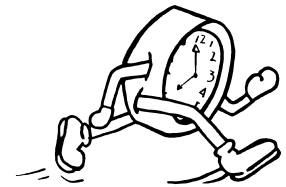


PJS

Personal Job Scheduler



Messages and Codes

Release 3.1

Draft 2/9/05

Personal Job Scheduler (PJS) 3.1.0 was released for distribution in February, 2005.

MVS is a trademark and IBM®, OS/390®, RACF®, and z/OS® are registered trademarks of International Business Machines Corporation.

Copyright © Northrop Grumman, 1990, 2005. All rights reserved.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with the Invariant Sections being “PJS Software Copyright and License” and “GNU General Public License”, with no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled “GNU Free Documentation License”.

Table of Contents

About This Manual.....	v
Acknowledgments.....	v
PJS Software Copyright and License.....	v
Comments and Suggestions.....	v
Organization of This Manual.....	vi
Related Publications.....	vii
 Chapter 1: PJS Messages.....	 1
PJS Message Identifiers.....	1
How to Use Message Descriptions.....	1
Message Descriptions.....	2
 Chapter 2: PJS Abend Codes.....	 51
 GNU General Public License.....	 79
 GNU Free Documentation License.....	 85

About This Manual

This manual describes the messages and userabend codes that can be issued by the Personal Job Scheduler (PJS) Release 2.2. An explanation is given for when and why the message or userabend was produced, and any action required for the message.

Acknowledgments

PJS was written by Tim Henness

The original version of this manual was written by Matthew ??? and Tim Henness. Extensive revisions have been made by Tim Henness

PJS Software Copyright and License

All PJS software is **Copyright © Northrop Grumman, 1990, 2005. All rights reserved.**

PJS is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

PJS is distributed in the hope that it will be useful, but **WITHOUT ANY WARRANTY**; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

A copy of the GNU General Public License is included in the back of this book.

Comments and Suggestions

The author welcomes your comments and suggestions. He can be contacted at:

Tim Henness
Northrop Grumman IT
Internal Information Services
Bldg. 521-2
4101 Washington Ave.
Newport News, VA 23607

E-mail: Tim.Henness@ngc.com

Organization of This Manual

This manual contains the following chapters:

- Chapter 1** “PJS Messages” describes the messages that can be produced by PJS.
- Chapter 2** “PJS Abend Codes” describes the userabend codes that can be produced by PJS.

Related Publications

PJS Documentation

Personal Job Scheduler (PJS) User's Guide

Personal Job Scheduler (PJS) Installation Guide

OS/390 Documentation

MVS System Messages, Volume 1 (ABA-ASA), GC27-1784

MVS System Messages, Volume 2 (ASB-ERB), GC27-1785

MVS System Messages, Volume 3 (EWX-IEB), GC27-1786

MVS System Messages, Volume 4 (IEC-IFD), GC27-1787

MVS System Messages, Volume 5 (IGD-IZP), GC27-1788

MVS System Codes, GC28-1780

MVS Programming: Authorized Assembler Services Guide, GC28-1763

MVS Programming: Authorized Assembler Services Reference, Volume 3
(LLA-SSU), GC28-1766

DFSMS Macro Instructions for Data Sets, SC26-7337

z/OS Documentation

MVS System Messages, Volume 1 (ABA-AOM), SA22-7631

MVS System Messages, Volume 2 (ARC-ASA), SA22-7632

MVS System Messages, Volume 3 (ASB-BPX), SA22-7633

MVS System Messages, Volume 4 (CBD-DMO), SA22-7634

MVS System Messages, Volume 5 (EDG-GFS), SA22-7635

MVS System Messages, Volume 6 (GOS-IEA), SA22-7636

MVS System Messages, Volume 7 (IEB-IEE), SA22-7637

MVS System Messages, Volume 8 (IEF-IGD), SA22-7638

MVS System Messages, Volume 9 (IGF-IWM), SA22-7639

MVS System Messages, Volume 10 (IXZ-IZP), SA22-7640

MVS System Codes, GC28-1780

DRAFT 2/9/05

MVS Programming: Authorized Assembler Services Guide, SA22-7608

MVS Programming: Authorized Assembler Services Reference, Volume 3
(LLA-SSU), SA22-7611

DFSMS Macro Instructions for Data Sets, SC26-7408

Chapter 1: PJS Messages

PJS Message Identifiers

PJS messages begin with an identifier, with the following format:

PJS*nnnns*

where ‘*nnnn*’ is a 4-digit message identifier, and ‘*s*’ is one of the following message severity codes:

- I** for **INFORMATION**. This is an informative message. There is no error, and no action is required.
- W** for **WARNING**. A condition that may lead to incorrect or unexpected results has been detected.
- E** for **ERROR**. An error that may terminate processing has occurred.

How to Use Message Descriptions

The following message descriptions contain the following information:

Message Text	Shows the message text. Fixed text is shown in boldface . Variable fields are shown in <i>italics</i> .
Explanation	Describes what caused the message to be issued.
Variable Fields	Describes contents of the variable fields in the message.
User Action	Describes what action is to be taken by the PJS user for this message.
Operator Action	Describes what action is to be taken by the system operator when the message is encountered.
Administrator Action	Describes what action is to be taken by the PJS Administrator. The PJS Administrator is the person (or persons) responsible for the installation and administration of PJS.

Message Descriptions

PJS0101I PJS OPTIONS IN EFFECT:

Explanation:

This message begins the PJS Options Table List in the SYSOUT data set from the PJS System Task. Message PJS0102I will follow.

PJS0102I *option = value*

Explanation:

The PJS option indicated has been assigned the indicated value.

Variable Fields:

option is the name of the option parameter on the PJSOPT macro.

value is the value assigned to the option parameter.

PJS0103E PJS OPTION *option* **IS INVALID**

Explanation:

The PJS option indicated has been assigned an invalid value. The value set is indicated by the preceding PJS0102I message.

Variable Fields:

option is the name of the option parameter on the PJSOPT macro.

System Action:

PJS execution is terminated.

Operator Action:

Report this message to the PJS Administrator.

Administrator Action:

Correct the options in the PJSOPT module and reinstall it. If the PJSOPTX Installation Options Exit is being used, make sure it is modifying the options correctly.

PJS1101I *jobname* - **PJS release** **SYSTEM TASK INITIALIZATION BEGINNING**

Explanation:

The PJS System Task is beginning initialization.

Variable Fields:

jobname is the jobname of the PJS Started Task.

release is the release and modification level of the PJS software.

PJS1102I *jobname* - **PJS SYSTEM TASK INITIALIZATION COMPLETE**

Explanation:

The PJS System Task initialization has completed.

Variable Fields:

jobname is the jobname of the PJS Started Task.

PJS1103E *jobname* - PJS SYSTEM TASK INITIALIZATION FAILED

Explanation:

The PJS System Task initialization has failed.

Variable Fields:

jobname is the jobname of the PJS Started Task.

System Action:

The PJS System Task is terminated.

Operator Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying messages for more information.

PJS1104I *jobname* - PJS SYSTEM TASK PARAMETERS - *parms*

Explanation:

The PJS System Task has been executed with the indicated parameters.

Variable Fields:

jobname is the jobname of the PJS Started Task.

parms is the execution parameter string.

PJS1105E *jobname* - INVALID PJS SYSTEM TASK PARAMETERS SPECIFIED

Explanation:

The PJS System Task execution parameters are invalid.

Variable Fields:

jobname is the jobname of the PJS Started Task.

System Action:

The PJS System Task is terminated.

Operator Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying messages for more information. Correct the PJS System Task execution parameters.

PJS1106E *jobname* - PJS SYSTEM TASK ALREADY STARTED

Explanation:

The operator tried to start the PJS System Task when the PJS System Task was already started in another address space.

Variable Fields:

jobname is the jobname of the task.

System Action:

The second copy of the PJS System Task is terminated.

Operator Action:

No action required at his time. In the future, check to see if the PJS System Task is already running before entering the start command.

PJS1107W *jobname* - PJS SYSTEM TASK NOT APF AUTHORIZED, TSO MESSAGES WILL NOT BE SENT

Explanation:

The PJS System Task is not running with APF authorization. As a result, the TSO SEND command cannot be issued to send messages to the TSO user.

Variable Fields:

jobname is the jobname of the PJS Started Task.

System Action:

PJS processing will continue, but the PJS System Task will not send messages to users.

Operator Action:

Report this message to the PJS Administrator.

Administrator Action:

Refer to the *PJS Installation Guide* and your MVS system documentation for information on how to APF authorize PJS.

PJS1108W *jobname* - PJS SYSTEM TASK WAS DOWN FOR *ddd* DAYS, *hh* HOURS, *mm* MINUTES

Explanation:

The PJS System Task was down for the indicated amount of time. This message is produced only if this time exceeds the DOWNWRN parameter of the PJS Installation Options Module (PJSOPT).

Variable Fields:

jobname is the jobname of the PJS Started Task.

ddd is the number of days the PJS System Task was down.

hh is the number of hours the PJS System Task was down.

mm is the number of minutes the PJS System Task was down.

System Action:

The PJS System Task will issue message PJS1109 to prompt the operator for the PJS Request Queue recovery option.

Operator Action

Check the time the PJS System Task was actually down. If the time indicated in the message is larger than the actual time the PJS System was down, notify the PJS Administrator.

PJS1109I *jobname* - ENTER REQUEST QUEUE RECOVERY OPTION ('NOHOLD', 'HOLD', OR 'HOLDALL')

Explanation:

This message is issued as a WTOR to prompt the operator for the PJS Request Queue recovery option.

Variable Fields:

jobname is the jobname of the PJS Started Task.

System Action:

The PJS System Task will wait for the operator to entry the PJS Request Queue recovery option.

Operator Action:

If you are certain that the PJS Request Queue has not been restored from a backup, reply NOHOLD. If you are not sure, report this message to the PJS Administrator.

Administrator Action:

If the PJS Request Queue has been restored from a backup, PJS cannot determine which job requests were submitted between the date and time of the backup and the current date and time. To hold any job requests that were scheduled for execution between the backup date and time and the current date and time, reply HOLD. To hold all job requests, reply HOLDALL. Held job requests will not be processed until they are enabled by the user.

If NOHOLD is specified, normal PJS processing is performed, which is to submit all jobs whose Next Run date and Time is past, unless the Window Time is exceeded.

PJS1110I *jobname* - REQUEST QUEUE RECOVERY OPTION SPECIFIED - *option*

Explanation:

The indicated PJS Request Queue recovery option has been specified.

Variable Fields:

jobname is the jobname of the PJS Started Task.

option is PJS Request Queue recovery option.

PJS1111E *jobname* - INVALID REQUEST QUEUE RECOVERY OPTION SPECIFIED

Explanation:

The PJS Request Queue recovery option specified is invalid.

Variable Fields:

jobname is the jobname of the PJS Started Task.

System Action:

The PJS System Task will issue message PJS1109 to prompt the operator for the PJS Request Queue recovery option.

Operator Action

Respond to message PJS1109 with the correct PJS Request Queue recovery option.

PJS1112W SYSTEM FAILURE OCCURRED DURING SUBMIT FOR JOB REQUEST *req-id*, SUBMIT STATUS CANNOT BE DETERMINED

Explanation:

When the PJS System Task was started, a job request in SUBMIT status was found. This occurs when the PJS System Task fails while trying to submit a job.

Variable Fields:

req-id is the request-ID for the processed job request.

System Action:

Because PJS doesn't know whether the job was submitted, it places the job request in HOLD status.

User Action:

Check to see whether the job ran. Modify the job request to reenable it. It may be necessary to recalculate the Next Run Date and Time.

PJS1113W JOB REQUEST *req-id* HELD BY PJS REQUEST QUEUE RECOVERY

Explanation:

When the PJS System Task was started, a PJS Request Queue recovery option of **HOLD** or **HOLDALL** was specified. If the **HOLD** option was specified, this job request had a Next Run Date and Time that was past, and the PJS System Task could not determine if the job request was submitted before the PJS Request Queue was lost. If the **HOLDALL** option was specified, all job requests are held, regardless of the Next Run date and Time.

Variable Fields:

req-id is the request-ID for the processed job request.

System Action:

The job request is placed in **HOLD** status.

User Action:

Check to see whether the job ran. Modify the job request to reenable it. It may be necessary to recalculate the Next Run Date and Time, or manually post or reset events.

PJS1201I *jobname* - **PJS SYSTEM TASK ENDING**

Explanation:

The PJS System Task is ending execution.

Variable Fields:

jobname is the jobname of the PJS Started Task.

PJS1202E **OPERATOR COMMUNICATION SUBTASK ABENDED, CODE=***code*

Explanation:

The operator communication subtask of the PJS System Task has abended. The PJS System Task is terminated. In most cases, a PJS internal error has generated this message.

