end), AJM must be started **before the initiator address spaces**.

AJM can be started before VTAM is active.

If an ACB error takes place when the communication is opened, the NCI tries to open the ACB again after a brief time.

9.1.2 Ending the AJM system

As described under "Collecting Events ("Event routing")" on page 16, all events are placed in a buffer area in the ECSA. The local AJM writes the data from this buffer into the event file. The buffer area remains even after the AJM system has been stopped. There is no problem to end and restart AJM while the system is running - no information is lost. After an IPL the buffer area, however, is no longer available since the ECSA is regenerated during IPL.

In order to make sure that AJM detects all events (job start / end) before an IPL, the local AJM must remain active at shutdown **until the initiator address space is ended**.

The scheduler part of the AJM can be switched off ahead of time (using the operator command "F AJM,SCHED=STOP") to prevent the automatic start of jobs. The administration of AJM remains possible.

After the IPL the scheduler part which was switched off must be switched back on again since otherwise no scheduling takes place. This can be done either with the operator command "F AJM,SCHED=START" (s. Appendix C. "Operator interface" on page 1) or using the parameter "SCHED=START" in the STC procedure.

If AJM is terminated, then the message "AJMBF11I" is output. Afterwards the started task is terminated.

9.2 Database expansion

The AJM system does not use the entire physically allocated space in the database after installation for performance reasons. A window is placed over the VSAM file which is expanded as necessary.

This window is initialized to about 2000 Jobs (approx. 1 MByte) when AJM is installed. The free space in this window decreases as a result of new definitions and changes, deletions and changes create gaps in the sequential entries (fragmented space). The administration function "System information" can be used to see this effect (see "System Information" on page 35).

As soon as the free space undershoots a threshold value, AJM automatically undertakes compression of the database. This process takes place in running operation under the control of the AJM system; it only takes a few seconds. The database is not available in this time, however there are no noticeable limitations for the end user.

After this compression, the entries are sequential again and arranged in the database without any gaps. The fragmented space is also zero.

If after compression the size of the free space is still determined to be under the threshold value, then the window is expanded with for another 2000 jobs. If necessary a physical expansion of the file automatically takes place at the same time.

If the window or the file cannot be expanded, then an error message is output. The administration command which triggered the expansion is rejected, however the AJM system continues to run. All administration functions which do not create any new entries in the database (e.g. queries, start commands, etc.) remain possible.

However, in order to make new entries possible, the database must be manually expanded, i.e. a new database of adequate size must be created. Afterwards a RESTORE run is performed (see also "Database restoration" on page 68).

The creation of the database is done using the parameterization dialog (see "Creating and initializing AJM files" on page 62). The number of jobs given is used as the basis for the calculation of the size (see also Appendix D. "Storage space requirements for the AJM files" on page 143).

9.3 Expansion of the restart file

The restart file is a VSAM-KSDS file. This file must be copied with the aid of a REPRO job to be expanded. The KSDS file can then be deleted and re-allocated and initialized using the parameterization dialog (see "Creating and initializing AJM files" on page 62). The number of jobs given is used as the basis for the calculation of the size (see also Appendix D. "Storage space requirements for the AJM files" on page 143). Afterwards the new file is loaded with REPRO.

These measures must be undertaken when AJM is not active.

9.4 Expansion of the log files

If the log files are to be expanded, then proceed as follows:

First check to which log file is being used. The free log file is then renamed or deleted and then re-allocated and initialized using the parameterization dialog (see "Creating and initializing AJM files" on

page 62). The given number of jobs is used as the basis for the calculation of the size (see also Appendix D. "Storage space requirements for the AJM files" on page 143).

As soon as AJM has switched to the new log file, the other log file can be enlarged the same way.

9.5 Expansion of the Event files

You must end AJM to expand the event files.

The file must then be deleted and re-allocated and initialized using the parameterization dialog (see "Creating and initializing AJM files" on page 62). An expansion on the basis of the existing file or a backup is not possible, this would lead to problems with the processing of events.

The given number of jobs is used as the basis for the calculation of the size (see also Appendix D. "Storage space requirements for the AJM files" on page 143).

9.6 Expansion of the AJM/P database

The AJM/P database is a VSAM-KSDS file. A copy of the job must be made with the aid of a REPRO job to expand it. The KSDS file can then be deleted and re-allocated and initialized using the parameterization dialog (see "Creating and initializing AJM files" on page 62). The number of jobs given is used as the basis for the calculation of the size (see also Appendix D. "Storage space requirements for the AJM files" on page 143).

Afterwards the new file is loaded with REPRO.

These measures must be undertaken when AJM is not active.

9.7 Database backup

The backup of the AJM and AJM/P databases as well as part of the restart file is done during running operation.

Backed up are:

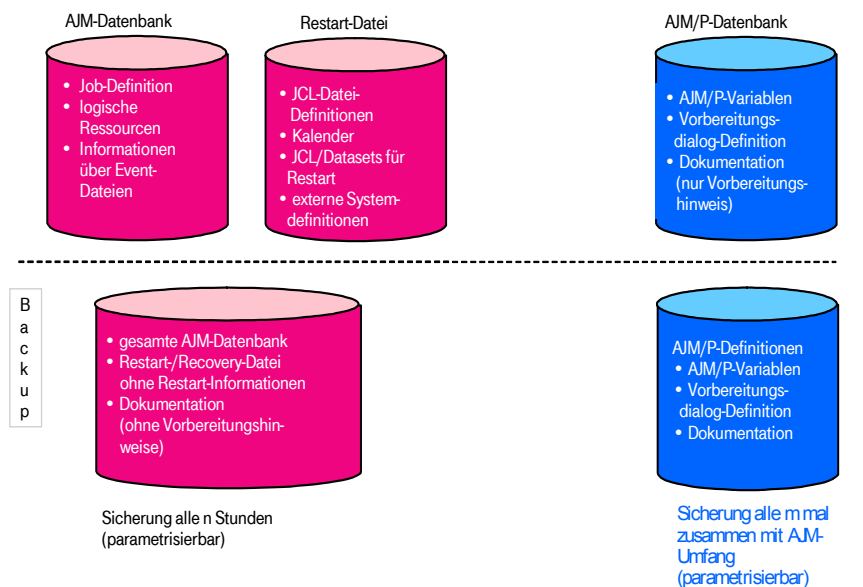
- Job and net definitions
- Logical resources
- JCL file definitions
- Calendars
- Information on event files
- Net / job documentation
- AJM/P dialog definitions
- AJM/P variables
- AJM/P file definitions

Not backed up are:

- The JCL of the last job run
- The names of the files which were created during the last job run

The following diagram gives you an overview:

AJM / AJM/P: Datenbanken und Sicherungskonzept



The backup is written to a generation file. The parameterization is used to define how often this backup run takes place and how many backup copies are to be maintained. In addition, the information on how often a full backup is to take place is filed there (i.e. a backup which also includes the AJM/P data). A detailed description is found under "Parameterics" on page 41.

The first (full) backup is created half an hour after the startup of the global AJM system, the other backups take place at the interval parameterized. In addition, backups can be created directly using the operator interface (see also "Overview of the operator commands" on page 143).

The message "AJMDB13I", which contains the name of the backup file, appears in the SYSLOG after the successful creation of a backup copy. Furthermore, the first record of the backup file contains the date and time of its creation and the dataset name of the last full backup.

9.8 Database restoration

If the AJM or AJM/P database or the restart file is destroyed, then these may be restored. The restoration is done with the aid of the backup file.

The following sample job is found in the file "ISSDC.Vw.AJM.SAJMSAMP(AJMDBRLD)":

```
//AJMDBRLD JOB (acct),,CLASS=Q,MSGCLASS=V
//*
//*-----
//*
//*      SAMPLE JOB FOR RELOAD OF AJM COMPONENTS
//*
//*      - BACKUP:   DSNAME OF BACKUP GDG
//*      - DATABASE: NAME OF AJM DATABASE
//*      - RESTART:  NAME OF RESTART/RECOVERY FILE (CONTAINS CALENDAR-
//*                  AND JCL-DEF. ENTRIES)
//*      - PREP:     NAME OF AJM/P DATABASE
//*      - REPLACE:  SPECIFIES IF EXISTING ENTRIES OF THE RESTART/
//*                  RECOVERY FILE OR OF THE AJM/P DATABASE
//*                  SHOULD BE REPLACED OR NOT (YES/NO)
//*
//* NOTE: FOR A SELECTIVE RELOAD YOU ONLY HAVE TO SPECIFY THE
//*        REQUIRED COMPONENTS
//*-----
//*
//DBRELOAD EXEC PGM=AJMDBRLD
//SYSIN DD *
        BACKUP (???? .AJM.DATABASE.BACKUP.G????V00) ,
        DATABASE (???? .AJM.DATABASE) ,
        RESTART (???? .AJM.RESTART) ,
        PREP (???? .AJM.PREP.DATABASE) ,
        REPLACE (YES)
/*
```

The parameters in the SYSIN file have the following meanings:

BACKUP Enter the dataset name of the backup generation from which the database is to be restored.

DATABASE Dataset name of the database to be newly created

This database must have been allocated and initialized beforehand using the parameterization function (see "Parameterics" on page 41).
After a successful restore the message "AJMDB70I" appears.

RESTART Dataset name of the restart file

This database must have been allocated and initialized beforehand using the parameterization function (see "Parameterics" on page 41).
After a successful restore the message "AJMDB65I" appears.

PREP Dataset name of the AJM/P database

This database must have been allocated and initialized beforehand using the parameterization function (see "Parameterics" on page 41).
After a successful restore the message "AJMDB72I" appears.

REPLACE Replacing previously defined calendar / JCL file definitions

If "REPLACE(YES)" is entered, then calendars and JCL definitions which are in both the restart file and the backup file are overwritten with the information from the backup file. With "REPLACE(NO)" double definitions are ignored. Definitions which are only found on the backup file are used in the restart file in every case.

A detailed description of restarting the AJM system after the restoration of a database or the restart file is found under "Loss of the database" on page 78.

9.9 Utility "AJMDBMRG"

If two AJM systems are to be merged, then this can be done with the aid of the AJMDBMRG utility. The target system is an active AJM system, either a generation of a backup file or a AJM database be used as an input. In addition, this utility can be used for the selective restoration of partial quantities, e.g. nets which were accidentally deleted. All AJM and AJM/P definitions can be restored from the appropriate backup file.

In contrast to the EXTRACT / UPDATE command (see "Command / Batch / Program Interface" on page 93), the statistics data from the source file is also brought onto the target system with the AJMDBMRG utility. This ensures that, e.g. after the restoration of a net definition the restored weekly processing does not start immediately, but an appropriate interval is maintained.

The following sample job is found in the file "ISSDC.Vw.AJM.SAJMSAMP(AJMDBMRG)":

```
//AJMDBMRG JOB                                00010003
//*
//*-----
//*
//*   SAMPLE JOB FOR:
//*   - MERGE OF AJM DATABASE INFORMATION INTO ANOTHER SYSTEM
//*   - RESTORE OF DEFINITIONS FROM THE BACKUP DATASET
//*
//*   CHANGE THE FOLLOWING PARAMETERS ACCORDING TO YOUR
//*   INSTALLATION'S DEFINITIONS:
//*   - DATABASE NAME
//*     (MUST NOT BE IN USE)
//*   - NAME OF BACKUP GENERATION DATASET
//*     (MUTUALLY EXCLUSIVE WITH DATABASE NAME)
//*   - NAME OF THE TARGET AJM SYSTEM
//*     (MUST BE DEFINED IN THE 'AJMNames' MODULE AND BE ACTIVE)
//*   - TYPE (JOB, LRS, CAL, JCD, XSD, PRP, VAR, PCD, DOC, ALL)
//*     (TYPE OF DEFINITIONS TO BE MERGED / RESTORED)
//*   - GROUP / NET / JOB / LRS / CALENDAR / XSD / PACK / VAR
//*     (CAN BE ABBREVIATED BY '*')
//*   - LOCK = YES/NO
//*     (LOCK DEFINITIONS ON TARGET SYSTEM)
//*   - ADDJOB = YES/NO
//*     (COPY JOBS EVEN IF NET EXISTS ON TARGET SYSTEM)
//*
//*   DATABASE OR BACKUP MUST BE SPECIFIED.
//*   IF TARGET IS NOT SPECIFIED, THE FIRST ENTRY IS TAKEN FROM
//*   THE 'AJMNames' MODULE.
//*-----
//*
//*DBMERGE EXEC PGM=AJMDBMRG
//AJMPRINT DD SYSOUT=*
//SYSIN DD *
        DATABASE(???? .AJM.DATABASE) ,
        BACKUP(???? .AJM.DATABASE.BACKUP.G????V??) ,
        TARGET(AJM?????) ,
        TYPE (JOB, DOC) ,
        GROUP(???????? .????????) ,
        NET(????????) ,
        LOCK (YES) ,
        ADDJOB (NO)
/*
```

The parameters in the SYSIN file have the following meanings:

DATABASE Name of the database

The nets selected from this database are copied into the AJM given under TARGET. Either the parameter BACKUP or the parameter DATABASE must be entered.

Note: The database given here must not be occupied, i.e. the "source" AJM system may not be active.

BACKUP Name of the backup file

The nets selected from this backup generation file are copied into the AJM given under TARGET. Either the parameter BACKUP or the parameter DATABASE must be entered. Calendars, JCL definitions, documentation texts and AJM/P definitions may only be restored from a backup file.

TARGET Name of the target AJM system

The name of the AJM system in which the nets selected are to be copied is given here. The name of the AJM system must be defined in the "AJMNames" module (see AJM Installation Manual).

TYPE Type of definitions which are to be copied / restored

The following entries are possible:

JOB Net / job definitions
DOC Documentation texts for nets /jobs
LRS Logical resources
JCD JCL definitions
CAL Calendars
XSD External system definitions
PRP AJM/P Dialog definitions
VAR AJM/P variables
PCD AJM/P file definitions
ALL All definitions

GROUP Group name

Only definitions that belong to the given group are copied. The group name can be abbreviated with "**". If this parameter is missing, the group name is not checked.

NET Net name

Only definitions that belong to the given net are copied. The net name can be abbreviated with "**". If this parameter is missing, the net name is not checked.

JOB Job name

Only definitions that belong to the given job are copied. The job name can be abbreviated with "*". If this parameter is missing, the job name is not checked.

CAL Calendar name

Only the appropriate calendars are copied. The calendar name can be abbreviated with "*". If this parameter is missing, the calendar name is not checked.

LRS Name of the logical resource

Only the appropriate logical resources are copied. The name can be abbreviated with "*". If this parameter is missing, the name of the logical resource is not checked.

PACK Name des AJM/P packets

Only definitions that belong to the given packet are copied. The packet name can be abbreviated with "*". If this parameter is missing, the packet name is not checked.

VAR Name of the AJM/P variables

Only the appropriate AJM/P variables are copied. The name can be abbreviated with "*". If this parameter is missing, the name of the AJM/P variables are not checked.

