

PJS 3.1.0 Read Me

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

PJS Requirements

Software Requirements

PJS Release 3.1 is intended to run on any currently supported release of OS/390 or z/OS. It may also run on older releases, but no effort will be made to maintain compatibility with unsupported operating systems.

The sample installation exits may have other software requirements. These requirements are noted in the documentation for the individual exits.

Hardware Requirements

PJS has no special hardware requirements, other than those required by its prerequisite software.

PJS uses the Immediate and Relative Instructions Facility, and may not run on a processor that does not support these instructions.

DASD Storage Requirements

PJS requires approximately 450 tracks on a 3390 for the software. Installation with SMP/E will require approximately an additional 500 tracks for DLIBs and SMP/E control data sets. In addition, approximately 650 tracks for non-SMP/E installation, or 1000 tracks for SMP/E installation, are required during the installation process, but can be deleted after installation is complete.

The PJS Request Queue is a VSAM KSDS that holds the scheduling information for all PJS users. Space requirements will vary for each installation, but 1 or 2 cylinders should be a sufficient starting point in most cases.

The PJS JCL Spool is a PDS (or PDSE) and needs to be large enough to hold all of the saved JCL for all of the PJS users, plus extra space for expansion between PDS compression. The directory requires 1 block for every 7 members.

The PJS Message History Log requirements may vary, depending on how many jobs are submitted by PJS and how long messages are retained. Each job submitted will require approximately 250 bytes. Multiply the number of jobs submitted in your message retention period by 250, then add an additional amount for jobs that have been submitted during that period. (PJS will always retain messages for the last submit for a job request, regardless for the retention period.)

PJS Installation Tasks

Task 1: Obtain the PJS Installation Package

The PJS Installation Package is a single sequential file that contains all the elements of PJS. This file may be obtained as part of a public software distribution tape, such as the MVS CBT Tape, or downloaded from the Internet. If you obtained PJS from the MVS CBT Tape, or another distribution tape, you should refer to the documentation for the distribution tape for specific details of how to extract the PJS Installation Package.

If you download PJS it from the Internet, it may be contained within a Zip file. If this is the case you will need to extract the PJS Installation Package file from the Zip file. In any case whenever the PJS Installation Package, or the Zip file that contains it, is transmitted, typically with FTP, it must be done as BINARY data.

When the PJS Installation Package file (unzipped if necessary) is finally loaded to the OS/390 or z/OS host it is to be installed on, it must be allocated as a physical sequential data set (DSORG=PS), fixed-length records (RECFM=FB), with a logical record length of 80 bytes (LRECL=80), and any appropriate block size. When FTP is used these data set characteristics are typically set using the FTP 'SITE' command. Your installation may also require additional parameters on the 'SITE' command to specify how and where the data set should be allocated. For a complete list of all the options available see the *IBM Communications Server: IP User's Guide* for your release of OS/390 or z/OS.

The following example shows the FTP commands that can be used to send the PJS Installation Package from a PC to an OS/390 or z/OS host:

```
C:\>FTP hostname
Connected to hostname.
220-FTPD1 IBM FTP CS V1R4 at hostname, 16:34:20 on 2005-01-04.
220 Connection will close if idle for more than 20 minutes.
User (hostname:(none)): userid
331 Send password please.
Password: password
230 userid is logged on. Working directory is "userid".
ftp> CD 'qual'
250 "qual." is the working directory name prefix.
ftp> LCD C:\dir
Local directory now C:\dir.
ftp> QUOTE SITE RECFM=FB LRECL=80 BLKSIZE=0 TRACKS PRI=450 SEC=75
200 SITE command was accepted
ftp> BINARY
200 Representation type is Image
ftp> PUT PJS310.XMIT
200 Port request OK.
125 Storing data set qual.PJS310.XMIT
250 Transfer completed successfully.
ftp: 17842560 bytes sent in 16.44Seconds 1085.12Kbytes/sec.
ftp> QUIT
221 Quit command received. Goodbye.
```

Task 2: Extract the PJS Installation Package PDS

The PJS Installation Package contains a partitioned data set that has been formatted as a TSO TRANSMIT file. This PDS, called the PJS Installation Package PDS may be extracted with the TSO 'RECEIVE' command with the 'INDATASET' parameter.