Variable Fields:

code is the abend code for the operator communication subtask. System abend codes are shown as 'Sxxx', where 'xxx' is the hexadecimal system abend code. User abend codes are shown as 'Unnnn', where 'nnnn' is the decimal abend code.

System Action:

The PJS System Task is terminated.

Operator Action:

Restart the PJS System Task. Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying messages for more information. For an explanation of user abend codes, refer to **Chapter 2: PJS Abend Codes**. For an explanation of system abend codes, refer to the *MVS System Codes* manual for your system.

PJS1203E **OPERATOR COMMUNICATION SUBTASK QEDIT ERROR, RC=***rc*,
REF=*ref-id*

Explanation:

A QEDIT macro issued by the operator communication subtask of the PJS System Task has failed. In most cases, a PJS internal error has generated this message.

Variable Fields:

rc is the QEDIT macro return code.

ref-id is a PJS function reference-ID.

System Action:

The PJS System Task is terminated.

Operator Action:

Restart the PJS System Task. Report this message to the PJS Administrator.

Administrator Action:

Refer to the *MVS Authorized Assembler Services Reference* manual for your system for an explanation of the QEDIT return code.

PJS1301I *jobname* - **READY FOR COMMAND**

Explanation:

The PJS System Task is now accepting operator commands.

Variable Fields:

jobname is the jobname of the task.

System Action:

PJS operator commands will be processed as they are entered.

Operator Action:

For information on operator commands, refer to the *PJS Installation Guide*.

PJS1302I *jobname* - *command*

Explanation:

The PJS System Task received the indicated command.

Variable Fields:

jobname is the jobname of the task.

command is the command text PJS received.

PJS1303E *jobname* - **INVALID COMMAND**

Explanation:

The last command received by the PJS System Task was not a valid command.

Variable Fields:

jobname is the jobname of the task.

System Action:

The command is ignored.

Operator Action:

Reenter the command, using the correct syntax. For information on operator commands, refer to the *PJS Installation Guide*.

PJS1501I **JOB SUBMIT FOR REQUEST** *req-id* **SUCCESSFUL**

Explanation:

The job submission for the indicated Job Request was successful.

Variable Fields:

req-id is the request-ID for the processed job request.

PJS1502E JOB SUBMIT FOR REQUEST *req-id* FAILED

Explanation:

The job submission for the indicated Job Request failed. A previous message should indicate the reason for the failure. The Job Request is placed in ERROR status.

Variable Fields:

req-id is the request-ID for the processed job request.

System Action:

The job request is placed in **ERROR** status.

User Action:

Check any accompanying messages for more information. Modify the job request to reenable it. It may be necessary to recalculate the Next Run Date and Time.

PJS1503E JOB SUBMIT SUBTASK ABENDED, CODE=*code*, REASON=*reason*

Explanation:

The job submit subtask abended while attempting to submit a job. Message PJS1502E will follow, indicating the Job Request being processed. It is possible that the job was actually submitted before the subtask abended in clean-up processing.

Variable Fields:

code is the abend code for the job submission subtask. System abend codes are shown as 'Sxxx', where 'xxx' is the hexadecimal system abend code. User abend codes are shown as 'Unnnn', where 'nnnn' is the decimal abend code.

reason is the reason code associated with the abend code.

System Action:

The job request is placed in **ERROR** status.

User Action:

Check any accompanying messages for more information. For an explanation of user abend codes, refer to **Chapter 2: PJS Abend Codes**. For an explanation of system abend codes, refer to the *MVS System Codes* manual for your system. Try to determine the cause of the error and fix the problem. Modify the job request to reenable it. It may be necessary to recalculate the Next Run Date and Time.

PJS1504E JOB SUBMIT WINDOW FOR REQUEST *req-id* MISSED

Explanation:

The PJS System Task could not submit the indicated job request within its job submit window time.

Variable Fields:

req-id is the request-ID for the processed job request.

System Action:

The action specified by the Window Option of job request is performed.

User Action:

Check the following message (PJS1505I, PJS1506I, or PJS1507I) for the disposition of the Job Request. It may be necessary to modify the job request to reenale it. It may also be necessary to recalculate the Next Run Date and Time.

PJS1505I JOB REQUEST *req-id* HAS BEEN RESCHEDULED, NEXT RUN DATE/TIME = *date time*

Explanation:

The indicated job request has been rescheduled to be submitted again at the indicated date and time.

Variable Fields:

req-id is the request-ID for the processed job request.

date is the next scheduled run date.

time is the next scheduled run time.

System Action:

The job request is placed in the **WAIT** status.

PJS1506I JOB REQUEST *req-id* HAS BEEN COMPLETED

Explanation:

The last scheduled run date and time for the indicated job request is past. The job will not be submitted again unless the job request is modified by the user.

Variable Fields:

req-id is the request-ID for the processed job request.

System Action:

The job request is placed in the **COMPLETE** status.

PJS1507I JOB REQUEST *req-id* HAS BEEN DISABLED

Explanation:

The indicated job request has been placed in the **DISABLED** status. A previous message should indicate the reason for the failure.

Variable Fields:

req-id is the request-ID for the processed job request.

System Action:

The job request is placed in the **DISABLED** status.

User Action:

Check any accompanying messages for more information. Modify the job request to reenable it. It may be necessary to recalculate the Next Run Date and Time.

PJS1508I JOB REQUEST *req-id* HAS BEEN DELAYED

Explanation:

The indicated job request has been delayed by a temporary submit error.

Variable Fields:

req-id is the request-ID for the processed job request.

System Action:

The job request is placed in the **WAIT** status. The job submit will be attempted again later.

User Action:

This message does not necessarily indicate an error. Check any accompanying messages for the reason the submit was delayed. This is usually caused by another address space holding an exclusive enqueue on the JCL data set for the job request. If possible, release the enqueue.

PJS1509I JOB REQUEST *req-id* HAS BEEN DISABLED IN ERROR

Explanation:

The indicated job request has been placed in the **ERROR** status. A previous message should indicate the reason for the failure.

Variable Fields:

req-id is the request-ID for the processed job request.

System Action:

The job request is placed in **ERROR** status.

User Action:

Check any accompanying messages for more information. Modify the job request to reenable it. It may be necessary to recalculate the Next Run Date and Time.

PJS1601E JOB REQUEST *req-id* VARIABLE *var-id* NOT FOUND

Explanation:

The indicated global variable could not be found. The global variable was specified (either directly or indirectly) as a variable substitution replacement value in the indicated job request.

Variable Fields:

req-id is the request-ID for the processed job request.

var-id is the variable-ID of the global variable that could not be found.

System Action:

The job submit is terminated, and the job request is placed in ther **ERROR** status.

User Action:

Add the missing global variable to the PJS Request Queue, or correct the replacement value specified by the job request variable.

PJS1602E JOB REQUEST *req-id* VARIABLE *var-id* SPECIFIES A RECURSIVE VALUE

Explanation:

The indicated global variable specifies a replacement value of another global variable, which in turn specifies another global variable, such that one of the global variables indirectly refers to itself.

Variable Fields:

req-id is the request-ID for the processed job request.

var-id is the variable-ID of the global variable specified by the job request variable.

System Action:

The job submit is terminated, and the job request is placed in ther **ERROR** status.

User Action:

Correct the global variables to resolve to either a literal value or a dynamic value.

PJS1603E JOB REQUEST *req-id* SPECIFIES AN INVALID DYNAMIC VARIABLE NAME *dyn-name*

Explanation:

The indicated job request specifies an invalid dynamic variable name. The dynamic variable may be specified either directly by a job request variable, or indirectly by a global variable specified by a job request variable.

Variable Fields:

req-id is the request-ID for the processed job request.

dyn-name is the dynamic value name specified.

System Action:

The job submit is terminated, and the job request is placed in ther **ERROR** status.

User Action:

Correct the dynamic value name.

PJS1702E OPEN ERROR FOR INTERNAL READER, RC=*rc*, REQID=*req-id*

Explanation:

An OPEN macro for the internal reader failed.

Variable Fields:

rc is the OPEN macro return code.

req-id is the request-ID for the processed job request.

System Action:

The job submit is terminated, and the job request is placed in ther **ERROR** status.

User Action:

Report this message to the PJS Administrator. Modify the job request to reenable it. It may be necessary to recalculate the Next Run Date and Time.

Administrator Action:

Check any accompanying messages for more information. Refer to the *DFSMS Macro Instructions for Data Sets* manual for your system for an explanation of the OPEN return code.

PJS1703E CLOSE ERROR FOR INTERNAL READER, RC=*rc*, REQID=*req-id*

Explanation:

An CLOSE macro for the internal reader failed. The submit may or may not be complete.

Variable Fields:

rc is the CLOSE macro return code.

req-id is the request-ID for the processed job request.

System Action:

The job submit is terminated, and the job request is placed in ther **ERROR** status.

User Action:

Report this message to the PJS Administrator. Check to see of the submit was completed successfully. Modify the job request to reenable it. It may be necessary to recalculate the Next Run Date and Time.

Administrator Action:

Check any accompanying messages for more information. Refer to the *DFSMS Macro Instructions for Data Sets* manual for your system for an explanation of the CLOSE return code.

PJS1704E ACCESS TO SUBMIT FUNCTION DENIED, REQID=*req-id*

Explanation:

The PJS Installation Security Exit denied access to the PJS System Task Submit function.

Variable Fields:

req-id is the request-ID for the processed job request.

System Action:

The job submit is terminated, and the job request is placed in ther **ERROR** status.

User Action:

Report this message to the PJS Administrator. Modify the job request to reenale it.

Administrator Action:

Check that the PJS Installation Security Exit is functioning correctly.

PJS1705E JOB SUBMISSION FOR REQUEST *req-id* PURGED BY INSTALLATION

Explanation:

The PJS Installation Submit Exit requested that the job for the indicated request-ID be purged before it could be executed.

Variable Fields:

req-id is the request-ID for the processed job request.

System Action:

The job submit is terminated, and the job request is placed in ther **ERROR** status.

User Action:

Report this message to the PJS Administrator. Modify the job request to reenale it.

Administrator Action:

Check that the PJS Installation Submit Exit is functioning correctly.

PJS1706E INSTALLATION SUBMIT EXIT ERROR, RC=*rc*, REQID=*req-id*

Explanation:

The PJS Installation Submit Exit returned with an invalid return code.

Variable Fields:

rc is the invalid return code generated by the exit.

req-id is the request-ID for the processed job request.

System Action:

The job submit is terminated, and the job request is placed in ther **ERROR** status.

Operator Action:

Report this message to the PJS Administrator.

User Action:

Report this message to the PJS Administrator. Modify the job request to reenale it.

Administrator Action:

Correct the PJS Installation Submit Exit to return with a valid return code.

PJS2001E ACCESS TO FUNCTION DENIED

Explanation:

The PJS Installation Security Exit (PJSSECX) has denied access to the requested TSO command.

System Action:

The command is terminated.

User Action:

Report this message to the PJS Administrator.

Administrator Action:

Check that the PJS Installation Submit Exit is functioning correctly.

PJS2002E INVALID SUBCOMMAND

Explanation:

The subcommand entered is invalid.

System Action:

The subcommand is ignored.

User Action:

Reenter the subcommand, using the correct syntax. For information on valid subcommands, refer to the *PJS User's Guide*.

PJS2003E PARAMETERS INVALID FOR SUBCOMMAND

Explanation:

Parameters were entered for a subcommand that does not accept any parameters.

System Action:

The subcommand is ignored.

User Action:

Reenter the subcommand, using the correct syntax. For information on valid subcommands, refer to the *PJS User's Guide*.

PJS2101I JOB REQUEST *req-id* SUCCESSFULLY ADDED

Explanation:

The indicated job request has been successfully added.

Variable Fields:

req-id is the request-ID for the added job request.

PJS2102I JOB REQUEST *req-id* SUCCESSFULLY MODIFIED

Explanation:

The indicated job request has been successfully modified.

Variable Fields:

req-id is the request-ID for the modified job request.

PJS2103I JOB REQUEST *req-id* SUCCESSFULLY DELETED

Explanation:

The indicated job request has been successfully deleted.

Variable Fields:

req-id is the request-ID for the deleted job request.

PJS2104W NO CHANGES MADE, JOB REQUEST NOT MODIFIED

Explanation:

The job request modify command did not specify any changes.

System Action:

The job request record is not modified.

User Action:

If you wish to modify the job request, reenter the job request modify command, then enter one or more subcommands to change the job request. For information on valid subcommands, refer to the *PJS User's Guide*.

PJS2121E JOB REQUEST *req-id* NOT FOUND

Explanation:

The indicated job request does not exist.

Variable Fields:

req-id is the request-ID for the specified job request.

System Action:

The command is ignored.

User Action:

Correct the request-ID and reenter the command.

PJS2122E NO JOB REQUESTS FOR SPECIFIED OWNER

Explanation:

The PJS List Job Requests TSO command (PJREQST) specified an owner-ID that has no job requests on the PJS Request Queue.