XSD Name of the external system definition

Only the appropriate external system definitions are copied. The name can be abbreviated with "*". If this parameter is missing, the name of the external system definition is not checked.

LOCK Locking the copied job

If YES is entered here, then the copied jobs are locked in the target system. If NO is given, then these job are not locked , i.e. they can start immediately when the schedule conditions are fulfilled. If this parameter is missing, YES is assumed.

ADDJOB Adding jobs to an existing net

If YES is entered here, then jobs are taken into the target system if such a net already exists. If NO is entered, then these jobs are not taken over. If this parameter is missing, NO is assumed.

Note: Jobs which already exist in the target system are generally not taken over. If the job in question does not exist, but the net already exists, then you can use the ADDJOB parameter (see above) to control whether this job is taken over or not.

The user under whom the job runs must have UPDATE authorization for the nets to be copied (see "Data Security" on page 91). Otherwise the net in question will not be copied.

After a job has been taken over in the target system, an implicit "CLR" command is executed. Afterwards the copied job has the status "not active".

Additional information on the "CLR" command is found in the users handbook.

All actions are logged in the AJMPRINT file.

Example:

All nets NET01* from the group MANDANT.GROUP1 are to be taken over from the AJM system AJM1 in the AJM system AJM2. The input is the database of AJM1, "HLQ.AJM1.DATABASE". In order to ensure that no net starts unintentionally, the nets copied in the system AJM2 are locked. In addition, if such a net already exists in AJM2, this net will remain unchanged, i.e. no jobs from AJM1 will be mixed in the existing nets of AJM2.

The SYSIN file then looks as follows:

```
//SYSIN DD *
        DATABASE (HLQ.AJM1.DATABASE) ,
        TARGET (AJM2) ,
        GROUP (MANDANT.GROUP1) ,
        NET (NET01*) ,
        LOCK (YES) ,
        ADDJOB (NO)
/*
```

9.10 Utility "AJMDBCLN"

The data from AJM and AJM/P is distributed over several databases. Data which is related to each other, e.g. the definition of a job, the associated documentation, and the recorded JCL of the last job run, is stored in different databases. If a job definition is deleted, the associated documentation remains for the time being. This ensures that after accidentally deleting a job that the other components are ready to be used after restoration of the job definition using AJMDBMRG.

This method of proceeding means that a tool must be available with the aid of which the AJM system administrator can completely or selectively clean up those elements which no longer need to be stored.

The following elements can be either listed or deleted directly with the aid of the utility AJMDBCLN:

- "Old JCL", i.e. the recorded JCL of the last job run (when no associated job definition exists any more or the last run is more than a defined number of days old)
- The recorded names of the datasets which were created in the last job run (when no associated job definition exists any more or the last run is more than a defined number of days old)
- Job definitions for which the last start was more than a defined number of days ago
- AJM/P dialog definitions for which no net exists
- Documentation for which the associated job or net no longer exists

The following sample member is found in the "smphlq.AJM.SAJMSAMP":

```
//AJMDBCLN JOB
//*
//*-----
//*
//*      SAMPLE JOB FOR DATABASE CLEANUP UTILITY
//*
//*      DATASETS:
//*      - AJMCMD      COMMAND(S), WILL BE IGNORED IF PARM SPECIFIED IN
//*                   EXEC CARD
//*      - AJMPRINT    MESSAGES (ERROR / INFORMATION), OUTPUT FOR "LIST"
//*                   OPTION
//*      - AJMOUT      GENERATED OUTPUT FOR "GEN" OPTION
//*                   (LRECL IS 80)
//*
//*      COMMANDS:
//*      - ADDRESS     ADDRESS THE AJM SYSTEM SPECIFIED IN THE AJM()
//*                   PARAMETER
//*      - OLDJCL      PROCESS "OLD JCL" FOR WHICH NO JOB DEFINITION
//*                   EXISTS (NOJOB) OR WHICH HAS A CERTAIN AGE (AGE)
//*      - DSTRACK     PROCESS "LIST OF DATASETS OF LAST JOB RUN" FOR
//*                   WHICH NO JOB DEFINITION EXISTS (NOJOB) OR WHICH
//*                   HAS A CERTAIN AGE (AGE)
//*      - JOB         PROCESS JOBS THAT HAVE NOT BEEN STARTED FOR A
//*                   CERTAIN TIME (NORUN)
//*      - PREP        PROCESS AJM/P DIALOG DEFINITIONS FOR WHICH NO JOB
//*                   DEFINITION EXISTS (NOJOB)
//*      - DOC         PROCESS DOCUMENTATION TEXTS FOR WHICH NO JOB
//*                   DEFINITION EXISTS (NOJOB)
//*
//*      PARAMETERS:
//*      - AGE / NORUN SPECIFY NUMBER OF DAYS
//*      - NOJOB       CHECK, IF NO JOB DEFINITION EXISTS
//*      - LIST        GENERATES A LIST OF MATCHING ENTRIES (AJMPRINT)
//*      - DEL         DELETES ALL MATCHING ENTRIES
//*      - GEN         GENERATES TEXT IN AJMOUT DATASET
//*      - TEXT        TEXT FOR GEN OPTION
//*                   ANY TEXT PARTS CAN BE COMBINED WITH VARIABLES
//*                   %1  GROUP NAME
//*                   %2  NET  NAME
//*                   %3  JOB  NAME
//*                   (E.G.: TEXT('DEL GROUP(%1) NET(%2) JOB(%3)')
//*                   WILL GENERATE INPUT FOR AJMAICMD INTERFACE)
//*
//*      WITHIN THE AJMCMD DATASET COMMENT CARDS ARE INDICATED BY
//*      '*', TO INDICATE A CONTINUATION END THE CARD THAT WILL BE
//*      CONTINUED IN THE NEXT LINE WITH A '-'.
//*-----
//*
//AJMDBCLN EXEC PGM=AJMDBCLN,
// PARM='ADDRESS AJM(AJM????);OLDJCL NOJOB LIST'
//AJMPRINT DD SYSOUT=*
//AJMOUT DD SYSOUT=*
//*
//AJMCMDDIN EXEC PGM=AJMDBCLN
//AJMPRINT DD SYSOUT=*
//AJMOUT DD DISP=(NEW,CATLG),UNIT=SYSALLDA,SPACE=(TRK,(10,10)),
// DSN=????
//AJMCMD DD *
*
* ADDRESS COMMAND: ADDRESS THE AJM SYSTEM TO SEND THE COMMANDS TO
* (DEFAULT: FIRST SYSTEM IN THE AJM NAMES TABLE)
*
* ADDRESS AJM(AJM?????)
```

```

*          -----
*
* COMMANDS FOR CLEANUP OF AJM DATABASES
*
* OLDJCL  AGE (????)  NOJOB          LIST GEN DEL -
*                                     TEXT (PARM1 %1 PARM2 %2 PARM3 %3)
*
* DSTRACK AGE (????)  NOJOB          LIST GEN DEL -
*                                     TEXT (PARM1 %1 PARM2 %2 PARM3 %3)
*
* JOB     NORUN (????) GROUP (*)      LIST GEN DEL -
*                                     TEXT (PARM1 %1 PARM2 %2 PARM3 %3)
*
* PREP    NOJOB      GROUP (?????????.?????????) -
*                                     LIST GEN DEL TEXT (PARM1 %1 PARM2 %2 PARM3 %3)
*
* DOC     NOJOB      GROUP (?????????.?????????) -
*                                     LIST GEN DEL TEXT (PARM1 %1 PARM2 %2 PARM3 %3)
*

```

The commands and their parameters have the following meanings:

ADDRESS Selection of the AJM system

Parameter:

AJM Name of the target system

This name is filed in the AJMNames module and serves as the key to finding the associated parameter module. If no ADDRESS command is given, then all commands go the AJM system which is defined first in the AJMNames module.

OLDJCL List / delete recorded JCL of the last job run

Master authorization is required for this function (at least READ for the profile #GLOBAL in the general resource class \$AJM).

Parameter:

NOJOB A check is made to see if the associated job definition still exists

AGE A check is made whether the last start of the associated job was at least the given number of days ago

DEL Delete the elements found

LIST A list of the elements found is output in the AJMPRINT file

GEN A record is generated in the AJMOUT file for each element found

TEXT For the option "GEN" (see above) you can enter in the TEXT parameter how the record which is generated in the AJMOUT file is to be structured. The variables group (%1), net (%2), and job (%3) can be placed in any position.

Example:

TEXT('%MYREXX %1 %2 %3') builds up a file that creates REXX calls for an own REXX. This output file can then be executed in a second step.

DSTRACK List / delete recorded files of the last job run

Master authorization is required for this function (at least READ for the profile #GLOBAL in the general resource class \$AJM).

Parameter:

- NOJOB** A check is made to see if the associated job definition still exists
- AGE** A check is made whether the last start of the associated job was at least the given number of days ago
- DEL** Delete the elements found
- LIST** A list of the elements found is output in the AJMPRINT file
- GEN** A record is generated in the AJMOUT file for each element found
- TEXT** For the option "GEN" (see above) you can enter in the TEXT parameter how the record which is generated in the AJMOUT file is to be structured. The variables group (%1), net (%2), and job (%3) can be placed in any position.

Example:

TEXT('%MYREXX %1 %2 %3') builds up a file that creates REXX calls for an own REXX. This output file can then be executed in a second step.

JOB Check / delete job definitions

READ authorization (option LIST / GEN) or UPDATE or ALTER authorization (Option DEL) for the appropriate net is required for this function.

Parameter:

- NORUN** A check is made to see if the last start of the job is at least the given number of days ago
- DEL** Delete the elements found
- LIST** A list of the elements found is output in the AJMPRINT file
- GEN** A record is generated in the AJMOUT file for each element found
- TEXT** For the option "GEN" (see above) you can enter in the TEXT parameter how the record which is generated in the AJMOUT file is to be structured. The variables group (%1), net (%2), and job (%3) can be placed in any position.

Example:

TEXT('DEL GROUP(%1) NET(%2) JOB(%3)') builds up a file that creates REXX calls for an own REXX. This output file can then be executed in a second step.

PREP Check / delete AJM/P dialog definitions

READ authorization (option LIST / GEN) or UPDATE or ALTER authorization (Option DEL) for the appropriate net is required for this function.

Parameter:

- NOJOB** A check is made to see if the associated job definition still exists
- DEL** Delete the elements found
- LIST** A list of the elements found is output in the AJMPRINT file
- GEN** A record is generated in the AJMOUT file for each element found
- TEXT** For the option "GEN" (see above) you can enter in the TEXT parameter how the record which is generated in the AJMOUT file is to be structured. The variables group (%1) and net (%2) can be placed in any position.

Example:

TEXT('%MYREXX %1 %2') builds up a file that creates REXX calls for an own REXX. This output file can then be executed in a second step.

DOC Check / delete documentation

READ authorization (option LIST / GEN) or UPDATE or ALTER authorization (Option DEL) for the appropriate net is required for this function.

Parameter:

- NOJOB** A check is made to see if the associated job definition still exists
- DEL** Delete the elements found
- LIST** A list of the elements found is output in the AJMPRINT file
- GEN** A record is generated in the AJMOUT file for each element found
- TEXT** For the option "GEN" (see above) you can enter in the TEXT parameter how the record which is generated in the AJMOUT file is to be structured. The variables group (%1), net (%2) and job (%3) can be placed in any position. The value of "%3" is for net documentation "-".

Example:

TEXT('%MYREXX %1 %2 %3') builds up a file that creates REXX calls for an own REXX. This output file can then be executed in a second step.

9.11 Disaster strategies

In the following chapters the measures which may be necessary due to, for example, hardware problems are discussed.

Both the necessary preparations for such a situation and the method of proceeding when such a situation occurs are presented.

9.11.1 Loss of individual job definitions

If jobs or nets were accidentally deleted, you can restore them to the state they were in on the last backup file with the aid of the utility AJMDBMRG. Run information is also restored.

A detailed description of AJMDBMRG is found under "Utility "AJMDBMRG"" on page 70.

9.11.2 Loss of the database

After the loss of the AJM database, e.g. through failure of the disk in question, the database can be restored to the state it was when the backup file was made. How old the backup file is can be seen from the first record which contains the date and time of creation.

If only the database itself, but not the restart database was destroyed, then a restore run can be performed so that only the database is restored. This is done by not coding the parameters "RESTART" and "PREP".

The exact description of the restore run is found under "Database restoration" on page 68.

After the start of AJM with the restored database, the event files are evaluated. Since the database contains references to the events last evaluated, all events which occurred after the creation of the database can be "re-done". No scheduling takes place in this time.

After this process is completed, AJM is unlocked internally for the scheduler, afterwards jobs may be started again.

9.11.3 Loss of the restart database

After the loss of the restart database it can be restored to the state it was when the backup file was made. How old the backup file is can be seen from the first record which contains the date and time of creation.

If only the restart database itself, but not the database was destroyed, then a restore run can be performed so that only the restart database is restored. This is done by not coding the parameters "DATABASE" and "PREP".

The exact description of the restore run is found under "Database restoration" on page 68.

Note:

Only the calendar and the JCL file definitions are re-loaded. Information on the JCL of the last job run or the files created in this run is not saved. Therefore the following functions are **not** available after the restore for the jobs run previously:

- Automatic restart
- Manual restart with old JCL
- Manual restart with deletion of the files created
- Administration command "RES" (deletion of the files created during the last run)

9.11.4 Loss of an event file

After the loss of an event file it must be re-created and initialized using the parameterization function. A restoration of the event file on the basis e.g. of an HSM backup **may never** be performed since this would lead to problems in operation.

All AJM systems which access this event file must then be restarted.

Events which had already been read out of the event buffer of the ECSA but had not been included in the database, are lost.

9.11.5 Loss of the AJM/P database

After the loss of the AJM/P database it can also be restored to the state it was at the last backup.

The net and job documentation is restored from the given backup dataset. The restoration of the other data in the AJM/P database (dialog definitions, AJM/P variables, and AJM/P file definitions) is done on the basis of the data included in the last full backup. Which backup file contains this full backup is automatically determined from the first record of the **given** backup dataset. How old the backup file is can be seen from the first record which contains the date and time of creation.

If only the AJM/P database was destroyed, but not the other files then the restore run can be performed such that only the AJM/P database is restored. This is done by not coding the parameters "DATABASE" and "RESTART".

The exact description of the restore run is found under "Database restoration" on page 68.

Interfaces

10 User exits in AJM

The following user exits are available for computer center-specific adjustment / expansion:

AJMJSUX1 Submit pre-processing

To process the JCL before the submit (variable parts are not replaced yet)

AJMJSUX2 Submit pre-processing

To process the JCL before the submit (variable parts are already replaced)

AJMLGUX1 Logging

To analyze log records

AJMADUX1 Interface to the print management system

To transfer a print file to the installed print management system

AJMADUX2 Interface to the SYSOUT management system

To query information from the installed SYSOUT management system

AJMADUX3 Interface to the job log system (e.g. SDSF / JVS)

To query information from the installed job log system

10.1 AJMJSUX1: Submit pre-processing (1)

This exit is used to process the JCL before the submit.