The following example shows the TSO 'RECEIVE' command that can be used to extract the PJS Installation Package PDS:

```
READY
RECEIVE INDATASET('qual.PJS310.XMIT')
INMR901I Dataset qual.DDNAME.PACKAGE from TCH03 on SYSTEME
INMR906A Enter restore parameters or 'DELETE' or 'END' +
DATASET('qual.PJS310.PACKAGE') NEW
INMR001I Restore successful to dataset 'qual.PJS310.PACKAGE'
READY
```

Your installation may also require additional parameters on the 'RECEIVE' command prompt response to specify how and where the data set should be allocated. For a complete list of all the options available see the *TSO/E Command Reference* for your release of OS/390 or z/OS.

The following table lists the members of the PJS Installation Package PDS:

Member Name	Description	Format
\$COPYRT	PJS Copyright Notice	Text
\$LICENSE	GNU General Public License	Text
\$LICPUB	GNU Free Documentation License	Text
\$README	“Getting Started” Instructions	Text
EXTRACT	Sample job to extract the PJS installation files	Text
PJSDOC	PJS Documentation	Zip
INSTALL	Sample Installation JCL Library	PDS
SMPMCS	SMP/E MCS Statements	Seq
JCLIN	SMP/E JCLIN	PDS
MACLIB	PJS Macro Library	PDS
SRCLIB	PJS Source Library	PDS
MODLIB	PJS Modules Library	PDS
TSOHELP	PJS TSO Help Library	PDS
ISPFPNL	PJS ISPF Panels Library	PDS
ISPFMSG	PJS ISPF Messages Library	PDS
ISPFMTBL	PJS ISPF Tables Library	PDS
SAMPLIB	PJS Samples Library	PDS
UCRLIB	PJS User Contributed Routines	PDS

The ‘Format’ column indicates the type of data in each member as follows:

Text = Text file
Zip = Zip file in TSO XMIT format
Seq = Sequential Data Set in TSO XMIT format
PDS = Partitioned Data Set in TSO XMIT format

The \$COPYRT, \$LICENSE, \$LICPUB, and \$README members are text files that contain important information about PJS that should be reviewed before completing the installation. The EXTRACT member contains a sample job that can be used to create the PJS installation files from the PJS Installation Package. Each of the other members contains one of the PJS installation files that has been formatted as a TSO TRANSMIT file.

Task 3: Review the \$COPYRT, \$LICENSE, and \$README Files

After extracting the PJS Installation Package you should carefully read the \$COPYRT and \$LICENSE members. These contain the copyright notice and license terms for using PJS. If you do not agree to these terms you will not be authorized to use PJS.

PJS is licensed under the GNU General Public License Version 2, or (at your option) any later version, as published by the Free Software Foundation. This license is widely used for many open-source software products, including Linux. A copy of the GNU GPL is also included in an appendix of this book. For more information about the GNU GPL see the Free Software Foundation website at <http://www.gnu.org/licenses/>.

You should also read the \$README member. This contains the latest information that might not have been included in the documentation.

Task 4: Extract the PJS Installation Files

The EXTRACT member of the PJS Installation Package PDS contains a job to extract the PJS installation files from the PJS Installation Package PDS. This job will run TSO as a background job and use the TSO 'RECEIVE' command with the 'INDATASET' parameter to extract each of the PJS installation files.

This job should be tailored to suit your installation standards, paying particular attention to the job card, and the allocation parameters on the TSO 'RECEIVE' command prompt responses. For a complete list of all the options available see the *TSO/E Command Reference* for your release of OS/390 or z/OS.

The following table lists the PJS installation files that are created by the EXTRACT job.

Data Set Name	Description	Format
qual.PJS310.PJSDOC	PJS Documentation	Zip
qual.PJS310.INSTALL	Installation JCL Library	PDS
qual.PJS310.SMPMCS	SMP/E MCS Statements	Seq
qual.PJS310.NPJ3100.F1	SMP/E JCLIN	PDS
qual.PJS310.NPJ3100.F2	PJS Macro Library	PDS
qual.PJS310.NPJ3100.F3	PJS Source Library	PDS
qual.PJS310.NPJ3100.F4	PJS Modules Library	PDS
qual.PJS310.NPJ3100.F5	PJS TSO Help Library	PDS
qual.PJS310.NPJ3100.F6	PJS ISPF Panels Library	PDS
qual.PJS310.NPJ3100.F7	PJS ISPF Messages Library	PDS
qual.PJS310.NPJ3100.F8	PJS ISPF Tables Library	PDS
qual.PJS310.NPJ3100.F9	PJS Samples Library	PDS
qual.PJS310.UCRLIB	PJS User Contributed Routines	PDS

These data sets are only required for the installation of PJS. Once installation is complete these data sets may be deleted, though it may be desirable to keep the INSTALL and UCRLIB data sets.