PJS2123E NO JOB REQUESTS IN REQUEST QUEUE

Explanation:

The PJS List Job Requests TSO command (PJREQST) specified that all job requests were to be listed, but there are no job requests on the PJS Request Queue.

PJS2124E ACCESS TO JOB REQUEST *req-id* DENIED BY INSTALLATION

Explanation:

The PJS Installation Security Exit has denied access to the specified job request.

Variable Fields:

req-id is the request-ID for the specified job request.

System Action:

The command or subcommand is ignored.

User Action:

Report this message to the PJS Administrator.

Administrator Action:

Check that the PJS Installation Submit Exit is functioning correctly.

PJS2125E JOB REQUEST *req-id* EXCEEDS MAXIMUM SIZE

Explanation:

PJS tried to add or modify a job request. The total length of the job request record exceeds the maximum PJS Request Queue record size of 4084 bytes.

Variable Fields:

req-id is the Request-ID for the job request being added or modified.

System Action:

The subcommand is terminated.

User Action:

Specify fewer calendars, events, or job request variables for the job request.

PJS2126E JOB REQUEST *req-id* IS IN SUBMIT STATUS

Explanation:

PJS tried to delete or modify a job request that is currently in SUBMIT status. The request is either being submitted or the PJS System Task abnormally terminated during submit processing for the job request and has not been restarted.

Variable Fields:

req-id is the request-ID for the specified job request.

System Action:

The command or subcommand is ignored.

User Action:

If the job is being submitted, wait until PJS finishes the submission. If the PJS System Task is down, wait for the System Task to be restarted (setting the status to HOLD). Then reenter the modify or delete command. If this message persists, report it to the PJS Administrator.

Administrator Action:

If the PJS System Task abnormally terminated, check for the reason for the termination. Have the PJS System Task restarted.

PJS2127E NO REQUEST ID'S AVAILABLE FOR OWNER *owner-id*

Explanation:

An attempt was made to add a job request with a generated request name, but the owner-ID already has 999 job requests with numeric request names. No generated request numbers are available.

Variable Fields:

owner-id is the owner-ID for the job request to be added.

System Action:

The add command is ignored.

User Action:

Specify an alphabetic request name, or delete one or more existing job requests with a numeric job request name.

PJS2128E JOB REQUEST *req-id* ALREADY EXISTS

Explanation:

The indicated job request was to be added, but another job request with the same request-ID already exists.

Variable Fields:

req-id is the request-ID for the specified job request.

System Action:

The command is ignored.

User Action:

Reenter the job request add command, specifying a different request-ID.

PJS2141E JCL DATA SET MUST BE SPECIFIED

Explanation:

The job request being added does not specify a JCL data set.

System Action:

The END subcommand is ignored.

User Action:

Specify a JCL data set with the SET JCLDSN subcommand. For information on valid subcommands, refer to the *PJS User's Guide*.

PJS2142E START DATE/TIME MUST BE SPECIFIED

Explanation:

The job request being added does not specify a start date and time.

System Action:

The END subcommand is ignored.

User Action:

Specify a start date and time with the SET STRTDATE and SET STRTTIME subcommand. For information on valid subcommands, refer to the *PJS User's Guide*.

PJS2201I CALENDAR *cal-id* SUCCESSFULLY ADDED

Explanation:

The indicated calendar has been successfully added.

Variable Fields:

cal-id is the calendar-ID for the added calendar.

PJS2202I CALENDAR *cal-id* SUCCESSFULLY MODIFIED

Explanation:

The indicated calendar has been successfully modified.

Variable Fields:

cal-id is the calendar-ID for the modified calendar.

PJS2203I CALENDAR *cal-id* SUCCESSFULLY DELETED

Explanation:

The indicated calendar has been successfully deleted.

Variable Fields:

cal-id is the calendar-ID for the deleted calendar.

PJS2204W NO CHANGES MADE, CALENDAR NOT MODIFIED

Explanation:

The calendar modify command did not specify any changes.

System Action:

The calendar record is not modified.

User Action:

If you wish to modify the calendar, reenter the calendar modify command, specifying one or more parameters to change the calendar. For information on valid parameters, refer to the *PJS User's Guide*.

PJS2221E CALENDAR *cal-id* NOT FOUND

Explanation:

The indicated calendar does not exist.

Variable Fields:

cal-id is the calendar-ID for the specified calendar.

System Action:

The command is ignored.

User Action:

Correct the calendar-ID and reenter the command.

PJS2222E NO CALENDARS FOR SPECIFIED OWNER

Explanation:

The PJS List Calendars TSO command (PJCALIST) specified an owner-ID that has no calendars on the PJS Request Queue.

PJS2223E NO CALENDARS IN REQUEST QUEUE

Explanation:

The PJS List Calendars TSO command (PJCALIST) specified that all calendars were to be listed, but there are no calendars on the PJS Request Queue.

PJS2224E ACCESS TO CALENDAR *cal-id* DENIED BY INSTALLATION

Explanation:

The PJS Installation Security Exit has denied access to the specified calendar.

Variable Fields:

cal-id is the calendar-ID for the specified calendar.

System Action:

The command or subcommand is ignored.

User Action:

Report this message to the PJS Administrator.

Administrator Action:

Check that the PJS Installation Submit Exit is functioning correctly.

PJS2225E CALENDAR *cal-id* EXCEEDS MAXIMUM SIZE

Explanation:

PJS tried to add or modify a calendar. The total length of the calendar record exceeds the maximum PJS Request Queue record size of 4084 bytes.

Variable Fields:

cal-id is the calendar-ID for the calendar being added or modified.

System Action:

The command is ignored.

User Action:

Eliminate the last dates selected by the calendar.

PJS2227E CALENDAR *cal-id* ALREADY EXISTS

Explanation:

An attempt was made to add a calendar with a calendar-ID that already exists on the PJS Request Queue.

Variable Fields:

cal-id is the calendar-ID for the specified calendar.

System Action:

The command is ignored.

User Action:

Reenter the calendar add command, specifying a different calendar-ID.

PJS2301I EVENT *event-id* SUCCESSFULLY POSTED

Explanation:

The indicated event has been posted. The individual job request events will be posted by the PJS System Task on its next PJS Request Queue Scan. Until then, the event record will be in POST PENDING status.

Variable Fields:

event-id is the event-ID for the posted event.

PJS2302I EVENT *event-id* SUCCESSFULLY RESET

Explanation:

The indicated event has been reset. The individual job request events will be reset by the PJS System Task on its next PJS Request Queue Scan. Until then, the event record will be in RESET PENDING status.

Variable Fields:

event-id is the event-ID for the reset event.

PJS2321E EVENT *event-id* NOT FOUND

Explanation:

The indicated event does not exist.

Variable Fields:

event-id is the event-ID for the specified event.

System Action:

The command is ignored.

User Action:

Correct the event-ID and reenter the command.

PJS2322E NO EVENTS FOR SPECIFIED OWNER

Explanation:

The PJS List Events TSO command (PJEVLIST) specified an owner-ID that has no events on the PJS Request Queue.

PJS2323E NO EVENTS IN REQUEST QUEUE

Explanation:

The PJS List Events TSO command (PJEVLIST) specified that all events were to be listed, but there are no events on the PJS Request Queue.

PJS2324E ACCESS TO EVENT *event-id* DENIED BY INSTALLATION

Explanation:

The PJS Installation Security Exit has denied access to the specified event.

Variable Fields:

event-id is the event-ID for the specified event.

System Action:

The command or subcommand is ignored.

User Action:

Report this message to the PJS Administrator.

Administrator Action:

Check that the PJS Installation Submit Exit is functioning correctly.

PJS2401I VARIABLE RECORD *var-id* SUCCESSFULLY ADDED

Explanation:

The indicated global variable has been successfully added.

Variable Fields:

var-id is the variable-ID for the added global variable.

PJS2402I VARIABLE RECORD *var-id* SUCCESSFULLY MODIFIED

Explanation:

The indicated global variable has been successfully modified.

Variable Fields:

var-id is the variable-ID for the modified global variable.

PJS2403I VARIABLE RECORD *var-id* SUCCESSFULLY DELETED

Explanation:

The indicated global variable has been successfully deleted.

Variable Fields:

var-id is the global variable-ID for the deleted global variable.

PJS2404W NO CHANGES MADE, VARIABLE RECORD NOT MODIFIED

Explanation:

The global variable modify command did not specify any changes.

System Action:

The global variable is not modified.

User Action:

If you wish to modify the global variable, reenter the global variable modify command, specifying one or more parameters to change the global variable. For information on valid parameters, refer to the *PJS User's Guide*.

PJS2421E VARIABLE RECORD *var-id* NOT FOUND

Explanation:

The indicated global variable does not exist.

Variable Fields:

var-id is the variable-ID for the specified global variable.

System Action:

The command is ignored.

User Action:

Correct the global variable-ID and reenter the command.

PJS2422E NO VARIABLE RECORDS FOR SPECIFIED OWNER

Explanation:

The PJS List Global Variables TSO command (PJVARLST) specified an owner-ID that has no global variables on the PJS Request Queue.

PJS2423E NO VARIABLE RECORDS IN REQUEST QUEUE

Explanation:

The PJS List Global Variables TSO command (PJVARLST) specified that all global variables were to be listed, but there are no global variables on the PJS Request Queue.

PJS2424E ACCESS TO VARIABLE RECORD *var-id* DENIED BY INSTALLATION

Explanation:

The PJS Installation Security Exit has denied access to the specified global variable.

Variable Fields:

var-id is the variable-ID for the specified global variable.

System Action:

The command or subcommand is ignored.

User Action:

Report this message to the PJS Administrator.

Administrator Action:

Check that the PJS Installation Submit Exit is functioning correctly.

PJS2427E VARIABLE RECORD *var-id* ALREADY EXISTS

Explanation:

An attempt was made to add a global variable with a variable-ID that already exists.

Variable Fields:

var-id is the variable-ID for the specified global variable.

System Action:

The command is ignored.

User Action:

Reenter the global variable add command, specifying a different variable-ID.

PJS3011E CONFLICTING END DATE AND TIME PARAMETERS SPECIFIED

Explanation:

Either ENDDATE was specified with NOENDTIME, or NOENDDATE was specified with ENDTIME.

System Action:

The command or subcommand is ignored.

User Action:

Reenter the command without one of the conflicting parameters.

PJS3012E END TIME SPECIFIED FOR JOB REQUEST WITHOUT END DATE

Explanation:

The ENDTIME was specified for a job request without an ENDDATE.

System Action:

The command or subcommand is ignored.

User Action:

Reenter the command either including the ENDDATE parameter, or without the ENDTIME parameter.

PJS3013E END DATE/TIME IS LESS THAN START DATE/TIME

Explanation:

The End Date and Time specified for the Job Request is less than the Start Date and Time.

System Action:

The command or subcommand is ignored.

User Action:

Specify an End Date and Time that is greater than the Start Date and Time.

PJS3014E NEXT RUN DATE/TIME EXCEEDS END DATE/TIME, OR IS ALREADY PAST

Explanation:

The Next Run Date and Time calculated for the job request exceeds the specified End Date and Time, or, for a one-time job request, is already past. The Next Run Date and Time is calculated from the Start Date and Start Time, the Frequency, and the current system date and time.

System Action:

The command or subcommand is ignored.

User Action:

Specify a Start Date and Time, End Date and Time, and Frequency that results in a Next Run Date and Time that is less than the End Date and Time.

PJS3015E CALENDAR HAS NO DATES

Explanation:

The calendar has no dates selected, or all of the selected dates are past.

System Action:

The command is ignored.

User Action:

Select one or more future dates.

PJS3021E REPLACEMENT VALUE IS REQUIRED

Explanation:

A replacement value was not specified for the global variable.

System Action:

The command is ignored.

User Action:

Specify a replacement value.

PJS3022E DYNAMIC VALUE NAME *dyn-name* IS INVALID

Explanation:

The global variable specifies an invalid dynamic value.

System Action:

The command is ignored.

User Action:

Specify a valid dynamic value name.

PJS3111E NOSAVE OPTION IS NOT ALLOWED BY INSTALLATION

Explanation:

NOSAVE was specified for a job request. When PJS was installed, the PJS Options Table was defined to allow only the SAVE option.

System Action:

The command or subcommand is ignored.

User Action:

Do not specify the NOSAVE option.

PJS3112E SAVE OPTION IS NOT ALLOWED BY INSTALLATION

Explanation:

SAVE was specified for a job request. When PJS was installed, the PJS Options Table was defined to allow only the NOSAVE option.

System Action:

The command or subcommand is ignored.

User Action:

Do not specify the SAVE option.

PJS3113E RESAVE SPECIFIED, BUT JCL IS NOT CURRENTLY SAVED

Explanation:

RESAVE was specified for a job request whose JCL was not previously saved.