Variable parts in the JCL (%TIME, %DATE, %JV, etc.) have not yet been replaced at this point in time.

This exit receives each card of the submitted JCL for checking and / or modification. It is called again for each card.

The exit uses the return code to inform the system whether

- The current card should be accepted (RC=0)
- The current card should be skipped (RC=4)
- Another card should be inserted instead of the current card (RC=8)

If a card is to be inserted, then Register 1 must point to an 80 character area which contains this card. After insertion the exit is called again, Register 1 then points to the card just inserted.

The exit must be linked to be re-enterable. It is called in AMODE 31.

Register 1 points to the current card when called, Register 0 points to a work area which is used jointly with the exit AJMJSUX2 (see below). Before the first call (i.e. before the first card), this area is initialized to X'00'. Afterwards information can be temporarily saved here which can also be called up over several jobs.

An example of the coding is found in the SAMPLIB member "AJMJSUX1".

10.2 AJMSUX2: Submit pre-processing (2)

This exit is used to process the JCL before the submit.

Variable parts in the JCL (%TIME, %DATE, %JV, etc.) have already been replaced at this point in time.

Format and call are like for "AJMSUX1" (see above).

An example of the coding is found in the SAMPLIB member "AJMSUX2".

10.3 AJMLGUX1: Logging

Log records can be analyzed using this exit.

It is always called when a log record is generated by the AJM system. Since this sometimes occurs in performance critical situations, (e.g. submit phase), I/O operations, for example, should be avoided in the exit.

The default exit supplied contains no functions.

The exit must be linked to be re-enterable. It is called in AMODE 31.

Register 1 points to the current log record whose structure and contents are described in the macro AJMALOGR. An example is coded in the SAMPLIB member "AJMLGUX1".

10.4 AJMADUX1: Interface to the print management system

This exit is called from the administration after a "PRT" command. The print file created earlier can now be passed on to any print management system.

The default exit supplied passes the file to PRMS from T-Systems ITS GmbH.

The exit must be linked to be re-enterable. It is called in AMODE 31.

Register 1 points to a 44 character field when called which contains the dataset name. An example is coded in the SAMPLIB member "AJMADUX1".

After being called the contents are interpreted by Register 15. If this register equals zero, then a successful completion is assumed. Otherwise an error message is displayed.

10.5 AJMADUX2: Interface to the SYSOUT management system

This exit is called from the administration after an "OUT" command. The selection by the user can now be passed to any SYSOUT management system.

The default exit supplied passes the file to AJM/S from T-Systems ITS GmbH.

The exit must be linked to be re-enterable. It is called in AMODE 31.

Register 1 shows a 33 character field when called which contains the group name (17 places), the net name (8 places), and the job name (8 places). An example is coded in the SAMPLIB member "AJMADUX2".

After being called the contents are interpreted by Register 15. If this register equals zero, then a successful completion is assumed. Otherwise an error message is displayed.

10.6 AJMADUX3: Interface to the job log system

This exit is called from the administration after a "JLG" command. The selection by the user can now be passed to any job log system.

The default exit supplied passes the file to SDSF (in a JES2 environment) or to JVS (in a JES3 environment).

The exit must be linked to be re-enterable. It is called in AMODE 31.

Register 1 shows a 33 character field when called which contains the group name (17 places), the net name (8 places), and the job name (8 places). An example is coded in the SAMPLIB member "AJMADUX3".

After being called the contents are interpreted by Register 15. If this register equals zero, then a successful completion is assumed. Otherwise an error message is displayed.

11 System exits

AJM uses the following SMF exits to detect events (job start / job end, cataloging datasets, etc.):

IEFUJV Module AJMUJV

IEFUJI Module AJMUJI

IEFU83 Module AJMU83

IEFU84 Module AJMU84

The linking for this module is undertaken with the product SEM from T-Systems ITS GmbH.

11.1 SEM (System Extensions and Modifications)

The product SEM (System Extensions and Modifications) from T-Systems ITS GmbH offers various functions, for example

- Job plausibility checks (e.g. check of the job name or account entries)
- Job control (e.g. controlling the region size or the maximal execution time)
- Exit routing (for simple implementation of exits)
- Variable and filter functions
- Message processing (automation)
- Making control block structure available
- Console command interface (across computers)
- Authorization check for system commands
- Control of computer center procedures (e.g. SVC dump process, BDT / NJE)

An extensive description of the product SEM and its parameters is found in the SEM handbook.

The following functions are used in AJM:

- Exit routing (for SMF exits)
- Variable function to determine the name of the job variable dataset
- Variable and filter functions for DS triggers

The following definitions must be completed in the SEM parameter file (the definitions below are found in the SAMPLIB member AJMSEM):

```

*****
* SEM PARAMETER DEFINITIONS FOR AJM *
*****
*
*-----*
* SMF EXITS *
*-----*
*
DEFINE EXIT NAME(IEFUJV), * JOB VALIDATION ROUTER-EXIT
                LMOD(AJMUJV) *
*
DEFINE EXIT NAME(IEFUJI), * JOB INITIATION ROUTER-EXIT
                LMOD(AJMUJI) *
*
DEFINE EXIT NAME(IEFU83), * SMF RECORD (SVC-ENTRY) ROUTER-EXIT
                LMOD(AJMU83), *
                PARM(YY) * WRITE RECORD TYPE 15 / 61/66
*
DEFINE EXIT NAME(IEFU84), * SMF RECORD (BR-ENTRY) ROUTER-EXIT
                LMOD(AJMU84) *
*
*-----*
* VARIABLES NEEDED FOR JOBVAR FUNCTION *
*-----*
*
DEFINE VARIABLES ID(JOBVAR),
                 NAME(JOBVAR),
                 VARS(JVARDSN(SAZ1.AJS.JOBVAR))
*
*-----*
* VARIABLES FOR AJM DS-TRIGGER CONTROL *
*-----*
*
DEFINE VARIABLES ID(AJMVARS),
                 NAME(AJMVARS),
                 VARS(,
                     DSTEXCLD(FLTEXCLD),
                     DSTCATLG(FLTCATLG),
                     DSTCLSNW(FLTCLSNW),
                     DSTCLSUP(FLTCLSUP))
*
*-----*
* FILTER-LIST FOR AJM DATASET-TRIGGERS *
*-----*
*
* EXCLUSION LIST: JOBS THAT WILL NOT PRODUCE A TRIGGER
*
DEFINE FILTER-LIST ID(AJM_DS_TRIGGER_EXCLD),
                  NAME(FLTEXCLD),
                  STRING(HSM*,
                        DEFRAG*)
*
* THE FOLLOWING DATASETS WILL TRIGGER WHEN BEING CATALOGUED
* (DISP=CATLG, RENAME ETC.)
*
DEFINE FILTER-LIST ID(AJM_DS_TRIGGER_CATLG),
                  NAME(FLTCATLG),
                  STRING(DSN1*)

```



```
*
*   THE FOLLOWING DATASETS WILL TRIGGER WHEN BEING CLOSED
*   (OPENED FOR UPDATE, NEW (NON-VSAM) DATASETS ONLY)
*
DEFINE FILTER-LIST ID(AJM_DS_TRIGGER_CLSNW),
                   NAME (FLTCLSNW),
                   STRING (DSN2*)
*
*   THE FOLLOWING DATASETS WILL TRIGGER WHEN BEING CLOSED
*   (OPENED FOR UPDATE, EXISTING (NON-VSAM) DATASETS ONLY)
*
DEFINE FILTER-LIST ID(AJM_DS_TRIGGER_CLSUP),
                   NAME (FLTCLSUP),
                   STRING (DSN3*)
```

The AJM... load module given is executed with the **DEFINE EXIT** entries when the corresponding exit is called.

Additional parameters are passed to the module AJMU83. They have the following meanings:

1st parameter Controls whether the SMF record type 15 is written.

Y The SMF records of type 15 are written.

N The SMF records of type 15 are not written.

2nd parameter Controls whether the SMF record types 61 and 66 are written.

Y The SMF records of type 61 and 66 are written.

N The SMF records of type 61 and 66 are not written.

Please remember that the parmlib in the SMFPRMxx member must not exclude record types 15, 30, 61, and 66.

If the records are suppressed by this member, then the exit receives no information.

DEFINE VARIABLES is used for two functions:

- The first set of variables supplies the name of the job variable dataset for the job variable function. The access to the job variables interfaces is implemented according to the information in this entry.
- With the second set of variables the names of the filters are determined which are used with the dataset trigger function. A "DEFINE FILTERLIST" entry must exist for every filter function entered.

DEFINE FILTERLIST supplies the parameters for the dataset trigger function. Any number of more or less generic entries can be made in each filter list.

The names of the filter lists are entered using "DEFINE VARIABLES".

The individual filter lists have the following meanings:

- Filter for variable **FLTEXCLD**

All jobs whose names correspond to this filter criterion do **not** read out any dataset triggers.

- Filter for variable **FLTCATLG**

As soon as a dataset is cataloged which fulfills these filter criteria, the AJM is informed by an event.

- Filter for variable **FLTCLSNW**

As soon as a new (non-VSAM) dataset is closed which has been opened with UPDATE and which fulfills these filter criteria, the AJM is informed by an event.

- Filter for variable **FLTCLSUP**

As soon as an existing (non-VSAM) dataset is closed which has been opened with UPDATE and which fulfills the filter criteria, the AJM is informed by an event.

The file information should not be too generic for performance reasons.

12 Interfaces to automation tools

If an error occurs in a process, then there are a number of ways to react to this error within AJM. These restart and recovery activities are determined by the definition of the process and automatically executed when the error occurs.

However, situations can occur which were not thought out in advance or for which, for example, the error cannot be remedied by multiple automatic restarts. In these cases various messages are generated by AJM which may be taken up by any automation tool to alarm the emergency services of a computer center.

The following messages are sent:

- **AJMEH80I - AJMEHPRC: group/net/job/compcode STOP**

This message is output when a job ends with the completion code "compcode" which is either not defined in the termination control definition or is filled there with "STOP". Individual portions of the net may still be running when this message is output.

- **AJMEH81I - AJMEHPRC: group/net M-RESTART**

This message is output when a net is running on manual restart. When this message is output the net is out of operation, i.e. no node of this net is still active.

- **AJMEH82I - AJMEHPRC: group/net/job/compcode R-LIMIT**

This message is output when a job ends with a completion code "compcode" which according to the termination control definition must lead to a restart, but the restart limit for this job has already been reached.

- **AJMEH82I - AJMEHPRC: group/net/job/compcode R-FAIL**

This message is output when a job ends with a completion code "compcode" which according to the termination control definition must lead to a restart, but the restart failed. This may be the result of a faulty definition (e.g. because the time of restart cannot be determined) or it could be, for example, the files from the last job run could not be deleted. The exact definitions can be seen in the AJM log.

"group", "net", "job" and "compcode" each contain the current value for the affected job.

13 Data Security

One of the characteristics - especially important for service computer centers - of AJM is its client capability. This means that users from different clients (e.g. from different customers or from different departments within one customer) can work independently of each other without seeing or especially without being able to change data from another client.

This function is implemented on the basis of the security server (e.g. RACF from IBM, TopSecret from CA). Every time a resource in AJM (job / net definitions, calendar, JCL library assignments, etc.) is accessed, a check is made whether the user has authorization for the access. Only when the corresponding security check allows it, the desired access is executed and the requested data displayed. This applies both to status queries as well as to definition changes.

In order to make this protection transparent and not require much administration effort, the expression group exists in AJM. Each client is clearly assigned to a group; the different groups are separated from each other on the access side. Finer structuring of the access rights is possible within a group so that different responsibilities within a client can be taken into account.

Depending on the object to be protected in AJM, different protection mechanisms which are explained below are used.

13.1 Protection of Jobs / Nets / Started Tasks / External Systems

For started tasks, external systems, jobs and nets, the access to the definitions, status queries, and the use of commands in the administration are protected. This is done using profiles in the **general resource class \$AJM**.

Furthermore, a check is made whether the user ID given in the job card ("USER=" entry) may be used by this user. This prevents a client from being started via AJM which would run under the user ID of another client without authorization. This check is also made using the general resource class \$AJM.

AJM must have the authorization to start a job under the user ID which is given on the job card in order to be able to start a job. The security server uses the profiles in the **resource class SURROGAT** for this purpose.

For started tasks a check is made whether the user who last changed the AJM job definitions on the local system has the start authorization for the procedure given.

For external systems a check is made whether the user who last changed the AJM job definitions is authorized to initiate jobs on this system.

AJM requires suitable authorization for the MVS start command for the start of a started task ("S proc..."). This authorization is checked using the **resource class OPERCMDS** or using **SEM**.

The profiles for the access to the net and job definitions have the following structure:

group-hlq.group-slq.net

They can be de-centrally administered.

The following authorization levels are possible:

NONE Allows no access to definitions
READ Allows read access to definitions
UPDATE Allows changing definitions
CONTROL Allows execution of commands
ALTER Allows both commands and modifications

The profile for protection of the "USER=" entry in the job card has the following structure:

group-hlq.group-slq.#SUB.net.job

They are administered centrally by the computer center.

The following levels of authorization are possible

NONE Forbids the use of the corresponding user in the job card
READ Allows the use of the corresponding user in the job card

The profile for the protection of starts of a stated task is checked on the local computer in question. The user ID that last changed the definition of the started task is used for this. The profiles have the following structure:

group-hlq.group-slq.#STC.proc

"proc" is the name of the start procedure.

The following levels of authorization are possible

NONE Forbids the start of the started task
CONTROL Allows the start of the started task
ALTER Allows the start of the started task

Generally the authorization NONE is assumed when either

1. the user ID checked is not authorized via the access list or
2. no suitable profile is found.

The profile for the protection of job starts on external systems is checked against the user ID which last changed the associated job definition in AJM. The profiles have the following structure:

group-hlq.group-slq.#XSD.xsdname

"xsdname" is the name of the external system definition.

The following levels of authorization are possible

NONE	Forbids the start of jobs on the external system and access to external system definitions.
CONTROL	Allows the start of jobs on external systems and read access to external system definitions.
READ	Allows the start of jobs on external systems and read access to external system definitions.
ALTER	Allows the start of jobs on external systems and changing or re-creating external system definitions.

Generally the authorization NONE is assumed when either

1. the user ID checked is not authorized via the access list or
2. no suitable profile is found.

13.2 Protection of Other AJM Resources

For calendars, logical resources, JCL definitions, and job variables a check is made as to who may view, use, and/or change these resources. This is done using the general resource classes \$AJM or \$JOBVAR (for job variables).

For calendars and logical resources we differentiate between *global* and *local* use. Global resources are assigned to the group #GLOBAL, local resources to the group of the client in question.

Global calendars are created and administered by the computer center. Their names begin with \$ (e.g. \$STD). These calendars can be read by all users and used within job definitions. Local calendars are created and administered by the client himself. They may only be used within the group to which they belong. Another group may define a calendar with the same name; AJM can differentiate between these calendars.