Task 5: Extract the PJS Documentation Files

If you download PJS it from the Internet as a Zip file, the PJS documentation may have been included as a separate file within that Zip file. If this is not the case you will need to get the PJS documentation from the PJS Installation Package.

In Task 4 you should have created a file named '*qual.PJS310.PJSDOC*'. This is a variable-length record sequential data set that is a Zip file containing the PJS documentation files. To extract and view these files you will need to download them to a workstation, typically with FTP. This transfer must be done in BINARY mode.

The following example shows the FTP commands that can be used to receive the PJS documentation Zip file to a PC from an OS/390 or z/OS host:

```
C:\>FTP hostname
Connected to hostname.
220-FTPD1 IBM FTP CS V1R4 at hostname, 16:41:20 on 2005-01-04.
220 Connection will close if idle for more than 20 minutes.
User (hostname:(none)): userid
331 Send password please.
Password: password
230 userid is logged on. Working directory is "userid".
ftp> CD 'qual.PJS310'
250 "qual.PJS310." is the working directory name prefix.
ftp> LCD C:\dir
Local directory now C:\dir.
ftp> BINARY
200 Representation type is Image
ftp> GET PJSDOC PJSDOC.ZIP
200 Port request OK.
125 Storing data set qual.PJS310.PJSDOC
250 Transfer completed successfully.
ftp: 3034126 bytes received in 3.70Seconds 820.92Kbytes/sec.
ftp> QUIT
221 Quit command received. Goodbye.
```

Once the file is on your workstation you will need to extract the documentation files using WinZip, or any other program that extracts Zip files. Once the documents have been extracted the documents shown in the following table will be available:

File Name	Description	Format
pjsinst.html	PJS Installation Guide	HTML
pjsinst.pdf		PDF
pjsinst.sxw		OOo
pjsmsg.html	PJS Messages and Codes	HTML
pjsmsg.pdf		PDF
pjsmsg.sxw		OOo
pjsuser.html	PJS User's Guide	HTML
pjsuser.pdf		PDF
pjsuser.sxw		OOo

The 'Format' column indicates the type of document in each file as follows:

- HTML = HTML Web Page. These files may be viewed with any standard web browser.
- OOo = OpenOffice.org 1.1.4 Text Document. These files may be viewed and edited with OpenOffice.org Writer. OpenOffice.org versions for Windows, Macintosh, Linux (X86 and PPC) Solaris (Sparc and X86), and FreeBSD are available free of charge at <http://www.openoffice.org>.
- PDF = Adobe Acrobat Portable Document File. To view these files you will need the free Adobe Acrobat Reader, available at <http://www.adobe.com/products/acrobat/readstep2.html>.

Continuing the Installation

From this point you should follow the instructions in Chapter 2 of the *PJS Installation Guide*.

Changes for PJS Release 3.1

Functional Changes

- PJS can now modify the JCL for a job request as it is being submitted. The user can specify a string appearing in the JCL that is to be replaced, and the replacement value. A replacement value may be a literal string, a '**Global Variable**' (which specifies a replacement value outside of the job request that uses it), or a '**Dynamic Value**' (which is computed at job submit time).