System Action:

The command or subcommand is ignored.

User Action:

Specify the SAVE parameter to save a copy of the job JCL on the PJS JCL Spool.

PJS3114E SAVED JCL LIMIT EXCEEDED FOR OWNER *owner-id*

Explanation:

The total number of records saved in the PJS JCL Spool for this owner exceeds the maximum allowed by the installation.

Variable Fields:

owner-id is the owner-ID for which the JCL Spool Limit is exceeded.

System Action:

The command or subcommand is ignored.

User Action:

Reduce the total number of JCL records saved for all job requests in the PJS JCL Spool.

PJS3121E WINDOW OPTION SPECIFIED FOR JOB REQUEST WITHOUT WINDOW TIME

Explanation:

The Window Option parameter was specified for a job request without a Window Time.

System Action:

The command or subcommand is ignored.

User Action:

Specify a Window Time for the job request.

PJS3131E ADDCALENDAR PARAMETER SPECIFIED FOR NON-CALENDAR FREQUENCY JOB REQUEST

Explanation:

The ADDCALENDAR parameter was specified for the job request, but the job request does not specify calendar scheduling.

System Action:

The command or subcommand is ignored.

User Action:

Use the CALENDARS parameter to specify calendar scheduling.

PJS3132E DELCALENDAR PARAMETER SPECIFIED FOR NON-CALENDAR FREQUENCY JOB REQUEST

Explanation:

The DELCALENDAR parameter was specified for the job request being added or modified, but the job request does not specify calendar scheduling.

System Action:

The command or subcommand is ignored.

User Action:

Do not specify the DELCALENDAR parameter. Use the CALENDARS parameter to specify calendar scheduling.

PJS3133E MAXIMUM NUMBER OF CALENDARS FOR JOB REQUEST EXCEEDED

Explanation:

The job request specifies more calendars than is allowed. The current maximum is 3 calendars.

System Action:

The command or subcommand is ignored.

User Action:

Specify fewer calendars for the job request.

PJS3134E ALL CALENDARS DELETED FROM JOB REQUEST

Explanation:

The DELCALENDARS parameter attempted to delete all calendars for the job request.

System Action:

The command or subcommand is ignored.

User Action:

To delete all calendars from a job request, specify another frequency option.

PJS3141E MAXIMUM NUMBER OF CALENDARS FOR JOB REQUEST EXCEEDED

Explanation:

The job request specifies more calendars than is allowed. The current maximum is 3 calendars.

System Action:

The command or subcommand is ignored.

User Action:

Specify fewer calendars for the job request.

PJS3142E JOB REQUEST EVENT-ID IS REQUIRED

Explanation:

An event-ID was not specified for an EVENT subcommand.

System Action:

The subcommand is ignored.

User Action:

Specify an event-ID for the subcommand.

PJS3143E JOB REQUEST EVENT *event-id* NOT FOUND

Explanation:

The indicated job request event does not exist.

Variable Fields:

event-id is the event-ID for the specified job request event.

System Action:

The command or subcommand is ignored.

User Action:

Correct the event-ID and reenter the subcommand.

PJS3144E JOB REQUEST EVENT *event-id* ALREADY EXISTS

Explanation:

An attempt was made to add a job request event with an event-ID that has already been specified for the job request.

Variable Fields:

event-id is the event-ID for the specified job request event.

System Action:

The command or subcommand is ignored.

User Action:

Specify an event-ID that is not already specified for the job request.

PJS3151E DAY-OF-WEEK SCHEDULING SPECIFIED, BUT NO DAYS SELECTED

Explanation:

The job request being added or modified specifies the WEEKDAYS parameter, but does not include any days of the week.

System Action:

The command or subcommand is ignored.

User Action:

Include at least one day of the week parameter with the WEEKDAYS parameter.

PJS3161E MAXIMUM NUMBER OF VARIABLES FOR JOB REQUEST EXCEEDED

Explanation:

The job request specifies more job request variables than is allowed. The current maximum is 20.

System Action:

The command or subcommand is ignored.

User Action:

Specify fewer job request variables for the job request.

PJS3162E SEARCH END COLUMN IS LESS THAN SEARCH START COLUMN

Explanation:

The search end column for a job request variable is less than the search start column.

System Action:

The command or subcommand is ignored.

User Action:

Specify a search end column that is greater than or equal to the search start column.

PJS3163E SHIFT END COLUMN IS LESS THAN SEARCH END COLUMN

Explanation:

The shift end column for a job request variable is less than the search end column.

System Action:

The command or subcommand is ignored.

User Action:

Specify a shift end column that is greater than or equal to the search end column.

PJS3164E JOB REQUEST VARIABLE FOR TARGET STRING NOT FOUND

Explanation:

No job request variable exists for the specified target string.

System Action:

The command or subcommand is ignored.

User Action:

Specify the correct target string.

PJS4001E DYNAMIC *function* FAILED FOR *data-set*, *error-desc*

Explanation:

An SVC 99 (dynamic allocation) function failed.

Variable Fields:

function is the SVC 99 function being performed:

**ALLOCATION
DEALLOCATION**

data-set indicates the data set for which the function is being performed:

**REQUEST QUEUE
MESSAGE HISTORY LOG
JCL SPOOL <(member)>
dsname<(member)> (JCL data set)
INTERNAL READER
TEMPORARY DATA SET**

error-desc describes the error encountered. The individual errors are described in the following paragraphs:

DATA SET NOT FOUND

Explanation:

The data set could not be found. One of the following occurred:

- The name of the data set was misspelled.
- The data set was not cataloged.
- The data set could not be found on the volume indicated by the catalog.

User Action:

If the data set is the users JCL data set, check that the data set name has been entered correctly. If the data set is **REQUEST QUEUE**, **MESSAGE HISTORY LOG**, or **JCL SPOOL**, report this message to the PJS Administrator.

Administrator Action:

If the data set is **REQUEST QUEUE**, **MESSAGE HISTORY LOG**, or **JCL SPOOL**, check that the PJS Options Module specifies the correct data set name and that the data set has been properly allocated and initialized.

DATA SET IN USE

Explanation:

The data set could not be allocated, because another address space has allocated the data set with exclusive control.

System Action:

If this message is issued by the PJS System Task for the JCL data set, and the maximum retry limit has not been exceeded, the PJS system task will retry the submit later.

User Action:

Find which address space is holding the enqueue and, if possible, release the enqueue. If this cannot be done, wait until exclusive control is released.

UNIT NOT AVAILABLE

Explanation:

All JES internal readers are currently allocated. The PJS System Task is unable to submit jobs.

System Action:

All job submits will be delayed until an internal reader is available. The job submit retry option limit does not apply to this situation.

Operator Action:

Report this message to the PJS Administrator.

Administrator Action:

Increase the number of JES internal readers, or reduce the number of tasks using internal readers. PJS will resume job submission when an internal reader is available.

SVC 99 ERROR, RC=*rc*, ERROR=X'*error*', INFO=X'*info*'

Explanation:

An unanticipated error occurred in SVC 99 processing.

Variable Fields:

rc is an SVC 99 Return Code.

error is an SVC 99 Error Reason Code.

info is an SVC 99 Information Reason Code.

User Action:

Report this message to the PJS Administrator.

Operator Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the SVC99 return code, error reason code, and information reason code, refer to the *MVS Programming: Assembler Services Guide* manual for your system.

PJS4181E VSAM ACB ERROR FOR REQUEST QUEUE, FUNC=*func*, RC=*rc*, REASON=*reason*, REF=*ref-id*

Explanation:

A VSAM ACB based macro for the PJS Request Queue has failed.

Variable Fields:

func is the VSAM ACB function being performed.

rc is an ACB Macro Return Code (from register 15).

reason is an ACB Error Reason Code (from ACB).

ref-id is a PJS function reference-ID.

User Action:

Report this message to the PJS Administrator.

Operator Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the VSAM return code, and error reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS4182E VSAM RPL ERROR FOR REQUEST QUEUE, FUNC=*func*, RC=*rc*, REASON=*reason*, REF=*ref-id*

Explanation:

A VSAM RPL based macro for the PJS Request Queue has failed.

Variable Fields:

func is the VSAM RPL function being performed.

rc is the RPL Macro Return Code (from register 15).

reason is the RPL Error Reason Code (from RPLERRCD).

ref-id is a PJS function reference-ID.

User Action:

Report this message to the PJS Administrator.

Operator Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the VSAM return code, and error reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS4183E VSAM CONTROL BLOCK ERROR FOR REQUEST QUEUE, FUNC=*func*, RC=*rc*, REASON=*reason*, REF=*ref-id*

Explanation:

A VSAM control block manipulation macro for the PJS Request Queue has failed.

Variable Fields:

func is the VSAM control block function being performed.

rc is the Control Block Macro Return Code (from register 15).

reason is the Control Block Macro Error Reason Code (from register 0).

ref-id is a PJS function reference-ID.

User Action:

Report this message to the PJS Administrator.

Operator Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the VSAM return code, and error reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS4281E VSAM ACB ERROR FOR MESSAGE HISTORY LOG, FUNC=*func*, RC=*rc*, REASON=*reason*, REF=*ref-id*

Explanation:

A VSAM ACB based macro for the PJS Message History Log has failed.

Variable Fields:

func is the VSAM ACB function being performed.

rc is an ACB Macro Return Code (from register 15).

reason is an ACB Error Reason Code (from ACB).

ref-id is a PJS function reference-ID.

User Action:

Report this message to the PJS Administrator.

Operator Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the VSAM return code, and error reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS4282E VSAM RPL ERROR FOR MESSAGE HISTORY LOG, FUNC=*func*, RC=*rc*, REASON=*reason*, REF=*ref-id*

Explanation:

A VSAM RPL based macro for the PJS Message History Log has failed.

Variable Fields:

func is the VSAM RPL function being performed.

rc is the RPL Macro Return Code (from register 15).

reason is the RPL Error Reason Code (from RPLERRCD).

ref-id is a PJS function reference-ID.

User Action:

Report this message to the PJS Administrator.

Operator Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the VSAM return code, and error reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS4283E VSAM CONTROL BLOCK ERROR FOR MESSAGE HISTORY LOG, FUNC=*func*, RC=*rc*, REASON=*reason*, REF=*ref-id*

Explanation:

A VSAM control block manipulation macro for the PJS Message History Log has failed.

Variable Fields:

func is the VSAM control block function being performed.

rc is the Control Block Macro Return Code (from register 15).

reason is the Control Block Macro Error Reason Code (from register 0).

ref-id is a PJS function reference-ID.

User Action:

Report this message to the PJS Administrator.

Operator Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the VSAM return code, and error reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS4301E JCL MEMBER NOT FOUND, DSN=*dsname(member)*

Explanation:

The JCL member could not be found in the JCL data set specified.

Variable Fields:

dsname is the name of the JCL data set.

member is the name of the JCL member.

System Action:

If this message is issued by a TSO command, the command or subcommand is ignored. If this message is issued by the ISPF interface, the JCL data set specified is highlighted as an error, and must be corrected. If this message is issued by the PJS Sustem Task, the job submit is terminated, and the job request is placed in the **ERROR** status.

User Action:

Check that the correct data set name and member name is specified for the job request.

PJS4302E JCL DATA SET ACCESS DENIED, DSN=*dsname*

Explanation:

The PJS Installation Security Exit (PJSSECX) has denied access to the specified JCL data set.

Variable Fields:

dsname is the name of the JCL data set.

System Action:

If this message is issued by a TSO command, the command or subcommand is ignored. If this message is issued by the ISPF interface, the JCL data set specified is highlighted as an error, and must be corrected. If this message is issued by the PJS Sustem Task, the job submit is terminated, and the job request is placed in the **ERROR** status.

User Action:

Check that the correct data set name and member name is specified for the job request. If so, report this message to the PJS Administrator.

Administrator Action:

Check that the PJS Installation Submit Exit is functioning correctly.

**PJS4303E JCL MEMBER SPECIFIED, DATA SET IS NOT PARTITIONED,
DSN=*dsname(member)***

Explanation:

A member name was specified for the JCL data set, but the JCL data set is not a partitioned data set.

Variable Fields:

dsname is the name of the JCL data set.

member is the name of the JCL member.

System Action:

If this message is issued by a TSO command, the command or subcommand is ignored. If this message is issued by the ISPF interface, the JCL data set specified is highlighted as an error, and must be corrected. If this message is issued by the PJS Sustem Task, the job submit is terminated, and the job request is placed in the **ERROR** status.

User Action:

Check that the correct data set name and member name is specified for the job request. If so, remove the member name.

**PJS4304E JCL MEMBER NOT SPECIFIED, DATA SET IS PARTITIONED,
DSN=*dsname***

Explanation:

No member name was specified for the JCL data set, but the JCL data set is a partitioned data set.

Variable Fields:

dsname is the name of the JCL data set.