Global logical resources are created and administered by the computer center. Their names begin with \$ (e.g. \$CTAPE). They are used to achieve serialization across clients. Local logical resources are assigned to a group like local calendars and are used to achieve serialization within clients.

The PROFILE for the access to the job variables must be administered centrally by the computer center, all other profiles can be administered centrally or de-centrally.

Profiles for AJM resources are administered using the **general resource class \$AJM**. They have the following structure:

For Calendars:

group-hlq.group-sql.#CAL.calendar

For logical resources:

group-hlq.group-sql.#LRS.lrs-hlq.lrs-sql

For JCL definitions

group-hlq.group-slq.#JCD

The following authorization levels are possible for all these profiles:

- NONE** Allows no access to definitions
- READ** Allows only read access to definitions
Additionally for logical resources: the use of this resource in a job definition
- UPDATE** Allows the change of definitions
- CONTROL** Allows the execution of commands
- ALTER** Allows both commands and modifications

Profiles for the security of job variables are administered in the **general resource class \$JOBVAR** and have the following structure:

jobvar-hlq.jobvar-slq

The following authorization levels are possible for these profiles:

- NONE** Allows no access to job variables
- READ** Allows only read access to job variables
- UPDATE** Allows the change of job variables
- ALTER** Allows the creation, deletion, and change of job variables

Generally the authorization NONE is assumed when either

1. the user ID checked is not authorized via the access list or
2. no suitable profile is found.

13.3 Protection of External Triggers

If an application wants to set an external trigger the following profile is checked:

#GLOBAL.#XTR.type.subtype

I trigger in

The following authorization levels are possible for these profiles:

- NONE** Forbids triggering an external trigger
- UPDATE** Allows triggering an external trigger
- ALTER** Allows triggering an external trigger

13.4 Protection of JCL Libraries

The JCL which AJM uses to start a job is usually located in libraries which belong to the client in question. The access security is performed here, like for other files, using the DATASET profile.

So that AJM can use the JCL from this file, it must be given the appropriate READ authorization.

13.5 Protection of Preparation Dialogs and Packages

For AJM/P preparation dialogs and packages a check is made as to who may view, change or delete them. This is done using profiles in the **general resource class \$AJM** which have the following structure:

For preparation dialogs:

group-hlq.group-slq.#PRP.net

For packages:

group-hlq.group-slq.#PAK.package

The following authorization levels are possible:

NONE	Allows no access to definitions
READ	Allows only read access to definitions
UPDATE	Allows the change of definitions
CONTROL	Allows the execution of preparation dialogs /packages
ALTER	Allows both modifications and execution of the preparation

13.6 Protection of AJM/P variables

For AJM/P variables a check is made who may view and change them. This is also done using profiles in the general resource class \$AJM which have the following structure:

For global variables:

#GLOBAL.#VRG

For group variables:

group-hlq.group-slq.#VRG

For package variables:

group-hlq.group-sql.#VRP.package

For net variables:

group-hlq.group-sql.#VRN.net

The following authorization levels are possible:

NONE Allows no access to definitions

READ Allows only read access to definitions

UPDATE Allows the change of definitions

ALTER Allows modifications

Please remember that during a preparation dialog **no** check of the access authorization takes place. The user is in a secure environment at this point, i.e. he is within a dialog that was thought out ahead of time by the dialog designer. He may only access those variables which the designer of the dialog has made available to him. An explicit check is therefore unnecessary here.

13.7 Protection of AJM/P file definitions

For AJM/P file definitions a check is made who may view and change them. This is also done using profiles in the general resource class \$AJM which have the following structure:

group-hlq.group-sql.#PCD

The following authorization levels are possible:

NONE Allows no access to definitions

READ Allows only read access to definitions

UPDATE Allows the change of definitions

ALTER Allows modifications

Please remember that during a preparation dialog these authorizations are **not** checked (see above). It is enough when the preparer has the necessary authorization in the resource class DATASET.

13.8 Summary

The authorization check is carried out by the AJM system which is to execute the administration command. If a user has no user ID on the computer where AJM is running, then this command is rejected.

Generally this procedure ensures that the awarding of authorization takes place where the "owner" of the resource to be protected is located.

The following authorizations are checked for the individual functions of the AJM administration interface:

Function	Authorization	Entity checked	Comments
<i>Read job definitions</i>	READ	group.net	
<i>Change job definitions</i>	UPDATE or ALTER	group.net	
<i>Create / Delete job definitions</i>	UPDATE or ALTER	group.net	
<i>The following may be in the job card as USER=</i>	READ	group.#SUB.net.job	see comment 1
<i>Read JCL of the last job run</i>	READ	group.#JCD	
<i>Start, block, clear, etc. jobs</i>	CONTROL or ALTER	group.net	
<i>Start started tasks</i>	CONTROL or ALTER	#GLOBAL.#STC.proc	see comment 2
<i>Start job on an external system</i>	READ or ALTER	group.#XSD.xsdname	see comment 3
<i>Read external system definition</i>	READ or CONTROL	group.#XSD.xsdname	
<i>Create / Delete external system definition</i>	UPDATE or ALTER	group.#XSD.xsdname	
<i>Change external system definition</i>	UPDATE or ALTER	group.#XSD.xsdname	
<i>Use logical resources</i>	READ	group.#LRS.lrsname	
<i>Change logical resources</i>	UPDATE or ALTER	group.#LRS.lrsname	
<i>Create / Delete logical resources</i>	UPDATE or ALTER	group.#LRS.lrsname	
<i>Display current user of a logical resource</i>	CONTROL or ALTER	group.#LRS.lrsname	
<i>Read JCL definition</i>	READ	group.#JCD	
<i>Change JCL definition</i>	UPDATE or ALTER	group.#JCD	
<i>Create / Delete JCL definition</i>	UPDATE or ALTER	group.#JCD	
<i>Lock / Unlock JCL definition</i>	CONTROL or ALTER	group.#JCD	
<i>Read calendar</i>	READ	group.#CAL.calendar	
<i>Change calendar</i>	UPDATE or ALTER	group.#CAL.calendar	
<i>Create / Delete calendar</i>	UPDATE or ALTER	group.#CAL.calendar	
<i>Read out external trigger</i>	UPDATE or ALTER	#GLOBAL.#XTR.type.subtype	
<i>View logging data for a</i>	READ	group.net	

<i>net</i>			
<i>"Master" start menu (see "Selection menu" on page 31)</i>	READ	#GLOBAL	Check is performed within the administration
<i>Send AJM control commands from the administration (e.g. start / stop scheduler, traces, ...)</i>	CONTROL or ALTER	#GLOBAL	
<i>AJM/P: Read definition of preparation dialogs</i>	READ	group.#PRP.net	
<i>AJM/P: Create / Delete preparation dialogs</i>	UPDATE or ALTER	group.#PRP.net	
<i>AJM/P: Change preparation dialogs</i>	UPDATE or ALTER	group.#PRP.net	
<i>AJM/P: Execute preparation dialogs</i>	CONTROL or ALTER	group.#PRP.packet	
<i>AJM/P: Change validity of preparation dialogs</i>	CONTROL or ALTER	group.#PRP.net	
<i>AJM/P: Read definition of packets</i>	READ	group.#PAK.packet	
<i>AJM/P: Create / Delete packets</i>	UPDATE or ALTER	group.#PAK.packet	
<i>AJM/P: Change packets</i>	UPDATE or ALTER	group.#PAK.packet	
<i>AJM/P: Prepare packets</i>	CONTROL or ALTER	group.#PAK.packet	
<i>AJM/P: Read global variables</i>	READ	#GLOBAL.#VRG	see comment 4
<i>AJM/P: Create / Delete global variables</i>	UPDATE or ALTER	#GLOBAL.#VRG	
<i>AJM/P: Change global variables</i>	UPDATE or ALTER	#GLOBAL.#VRG	see comment 4
<i>AJM/P: Read group variables</i>	READ	group.#VRG	see comment 4
<i>AJM/P: Create / Delete group variables</i>	UPDATE or ALTER	group.#VRG	
<i>AJM/P: Change group variables</i>	UPDATE or ALTER	group.#VRG	see comment 4
<i>AJM/P: Read net variables</i>	READ	group.#VRN.net	see comment 4
<i>AJM/P: Create / Delete net variables</i>	UPDATE or ALTER	group.#VRN.net	
<i>AJM/P: Change net variables</i>	UPDATE or ALTER	group.#VRN.net	see comment 4
<i>AJM/P: Read packet variables</i>	READ	group.#VRP.packet	see comment 4
<i>AJM/P: Create / Delete packet variables</i>	UPDATE or ALTER	group.#VRP.packet	
<i>AJM/P: Change packet</i>	UPDATE or	group.#VRP.packet	see comment 4

<i>variables</i>	ALTER		
<i>AJM/P: Read file definitions</i>	READ	group.#PCD	see comment 4
<i>AJM/P: Change file definitions</i>	UPDATE or ALTER	group.#PCD	

Comment 1: The check of the USER parameter in the job card is done using the profile "group.#SUB.net.job" for which the user ID given there must have at least READ authorization.

These profiles may **not** be administered decentrally since otherwise a user from a client could enter the user ID of another client and then start jobs using AJM that run under his user ID. All other profiles can be administered decentrally (with the option GENERICOWNER).

Comment 2: When starting a started task (STC) a check is made on the corresponding local computer whether authorization for the start exists. The user ID which last changed the STC definition in the AJM is used for this check.

Comment 3: When starting a job on an external system a check is made on the corresponding local computer whether authorization for the start exists. The user ID which last changed the job definition in the AJM is used for this check.

Comment 4: During a preparation dialog the user is in a "secure environment", i.e. within a process thought out by the dialog designer. Therefore, for example, there is **no** check here whether the user may access the AJM/P file definition or if he may change an AJM/P variable.

In other words: The user may access elements which he may not normally have access to. In detail this means:

- He can change the contents of the variables belonging to the current packet without having authorization for the profile *group.#VRP.packet*.
- He can change the contents of the variables belonging to the current net without having authorization for the profile *group.#VRN.net*.
- He does not need authorization for *group.#PCD* to use AJM/P link names.
- He does not need authorization for *group.net* to indirectly address files (using the AJM job definition and the AJM link names).

Note: If it is not clear which profile is checked in a specific case, then a trace function can be activated which writes a corresponding message into the AJM log for each check. For more information on this, see "Trace options in the AJM system" on page 135.

14 Command / Batch / Program Interface

14.1 Introduction

With the aid of this interface it is possible to transfer *commanddefinitions* from the following environments to AJM:

- From a batch job
- From a CLIST or REXX or from any place under TSO
- From a program.

The authorization checks for the access generally take place on the computer on which the addressed AJM system runs. Accesses on job variables are only possible within the same complex, the authorization check takes place on the computer on which the job runs which performs the access.

The following table shows what each call looks like:

Call from	Program	Parameters via	Example
<i>Batch job</i>	AJMAICMD	Parm entry or AJMCMD file	see SAMPLIB member AJMBATCH (SYS4.SAMPLIB)
<i>TSO / ISPF / CLIST / REXX</i>	AJMCMD	Command parameter	TSO AJMCMD LOCK GROUP(...) NET(...) JOB(...)
<i>Program</i>	AJMCMD	Program area (2 byte length field (length of the command string) followed by commands (separated by a semi-colon))	... 01 CMD. 02 CMDLEN PIC 9(2) COMP. 02 CMDVAL PIC X(250). ... CALL "AJMCMD" USING CMD. ...

14.2 Return codes

Error conditions are reported back as return codes. In addition, messages are output in AJMPRINT (provided they are not suppressed via PROFILE MSGLEVEL(...)).

If a return code of 4 or 8 occurs (error in the execution of a command), then the following commands are only executed when this is explicitly desired using "PROFILE ONERROR(CONT)". If this was not set or a higher return code occurred (e.g. syntax error), then the execution breaks off at the faulty position.

The following return codes are possible:

- 0 All commands from the AJMCMD file or the PARM entry were executed successfully
- 4 Warning when the command is executed
(e.g. EXTRACT data incomplete)
- 8 Error on execution of the command
(e.g. Job not found, lack of authorization, etc.)
- 12 Faulty command
(e.g. Command not recognized, incorrect parameters, etc..)
- 16 Command too long
(a command from the AJMCMD file is longer than 1024 bytes)
- 20 File end reached before the command ended
(a line in the AJMCMD file ended with '-' (continuation), but no continued line followed)

14.3 DD names

The following files are used for the batch / program interface:

DD name	Use	RECFM	LRECL
<i>AJMPRINT</i>	Log file (messages)	VBA	137
<i>AJMCMD</i>	Commands (only columns 1-72 are interpreted) If a PARM entry is specified on call, the file AJMCMD is ignored	FB	80
<i>AJMIN</i>	Input file for UPDATE commands (Default, DD name can be overwritten by INDD entry)	FB	80
<i>AJMOUT</i>	Output file for EXTRACT, LOGREAD, and LOGLIST commands (Default, DD name can be overwritten by OUTDD)	FB	80 133

14.4 Examples

A job is locked in AJM with the following JCL:

```
//jobname JOB ...
//TEST EXEC PGM=AJMAICMD,
// PARM='LOCK GROUP(group) NET(net) JOB(job)'
//AJMPRINT DD SYSOUT=*
//
```


Several commands can be sent using the AJMCMD file:

```
//jobname JOB ...  
//TEST EXEC PGM=AJMAICMD  
//AJMPRINT DD SYSOUT=*  
//AJMCMD DD *  
    PROFILE LANGUAGE(GE)  
    UNLOCK GROUP(group) NET(net) JOB(job)  
    START GROUP(group) NET(net) JOB(job)  
//
```

The call looks as follows in a CLIST:

```
PROC 0  
  
    ALLOC DD(AJMPRINT) DS(*)  
    AJMCMD LOCK GROUP(group) NET(net) JOB(job)  
  
    ...
```

The call looks as follows in a REXX:

```
/* REXX */  
  
address TSO  
"ALLOC DD(AJMPRINT) DS(*) "  
"AJMCMD LOCK GROUP(group) NET(net) JOB(job) "  
  
    ...
```

14.5 Syntax

The following rules apply to syntax:

Commands via	Rules
<i>AJMCMD</i>	<ol style="list-style-type: none">1. Every command begins with a command keyword (e.g. LOCK) which is followed by the corresponding parameters (e.g. NET(net)).2. If a command does not fit onto a line, then this line must be ended with a blank followed by a hyphen ('-'). The continuation must be in the next line that is not a comment line.3. Parameters cannot be continued over several lines.

	<ol style="list-style-type: none"> 4. If a line does not end with a continuation sign it is interpreted as a new command. 5. Comment lines begin with a star (*). They may be placed in any position.
<i>PARM entry or Program area</i>	<ol style="list-style-type: none"> 1. Every command begins with a keyword (e.g. LOCK) which is followed by the appropriate parameters (e.g. NET(net)). 2. Commands are separated with a semi-colon (;).

14.6 Commands and their parameters

The following table gives an overview of the commands which are possible using the interface.