- A **PJS Message History Log** is provided to record messages sent to the user by the **PJS System Task**.
- The **Job Request-ID** can be specified by the user, instead of being generated by PJS. The Request Number is replaced by an alphanumeric **Request Name**. If a request name is not specified by the user when a job request is added, a numeric **Request Number** will still be generated by PJS.
- A 50-character **Job Request Description** may be specified for a job request. This description is used for documentation purposes only, and does not affect PJS processing in any way.
- A **Notify Message Level** may be specified for a job request, to specify the minimum message severity level for messages sent to the user by the PJS System Task. Messages not sent can still be retrieved from the **PJS Message History Log**.
- A **Notify Userid** may be specified for a job request, to specify the TSO Userid to which the PJS System Task is to send messages about the job request. The default is the **Owner-ID** of the job request.
- The user can now explicitly specify that the **Next Run Date and Time** is to be recalculated, without changing the **Start Date and Time**, **End Date and Time**, or **Frequency**.
- The PJS ISPF job request dialog panels have been reorganized to make room for new options, and to display more complete information about a job request as it is being added/modified. Several new sub-dialog panels have been implemented. The **Add Job Request** and **Modify Job Request** panels now display summary information about the job request, while most job request parameters are updated on sub-dialog panels.
- The PJS **Job Request Add** and **Job Request Modify** TSO commands have been changed to use a sub-command processing mode. The job request to be added or modified is created or read into storage when the **PJREQADD** or **PJREQMOD** command is entered. One or more sub-commands are then used to update the job request. When the job request is complete, the **END** sub-command is entered to complete the add or modify.
- When the **Next Run Date and Time** is recalculated by the ISPF interface, it is recalculated immediately, rather than waiting until the add or modify is completed.
- A **Display Job Request Internal Information** ISPF panel is provided to display internal information about a job request. This information can be valuable in debugging problems.
- The **Job Request List** TSO command DETAIL format has been changed to include the Added Status Change Date/Time, Spool Member, Spool Record Count, and Retry Count.
- When a **Calendar-ID** is entered on the **Specify Job Request Frequency** panel, the Calendar-ID will be checked for validity immediately, rather than waiting until the add or modify is completed.
- A 50-character **Calendar Description** may be specified for a calendar. This description is used for documentation purposes only, and does not affect PJS processing in any way.
- Security access to Job Requests, Calendars and Events can now be checked when they are entered by the user, instead of waiting until the add or modify is completed.

- The PJS messages have been renumbered. PJS messages are now numbered **PJSnnnnn** where *nnnn* is a 4 digit number.

Installation Changes

- The PJS System Task will now periodically record the current date and time in the PJS Request Queue. If the PJS Request Queue is damaged and restored from a backup tape, the PJS System task can detect that a long period of time has passed since PJS last processed the PJS Request Queue. The operator will be prompted for a recovery option. If the JOB Request Queue was restored from a backup, job requests that were scheduled to be submitted between the backup and the current date and time (which may have already been submitted before the PJS Request Queue was lost) can be placed in the HOLD status, to prevent them from being processed again until the user can 'clean-up' the job requests.
- PJS abend handling has been improved. When an abend occurs in the PJS System Task while submitting a job, the abend will be analyzed to determine if the PJS System Task can continue processing with the next job request, or if the PJS System Task should terminate. PJS will optionally produce an SVC dump when the abend occurs in an APF authorized environment. The PJS TSO command processors and the PJS utilities are now link-edited with AC=1. The PJS Installation Security Exit will be called when an abend occurs to allow the exit to clean-up any resources it has acquired.
- The PJS Request Queue records have been reformatted. A utility is provided to convert the PJS Release 2.x PJS Request Queue to the Release 0 format at installation time.
- Sample security and submit exits are provided to implement security in an ACF2 environment.
- The **DOWNWRN**, **HISTDSN**, **HISTRET**, **RTBLSZ**, and **SVCDUMP** parameters have been added to the PJS Installation Options Table.
- PJS may use system services that are not available on unsupported operating systems. OS/390 R10 is the earliest supported release. Compatibility with earlier releases has not been established.
- PJS is now using the Immediate and Relative Instructions Facility. PJS may not run hardware that does not support these instructions. It is believed that all such hardware is now out of support.
- The PJS documentation is now maintained with OpenOffice.org 1.1.4, rather than Microsoft Word. OpenOffice.org versions for Windows, Macintosh, Linux (X86 and PPC) Solaris (Sparc and X86), and FreeBSD are available free of charge at <http://www.openoffice.org>. The documentation is also provided in Adobe Acrobat PDF, and HTML formats.
- The internal formats of the PJS Options Table (**PJSOPT**) and the PJS Vector Table (**PJSVT**) have changed. The PJS Options Table and any user exits must be reassembled.

Problems Fixed

- The PJS Request Queue Maintenance Utility was deleting Job Requests in the ERROR and DISABLED status based on the Last Submit Date. This is frequently earlier than the date the Job Request was placed in the ERROR or DISABLED status. As a result these records were being deleted prematurely.

A 'Status Change Date and Time' has been added to the Job Request record. The PJS Request Queue Maintenance Utility will now delete Job Requests based on this date, rather than the Last Submit Date.

- PJS APAR OPJ0034 (List Events for Owner does not list Events when the corresponding Owner record does not exist) is fixed correctly.

The Job Request Add and Modify processes are changed to check that any added Events have Owner records. If not one will be created.