System Action:

If this message is issued by a TSO command, the command or subcommand is ignored. If this message is issued by the ISPF interface, the JCL data set specified is highlighted as an error, and must be corrected. If this message is issued by the PJS Sustem Task, the job submit is terminated, and the job request is placed in the **ERROR** status.

User Action:

Check that the correct data set name and member name is specified for the job request. If so, specify the name of the member of the JCL data set that contains the JCL.

PJS4305E JCL DATA SET HAS INVALID ORGANIZATION, DSN=*dsname*

Explanation:

The JCL data set organization (DSORG) is not sequential (PS) or partitioned (PO).

Variable Fields:

dsname is the name of the JCL data set.

System Action:

If this message is issued by a TSO command, the command or subcommand is ignored. If this message is issued by the ISPF interface, the JCL data set specified is highlighted as an error, and must be corrected. If this message is issued by the PJS Sustem Task, the job submit is terminated, and the job request is placed in the **ERROR** status.

User Action:

Check that the correct data set name and member name is specified for the job request. Specify a sequential or partitioned data set for the JCL data set.

PJS4306E JCL DATA SET HAS INVALID RECORD FORMAT, DSN=*dsname*

Explanation:

The JCL data set record format (RECFM) is not fixed (F, FB, or FBS).

Variable Fields:

dsname is the name of the JCL data set.

System Action:

If this message is issued by a TSO command, the command or subcommand is ignored. If this message is issued by the ISPF interface, the JCL data set specified is highlighted as an error, and must be corrected. If this message is issued by the PJS Sustem Task, the job submit is terminated, and the job request is placed in the **ERROR** status.

User Action:

Check that the correct data set name and member name is specified for the job request. Specify a JCL data set with 80-byte fixed length records.

PJS4307E JCL DATA SET HAS INVALID RECORD LENGTH, DSN=*dsname*

Explanation:

The JCL data set record length (LRECL) is not 80 bytes.

Variable Fields:

dsname is the name of the JCL data set.

System Action:

If this message is issued by a TSO command, the command or subcommand is ignored. If this message is issued by the ISPF interface, the JCL data set specified is highlighted as an error, and must be corrected. If this message is issued by the PJS Sustem Task, the job submit is terminated, and the job request is placed in the **ERROR** status.

User Action:

Check that the correct data set name and member name is specified for the job request. Specify a JCL data set with 80-byte fixed length records.

PJS4381E OPEN ERROR FOR JCL DATA SET, RC=*rc*, DSN=*dsname*

Explanation:

An OPEN macro for the JCL data set failed.

Variable Fields:

rc is an OPEN macro return code.

dsname is the name of the JCL data set.

User Action:

Check that the correct data set name and member name is specified for the job request. If so, report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the OPEN return code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS4382E CLOSE ERROR FOR JCL DATA SET, RC=*rc*, DSN=*dsname*

Explanation:

A CLOSE macro for the JCL data set failed.

Variable Fields:

rc is the CLOSE macro return code.

dsname is the name of the JCL data set.

User Action:

Check that the correct data set name and member name is specified for the job request. If so, report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the CLOSE return code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS4383E FIND ERROR FOR JCL DATA SET, RC=*rc*, REASON=*reason*, DSN=*dsname* (*member*)

Explanation:

A FIND macro for the JCL data set failed.

Variable Fields:

rc is the FIND macro return code.

reason is the FIND macro reason code.

dsname is the name of the JCL data set.

member is the name of the JCL member.

User Action:

Check that the correct data set name and member name is specified for the job request. If so, report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the FIND return code and reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS4384E I/O ERROR FOR JCL DATA SET, DSN=*dsname*<(*member*)>, SYNAD MESSAGE=*message*

Explanation:

A I/O error occurred for the JCL data set.

Variable Fields:

dsname is the name of the JCL data set.

member is the name of the JCL member.

message is the message returned by the SYNADAF function describing the error.

User Action:

Check that the correct data set name and member name is specified for the job request. If so, report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the SYNADAF error message, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS4401E JCL SPOOL MEMBER NOT FOUND, REQID=*req-id*, MEMBER=*member*

Explanation:

The PJS JCL Spool member could not be found, or the request-ID in the PJS JCL Spool directory entry user data did not match the job request record. This situation can occur if an abend or system crash occurs while PJS is updating the PJS JCL Spool, or if a utility other than PJS updates the PJS JCL Spool.

Variable Fields:

req-id is the request-ID of the job request associated with the PJS JCL Spool member.

member is the name of the PJS JCL Spool member.

User Action:

Delete the job request involved, then re-add it. Report this message to the PJS Administrator.

Administrator Action:

Make sure non-PJS utilities do not update the user data of the directory entries PJS JCL Spool data set.

PJS4481E JCL SPOOL DATA SET IS INVALID

Explanation:

The PJS JCL Spool data set is invalid. The PJS JCL Spool must be a partitioned data set with RECFM=FB, LRECL=80, and any appropriate blocksize.

User Action:

Report this message to the PJS Administrator.

Administrator Action:

Check that the PJS Options Module specifies the correct PJS JCL Spool data set name and that the data set has been properly allocated and initialized.

PJS4482E OPEN ERROR FOR JCL SPOOL, RC=*rc*

Explanation:

An OPEN macro for the PJS JCL Spool data set failed.

Variable Fields:

rc is an OPEN macro return code.

User Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the OPEN return code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS4483E CLOSE ERROR FOR JCL SPOOL, RC=*rc*

Explanation:

A CLOSE macro for the PJS JCL Spool data set failed.

Variable Fields:

rc is the CLOSE macro return code.

User Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the CLOSE return code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

**PJS4484E BLDL ERROR FOR JCL SPOOL, RC=*rc*, REASON=*reason*,
MEMBER=*member*, REQID=*req-id***

Explanation:

A BLDL macro for the PJS JCL Spool data set failed.

Variable Fields:

rc is the BLDL macro return code.

reason is the BLDL macro reason code.

member is the name of the PJS JCL Spool member.

req-id is the request-ID of the job request associated with the PJS JCL Spool member.

User Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the BLDL return code and reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

**PJS4485E FIND ERROR FOR JCL SPOOL, RC=*rc*, REASON=*reason*,
MEMBER=*member*, REQID=*req-id***

Explanation:

A FIND macro for the PJS JCL Spool data set failed.

Variable Fields:

rc is the FIND macro return code.

reason is the FIND macro reason code.

member is the name of the PJS JCL Spool member.

req-id is the request-ID of the job request associated with the PJS JCL Spool member.

User Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the FIND return code and reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

**PJS4486E STOW ERROR FOR JCL SPOOL, RC=*rc*, REASON=*reason*,
MEMBER=*member*, REQID=*req-id***

Explanation:

A STOW macro for the PJS JCL Spool data set failed.

Variable Fields:

rc is the STOW macro return code.

reason is the STOW macro reason code.

member is the name of the PJS JCL Spool member.

req-id is the request-ID of the job request associated with the PJS JCL Spool member.

User Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the STOW return code and reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS4487E I/O ERROR FOR JCL SPOOL, MEMBER=member, REQID=req-id, SYNAD MESSAGE=message

Explanation:

A I/O error occurred for the PJS JCL Spool data set.

Variable Fields:

member is the name of the PJS JCL Spool member.

req-id is the request-ID of the job request associated with the PJS JCL Spool member.

message is the message returned by the SYNADAF function describing the error.

User Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the SYNADAF error message, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS5101E MAXIMUM SPOOL RECORD COUNT EXCEEDED

Explanation:

The number of records saved in the PJS JCL Spool for a job request exceeds the maximum allowed by the installation.

System Action:

The command or subcommand is ignored.

User Action:

Reduce the number of JCL records for the job request, or specify the NOSAVE option.

PJS5281E OPEN ERROR FOR TEMPORARY DATA SET, RC=*rc*, DSN=*dsname*

Explanation:

An OPEN macro for a temporary data set failed.

Variable Fields:

rc is an OPEN macro return code.

dsname is the data set name of the temporary data set.

User Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information.

PJS5282E CLOSE ERROR FOR TEMPORARY DATA SET, RC=*rc*, DSN=*dsname*

Explanation:

A CLOSE macro for a temporary data set failed.

Variable Fields:

rc is a CLOSE macro return code.

dsname is the data set name of the temporary data set.

User Action:

Report this message to the PJS Administrator.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the CLOSE return code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS7101I PJSQINIT - REQUEST QUEUE DATA SET INITIALIZATION

Explanation:

The PJS Request Queue initialization utility is starting.

PJS7102I PJSQINIT - REQUEST QUEUE DATA SET SUCCESSFULLY INITIALIZED

Explanation:

The PJS Request Queue has been successfully initialized.

**PJS7181E PJSQINIT - VSAM OPEN ERROR FOR QUEUE DATA SET, RC=*rc*
REASON=*reason***

Explanation:

The VSAM OPEN macro for the PJS Request Queue has failed.

Variable Fields:

rc is the OPEN Macro Return Code (from register 15).

reason is the ACB Error Reason Code (from ACB).

System Action:

PJS Request Queue initialization is terminated.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the VSAM OPEN return code and error reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

**PJS7182E PJSQINIT - VSAM I/O ERROR FOR QUEUE DATA SET, RC=*rc*
REASON=*reason***

Explanation:

A VSAM PUT macro for the PJS Request Queue has failed.

Variable Fields:

rc is the PUT Macro Return Code (from register 15).

reason is the RPL Error Reason Code (from RPLERRCD).

System Action:

PJS Request Queue initialization is terminated.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the VSAM PUT return code and error reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

**PJS7183E PJSQINIT - VSAM CLOSE ERROR FOR QUEUE DATA SET, RC=*rc*
REASON=*reason***

Explanation:

The VSAM CLOSE macro for the PJS Request Queue has failed.

Variable Fields:

rc is the CLOSE Macro Return Code (from register 15).

reason is the ACB Error Reason Code (from ACB).

System Action:

PJS Request Queue initialization is terminated.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the VSAM CLOSE return code and error reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

**PJS7184E PJSQINIT - VSAM CONTROL BLOCK ERROR FOR QUEUE DATA SET,
RC=*rc* REASON=*reason***

Explanation:

A VSAM control block manipulation macro for the PJS Request Queue has failed. In most cases, this is a PJS internal error.

Variable Fields:

rc is the Control Block Macro Return Code (from register 15).

reason is the Control Block Macro Error Reason Code (from register 0).

System Action:

PJS Request Queue initialization is terminated.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the VSAM return code and error reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS7201I PJSQMNT - REQUEST QUEUE DATA SET MAINTENANCE - START

Explanation:

The PJS Request Queue Maintenance utility is starting.

PJS7202I PJSQMNT - REQUEST QUEUE DATA SET MAINTENANCE - END

Explanation:

The PJS Request Queue Maintenance utility completed successfully.

PJS7203I PJSQMNT - COMPLETED JOB REQUEST DELETED - REQID=*req-id*

Explanation:

The indicated job request was in COMPLETE status for more than the time specified by the CMPTIME parameter of the PJSOPT macro. The job request is deleted.

Variable Fields:

req-id is the Request-ID of the deleted job request.

PJS7204I PJSQMNT - DISABLED JOB REQUEST DELETED - REQID=*req-id*

Explanation:

The indicated job request was in DISABLED status for more than the time specified by the DISTIME parameter of the PJSOPT macro. The job request is deleted.

Variable Fields:

req-id is the Request-ID of the deleted job request.

PJS7205I PJSQMNT - ERROR JOB REQUEST DELETED - REQID=*req-id*

Explanation:

The indicated job request was in ERROR status for more than the time specified by the ERRTIME parameter of the PJSOPT macro. The job request is deleted.

Variable Fields:

req-id is the Request-ID of the deleted job request.

PJS7206I PJSQMNT - UNREFERENCED EVENT DELETED - EVENTID=*event-id*

Explanation:

The indicated event is not referenced in any job request. The event is deleted.

Variable Fields:

event-id is the Event-ID of the deleted event.

PJS7207I PJSQMNT - UNREFERENCED OWNER DELETED - OWNERID=*owner-id*

Explanation:

The indicated owner has no job requests, calendars, events, or global variables. The owner is deleted.

Variable Fields:

owner-id is the Owner-ID of the deleted owner.

PJS7208I PJSQMNT - UNREFERENCED SPOOL MEMBER DELETED - MEMBER=*member*

Explanation:

The indicated PJS JCL Spool member was not owned by any job request record. The spool member is deleted.

Variable Fields:

member is the name of the deleted PJS JCL Spool member.

PJS7209I PJSQMNT - SAVED JCL RECORD COUNT UPDATED FOR JOB REQUEST - REQID=*req-id*

Explanation:

The PJS JCL Spool Record Count for the indicated job request was updated to reflect the correct number of saved JCL records.

Variable Fields:

req-id is the Request-ID of the updated job request.

PJS7210I PJSQMNT - SAVED JCL RECORD COUNT UPDATED FOR OWNER - OWNERID=owner-id

Explanation:

The PJS JCL Spool Record Count for the indicated owner was updated to reflect the correct number of saved JCL records.