Command	Function	GROUP	NET	JOB	other parameters
<i>ADDRESS</i>	Select AJM system				AJM(ajmname)
<i>PROFILE</i>	Set profile values				LANGUAGE(language) MSGLEVEL(msglvl) LOCKNEW(YES/NO) BUFFERS(number) ONERROR(TERM/CONT)
<i>AJMSTAT</i>	Query AJM status				
<i>START</i>	Start job	M	M	M	
<i>RUN</i>	Start job	M	M	M	
<i>EXECUTE</i>	Start job	M	M	M	
<i>LOCK</i>	Lock net/job	M	M	O/G	
<i>UNLOCK</i>	Unlock net/job	M	M	O/G	
<i>DELETE</i>	Delete net/job	M	M	O/G	
<i>CLEAR</i>	Manual net/job clear	M	M	O/G	
<i>BYPASS</i>	Bypass job on next run	M	M	M	
<i>BYPRUN</i>	Bypass with run	M	M	M	
<i>RESET</i>	Delete files of the last run	M	M	M	
<i>UPDATE</i>	Add / change definition				INDD

<i>EXTRACT</i>	Save definition in file	M/G	M/G	O/G	OUTDD
<i>PLSCHECK</i>	Plausibility check	M/G	M/G		
<i>STATUS</i>	Query status of a job	M	M	M	
<i>LRSUPD</i>	Set total counter of a logical resource	M			LRS(Irsname) TOTAL(number)
<i>LRSWHO</i>	Display current "user" of a logical resource	M			LRS(Irsname)
<i>JCDLOCK</i>	Lock link name	M			LINKNAME(link name)
<i>JCDUNL</i>	Unlock link name	M			LINKNAME(link name)
<i>JVRSET</i>	Define / set job variable name				NAME(jvrname) VALUE(value) OFFSET(offset) LENGTH(length) LIFETIME(validity)
<i>JVRDEL</i>	Delete job variable				NAME(jvrname)
<i>XTRSET</i>	Red out external trigger				TYPE(type) SUBTYPE(subtype) LEN01(length) VAL01(value) LEN02(length) VAL02(value)
<i>NDOCUPD</i>	Add / change net documentation				INDD
<i>JDOCUPD</i>	Add / change job documentation				INDD
<i>NDOCEXT</i>	Save net documentation in a file	M/G	M/G	O/G	OUTDD
<i>JDOCEXT</i>	Save job documentation in a file	M/G	M/G	O/G	OUTDD
<i>LOGREAD</i>	Output AJM log in a file	O/G	O/G	O/G	OUTDD LIMIT FORMAT FROMDATE FROMTIME TODATE TOTIME LAST LRS VAR EXEC ADMI
<i>LOGLIST</i>	List available AJM log files				OUTDD

M Mandatory parameter

O Optional parameter

G Generic input possible

14.6.1 The parameters GROUP / NET / JOB

In the table above it indicates whether the command in question supports the parameters GROUP / NET / JOB, whether these are mandatory parameters (M) and whether they may be abbreviated (G).

These parameters have the following meanings:

Parameter	Meaning
<i>GROUP</i>	<p>Name of the group which the command in question refers to</p> <p>The group name consists of two expression each of which is up to 8 characters long which are separated by ".".</p>
NET	<p>Name of the net which the command in question refers to.</p> <p>The net name is a maximum of 8 places.</p>
JOB	<p>Name of the job which the command in question refers to.</p> <p>The job name is a maximum of 8 places.</p>

14.6.2 The ADDRESS command

The ADDRESS command is used to "address" the AJM system which is now to execute the following commands.

Note: The ADDRESS command does not work on commands which refer to job variables or to external triggers.

Parameter	Meaning
<i>AJM</i>	<p>Name of the AJM system</p> <p>This name is kept in the AJMNames module and serves as the key to finding the associated parameter module. If no ADDRESS command is given, then all commands go to the AJM system which is defined first in the AJMNames module.</p>

14.6.3 The PROFILE command

The PROFILE command is used to set the run parameters for the following commands. It may be used as often as you like, e.g. to set a sequence of commands ONERROR(CONT) and afterwards to change to ONERROR(TERM).

Parameter	Meaning										
<i>LANGUAGE</i>	<p>Language in which the messages are output and definitions are to be implemented.</p> <p>All subsequent messages are output in this language. For the UPDATE command the definitions are interpreted with the language given, the EXTRACT command supplies the data in the language given.</p> <p>Possible inputs are:</p> <table border="1"> <tr> <td>EN</td> <td>English (Default)</td> </tr> <tr> <td>GE</td> <td>German</td> </tr> </table>	EN	English (Default)	GE	German						
EN	English (Default)										
GE	German										
<i>MSGLEVEL</i>	<p>Selection of the messages which are to be logged in the AJMPRINT file</p> <p>Possible inputs are:</p> <table border="1"> <tr> <td>INFO</td> <td>Error messages, warnings, and information messages (default)</td> </tr> <tr> <td>WARNING</td> <td>Error messages and warnings</td> </tr> <tr> <td>ERROR</td> <td>Error messages</td> </tr> <tr> <td>NONE</td> <td>No messages</td> </tr> <tr> <td>TRACE</td> <td>Like INFO, but additional messages</td> </tr> </table>	INFO	Error messages, warnings, and information messages (default)	WARNING	Error messages and warnings	ERROR	Error messages	NONE	No messages	TRACE	Like INFO, but additional messages
INFO	Error messages, warnings, and information messages (default)										
WARNING	Error messages and warnings										
ERROR	Error messages										
NONE	No messages										
TRACE	Like INFO, but additional messages										
<i>ONERROR</i>	<p>Reaction on errors</p> <p>If a return code > 8 occurs, processing is cancelled. For return codes 4 and 8 (Error executing a command) the following may be defined:</p> <table border="1"> <tr> <td>TERM</td> <td>The processing is cancelled.</td> </tr> <tr> <td>CONT</td> <td>The processing is continued with the next command. The job ends with the highest return code that occurred.</td> </tr> </table>	TERM	The processing is cancelled.	CONT	The processing is continued with the next command. The job ends with the highest return code that occurred.						
TERM	The processing is cancelled.										
CONT	The processing is continued with the next command. The job ends with the highest return code that occurred.										
<i>LOCKNEW</i>	<p>Locking new definitions in the AJM database</p> <p>Possible inputs are:</p> <table border="1"> <tr> <td>YES</td> <td>New definitions are locked in the AJM database (default), they must be unlocked using an UNLOCK command (online or via the interface) before new jobs can be started by AJM.</td> </tr> <tr> <td>NO</td> <td>New definitions are not locked in the AJM database, they may be selected immediately by the scheduler.</td> </tr> </table>	YES	New definitions are locked in the AJM database (default), they must be unlocked using an UNLOCK command (online or via the interface) before new jobs can be started by AJM.	NO	New definitions are not locked in the AJM database, they may be selected immediately by the scheduler.						
YES	New definitions are locked in the AJM database (default), they must be unlocked using an UNLOCK command (online or via the interface) before new jobs can be started by AJM.										
NO	New definitions are not locked in the AJM database, they may be selected immediately by the scheduler.										

<i>BUFFERS</i>	Maximal number of response buffers of the AJM system
	The default response of the AJM system, for example to an EXTRACT command, is to send a 32 KB buffer. If a great deal of data is extracted, then this buffer may not be large enough. The BUFFERS command can be used to specify the maximum number of buffers per command. A value between 1 and 64 is permitted.

14.6.4 The LRSUPD command

The total counter of a logical resource can be set using the LRSUPD command.

Parameter	Meaning
<i>GROUP</i>	Group to which the logical resource belongs
<i>LRS</i>	Name of the logical resource
<i>TOTAL</i>	New value of the total counter of the logical resource
	This value must be numeric and may include up to 4 places.

14.6.5 The LRSWHO command

The LRSWHO command can be used to query who is occupying a logical resource.

The list of users is output on the AJMPRINT file.

Parameter	Meaning
<i>GROUP</i>	Group to which the logical resource belongs
<i>LRS</i>	Name of the logical resource

14.6.6 The JCDLOCK command

A link name can be locked for AJM with the JCDLOCK command. All files assigned are then released and may then be moved or compressed.

Parameter	Meaning
<i>GROUP</i>	Group to which the JCL definition belongs in which the link name can be locked
<i>LINKNAME</i>	Name of the link name

14.6.7 The JCDUNL command

A link name for AJM can be released again using the JCDUNL command.

Parameter	Meaning
<i>GROUP</i>	Group to which the JCL definition belongs in which the link name can be unlocked
<i>LINKNAME</i>	Name of the link name

14.6.8 The JVRDEL command

A job variable can be deleted using the JVRDEL command. The return code of JVRCMD is recorded in a message and changed to one of the return codes listed above.

Note: The ADDRESS command has no influence on the JVRDEL command. This always refers to job variables on the computer on which the command was issued.

Parameter	Meaning
<i>NAME</i>	Name of the job variable which is to be deleted

14.6.9 The JVRSET command

A job variable can be defined and changed using the JVRSET command. The return code of JVRCMD is recorded in a message and changed to one of the return codes listed above.

Note: The ADDRESS command has no influence on the JVRSET command. This always refers to job variables on the computer on which the command was issued.

Parameter	Meaning
<i>NAME</i>	Name of the job variable which is to be defined / changed
<i>VALUE</i>	New value of the of the job variable
<i>OFFSET</i>	Offset as of which the value is to be set If this value is not given then OFFSET(0) is assumed.
<i>LENGTH</i>	Length of the value to be set If this value is missing then the length of the VALUE entry is used.
<i>LIFETIME</i>	Period of validity of the job variable The entry has the format days/hours/minutes and each entry may have up to two digits or may be missing. In addition, * can be entered for infinite. If no lifetime is specified with a new job variable, LIFETIME(*) is assumed. If no lifetime is specified with an existing job variable, the lifetime remains unchanged.

14.6.10 The XTRSET command

An external trigger can be triggered using the XTRSET command.

Note: The ADDRESS command has no influence on the XTRSET command. The JVRSET command always refers to all global AJM systems.

Parameter	Meaning
<i>TYPE</i>	Type of the external trigger The type is used to group the various types of triggers (e.g. MQSeries, RFTS, User). It is used to check authorization.
<i>SUBTYPE</i>	Subtype of the external trigger The subtype is used to subdivide triggers of the same kind. It is also used to check authorization.
<i>LEN01</i>	Length of the first parameter If this entry is missing, then a length of "0" is assumed. VAL01 may not be used then.
<i>VAL01</i>	Value of the first parameter This value may contain as many characters as are entered in the parameter LEN01. If the value is shorter, then it is filled with blanks.
<i>LEN02</i>	Length of the second parameter If this entry is missing, then a length of "0" is assumed. VAL02 may not be used then.
<i>VAL02</i>	Value of the second parameter This value may contain as many characters as are entered in the parameter LEN02. If the value is shorter, then it is filled with blanks.

14.6.11 The LOGREAD command

Data from AJM log files can be written into an output file using the LOGREAD command. In addition, the available log files are listed (see also "The LOGLIST command" on page 113).

The LOGREAD command allows the following parameters:

Parameter	Meaning						
<i>GROUP</i>	Name of a group for which the log files are to be evaluated The group name may be abbreviated with '*'. The net name may be abbreviated with '*'. The job name may be abbreviated with '*'. The logical resource name may be abbreviated with '*'. The variable name may be abbreviated with '*'. The date must be given in the format DD.MM.YY. The time must be given in the format HH:MM.						
<i>NET</i>	Name of a net for which the log files are to be evaluated						
<i>JOB</i>	Name of a job for which the log files are to be evaluated						
<i>LRS</i>	Name of a logical resource for which the log files are to be evaluated						
<i>VAR</i>	Name of an AJM/P variable for which the log files are to be evaluated						
<i>FROMDATE</i>	Date as of which the log files are to be evaluated						
<i>FROMTIME</i>	Time as of which the log files are to be evaluated						
<i>TODATE</i>	Date until which the log files are to be evaluated (default: current day)						
<i>TOTIME</i>	Time until which the log files are to be evaluated (default: 23:59)						
<i>LAST</i>	Relative entry indicating which time period is to be evaluated The entry consists of two parts: a two digit number followed by the unit (MIN for minutes, HRS for hours, DAY for days, WK for weeks or MON for months). The entries for number and unit must be separated by a blank. It always goes back to the beginning of the unit entered (e.g. 01 DAY goes back to yesterday, 00:00 o'clock). If this parameter is entered the entries for FROMDATE / FROMTIME or TODATE / TOTIME are ignored.						
<i>EXEC</i>	Selection of information on nets / job runs The following entries are possible (multiple entries are separated by commas): <table border="1" data-bbox="295 1960 1034 2092"> <tr> <td>JOB</td> <td>Start / end of nets and jobs</td> </tr> <tr> <td>TRC</td> <td>Additional information on processes (e.g. job trace, DS trigger)</td> </tr> <tr> <td>SYS</td> <td>Messages of the AJM system</td> </tr> </table>	JOB	Start / end of nets and jobs	TRC	Additional information on processes (e.g. job trace, DS trigger)	SYS	Messages of the AJM system
JOB	Start / end of nets and jobs						
TRC	Additional information on processes (e.g. job trace, DS trigger)						
SYS	Messages of the AJM system						

	ALL All record types listed above are evaluated		
<i>ADMI</i>	Selection of information on user actions The following entries are possible (multiple entries are separated by commas):		
	JOB	Administration of nets and jobs (new system, changes, manual start, locking, unlocking, etc.)	
	LRS	Administration of logical resources	
	JCD	Administration of JCL definitions	
	CAL	Administration of calendars	
	XSD	Administration of external system definitions	
	PRP	Administration of AJM/P dialogs (new systems, changes, execution)	
	VAR	Administration of AJM/P variables	
	PCD	Administration of AJM/P file definitions	
	ALL	All record types listed above are evaluated	
<i>FORMAT</i>	Selection of the list format S Standard format (prepared language-dependent texts) R Report format (standard structure as the basis for evaluations) L List format (equivalent to the result from LOGLIST)		OUTDD DD name of the output file (default: AJMOUT)

14.6.12 The LOGLIST command

The available AJM log files can be listed using the LOGLIST command. Output of the list is done in the file AJMOUT file or in the file whose DD name is given in the parameter OUTDD.

The list displayed has the following structure:

LIST OF AVAILABLE LOG FILES			

DATASET		FROM	TO
hlq.AJM.LOGFILE1	*	18.09.97 17:56	29.09.97 15:06
hlq.AJM.LOGFILE2		05.09.97 12:00	18.09.97 17:56
hlq.AJM.LOGFILE.ARCHIVE.G0005V00 (M1)		05.09.97 12:00	18.09.97 17:56
hlq.AJM.LOGFILE.ARCHIVE.G0004V00 (M2)		31.08.97 11:17	05.09.97 12:00

The active AJM log files are listed first followed by the archived ones. Each line contains information on the time period which the log file covers. The list is sorted in descending order, i.e. the latest data is listed first.