Variable Fields:

owner-id is the Owner-ID of the updated owner.

PJS7211I PJSQMNT - JCL SPOOL DATA SET COMPRESSED

Explanation:

The PJS JCL Spool data set was successfully compressed.

PJS7301I PJSHINIT - MESSAGE HISTORY LOG DATA SET INITIALIZATION

Explanation:

The PJS Message History Log initialization utility is starting.

PJS7302I PJSHINIT - MESSAGE HISTORY LOG DATA SET SUCCESSFULLY INITIALIZED

Explanation:

The PJS Message History Log has been successfully initialized.

PJS7381E PJSHINIT - VSAM OPEN ERROR FOR MESSAGE HISTORY LOG DATA SET, RC=rc REASON=reason

Explanation:

The VSAM OPEN macro for the PJS Message History Log has failed.

Variable Fields:

rc is the OPEN Macro Return Code (from register 15).

reason is the ACB Error Reason Code (from ACB).

System Action:

PJS Request Queue initialization is terminated.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the VSAM OPEN return code and error reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS7382E PJSHINIT - VSAM I/O ERROR FOR MESSAGE HISTORY LOG DATA SET, RC=rc REASON=reason

Explanation:

A VSAM PUT macro for the PJS Message History Log has failed.

Variable Fields:

rc is the PUT Macro Return Code (from register 15).

reason is the RPL Error Reason Code (from RPLERRCD).

System Action:

PJS Message History Log initialization is terminated.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the VSAM PUT return code and error reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS7383E PJSHINIT - VSAM CLOSE ERROR FOR MESSAGE HISTORY LOG DATA SET, RC=*rc* REASON=*reason*

Explanation:

The VSAM CLOSE macro for the PJS Message History Log has failed.

Variable Fields:

rc is the CLOSE Macro Return Code (from register 15).

reason is the ACB Error Reason Code (from ACB).

System Action:

PJS Message History Log initialization is terminated.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the VSAM CLOSE return code, and error reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS7384E PJSHINIT - VSAM CONTROL BLOCK ERROR FOR MESSAGE HISTORY LOG DATA SET, RC=*rc* REASON=*reason*

Explanation:

A VSAM control block manipulation macro for the PJS Message History Log has failed. In most cases, this is a PJS internal error.

Variable Fields:

rc is the Control Block Macro Return Code (from register 15).

reason is the Control Block Macro Error Reason Code (from register 0).

System Action:

PJS Request Queue initialization is terminated.

Administrator Action:

Check any accompanying system messages for more information. For an explanation of the VSAM return code and error reason code, refer to the *DFSMS Macro Instructions for Data Sets* manual for your system.

PJS7801I PJSQCONV - REQUEST QUEUE DATA SET CONVERSION - START

Explanation:

The PJS Request Queue Conversion utility is starting.

PJS7802I PJSQCONV - INPUT FILE IS FROM PJS *Vversion*

Explanation:

The input data set to the PJS Request Queue Conversion utility is at the indicated version level.

Variable Fields:

version is the level of the input PJS Request Queue data set.

PJS7803I PJSQCONV - REQUEST QUEUE DATA SET CONVERSION - END

Explanation:

The PJS Request Queue has been successfully converted.

PJS7804I PJSQCONV - REQUEST QUEUE CONTROL RECORD CONVERTED

Explanation:

The PJS Request Queue Control Record has been successfully converted.

PJS7805I PJSQCONV - OWNER CONVERTED - OWNERID=*owner-id*

Explanation:

The indicated PJS Request Queue owner record was successfully converted.

Variable Fields:

owner-id is the owner-ID of the converted owner record.

PJS7806I PJSQCONV - JOB REQUEST CONVERTED - REQID=*req-id*

Explanation:

The indicated PJS Request Queue job request record was successfully converted.

Variable Fields:

req-id is the Request-ID of the converted job request record.

PJS7807I PJSQCONV - CALENDAR CONVERTED - CALID=*cal-id*

Explanation:

The indicated PJS Request Queue calendar record was successfully converted.

Variable Fields:

cal-id is the calendar-ID of the converted calendar record.

PJS7808I PJSQCONV - EVENT CONVERTED - EVENTID=*event-id*

Explanation:

The indicated PJS Request Queue event record was successfully converted.

Variable Fields:

event-id is the event-ID of the converted event record.

PJS7809I PJSQCONV - VARIABLE CONVERTED - VARID=*var-id*

Explanation:

The indicated PJS Request Queue global variable record was successfully converted.

Variable Fields:

var-id is the variable-ID of the converted global variable record.

Chapter 2: PJS Abend Codes

0101 ABND_PARMERR_EXT

Explanation:

PJS was called with an invalid parameter list.

Action:

Check that PJS is being executed correctly. The PJS TSO commands must be executed as TSO command processors (and must be passed a TSO CPPL).

0102 ABND_PARMERR_INT

Explanation:

An internal PJS program was called with an invalid parameter list.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed and executed correctly.

0103 ABND_FUNCERR

Explanation:

A PJS program was called with an invalid function code.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed and executed correctly.

0104 ABND_CMDERR

Explanation:

A PJS internal TSO command (used by the PJS ISPF interface) was called with an invalid command.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed and executed correctly.

0111 ABND_ESTAEERR

Explanation:

An ESTAE macro failed.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

0112 ABND_TSFERR

Explanation:

A TSO service facility request failed.

Action:

This abend usually occurs when the PJS ISPF Interface TSO commands cannot be accessed by the PJS ISPF Interface dialog. Check that PJS has been properly installed.

0121 ABND_SPLMERR

Explanation:

The maximum PJS JCL Spool member number has been exceeded.

Action:

Since the maximum PJS JCL Spool member number is currently 9999999, this is not expected occur.

0131 ABND_QIOERR

Explanation:

A call to the PJS Request Queue I/O service routine (PJSQIO) returned with an unexpected return code.

Action:

Check for previous error messages and respond as specified for those messages.

0132 ABND_HIOERR

Explanation:

A call to the PJS Message History Log I/O service routine (PJSHIO) returned with an unexpected return code.

Action:

Check for previous error messages and respond as specified for those messages.

0133 ABND_SIOERR

Explanation:

A call to the PJS JCL Spool I/O service routine (PJSSIO) returned with an unexpected return code.

Action:

Check for previous error messages and respond as specified for those messages.

0134 ABND_JIOERR

Explanation:

A call to the PJS JCL Data Set I/O service routine (PJSJIO) returned with an unexpected return code.

Heading:

Check for previous error messages and respond as specified for those messages.

0141 ABND_CALUERR

Explanation:

A call to the PJS Calendar Record service routine (PJSCALU) returned with an unexpected return code.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

0151 ABND_DYNXERR

Explanation:

A call to the PJS Installation Dynamic Variables Exit (PJSDYNX) returned with an invalid return code.

Action:

Check the PJS Installation Dynamic Variables Exit. Make sure that a valid return code is always set.

0152 ABND_SECXERR

Explanation:

A call to the PJS Installation Security Exit (PJSSECX) returned with an invalid return code.

Action:

Check the PJS Installation Security Exit. Make sure that a valid return code is always set.

0153 ABND_SUBXERR

Explanation:

A call to the PJS Installation Submit Exit (PJSSUBX) returned with an invalid return code.

Action:

Check the PJS Installation Submit Exit. Make sure that a valid return code is always set.

0161 ABND_PQJRQ_SEGERR

Explanation:

An invalid segment was found in the PJSQ Job Request Record.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

0162 ABND_PQJRQ_LENERR

Explanation:

The segments created could not fit in the PJSQ Job Request Record.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

0171 ABND_ISPF_PQERR

Explanation:

An ISPF PQUERY service failed.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

0172 ABND_ISPF_STGERR

Explanation:

Insufficient storage is available to build an ISPF list table.

Action:

Increase the region size for the TSO session, and try again.

0181 ABND_ISPFTSO_RECERR

Explanation:

The record area passed to an internal TSO command by the PJS ISPF interface is invalid.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

0182 ABND_ISPFTSO_TBLERR

Explanation:

The record table area passed to an internal TSO command by the PJS ISPF interface is invalid.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

0501 ABND_PJSTASK_OPERERR

Explanation:

The PJS System Task operator communication subtask abended.

Action:

Respond as specified for the preceeding PJS1202 message.

0511 ABND_PJSTASKI_ENQERR

Explanation:

The ENQ for the PJS System Task resource failed. The PJS System Task was probably started when it was already running in another address space.

Action:

Respond as specified for the preceeding PJS1106 message.

0521 ABND_PJSTASKB_PTBLERR

Explanation:

The PJS System Task Event Post/Reset table has overflowed.

Action:

Increase the PTBLSZ parameter of the PJSOPT macro, reassemble the PJS Options Table, and restart the PJS System Task.

0522 ABND_PJSTASKB_RRTBLERR

Explanation:

The PJS System Task Ready Request table has overflowed.

Action:

Increase the RRTBLSZ parameter of the PJSOPT macro, reassemble the PJS Options Table, and restart the PJS System Task.

0531 ABND_PJSTASKJ_SUBABND

Explanation:

The PJS Submit Job Subtask (PJSSUB) failed with an unrecoverable error.

Action:

Check for previous error messages and subtask abend codes and respond as specified for those messages.

0541 ABND_PJTASKV_VTBLERR

Explanation:

The PJS System Task Variable Substitution table has overflowed.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

0542 ABND_PJTASKV_VRTBLERR

Explanation:

The PJS System Task Global Variable Record table has overflowed.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

0543 ABND_PJTASKV_JRQERR

Explanation:

A job request record contains invalid data.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

0544 ABND_PJTASKV_VARERR

Explanation:

A global variable record contains invalid data.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

0561 ABND_PJSSUB_OPENERR

Explanation:

An OPEN macro for the internal reader failed. This abend is issued by the PJS System Task job submit subtask.

Action:

Respond as specified for the preceeding PJS1702 message.

0562 ABND_PJSSUB_CLOSERR

Explanation:

A CLOSE macro for the internal reader failed. This abend is issued by the PJS System Task job submit subtask.

Action:

Respond as specified for the preceeding PJS1703 message.

0581 ABND_PJSOPER_QEDITERR

Explanation:

A QEDIT function by the PJS System Task operator communication subtask failed.

Action:

Respond as specified for the preceeding PJS1203 message.

1001 ABND_PJSQMNT_SPLDERR

Explanation:

An OPEN macro for the PJS JCL Spool directory failed. This abend is issued by the PJS Request Queue Maintenance utility.

Action:

Check for previous error messages and respond as specified for those messages.

1002 ABND_PJSQMNT_SPLOPERR

Explanation:

An OPEN macro for the PJS JCL Spool data set (using BPAM) failed. This abend is issued by the PJS Request Queue Maintenance utility.

Action:

Check for previous error messages and respond as specified for those messages.

1003 ABND_PJSQMNT_HBKOOERR

Explanation:

An OPEN OUTPUT macro for the PJS Message History Log Backup data set failed. This abend is issued by the PJS Request Queue Maintenance utility.

Action:

Check for previous error messages and respond as specified for those messages.

1004 ABND_PJSQMNT_HBKOIERR

Explanation:

An OPEN INPUT macro for the PJS Message History Log Backup data set failed. This abend is issued by the PJS Request Queue Maintenance utility.

Action:

Check for previous error messages and respond as specified for those messages.

1005 ABND_PJSQMNT_CTLOERR

Explanation:

An OPEN macro for the IEBCOPY SYSIN Control data set failed. This abend is issued by the PJS Request Queue Maintenance utility.

Action:

Check for previous error messages and respond as specified for those messages.

1011 ABND_PJSQMNT_STOWERR

Explanation:

A STOW macro for the PJS JCL Spool directory failed. This abend is issued by the PJS Request Queue Maintenance utility.

Action:

Check for previous error messages and respond as specified for those messages.

1012 ABND_PJSQMNT_COPYERR

Explanation:

The IEBCOPY utility was called to compress the PJS JCL Spool data set, and returned a non-zero return code. This abend is issued by the PJS Request Queue Maintenance utility.

Action:

Check for previous error messages and respond as specified for those messages.

1021 ABND_PJSQMNT_STBLOVFL

Explanation:

The PJS Request Queue Maintenance Utility Spool Directory table has overflowed.

Action:

Increase the STBLSZ parameter of the PJSOPT macro, reassemble the PJS Options Table, and rerun the PJS Request Queue Maintenance utility.

1022 ABND_PJSQMNT_RTBLOVFL

Explanation:

The PJS Request Queue Maintenance Utility Request table has overflowed.

Action:

Increase the RTBLSZ parameter of the PJSOPT macro, reassemble the PJS Options Table, and rerun the PJS Request Queue Maintenance utility.

1023 ABND_PJSQMNT_OTBLOVFL

Explanation:

The PJS Request Queue Maintenance Utility Owner table has overflowed.

Action:

Increase the OTBLSZ parameter of the PJSOPT macro, reassemble the PJS Options Table, and rerun the PJS Request Queue Maintenance utility.

1024 ABND_PJSQMNT_ETBLOVFL

Explanation:

The PJS Request Queue Maintenance Utility Event table has overflowed.

Action:

Increase the ETBLSZ parameter of the PJSOPT macro, reassemble the PJS Options Table, and rerun the PJS Request Queue Maintenance utility.

1101 ABND_PJSQINIT_OPENERR

Explanation:

A VSAM OPEN macro for the PJS Request Queue failed. This abend is issued by the PJS Request Queue Initialization utility.

Action:

Respond as specified for the preceeding PJS7181 message.

1102 ABND_PJSQINIT_PUTERR

Explanation:

A VSAM PUT macro for the PJS Request Queue failed. This abend is issued by the PJS Request Queue Initialization utility.

Action:

Respond as specified for the preceeding PJS7182 message.

1103 ABND_PJSQINIT_CLOSERR

Explanation:

A VSAM CLOSE macro for the PJS Request Queue failed. This abend is issued by the PJS Request Queue Initialization utility.

Action:

Respond as specified for the preceeding PJS7183 message.

1201 ABND_PJSHINIT_OPENERR

Explanation:

A VSAM OPEN macro for the PJS Message History Log failed. This abend is issued by the PJS Message History Log Initialization utility.

Action:

Respond as specified for the preceeding PJS7381 message.

1202 ABND_PJSHINIT_PUTERR

Explanation:

A VSAM PUT macro for the PJS Message History Log failed. This abend is issued by the PJS Message History Log Initialization utility.

Action:

Respond as specified for the preceeding PJS7382 message.

1203 ABND_PJSHINIT_CLOSERR

Explanation:

A VSAM CLOSE macro for the PJS Message History Log failed. This abend is issued by the PJS Message History Log Initialization utility.

Action:

Respond as specified for the preceeding PJS7383 message.

1204 ABND_PJSHINIT_CBERR

Explanation:

A VSAM control block manipulation macro for the PJS Message History Log failed. This abend is issued by the PJS Message History Log Initialization utility.

Action:

Respond as specified for the preceeding PJS7384 message.

1301 ABND_PJSQCONV_INITERR

Explanation:

The PJS Release 3.1 Request Queue to used by the PJS Request Queue Conversion Utility is not a newly initialized data set.

Action:

Check that the PJS Request Queue Initialization utility ran successfully and that the PJS Request Queue data set was not otherwise updated. Check that the PJSQDSN parameter of the PJSOPT macro specifies the correct data set name.

1302 ABND_PJSQCONV_OPENERR

Explanation:

An OPEN macro for the PJS Release 2.x Request Queue sequential data set failed. This abend is issued by the PJS Request Queue Conversion utility.

Action:

Check for previous error messages and respond as specified for those messages. Registers 2 and 3 contain the contents of registers 0 and 15 from the OPEN macro.

1303 ABND_PJSQCONV_CNTLERR

Explanation:

An invalid Queue Control record was found in the PJS Release 2.x Request Queue sequential data set. This abend is issued by the PJS Request Queue Conversion Utility.

Action:

Check that the PJS Request Queue Conversion utility is run correctly. Check that the IDCAMS REPRO step runs correctly. Check that the correct data set is used as input to the PJS Request Queue Conversion utility.

1304 ABND_PJSQCONV_TYPEERR

Explanation:

An invalid record type was found in the PJS Release 2.x Request Queue sequential data set. This abend is issued by the PJS Request Queue Conversion Utility.

Action:

Check that the PJS Request Queue Conversion utility is run correctly. Check that the IDCAMS REPRO step runs correctly. Check that the correct data set is used as input to the PJS Request Queue Conversion utility.

1305 ABND_PJSQCONV_LENERR

Explanation:

The segments of a converted job request could not fit in the PJSQ Job Request Record. This abend is issued by the PJS Request Queue Conversion Utility.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

1306 ABND_PJSQCONV_VTYPERR

Explanation:

A Global Variable record with an invalid Variable Value Type was found in the PJS Release 2.x Request Queue sequential data set. This abend is issued by the PJS Request Queue Conversion Utility.

Action:

Check that the PJS Request Queue Conversion utility is run correctly. Check that the IDCAMS REPRO step runs correctly. Check that the correct data set is used as input to the PJS Request Queue Conversion utility.

1501 ABND_PJSTJCMD_PGMERR

Explanation:

The PJS Job Request TSO Subcommand Processor was not called from a valid main command processor.

Action:

Check that PJS has been installed and executed correctly.

1502 ABND_PJSTJCMD_PUTGERR

Explanation:

A PUTGET service for the PJS Job Request TSO Subcommand Processor failed. The reason code contains the return code from PUTGET.

Action:

Check any accompanying system messages for more information. Check the documentation on the PUTGET return code.

1503 ABND_PJSTJCMD_SCANERR

Explanation:

An IKJSCAN service for the PJS Job Request TSO Subcommand Processor failed. The reason code contains the return code from IKJSCAN.

Action:

Check any accompanying system messages for more information. Check the documentation on the IKJSCAN return code.

1504 ABND_PJSTJCMD_STAXERR

Explanation:

A STAX service for the PJS Job Request TSO Subcommand Processor failed. The reason code contains the return code from STAX.

Action:

Check any accompanying system messages for more information. Check the documentation on the STAX return code.

1511 ABND_PJSTJEND_PGMERR

Explanation:

The PJS Job Request END TSO Subcommand Processor was not called from a valid main command processor.

Action:

Check that PJS has been installed and executed correctly.

2001 ABND_PJSISPF_NOISPF

Explanation:

The PJS ISPF Interface was not called from a valid ISPF environment.

Action:

Check that PJS has been installed and executed correctly. Check that the PJS ISPF Interface is executed from a valid ISPF environment.

2101 ABND_PJSIDJB_LMERR

Explanation:

An ISPF Library Management service for the PJS Browse JCL for Job Request dialog failed.

Action:

Check for any ISPF error messages in the ISPF Log data set.

2501 ABND_PJSITJJ_DDNERR

Explanation:

The DDNAME area passed to the PJS/ISPF Get Saved JCL for Job Request TSO command processor (PJSITJJ) is invalid.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3001 ABND_PJSMAIN_OPTERR

Explanation:

One or more options specified by the PJSOPT macro (or modified by the PJSOPTX Installation Options Exit) is invalid.

Action:

Respond as specified for the preceeding PJS0103 message.

3011 ABND_PJSALLOC_S99ERR

Explanation:

SVC 99 (dynamic allocation) returned with an unexpected return code. This abend is issued by the PJS Dynamic Allocation Service Routine (PJSALLOC).

Action:

Respond as specified for the preceeding PJS4001 message.

3021 ABND_PJSTEMP_OPENERR

Explanation:

An OPEN macro for a Temporary Data Set failed. This abend is issued by the Copy JCL Spool Member to Temporary Data Set Service Routine (PJSTEMP).

Action:

Respond as specified for the preceeding PJS5281 message.

3022 ABND_PJSTEMP_CLOSERR

Explanation:

A CLOSE macro for a Temporary Data Set failed. This abend is issued by the Copy JCL Spool Member to Temporary Data Set Service Routine (PJSTEMP).

Action:

Respond as specified for the preceeding PJS5282 message.

3031 ABND_PJSFREQ_DATEERR

Explanation:

The PJSQ Job Request record passed to the Calculate Next Run Date/Time Service Routine (PJSFREQ) contains an invalid date or time field.

Action:

Check the PJS Request Queue for corrupted records. This can also be caused by an internal PJS error. Check that PJS has been installed correctly.

3032 ABND_PJSFREQ_FREQERR

Explanation:

The PJSQ Job Request record passed to the Calculate Next Run Date/Time Service Routine (PJSFREQ) contains an invalid frequency type code.

Action:

Check the PJS Request Queue for corrupted records. This can also be caused by an internal PJS error. Check that PJS has been installed correctly.

3033 ABND_PJSFREQ_PRDERR

Explanation:

The PJSQ Job Request record passed to the Calculate Next Run Date/Time Service Routine (PJSFREQ) contains an invalid frequency period units code.

Action:

Check the PJS Request Queue for corrupted records. This can also be caused by an internal PJS error. Check that PJS has been installed correctly.

3034 ABND_PJSFREQ_QTYERR

Explanation:

The PJSQ Job Request record passed to the Calculate Next Run Date/Time Service Routine (PJSFREQ) contains an invalid frequency period quantity.

Action:

Check the PJS Request Queue for corrupted records. This can also be caused by an internal PJS error. Check that PJS has been installed correctly.

3035 ABND_PJSFREQ_WKDAYERR

Explanation:

The PJSQ Job Request record passed to the Calculate Next Run Date/Time Service Routine (PJSFREQ) contains an invalid days-of-week flag.

Action:

Check the PJS Request Queue for corrupted records. This can also be caused by an internal PJS error. Check that PJS has been installed correctly.

3036 ABND_PJSFREQ_EOMERR

Explanation:

The PJSQ Job Request record passed to the Calculate Next Run Date/Time Service Routine (PJSFREQ) contains an invalid number of days before end-of-month value.

Action:

Check the PJS Request Queue for corrupted records. This can also be caused by an internal PJS error. Check that PJS has been installed correctly.

3037 ABND_PJSFREQ_CALERR

Explanation:

The PJSQ Job Request record passed to the Calculate Next Run Date/Time Service Routine (PJSFREQ) contains an invalid or missing Calendar Table segment.

Action:

Check the PJS Request Queue for corrupted records. This can also be caused by an internal PJS error. Check that PJS has been installed correctly.

3041 ABND_PJSCALU_LENERR

Explanation:

The PJSQ Calendar record passed to the Calendar Data Utility Service Routine (PJSCALU) contains an invalid selected dates string.

Action:

Check the PJS Request Queue for corrupted records. This can also be caused by an internal PJS error. Check that PJS has been installed correctly.

3101 ABND_PJSJRQFM_ALENERR

Explanation:

The output data area passed to the Format Job Request Record Service Routine (PJSJRQFM) is not large enough to contain the formatted Job Request.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3102 ABND_PJSJRQFM_TLENERR

Explanation:

A token formatted by the Format Job Request Record Service Routine (PJSJRQFM) is larger than the area allocated for it.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3111 ABND_PJSCALFM_ALENERR

Explanation:

The output data area passed to the Format Calendar Record Service Routine (PJSCALFM) is not large enough to contain the formatted Calendar.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3112 ABND_PJSCALFM_TLENERR

Explanation:

A token formatted by the Format Calendar Record Service Routine (PJSCALFM) is larger than the area allocated for it.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3121 ABND_PJSEVNFM_ALENERR

Explanation:

The output data area passed to the Format Event Record Service Routine (PJSEVNFM) is not large enough to contain the formatted Event.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3122 ABND_PJSEVNFM_TLENERR

Explanation:

A token formatted by the Format Event Record Service Routine (PJSEVNFM) is larger than the area allocated for it.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3131 ABND_PJSVARFM_ALENERR

Explanation:

The output data area passed to the Format Global Variable Record Service Routine (PJSVARFM) is not large enough to contain the formatted Global Variable.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3132 ABND_PJSVARFM_TLENERR

Explanation:

A token formatted by the Format Global Variable Record Service Routine (PJSVARFM) is larger than the area allocated for it.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3141 ABND_PJSHSTFM_ALENERR

Explanation:

The output data area passed to the Format Message History Log Record Service Routine (PJSHSTFM) is not large enough to contain the formatted message.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3142 ABND_PJSHSTFM_TLENERR

Explanation:

A token formatted by the Format Message History Log Record Service Routine (PJSHSTFM) is larger than the area allocated for it.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3501 ABND_PJSQIO_OPTERR

Explanation:

The PJS Request Queue I/O Service Routine (PJSQIO) was called with an invalid option code.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3502 ABND_PJSQIO_TYPEERR

Explanation:

The PJS Request Queue I/O Service Routine (PJSQIO) was called with an invalid record type code.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3503 ABND_PJSQIO_LENERR

Explanation:

A PJS Request Queue record passed to the PJS Request Queue I/O Service Routine (PJSQIO) had an invalid record length.

Action:

If your PJS Installation Security Exit (PJSSECX) updates the Installation Data Area, make sure this code is correct. This can also be caused by an internal PJS error. Check that PJS has been installed correctly.

3504 ABND_PJSQIO_KEYERR

Explanation:

A PJS Request Queue record passed to the PJS Request Queue I/O Service Routine (PJSQIO) had an invalid record key.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3505 ABND_PJSQIO_VERERR

Explanation:

A PJS Request Queue record read from the PJS Request Queue, or a record passed to the PJS Request Queue I/O Service Routine (PJSQIO) had an invalid record version.