A * before the date entry indicates which files were selected based on the time parameters given (FROMDATE/TIME, TODATE/TIME, or LAST).

If (M1) or (M2) is displayed behind the file name, then the file has been migrated (migration level 1/2). A selection of log files from this file leads to an automatic recall by AJM.

14.7 AJM Definition File (AJMIN)

The input records have the following structure:

)Subsection Parameter(value) Parameter(value)

The following rules apply to this:

1. The beginning of a subsection is marked with a right bracket) in column 1.
2. Continuation lines of a subsection must begin with a blank in column 1, as of column 2 the parameter entries may follow.
3. Parameter entries which do not fit completely into one line may be continued in the continuation line. To do this, add a * in column 72. The parameter value must be continued in the following line as of column 12.
4. Comment lines are marked with * in column 1.
5. If there are definitions for several jobs in an input file, then the end of a job is to be marked with // as of column 1.

14.7.1 AJMIN Structure for Job Definitions

Subsection / Parameter	Meaning
<i>JOB-ID</i>	Job definitions
<i>JOB</i>	Job name
<i>NET</i>	net name
<i>GROUP</i>	Group name
<i>SEQ</i>	Sequential number (determines the display sequence on the screen)
<i>CALENDAR</i>	Calendar name
<i>TRACE</i>	Trace Y/N
<i>JCL</i>	JCL information
<i>LINK</i>	Link name
<i>MEMBER</i>	Member name
<i>JCLT</i>	Temporary JCL information
<i>LINK</i>	Link name
<i>MEMBER</i>	Member name
<i>FROMDATE</i>	Date of validity (from)
<i>FROMTIME</i>	Time of validity (from)
<i>TODATE</i>	Date of validity (to)

<i>TOTIME</i>	Time of validity (to)
STC	Definitions for started tasks
<i>SYSTEM</i>	System name
<i>CMD</i>	Start command string
EXTSYS	Definition for external systems
<i>NAME</i>	Name of the external system
JOBPREP	Preparation information
<i>REQ</i>	Preparation required y/n
RESTART	Restart definitions
<i>ATTR</i>	Restart attribute (RR, RS, none)
<i>LIMIT</i>	Restart limit (max. number of restart attempts)
<i>DSN-TRACK</i>	Record datasets
<i>DSN-DEL</i>	Delete datasets (on automatic restart)
TERM-CNTL	Termination control definitions
<i>OP</i>	Comparison operator (e.g. =, >)
<i>CODE</i>	Code (e.g. C0004, U1111, S0C4)
<i>ACTION</i>	Action (e.g. OK, R-RST, J-WAIT)
<i>PARAM</i>	Parameter for action
<i>RECNET</i>	Name of a recovery net
SCHEDLST	Schedule list
<i>NR</i>	Number of the schedule list
<i>TYPE</i>	Type of the schedule list (S(ubmit), B(ypass), L(ate))
<i>WD-BEFORE</i>	Work day before ?
<i>VALID</i>	Valid y/n
<i>FROMDATE</i>	Date of validity (from)
<i>FROMTIME</i>	Time of validity (from)
<i>TODATE</i>	Date of validity (to)
<i>TOTIME</i>	Time of validity (to)
INTERNAL	Internal precedent
<i>JOB</i>	Job name of the internal precedent
<i>LOGDEP</i>	Logical link to previous definition (AND, OR)
<i>ALT</i>	Number of the alternative path or ANY
EXTERNAL	External dependency
<i>NET</i>	net name of the external dependency
<i>JOB</i>	Job name of the external dependency
<i>LOGDEP</i>	Logical link to previous definition (AND, OR)
<i>ACTIVE</i>	Active y/n
<i>HOURSVAL</i>	Validity (in hours)
<i>MINSVAL</i>	Validity (in minutes)
<i>OP</i>	Comparison operator (e.g. =, >)
<i>CODE</i>	Code (e.g. C0004, U1111, S0C4)
INTERVAL	Interval
<i>MINUTE</i>	Interval in minutes
<i>HOUR</i>	Interval in hours
<i>DAY</i>	Interval in days
<i>WDAY</i>	Interval in work days

<i>WEEK</i>	Interval in weeks
<i>MONTH</i>	Interval in months
<i>YEAR</i>	Interval in years
<i>DAYTYPE</i>	Type of day
<i>VALID</i>	y/n
<i>WD</i>	Work day
<i>HD</i>	Holiday
<i>RD</i>	Day of rest
<i>DAYNAME</i>	Name of day
<i>VALID</i>	y/n
<i>MO</i>	Monday
<i>TU</i>	Tuesday
<i>WE</i>	Wednesday
<i>TH</i>	Thursday
<i>FR</i>	Friday
<i>SA</i>	Saturday
<i>SU</i>	Sunday
<i>RELDAY</i>	Relative day
<i>VALID</i>	y/n
<i>LOGDEP</i>	Logical link to previous definition (AND, OR)
<i>TYPE</i>	Type of day
<i>WEEK</i>	Relative day within a week
<i>MONTH</i>	Relative day within a month
<i>YEAR</i>	Relative day within a year
<i>TIME</i>	Time window
<i>VALID</i>	y/n
<i>LOGDEP</i>	Logical link to previous definition (AND, OR)
<i>AFTER</i>	Time window(from)
<i>BEFORE</i>	Time window (to)
<i>ONCE</i>	One time
<i>DATE</i>	Date window
<i>VALID</i>	y/n
<i>LOGDEP</i>	Logical link to previous definition (AND, OR)
<i>FROM</i>	Date window (from)
<i>TO</i>	Date window (to)
<i>JOBVAR</i>	Job variable
<i>NAME</i>	Name of the job variable
<i>LOGDEP</i>	Logical link to previous definition (AND, OR)
<i>DISP</i>	Offset after which check is to take place
<i>LENGTH</i>	Length in which check is to take place
<i>OP</i>	Comparison operator
<i>VALUE</i>	Value to be checked against
<i>LOGRES</i>	Logical resource
<i>NAME</i>	Name of the logical resource
<i>LOGDEP</i>	Logical link to previous definition (AND, OR)
<i>NUMBER</i>	Number to be filled (number or ALL)

<i>DSTRIGG</i>	Dataset trigger
<i>DSN</i>	Dataset name (also generic)
<i>LOGDEP</i>	Logical link to previous definition (AND, OR)
<i>HOURSVAL</i>	Validity (in hours)
<i>MINSVAL</i>	Validity (in minutes)
<i>EXTTRIG</i>	External trigger
<i>TYPE</i>	Type of the external trigger
<i>SUBTYPE</i>	Subtype of the external trigger
<i>LOGDEP</i>	Logical link to previous definition (AND, OR)
<i>HOURSVAL</i>	Validity (in hours)
<i>MINSVAL</i>	Validity (in minutes)
<i>LEN01</i>	Length of the first parameter
<i>VAL01</i>	Value of the first parameter
<i>LEN02</i>	Length of the second parameter
<i>VAL02</i>	Value of the second parameter

14.7.2 AJMIN Structure for Documentation

Subsection / Parameter	Meaning
<i>JOB-ID</i>	Job information (only for job documentation)
<i>GROUP</i>	Group name
<i>NET</i>	net name
<i>JOB</i>	Job name
<i>NET-ID</i>	net information (only for net documentation)
<i>GROUP</i>	Group name
<i>NET</i>	net name
<i>DOC</i>	Documentation text
<i>TEXT</i>	Text contents (may be repeated)
<i>ACT</i>	Action note
<i>TEXT</i>	Text contents (may be repeated)

15 Batch/Command interface for job variables

Job variables can be created, deleted, changed, read or queried with the program JVRAICMD. The program runs in batch as well as under a CLIST / REXX. The appropriate function is controlled by the PARM entry.

Access to job variables is only possible within the same complex; the authorization check takes place on the computer on which the job which performs the access runs.

The call looks as follows:

```
//jobname JOB ...  
//TEST EXEC PGM=JVRAICMD,  
// PARM='SET USER1.V3(10,5)=TEST1 LIFETIME(2/5)'  
//
```

In a CLIST the call looks as follows:

```
ISPEXEC SELECT PGM(JVRAICMD) PARM(SET USER1.V3(10,5)=TEST1 LIFETIME(2/5))
```

The following functions are available:

SET Create or change a job variable

IF Query a job variable

GET Read the value of a job variable

DEL Delete a job variable

15.1 SET - Creating or changing job variables

A job variable can be created and filled with a value and a lifetime using the SET command. If a job variable already exists, then SET can be used to change its value and lifetime.

The trace options of a job variable and the descriptive comment can only be modified using the administration interface.

The SET command is structured as follows:

```
SET jvr-hlq.jvr-llq(d,l)=value [LIFETIME(days/hours/minutes)]  
[ABEND]
```

The individual parameters have the following meanings:

jvr-hlq High level qualifier of the job variable name

jvr-llq Low level qualifier of the job variable name

d Offset

The value is overwritten as of this offset. Offset 0 is the beginning of the job variable value, offset 59 the last place.

l Length

The value is entered into the job variable in this length. Any previously existing value is overwritten. The maximum length that may be entered is 60. The entry for value must correspond to this length.

value New contents of the job variable

The contents of a job variable cover 60 places. It may consist of any characters. Parts of the value that are not defined by explicit setting contain blanks.

LIFETIME Period of validity

The lifetime can be given in days / hours / minutes. Valid numeric values are 0 - 99. The time entries can also be combined with each other or be individual entries (see following examples).

During the LIFETIME(*) the job variable retains unlimited validity.

If this entry is missing, then the previous lifetime is maintained for an existing job variable and an unlimited lifetime is assumed for a new job variable.

After expiry of the lifetime the job variable has an undetermined value and cannot be used in a JCL.

Examples for the parameter LIFETIME:

(0) Valid until 24:00 of the current day

(1) Valid until 24:00 of the next day

(/12) Valid for the next 12 hours

(//30) Valid for the next 30 minutes

(/5/30) Valid for the next 5 1/2 hours

(1/5/30) Valid for a day (24 hours) + 5 1/2 hours

(1//30) Valid for a day (24 hours) + 1/2 hour

ABEND Conversion of condition codes into user abends

This entry means that a condition code other than 0 is converted into a user abend. Instead of a condition code C0012 a user abend U0012 would be output.

The following return codes may occur:

- 0 Processing without errors
- 12 Lack of access authorization. See message ICH408I.
- 16 Access to job variable file (VSAM) not possible.
- 20 Syntax error on calling JVRAICMD

15.2 IF-Querying a job variable

To query the value of a job variable you have the IF command available. It generates a return code based on the query which triggers a reaction, for example, using "Dependent Job Control".

The IF command is structured as follows:

```
IF jvr-hlq.jvr-llq(d,l)=value [THEN(rc) ELSE(rc)] [ABEND]
```

The individual parameters have the following meanings:

jvr-hlq High level qualifier of the job variable name

jvr-llq Low level qualifier of the job variable name

d Offset

The value is queried as of this offset. Offset 0 is the beginning of the job variable value, offset 59 the last place.

l Length

The value is checked to this length. The maximum length that may be entered is 60. The entry for value must correspond to this length.

value Comparison value

THEN Return code for true

If the job variable exists, is currently valid, and agrees with the entry for value in the area given, then the return code entered is set. Without a THEN entry the return code 0 is used.

ELSE Return code for false

If the conditions listed under THEN are not fulfilled or an error occurs accessing the job variable (e.g. lack of authorization, VSAM file access unsuccessful), then the return code entered here is set. Without an ELSE entry the return code 12 is used.

ABEND Conversion of condition codes into user abends

This entry means that a condition code other than 0 is converted into a user abend. Instead of a condition code C0012 a user abend U0012 would be output.

15.3 GET - Reading a Job Variable

In order to read the value of a job variable the GET command can be used. Either the value can be placed in an ISPF variable in the profile pool or the address of the value can be made available in register 1.

The GET command is structure as follows:

```
GET jvr-hlq.jvr-llq(d,l) [ispf-variable]
```

The individual parameters have the following meanings:

jvr-hlq High level qualifier of the job variable name

jvr-llq Low level qualifier of the job variable name

d Offset

The value is read as of this offset. Offset 0 is the beginning of the job variable value, offset 59 the last place.

l Length

The value is read to this length. The maximum length that may be entered is 60.

ispf-variable Name of an ISPF variable

The value of the job variable is set in the ISPF variable here (as of place d in the length l). This can now be read out of the profile pool with **VGET ... PROFILE** and, for example, further processed in a CLIST. The entry of an ISPF variable only makes sense under ISPF and is only allowed there.

After a GET command the following condition codes may occur:

0 Processing without errors

4 Lifetime of the job variable expired

8 Job variable not found

12 Lack of access authorization. See message ICH4081.

16 Access to job variable file (VSAM) not possible.

20 Syntax error on calling JVRAICMD

15.4 DEL - Deleting a job variable

The DEL command is to delete job variables.

The DEL command is structured as follows:

```
DEL jvr-hlq.jvr-llq [ABEND]
```

The individual parameters have the following meanings:

- jvr-hlq** High level qualifier of the job variable name
- jvr-llq** Low level qualifier of the job variable name
- ABEND** Conversion of condition codes into user abends

This entry means that a condition code other than 0 is converted into a user abend. Instead of a condition code C0012 a user abend U0012 would be output.

The following condition codes are possible:

- 0** Processing without errors
- 8** Job variable not found
- 12** Lack of access authorization. See message ICH408I.
- 16** Access to job variable file (VSAM) not possible.
- 20** Syntax error on calling JVRAICMD

15.5 Examples

```
//SETJV EXEC PGM=JVRAICMD,PARM='SET USER1.TEST(4,1)=T LIFETIME=(0/12)'
```

The value of the job variable USER1.TEST is changed as of the fifth place to T. The job variable remains valid for 12 hours after the change.

```
//IFJV EXEC PGM=JVRAICMD,  
// PARM='IF USER1.TEST(4,1)=T THEN(4) ELSE(0) ABEND'
```

If the job variable USER1.TEST is valid and has a value of T in the fifth place, then the job is cancelled with the user abend U0004, otherwise it ends with condition code C0000.

```
ISPEXEC SELECT PGM(JVRAICMD) PARM(GET USER1.TEST(4,1) VALUE)
ISPEXEC VGET VALUE PROFILE
IF &LASTCC. = 0 THEN
  WRITE &VALUE.
```

The value which is in the fifth place in the job variable USER1.TEST is read into the ISPF variable VALUE, taken out of the profile pool, and can then be used further. If the return code is not zero, then the value could not be read out.

16 Setting external triggers from programs

External triggers may be set from any program using the program AJMAIXTR. The authorization to set such a trigger is checked using the profile The authorization check is done on the computer on which the application is running which sets the trigger.