Action:

For a READ operation, check that you are not using a back-level PJS Request Queue. Check the PJS Request Queue for corrupted records.

For an ADD or UPDATE operation, in most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3506 ABND_PJSQIO_DUOPEN

Explanation:

The PJS Request Queue I/O Service Routine (PJSQIO) was called to open the PJS Request Queue, but the PJS Request Queue was already open.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3507 ABND_PJSQIO_NOTOPEN

Explanation:

The PJS Request Queue I/O Service Routine (PJSQIO) was called for a function requiring the PJS Request Queue to be open, but the PJS Request Queue was not open.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3508 ABND_PJSQIO_CURERR

Explanation:

The PJS Request Queue I/O Service Routine (PJSQIO) attempted to reestablish its position in the PJS Request Queue, but no current positioning has been established.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3509 ABND_PJSQIO_UPDERR

Explanation:

The PJS Request Queue I/O Service Routine (PJSQIO) was called to update or delete a PJS Request Queue record, but the record had not been previously read for update processing.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3510 ABND_PJSQIO_UNKERR

Explanation:

The PJS Request Queue I/O Service Routine Error Handler (PJSQIO9) was called with an invalid error type code.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3511 ABND_PJSQIO_ACBERR

Explanation:

A VSAM ACB based macro for the PJS Request Queue failed with an unexpected reason code. This abend is issued by the PJS Request Queue I/O Service Routine Error Handler (PJSQIO9).

Action:

Respond as specified for the preceeding PJS4181 message.

3512 ABND_PJSQIO_RPLERR

Explanation:

A VSAM RPL based macro for the PJS Request Queue failed with an unexpected reason code. This abend is issued by the PJS Request Queue I/O Service Routine Error Handler (PJSQIO9).

Action:

Respond as specified for the preceeding PJS4182 message.

3513 ABND_PJSQIO_CBERR

Explanation:

A control block manipulation macro for the PJS Request Queue failed. This abend is issued by the PJS Request Queue I/O Service Routine Error Handler (PJSQIO9).

Action:

Respond as specified for the preceeding PJS4183 message.

3514 ABND_PJSQIO_QIO9ERR

Explanation:

The PJS Request Queue I/O Service Routine Error Handler (PJSQIO9) unexpectedly returned control to the PJS Request Queue I/O Service Routine.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3521 ABND_PJSHIO_DUOPEN

Explanation:

The PJS Message History Log Service Routine (PJSHIO) was called to open the PJS Message History Log, but the PJS Message History Log was already open.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3522 ABND_PJSHIO_NOTOPEN

Explanation:

The PJS Message History Log I/O Service Routine (PJSHIO) was called for a function requiring the PJS Message History Log to be open, but the PJS Message History Log was not open.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3523 ABND_PJSHIO_UNKERR

Explanation:

The PJS Message History Log I/O Service Routine Error Handler (PJSHIO9) was called with an invalid error type code.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3524 ABND_PJSHIO_ACBERR

Explanation:

A VSAM ACB based macro for the PJS Message History Log failed with an unexpected reason code. This abend is issued by the PJS Message History Log I/O Service Routine Error Handler (PJSHIO9).

Action:

Respond as specified for the preceeding PJS4281 message.

3525 ABND_PJSHIO_RPLERR

Explanation:

A VSAM RPL based macro for the PJS Message History Log failed with an unexpected reason code. This abend is issued by the PJS Message History Log I/O Service Routine Error Handler (PJSHIO9).

Action:

Respond as specified for the preceeding PJS4282 message.

3526 ABND_PJSHIO_CBERR

Explanation:

A control block manipulation macro for the PJS Message History Log failed. This abend is issued by the PJS Message History Log I/O Service Routine Error Handler (PJSHIO9).

Action:

Respond as specified for the preceeding PJS4283 message.

3527 ABND_PJSHIO_HIO9ERR

Explanation:

The PJS Message History Log I/O Service Routine Error Handler (PJSHIO9) unexpectedly returned control to the PJS Message History Log I/O Service Routine.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3541 ABND_PJSSIO_SIODERR

Explanation:

The PJS JCL Spool I/O Service Routine (PJSSIO) either found a PJSSIO Data Area already allocated before the open, or could not locate the PJSSIO Data Area after the open.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3542 ABND_PJSSIO_DUOPEN

Explanation:

The PJS JCL Spool I/O Service Routine (PJSSIO) was called to open the PJS JCL Spool, but the PJS JCL Spool was already open.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3543 ABND_PJSSIO_NOTOPEN

Explanation:

The PJS JCL Spool I/O Service Routine (PJSSIO) was called to access or close the PJS JCL Spool, but the PJS JCL Spool was not open.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3544 ABND_PJSSIO_INVPOOL

Explanation:

The PJS JCL Spool Data Set specified by the PJS Installation Options Module (PJSOPT) is invalid. This abend is issued by the PJS JCL Spool I/O Service Routine (PJSSIO).

Action:

Respond as specified for the preceeding PJS4481 message.

3545 ABND_PJSSIO_OPENERR

Explanation:

An OPEN macro for the PJS JCL Spool failed. This abend is issued by the PJS JCL Spool I/O Service Routine (PJSSIO).

Action:

Respond as specified for the preceeding PJS4482 message.

3546 ABND_PJSSIO_BLDLERR

Explanation:

A BLDL macro for the PJS JCL Spool failed. This abend is issued by the PJS JCL Spool I/O Service Routine (PJSSIO).

Action:

Respond as specified for the preceeding PJS4484 message.

3547 ABND_PJSSIO_FINDERR

Explanation:

A FIND macro for the PJS JCL Spool failed. This abend is issued by the PJS JCL Spool I/O Service Routine (PJSSIO).

Action:

Respond as specified for the preceeding PJS4485 message.

3548 ABND_PJSSIO_STOWERR

Explanation:

A STOW macro for the PJS JCL Spool failed. This abend is issued by the PJS JCL Spool I/O Service Routine (PJSSIO).

Action:

Respond as specified for the preceeding PJS4486 message.

3549 ABND_PJSSIO_CLOSERR

Explanation:

A CLOSE macro for the PJS JCL Spool failed. This abend is issued by the PJS JCL Spool I/O Service Routine (PJSSIO).

Action:

Respond as specified for the preceeding PJS4483 message.

3550 ABND_PJSSIO_SYNADERR

Explanation:

An I/O error was encountered while processing the PJS JCL Spool. This abend is issued by the PJS JCL Spool I/O Service Routine (PJSSIO).

Action:

Respond as specified for the preceeding PJS4487 message.

3561 ABND_PJSJIO_JIODERR

Explanation:

The PJS JCL Data Set I/O Service Routine (PJSJIO) either found a PJSJIO Data Area already allocated before the open, or could not locate the PJSJIO Data Area after the open.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3562 ABND_PJSJIO_DUPOPEN

Explanation:

The PJS JCL Data Set I/O Service Routine (PJSJIO) was called to open the JCL Data Set, but the JCL Data Set was already open.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3563 ABND_PJSJIO_NOTOPEN

Explanation:

The PJS JCL Data Set I/O Service Routine (PJSJIO) was called to access or close the JCL Data Set, but the JCL Data Set was not open.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3564 ABND_PJSJIO_OPENERR

Explanation:

An OPEN macro for the JCL Data Set failed. This abend is issued by the PJS JCL Data Set I/O Service Routine (PJSJIO).

Action:

Respond as specified for the preceeding PJS4381 message.

3565 ABND_PJSJIO_FINDERR

Explanation:

A FIND macro for the JCL Data Set failed. This abend is issued by the PJS JCL Data Set I/O Service Routine (PJSJIO).

Action:

Respond as specified for the preceeding PJS4383 message.

3566 ABND_PJSJIO_CLOSERR

Explanation:

A CLOSE macro for the JCL Data Set failed. This abend is issued by the PJS JCL Data Set I/O Service Routine (PJSJIO).

Action:

Respond as specified for the preceeding PJS4382 message.

3567 ABND_PJSJIO_SYNADERR

Explanation:

An I/O error was encountered while processing the JCL Data Set. This abend is issued by the PJS JCL Data Set I/O Service Routine (PJSJIO).

Action:

Respond as specified for the preceeding PJS4384 message.

3701 ABND_PJSMSG_MSGIDERR

Explanation:

The PJS Message Service Routine (PJSMSG) was called with an invalid message-ID.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3702 ABND_PJSMSG_DESTERR

Explanation:

The PJS Message Service Routine (PJSMSG) was called with an invalid message destination list.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3703 ABND_PJSMSG_MSGERR

Explanation:

The Message Format Service Routine (PJSMSGF) returned with an unexpected return code.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3704 ABND_PJSMSG_LENERR

Explanation:

The message buffer passed to the PJS Message Service Routine (PJSMSG) was not large enough to contain the formatted message.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3711 ABND_PJSMSGI_ISPFERR

Explanation:

An ISPF variable service issued by the PJS Display ISPF Message Service Routine (PJSMSGI) failed.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3721 ABND_PJSMSGP_OPENERR

Explanation:

An OPEN macro for the printer data set failed. This abend is issued by the PJS Display Message on Printer Service Routine (PJSMSGP).

Action:

Make sure the SYSOUT DD statement is specified correctly. Check for previous error messages and respond as specified for those messages.

3722 ABND_PJSMSGP_CLOSERR

Explanation:

A CLOSE macro for the printer data set failed. This abend is issued by the PJS Display Message on Printer Service Routine (PJSMSGP).

Action:

Make sure the SYSOUT DD statement is specified correctly. Check for previous error messages and respond as specified for those messages.

3723 ABND_PJSMSGP_NOTOPEN

Explanation:

The PJS Display Message on Printer Service Routine (PJSMSGP) was called to print a message, or close the print data set, but the printer data set was not open.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3724 ABND_PJSMSGP_CLOSED

Explanation:

The PJS Display Message on Printer Service Routine (PJSMSGP) was called to open the printer data set, but it had previously been open and closed.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3725 ABND_PJSMSGP_INVRECFM

Explanation:

The printer data set has an invalid record format. This abend is issued by the PJS Display Message on Printer Service Routine (PJSMSGP).

Action:

Make sure the SYSOUT DD statement is specified correctly. The record format must be F, FA, FM, FB, FBA, FBM, V, VA, VM, VB, VBA, or VBM.

3726 ABND_PJSMSGP_INVLRECL

Explanation:

The printer data set has an invalid record length.

Action:

Make sure the SYSOUT DD statement is specified correctly. The data portion of the record, which does not include the RDW or carriage control character, must be at least 80 bytes long but no more than 255 bytes long.

3731 ABND_PJSMSGT_NOCPPPL

Explanation:

The PJS Display Message on TSO Terminal Service Routine (PJSMSGT) was called to send a message to the TSO terminal, but the CPPPL address in the PJSVT was zero.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3732 ABND_PJSMSGT_PUTLERR

Explanation:

The TSO PUTLINE service routine failed. Register 15 contains the PUTLINE return code. This abend is issued by the PJS Display Message on TSO Terminal Service Routine (PJSMSGT).

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

3741 ABND_PJSWTOR_LENERR

Explanation:

The PJS Get Operator Reply Service Routine (PJSWTOR) was called to send a WTOR message on the system console, but the length of the message to be displayed exceeded the maximum allowed.

Action:

In most cases this is caused by an internal PJS error. Check that PJS has been installed correctly.

40nn

Explanation:

This abend code ranges from 4000 through 4095; this range of abend codes is reserved for use in installation exits. Their meaning depends on the type of exit installed at your site.

Action:

Note the actions that preceded generation of the abend code, and contact the person who installed PJS. If you are the PJS installer, check the installation exits for the source of the abend, and fix the problem. For more information on the PJS Installation Exits, please refer to the *PJS Installation Guide*.

GNU General Public License

Version 2, June 1991

Copyright © 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of
this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program

or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a)** You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b)** You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c)** If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a)** Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b)** Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c)** Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For

example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
one line to give the program's name and a brief idea of what it does.
Copyright (C)  yyyy  name of author
```

```
This program is free software; you can redistribute it and/or modify it
under the terms of the GNU General Public License as published by the
Free Software Foundation; either version 2 of the License, or (at your
option) any later version.
```

```
This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
General Public License for more details.
```

```
You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
```

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
'show w'.
This is free software, and you are welcome to redistribute it under
certain conditions; type 'show c' for details.
```

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

DRAFT 2/9/05

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.
```

```
signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

GNU Free Documentation License

Version 1.2, November 2002

Copyright (C) 2000, 2001, 2002 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of
this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front

cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.

- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (c)  YEAR  YOUR NAME.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.2 or
any later version published by the Free Software Foundation; with no
Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.  A
copy of the license is included in the section entitled "GNU Free
Documentation License".
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

```
with the Invariant Sections being LIST THEIR TITLES, with the Front-
Cover Texts being LIST, and with the Back-Cover Texts being LIST.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