The parameter bar for the assembler call is found in the member AJMAAIXT in MACLIB. For COBOL the call looks as follows:

```
WORKING-STORAGE SECTION.  
  
01  XTR-PARAMS.  
   10  XTR-TYPE                               PIC X(8) .  
   10  XTR-SUBTYPE                            PIC X(8) .  
   10  XTR-PARM01.  
       20  XTR-PARM01-LEN                     PIC 9(2) BINARY .  
       20  XTR-PARM01-VALUE                   PIC X(nn) .  
   10  XTR-PARM02.  
       20  XTR-PARM02-LEN                     PIC 9(2) BINARY .  
       20  XTR-PARM02-VALUE                   PIC X(nn) .  
  
   ...  
  
LINKAGE SECTION.  
  
   ...  
  
CALL "AJMAIXTR" USING XTR-TYPE XTR-SUBTYPE  
                     XTR-PARM01-LEN XTR-PARM01-VALUE  
                     XTR-PARM02-LEN XTR-PARM02-VALUE .  
  
   ...
```

The parameters must be filled as follows when called:

XTR-TYPE	Type of the external trigger
XTR-SUBTYPE	Subtype of the external trigger
	Type and subtype are used for the authorization check.
XTR-PARM01-LEN	Length of the first parameter
XTR-PARM01-VALUE	Value of the first parameter
XTR-PARM02-LEN	Length of the second parameter
XTR-PARM02-VALUE	Value of the second parameter

Note: Length and value of a parameter must be given in pairs. Between 0 and 2 parameters can be defined depending on the type of trigger.

17 Program AJMTEST

AJMTEST is an auxiliary program that is used to simulate a job run. Both the runtime (WAIT=) as well as the termination code (TERM=) can be parameterized.

In the SAMPLIB library you will find the following sample job in the member AJMTEST:

```
//AJMTEST JOB (ACCNT, . . .), AJM, CLASS=Q, MSGCLASS=V,  
//      USER=????????  
//*  
//*-----  
//*  
//*      SAMPLE JOB FOR AJMTEST  
//*  
//*-----  
//*  
//TEST   EXEC PGM=AJMTEST, PARM='WAIT=5, TERM=C0008'  
/*
```

The following parm entries are possible:

WAIT=NNN NNN is the number of seconds which the job should be active.

TERM=Cxxx Condition code with which the job should end. For xxx values from 0000 to 4095 are possible.

TERM=Uxxx User abend code with which the job should end. For xxx values from 0000 to 4095 are possible.

TERM=Sxxx System abend code with which the job should end. For xxx values from 000 to FFF are possible.

The program ends with the system abend SFFF on incorrect or incomplete parm entries.

You can check whether the net definitions made in AJM make sense with the help of the program AJMTEST. In the example above the job AJMTEST ends with the condition code C0008 after 5 seconds. You can use this to test termination code controls, for example to check whether the desired alternative path is taken after the occurrence of the condition code.

Appendix

18 Appendix A. Notes on troubleshooting

18.1 SVC dump for problems in the AJM system

In order to document problems within the AJM system, an SVC dump is automatically generated by the monitor task of the AJM when a subtask has an abnormal end. In addition, a message is output which documents the generation of the SVC dump (Message number AJMBF99A).

If the task is of vital importance for the operation of AJM, the message

is output at regular intervals.

The dump generated contains all areas necessary for the analysis. The computer center should prepare the dump for the analysis using IPCS and make it accessible to the AJM development team.

Afterwards the AJM system affected must be terminated and restarted.

If the generation of an SVC dump is not desired, then it may be suppressed with the aid of the MVS-SLIP command. However, this means that in the event of an error, analysis is almost impossible.

18.2 SYSMDUMP for problems in the AJM administration

In order to document problems which lead to termination of the AJM administration interface, proceed as follows:

1. Call the REXX SEMIDMPA (input "TSO SEMIDMPA" in the command line). You see the following messages:

```
SEMIDMPA002I New dump data set allocated to SYSMDUMP  
SEMIDMPA003I Data set name is 'userid.ISPF.DUMP.Ddddddd.Tttttt'  
SEMIDMPA004I ISPF ENVIRON dump now enabled
```

2. Go through the steps that led to the termination.
3. After the dump is generated, exit the ISPF and TSO (new LOGON).
4. Report the problem and enter the name of the dump file generated (see above). The AJM development team requires READ access to this file to conduct an analysis.

18.3 "DIA" line command

In any job line the line command DIA can be entered. An extract from the AJM database in the dump format is displayed that contains all definitions for the selected job.

When a problem with a job definition occurs, create a current extract and make it available to the AJM development team.

Please remember that all actions that change the status of a job (e.g. STR, RUN, CLR) also change the information in the database. The database extract must therefore be created **before** such actions are undertaken since otherwise an analysis of the situation is no longer possible.

18.4 Program AJMUTEVT

The program AJMUTEVT can be used for evaluations and troubleshooting. It offers the ability to filter events from one or more event files and to sort them by time in a sequential output file. The name of this file is 'Userid.AJM.EVENTx.LIST' (x=number). Up to 5 event files can be searched.

The call is done using the "Utilities" menu in the administration or directly with 'TSO AJMUTEVT'. Afterwards the following panel appears:

```

AJM ----- CHECK EVENT DATASET -----
CMD====>

AJM Parm Module : AJM1PARM   and           AJM-ID: AJM1
or
Event Dataset(s) : _____
                  : _____
                  : _____
                  : _____
                  : _____

Start Date      : 18.09.95 (dd.mm.yy)      Time: 00:00 (hh:mm)
End Date       : 18.09.95 (dd.mm.yy)      Time: 23:59 (hh:mm)

Event Type(s)  : Job Start: Y (y/n)   Job End : Y (y/n)   AJM/S   : Y (y/n)
                  Net Prep.: Y (y/n)   Netname : * _____ Jobname : * _____
                  Sys. IPL : Y (y/n)
                  Jobvar  : Y (y/n)   Name    : * _____
                  DSN Trig: Y (y/n)
                  Data Set : * _____
                  Ext Trig : Y (y/n)   Type    : * _____ Subtype : * _____

Output        : readable : Y (y/n)   hexadec.: Y (y/n)   original: Y (y/n)
                  primary / secondary Space : 10 / 05   tracks

```

The panel consists of three sections. In the first part you determine which event files are to be evaluated. To do this you either enter the name of your AJM parameter module and the AJM ID or the names of the event files which are to be searched.

In the middle section you can enter various selection criteria for the events being searched for.

In the lower section of the panel you decide on the display of the data and the size of the output file.

AJM Parm Module Name of the parameter module of the global AJM

The names of the event files are stored in the parameter module. If the appropriate AJM supplies event routing (i. e. local AJM, too), the parameter module lists also the event files of the other AJM systems that run on the same computer.

AJM ID Name of the AJM system (AJM1/2/3/4) for which the event files are to be evaluated

Event Dataset(s) Name of the event file(s)

Up to 5 event files can be entered.

Start Date / Time Date and time

As of this time all desired events are output.

End Date / Time Date and time

All desired events are output up to this time.

Event Type(s) Event types

All event types can be entered which are to be extracted. At least one event type that contains the date and time must be selected.

The following entries are possible:

Job Start Job start

The selection of the job can be limited with

- Date
- Time
- Net name
- Job name

Job End Job end

The selection of the job can be limited with

- Date
- Time
- Net name
- Job name

AJM/S O-Codes (from AJM/S)

The selection of the job can be limited with

- Date
- Time
- Net name
- Job name

Net Prep. Job preparation valid (from AJM/P)

The selection of the job can be limited with

- Date
- Time
- Net name
- Job name

**Sys.
IPL** System IPL

The selection of the records can be limited with

- Date
- Time

Jobvar Job variable and contents

Since no time is stored for the job variable events, all job variables before the first time period searched and all job variables after the last event in the searched time period are taken into account. This means that it is possible that too many job variable events are displayed.

The selection of the records can be limited with

- Name

**DSN
Trig** Dataset trigger

Depending on the type of dataset trigger this event is generated as soon as a dataset which was defined as a dataset trigger for AJM is cataloged or closed after an update.

The selection of the records can be limited with

- Date
- Time
- Dataset

**Ext
Trig** External trigger

This event is generated when an external trigger for AJM is triggered from the interface AJMAIXTR.

The selection of the records can be limited with

- Date
- Time
- Type

- Subtype

Output Output format

The collected events can be displayed in three forms.

The following entries are possible:

readable Events are displayed in readable form

hexadecimal Events are displayed in hexadecimal form

original Copy of the events from the event file

Be careful with hardcopy, the data may contain characters which cannot be printed.

prim/sec space Primary and secondary space

These parameters determine the size of the output file. If your space entries are not large enough, then you may have to delete the output file already created. Afterwards you can call AJMUTEVT again with enlarged space entries.

18.5 Trace options in the AJM system

The following traces can be switched on in necessary for troubleshooting in the AJM system:

Trace type	Function	Start / Stop
<i>Event</i>	<p>Every event that is processed by the corresponding AJM generates a log record in the AJM log. These log records are only displayed to administrators with master authorization.</p> <p>The file name from the event file from which this event was read is in the log as well as the display of the exact position of the event record. In addition, the beginning of the event record is displayed.</p> <p>Events in AJM can be followed using this trace.</p> <p>Special events (job initiation and termination) can be recorded using their own traces.</p>	F ajm,EVENTTRC=ON/OFF
<i>Job initiation</i>	<p>Every job init event which is processed by the AJM system generates a message in the SYSLOG (message number AJMEH99I).</p> <p>The net and job name can be seen there.</p> <p>Which net or job name was reported by the corresponding SMF exit on job initiation can be seen using this trace.</p>	F ajm,JOBI=ON/OFF

<p><i>Job termination</i></p>	<p>Every job term event which is processed by the AJM system generates a message in the SYSLOG (message number AJMEH99I).</p> <p>The net and job name can be seen there.</p> <p>Which net or job name was reported by the corresponding SMF exit on job termination can be seen using this trace.</p>	<p>F ajm,JOBT=ON/OFF</p>								
<p><i>Database writing</i></p>	<p>Each DIV SAVE of the AJM database generates a message in the SYSLOG (message number AJMDB14I).</p> <p>Which interval is used to physically save the AJM database can be seen using this trace.</p>	<p>F ajm,DBSAVE=ON/OFF</p>								
<p><i>Security check</i></p>	<p>Each security check in the global AJM generates a message in the AJM log (message number AJMBF40I).</p> <p>Which profile was used for the authorization check can be seen using this trace.</p> <p>Displayed are</p> <ul style="list-style-type: none"> • The user ID checked • The profile which the check is made against • The function code: <table border="1" data-bbox="384 1272 927 1480"> <tr> <td style="text-align: center;">1</td> <td>Check for read authorization</td> </tr> <tr> <td style="text-align: center;">2</td> <td>Check for update authorization</td> </tr> <tr> <td style="text-align: center;">3</td> <td>Check for command authorization</td> </tr> <tr> <td style="text-align: center;">4</td> <td>Fetching the authorization level (for the display in the administration)</td> </tr> </table>	1	Check for read authorization	2	Check for update authorization	3	Check for command authorization	4	Fetching the authorization level (for the display in the administration)	<p>F ajm,RACFTRC=ON/OFF</p>
1	Check for read authorization									
2	Check for update authorization									
3	Check for command authorization									
4	Fetching the authorization level (for the display in the administration)									

Note: Please remember that traces always degrade performance.

18.6 Special job status information

In the job status panel (see user handbook) some information is displayed in the field "Scheduler status" which requires intervention by the computer center. The following is a list of these status entries and the actions necessary for them are explained.

Logical error in ...

The job definition is not correctly stored in the AJM database.

If this message repeats after changing the definition, then you can use the line command "DIA" to generate a dump of the appropriate database entry. Print out the file generated and send it to the AJM development team.

Scheduler locked by operator

The scheduler part of the AJM was locked. No more jobs can be started by AJM. A warning is output for each access to the AJM system in the administration interface.

The scheduler part of the AJM can be reactivated using the modify command

(see also "Overview of the operator commands" on page 143).

Note: Using the line commands "STR", "RUN", and "EXE" in the user dialog (see user handbook) jobs can be started manually regardless of the schedule conditions. Please remember that the commands "RUN" and "STR" affect other jobs. If the job is started this way, then the condition "internal precedent" is fulfilled within a net for dependent jobs. If there are dependencies on jobs in other nets, then the conditions "external precedent" is also fulfilled for the time span given. As soon as the schedule algorithm is activated again then these conditions are evaluated; a "RUN" or "STR" may cause a delayed net run.

Memory bottleneck in JCL preparation

A memory bottleneck occurred in the preparation of the JCL for the submit.

Possibly the REGION entry of the global AJM system is too small to prepare the JCL for the submit. AJM searches the given JCL member for the desired job and loads the JCL cards in the main storage. If the job is very large or if too many cards are inserted by the submit user exits AJMJSUX1 and AJMJSUX2 (possibly as a consequence of a program loop in the exit), then a bottleneck may occur. First check the JCL member of the job, perhaps the job contains a large number of instream data cards. Then check the logic of the submit exit you used for a program loop.

Error in writing the restart JCL

An error occurred when writing the JCL of the job to the restart database in which the JCL is maintained if a restart should become necessary.

A message **AJMRR...** is written into the AJM job log and into the system log. It gives more information on the cause of the error. If the error occurs again, save the AJM job log and contact the AJM development team for diagnosis of the problem.

Error in reading the restart JCL

An error occurred when reading the JCL of the job to the restart database in which the JCL is maintained if a restart should become necessary.

A message **AJMRR...** is written into the AJM job log and into the system log. It gives more information on the cause of the error. If the error occurs again save the AJM job log and contact the AJM development team for diagnosis of the problem.

JCL not found for restart

A restart was requested, but the restart database in which the JCL of the last job run is kept for a possible restart does not contain any data for this job.

This situation may occur when the restart database (e.g. as a consequence of a disk defect) was regenerated. It does not contain any images of the JCL of the last run.

Error on open for INTERNAL READER

An error occurred during the transfer of the JCL to JES.

Check whether there are general problems in JES.

18.7 Checklists for common problems

18.7.1 Jobs in status "submitted"

If job started by AJM got stuck in the status "submitted", please check the following points:

Job definition:

1. Has the job been started or is it stuck in the status "waiting for execution" or in the operator hold ?
2. Has the job ended or was it cancelled before it was begun (e.g. because of a faulty job card or other JCL error) ?

This problem can no longer occur when you have installed AJM/S and the MSGCLASS given in this job is processed by AJM/S.

3. Was the job cancelled before it was started ?

This problem can no longer occur when you have installed AJM/S and the MSGCLASS given in this job is processed by AJM/S.

4. Does the job consist of at least three non-comment cards ?

Problems may occur when JES2 / JES3 control cards are found among the first three cards or when cards are added (by any own exit).

AJM parameterization:

5. Does at least one AJM have the "event routing" function (parameter "Event routing function" = "Y" when generating the parameter module) on the computer on which the job ran ?
6. Are event files assigned for each computer and for each AJM system ?
7. Are the correct event files evaluated by the right AJM system ?

Caution: Event files for the AJM system AJM1 are in the first event file line of each local definition, the event files for the AJM system AJM2 are in the second, etc.

AJM systems on the appropriate computer:

8. Is an local AJM started on the computer on which the job ran ?
9. Did the job run before an IPL and was the associated AJM system still active ?
10. Did the job run after an IPL and was the associated AJM system already active ?

Miscellaneous parameterization:

11. Are all necessary record types active in the parmlib member SMFPRMxx (SMF records 15, 30, 61, and 66) ?
12. Are all necessary exits active in the parmlib member (Exits IEFUJV, IEFUJI, IEFU83, IEFU84) ?
13. Are these exits still active (SMF deactivates exits which cause abends) ?
14. Are all exits correctly defined in the SEM parameter member (Exits IEFUJV, IEFUJI, IEFU83, IEFU84 with the appropriate AJMxxx exits) ?
15. Are the router exits of the SEM correctly installed or are the IBM modules being used (Exits IEFUJV, IEFUJI, IEFU83, IEFU84) ?

Note: The utility AJMDFBAS can be used to check the SEM definitions (see AJM Installation Manual).

External systems:

16. Has the agent on the corresponding UNIX system been started? If yes, then is it still working? (see "Communication error with an external system" on page 140 for additional causes of errors)

18.7.2 Dataset trigger is not triggered

If dataset triggers are not triggered although the corresponding file is generated, updated or renamed, then please check the following points:

Parameterization in SEM:

1. Are the filters for the dataset names correctly defined in the SEM ?
2. Are the variable and dataset names in the "DEFINE VARIABLES" statement correct ?
3. Was the job which generated or modified the dataset suppressed by the job filter ?
4. Are all exits correctly defined in the SEM parameter member (Exits IEFUJV, IEFUJI, IEFU83, IEFU84 with the corresponding AJMxxx exits) ?
5. Are the router exits of the SEM correctly installed or were the IBM modules used (Exits IEFUJV, IEFUJI, IEFU83, IEFU84) ?

Note: The utility AJMDFBAS can be used to check the SEM definitions (see AJM Installation Manual).

AJM parameterization:

6. Does at least one AJM have the "event routing" function (parameter "Event routing function" = "Y" when generating the parameter module) on the computer on which the job ran ?
7. Are event files assigned for each computer and for each AJM system ?
8. Are the correct event files evaluated by the right AJM system ?

Caution: Event files for the AJM system AJM1 are in the first event file line of each local definition, the event files for the AJM system AJM2 are in the second, etc.

AJM systems on the appropriate computer:

9. Is an local AJM started on the computer on which the dataset was created, renamed or modified ?
10. Was the dataset created, renamed or modified just before an IPL and was the associated AJM system still active ?
11. Was the dataset created, renamed or modified just after an IPL and was the associated AJM system already active ?

Miscellaneous parameterization:

12. Are all necessary record types active in the parmlib member SMFPRMxx (SMF records 15, 30, 61, and 66) ?
13. Are all necessary exits active in the parmlib member SMFPRMxx (Exits IEFUJV, IEFUJI, IEFU83, IEFU84) ?
14. Are these exits still active (SMF deactivates exits which cause abends) ?

Other causes:

15. Was the dataset trigger condition defined after the file was generated ?
16. Is the dataset name given incorrectly in the job definition ?
17. Was the dataset created on a volume BACKUP ? Datasets on this volume do not cause dataset triggers since the HSMMAIN program uses this information for administration purposes.

If the problem cannot be traced to one of the causes listed above, then the SMF records created during cataloging or closing the datasets (record types 15, 61, and 66) must be copied into a file and sent to the AJM development team for analysis.

18.7.3 Communication error with an external system

It is difficult to list all possible causes for operational errors here due to the many components which run between the AJM and the agents. Furthermore, problems which, for example, show up in the network which may be of only a very short duration can be hard to track down. This overview is intended to show how some sources of error can be found.

1. Update the definition of the external system to reset the communication error to make sure that the problem persists.
2. Check the AJM agent to make sure it is still working. The message "Waiting for work..." is written in the log file "ajmagt.log" every minute. If the AJM agent is no longer working then it should be restarted. If the problem persists, then you should check the points listed below.
3. If the agent hangs up and the last message in the agent log is an XMI logon, then there are currently problems in the corresponding SAP system. The logon was initiated but no reply was received from SAP. You can check this by logging on to the appropriate SAP.
4. If the message "RFC-Error..." appears repeatedly in the log for this SAP system, then there may be internal problems with SAP, for example, logon using the XMI interface may not be possible or there may be connection problems between the UNIX of the agent and the system on which SAP is running.
5. If there are a number of identical records in the file "ajmagt.tab" with the status "SUBP" for the same job, then it is advisable to remove these records except the most recently for performance reasons. If the job is not active in AJM, all records may be deleted. The AJM agent must be ended to delete the records. **Caution:** This file contains internal information on the status of the active processes. Records may only be removed completely, but may never be changed. If you have accidentally deleted the record of an active job, it then falls out of the monitoring of the agent and will not reach a job end in AJM.

19 Appendix B. Messages

Error messages in AJM have the following structure

AJMnnns - module name: message text

Error messages in AJM/P have the structure

AJPnnns - module name: message text

Whereby:

AJM	Message prefix for AJM messages
AJP	Message prefix for AJM/P messages
nnnn	Message number
s	Severity Code :p- The following values are possible: I Information W Warning E Error A Critical error Q Error, reply necessary

Module name Name of the module causing the message

Message text Explanatory text

The messages are available in two languages. The language for the messages from the AJM system (which go, for example into the SYSLOG) is determined in the parameterization. Following message modules are available

- AJMMSGGEN (AJM messages, English)
- AJMMSGGE (AJM messages, German)
- AJPMSGGEN (AJM/P messages, English)
- AJPMSGGE (AJM/P messages, German)

The language for messages within the administration interface can be determined for each user (selection 0, session options).

The language is determined for the command / batch interface using the PROFILE command (see "The PROFILE command" on page 108).

The language selected for the AJM system is used as the default language.

20 Appendix C. Operator interface

20.1 Overview of the operator commands

The following commands can be sent to the AJM system using the console:

Command	Function
<i>P ajm</i>	End AJM system
<i>F ajm,HELP</i>	Display available operator commands
<i>F ajm,STAT</i>	Display current status of the AJM system
	You can see whether the scheduler is switched on or off (see next command) and
<i>F ajm,SCHED=START/STOP</i>	Switch scheduler function on / off
	Using this command you can switch the scheduler part of the AJM system on or off. This command can be used, for example, to maintain AJM before termination for the processing of the log files.
	Please remember that the scheduler remains switched off after a SCHED=STOP command.
<i>F ajm,BACKUP</i>	Create backup file
<i>F ajm,LOGINDEX</i>	Create index record for AJM log files
	AJM checks which generations of the log file backup exist and updates the index. If the index of the log files is no longer correct in AJM (e.g. after manual deletion of a file from the log file), this command can be used to update the index.
<i>F ajm,LOGARCH=xxx</i>	Create archive file(s) for AJM log
	The following may be entered for xxx:
	ACT Archive copy of the active log file
	INACT Archive copy of the inactive log file
	BOTH Archive copy of both log files
<i>F ajm,EVENTTRC=ON/OFF</i>	Switch event trace on / off (see "Trace options in the AJM system" on page 135).
<i>F ajm,JOBI=ON/OFF</i>	Switch job initiation trace on / off (see "Trace options in the AJM system" on page 135).
<i>F ajm,JOBT=ON/OFF</i>	Switch job termination trace on / off (see "Trace options in the AJM system" on page 135).
<i>F ajm,DBSAVE=ON/OFF</i>	Switch database storage trace on / off (see "Trace options in the AJM system" on page 135).
<i>F ajm,RACFTRC=ON/OFF</i>	Switch RACF check trace on / off (see "Trace options in the AJM system" on page 135).

21 Appendix D. Storage space requirements for the AJM files

The following table is used to calculate the storage space necessary for the installation of AJM.

The files are created and initialized in the parameterization dialog. How large the files to be created are depends on the number of jobs to be administered in the AJM system. This value is to be given by the AJM administrator during the parameterization (**ESTIMATED NUMBER OF JOBS**).

The sizes given in the table apply to an entry of **1000** for the number of jobs to be administered. The sizes are to be recalculated for a greater number of jobs.

Note: The AJM database and the restart database expand themselves as needed. It is recommended that the number of jobs that is expected be chosen large enough so that the generation of any unnecessary extends is avoided.

File	DSORG	Number	Size Primary/Secondary	Comments				
<i>AJM database</i>	LDS (VSAM)	1 per AJM complex	25/5 (TRK)	AJM expands the file itself as needed.				
<i>AJM restart database</i>	KSDS (VSAM)	1 per AJM complex	Index: 1/1 (CYL) Data: 11/6 (CYL)					
<i>AJM/P database</i>	KSDS (VSAM)	1 per AJM complex when AJM/P is installed	Index: 2/2 (TRK) Data: 6/3 (CYL)					
<i>Event file</i>	DA	1 per MVS system and AJM complex	12/0 (TRK)	File is formatted and written in "wrap around".				
<i>Log file</i>	PS	2 per AJM complex	9/0 (TRK)	File is formatted and written in "wrap around".				
<i>Backup file</i>	PS (GDG)	Number of generations parameterizable	n/m (BLK,RLSE) <table border="1" style="margin-left: 20px;"> <tr> <td>n</td> <td>Size of AJM-DB in 4K blocks + 10%</td> </tr> <tr> <td>m</td> <td>n/2</td> </tr> </table>	n	Size of AJM-DB in 4K blocks + 10%	m	n/2	Actual storage capacity depends on: <ul style="list-style-type: none"> • Size of the AJM database. • Size of the AJM/P database. • Type of backup (with/without AJM/P)
n	Size of AJM-DB in 4K blocks + 10%							
m	n/2							
<i>Log archive file</i>	PS (GDG)	Number of generations parameterizable	Same size as the backed up log file					

22 Index

Special Characters
A B C D E G J L M O R S T U

Special Characters

- #GLOBAL
 - Data security 95

A

- Administration
 - Cross-system - 16
- AJM
 - Global - 15
 - Local - 15
- AJM Agent
 - Characteristics 21
 - Checking the status 24
 - Ending the agent 24
 - Error analysis 26
 - Files in operation 24
 - Installation 23
 - Messages in ajmagt.log 28
 - Operation 24
 - Starting the agent 24
- AJM Database
 - Allocation 56
 - Backup 67
 - Clean up 73
 - Expansion 66
 - Gr%(e 145
 - Mixing 70
 - Overview 17
 - Procedure on loss 78
 - Restoration 68
 - Restoration of individual jobs 77
 - Selective restoration 70
- AJM/P Database
 - Allocation 58
 - Clean up 73
 - Expansion 67
 - Gr%(e 145
 - Overview 17
 - Restoration 78
- AJMAICMD
 - *See* Batch / Command Interface
- AJMCMD
 - *See* Batch / Command Interface
- AJMTEST 127
- AJMUTEVT 132

B

- Backup Datasets
 - Allocation 56
 - Create via console 143
 - Gr%(e 145
 - Overview 18
- Batch / Command Interface
 - ADDRESS 106, 108

- AJMSTAT 106
- BYPASS 106
- BYPRUN 106
- CLEAR 106
- DELETE 106
- Examples 104
- EXECUTE 106
- EXTRACT 107
- for job variables (JVRAICMD) 119
- JCDLOCK 107, 110
- JCDUNL 107, 110
- JDOCEXT 107
- JDOCUPD 107
- JVRDEL 107, 110
- JVRSET 107, 110
- LOCK 106
- LOGLIST 107, 113
- LOGREAD 107, 112
- LRSUPD 107, 110
- LRSWHO 107, 110
- NDOCEXT 107
- NDOCUPD 107
- Overview 103
- PLSCHECK 107
- PROFILE 106, 108
- RESET 106
- RUN 106
- START 106
- STATUS 107
- Syntax 105
- UNLOCK 106
- UPDATE 106
- XTRSET 107, 111

C

- Commands
See Batch / Command Interface, Line Commands, Operator Interface
- Console Interface
See Operator Interface
 - Traces
See Trace Options

D

- Data Security
 - "USER=" entry 93
 - AJM/P File Definitions 98
 - AJM/P Variables 97
 - Calendar 95
 - Decentral administration 101
 - External system 93
 - External systems 101
 - External triggers 96
 - GENERICOWNER 101
 - JCL definitions 95
 - JCL files 97
 - Job Variables 96
 - Jobs 93

- Logical resources 95
- nets 93
- Overview 93, 98
- Packages 97
- Preparation dialogs 97
- Started Tasks 93, 101
- STC 93
- STCs 101
- Submit user 93
- Trace 101
- Datasets
 - See also* JCL Datasets, AJM Database, AJM/P Database, Restart Database, Event Files, Log-Dateien, Backup Datasets, Job Variable Dataset
 - Allocation 55
 - Overview 17

E

- Event Files
 - Allocation 57
 - Expansion 67
 - Gr%(e 145
 - Loss 78
 - Overview 17
- Events
 - Buffer in ECSA 16
 - Event routing 16
- External Systems
 - Characteristics 21
 - SAP/R3 22
- External Trigger
 - Batch/Command interface 111
 - Data security 96
 - Setting from programs 125

G

- Global AJM 15
- Group
 - #GLOBAL
 - See* #GLOBAL
 - Data security 93

J

- JCL Datasets
 - Data security 97
- JES2
- JES3
- Job
 - Status 136
- Job Variable
 - Batch/Command interface 110, 119
- Job Variable Dataset
 - Overview 18
- JVRAICMD
 - See* Batch / Command Interface

L

- Line Commands
 - DIA 131
 - STA 136

- Local AJM 15
- Log-Dateien
 - Allocation 56
 - Expansion 66
 - Gr%(e 145
 - Logged events 50
 - Overview 18

M

- Messages
 - Structure 141
- Migration

O

- Operator Interface
 - Create backup file 143
 - Create log archive file 143
 - Create log index 143
 - End AJM 143
 - Help information 143
 - Overview 143
 - Start scheduler 143
 - Status information 143
 - Stop scheduler 143

R

- RACF
 - See* Data Security
- Restart Database
 - Allocation 56
 - Clean up 73
 - Expansion 66
 - Gr%(e 145
 - Overview 17
 - Restoration 78

S

- SEM 87
- Submit User
 - Data security 93
- submitted
 - Jobs in status 138
- SVC Dump
 - To analyze problems 131
- SYSMDUMP
 - To analyze problems 131

T

- Trace Options
 - Database save 136
 - Events 135
 - Job initiation 135
 - Job termination 136
 - Overview 135
 - RACF check 136

U

- USER Parameter

See Submit User

Backpage

Copyright T-Systems International GmbH 2005

**T-Systems International GmbH
Computing Services & Solutions (CSS), System Products & Solutions (MSY-PS)
Fasanenweg 9, 70771 Leinfelden-Echterdingen, Germany**

**Phone: +49 89/1011-4685
Fax : +49 711/972-91622
E-mail: cc.jobmgmt@t-systems.com
Intranet : <http://t-systems.telekom.de/sps>
Internet : www.t-systems-systemproducts.com**

..... **T** .. Systems ..