

[print](#) | [close](#)

## JOBLOGS EXPLAINED: Cleaning Up Pending Job Logs Using RMVPNDJOBL

[System iNetwork Systems Management Newsletter](#)

[Carsten Flensburg](#)

Carsten Flensburg

Wed, 01/07/2009 (All day)

In this issue, I will present the Remove Pending Job Log (RMVPNDJOBL) command, which allows you to easily perform a manual or scheduled cleanup of pending job logs. To illustrate the circumstances that make the RMVPNDJOBL command useful, I've gathered information from IBM's documentation of the release 5.4 introduction of the QLOGOUTPUT system value and the LOGOUTPUT job description and job attribute.

As of release 5.4, job logs are brought into the "on demand" world. The job logs are available when needed, but no work is done to produce job logs for which there is no need. Here's the story about how the different components act together.

A job's LOG parameter has three elements: the message (or logging) level, the message severity, and the level of message text. Each of these elements have specific values that, when combined, determine the amount and type of information sent to the job log by the job.

For example, the \*NOLIST value of the Text element causes no job log to be produced if the job ends normally. (The job log does not go into pending.) If the job ends abnormally (if the job end code is 20 or higher), a job log is produced. The messages that appear in the job log contain both the message text and the message help.

You can control what produces the job log. This is done with the job LOGOUTPUT parameter. When a job completes, one of three actions occur that affects how the job log is created. The following are values of the LOGOUTPUT parameter:

- \*JOBLOGSVR: The job log server produces the job log. Job logs will be generated by a server job after the job has completed its activity.
- \*JOBEND: The job itself produces the job log. If the job cannot produce its own job log, the job log will be produced by a job log server, but basically job logs will be generated as they are today during job termination.
- \*PND: The job log is not produced. The job log remains in pending until it is removed.

**Note:** These values do not affect job logs that are produced when the message queue is full and the job message queue full action specifies \*PRTWRAP.

When the job log is produced, messages in the job message queue are written to a spooled file from which the job log can be printed, unless the Control Job Log Output (QMHCTLJL) API was used in the job to specify that the messages in the job log are to be written to a database file. On a related note--if you are using the QMHCTLJL API, you should not use pending job logs. Using the control job log output API makes sense only if you are creating the job log when the job ends.

IBM generally recommends using \*JOBLOGSVR for best performance. Using \*JOBLOGSVR is fairly close to using \*JOBEND and tends to operate more smoothly (with less contention and less CPU spike).

Only use \*JOBEND if you have a dependency on the job log being available immediately after the job ends. When many jobs end at the same time, using \*JOBEND can result in a CPU spike as well as potential contention on the output queue.

\*PND keeps the job logs in a pending state until they are spooled or deleted from the system. Pending job logs reduce CPU and contention. However, they also consume a job structure even if you are detaching spooled files. The job descriptions for several server jobs are configured to use LOGOUTPUT(\*PND), helping to avoid CPU spikes and contention when a large number of server jobs end at the same time.

When a job starts, it gets its LOGOUTPUT value from the job description. If the job description specifies \*SYSVAL (the default for CRTJOB), the job uses the job log output value that is specified in the job log output (QLOGOUTPUT) system value. The QLOGOUTPUT system value defaults to \*JOBEND for compatibility with prior releases.

After the job has established its LOGOUTPUT job attribute, any changes to the job description or system value do not affect the active job. Changes to the system value or to the job description take effect for jobs entering the system after the change. You can use the Change Job (CHGJOB) command or API (QWTRCHGJB) to change the LOGOUTPUT job attribute after it has already been set in the job. Changes to the job take effect immediately.

Regardless of the method that you choose, the options for handling job logs are the same. You can set the job to not produce a job log (\*PND), have the job produce the job log (\*JOBEND), or have the job log server produce the job log (\*JOBLOGSVR).

As for pending job logs, there are a few options to clean up and remove these at the point where you have decided they are no longer needed:

- You can end the job with a value of 0 for the Maximum log entries (LOGLMT) parameter.
- 5.4 introduced the WRKJOBLOG command. You can delete job logs from this interface. In 6.1, you have the option to delete all job logs.
- If the job is already ended, you can run the Remove Pending Job Log (QWTRMVJL) API. The Remove Pending Job Log (RMVPNDJOB) command presented today is using this API to ease the process of cleaning up pending job logs. Specify a zero value for the "Pending job log retain days" parameter to clean up the job logs immediately.
- The automatic cleanup function (GO CLEANUP) deletes pending job logs in the same manner as spooled job logs; however, if you are not using the system's cleanup feature, then you must do your own cleanup, using one of the above options. Please note that cleanup is not optional, because if cleanup is not done, the system can run out of job structures or can fill with spooled files.

**On a side note:** Besides the automatic cleanup of system objects, you can also automatically clean up your own objects. To clean up user objects, do the following:

1. Use the Retrieve CL Source (RTVCLSRC) command to retrieve CL program QEZUSRCLNP.
2. Change the QEZUSRCLNP program source to do any cleanup of user objects.
3. Compile the QEZUSRCLNP program using the Create CL Program (CRTCLPGM) command.

The QEZUSRCLNP program will automatically be called whenever you specify Yes to "Allow automatic cleanup" on the Change Cleanup Options display presented when you run the Change Cleanup (CHGCLNUP) command.

Back to the RMVPNDJOBL command--here's what the command's prompt panel looks like:

```

                                Remove Pending Job Log (RMVPNDJOBL)

Type choices, press Enter.

Job name . . . . . *ALL          Name, generic*,
*ALL
User . . . . . *CURRENT      Name, generic*,
*CURRENT...
Number . . . . . *ALL        000000-999999, *ALL

Job log output . . . . . *PND      *ALL, *PND

Pending job log retain days . . *MAX      0-1827, *MAX

```

The Remove Pending Job Log (RMVPNDJOBL) command changes a completed job whose job log has not yet been written. The job messages are removed. The job log can no longer be produced or displayed. If the spooled file action for the job specifies that spooled files are to be detached or if the only remaining spooled files for the job are in independent auxiliary storage pools (ASPs), the job is removed from the system.

Please note that the RMVPNDJOBL command can remove pending job logs for all completed jobs, even those jobs that specify \*JOBEND or \*JOBLOGSVR for the job log output (LOGOUTPUT) job attribute.

The command and all its parameters are explained in detail in the accompanying help text panel group. Also note that the following restrictions apply to the RMVPNDJOBL command:

- The caller of this command must be running under a user profile that is the same as the user name specified in the command's job name (JOB) parameter.
- If the caller of this command is not the same as the user specified in the command's job name (JOB), the caller must be running under a user profile that has job control (\*JOBCTL) special authority.

The RMVPNDJOBL command is created from the following sources:

```

CBX985  -- RPGLE  -- Remove Pending Job Log - CPP
CBX985H -- PNLGRP -- Remove Pending Job Log - Help
CBX985X -- CMD    -- Remove Pending Job Log

```

Simply upload the source members to their respective source files and follow the compilation instructions in the source headers.

### **IBM job log documentation**

[Job logs](#)

[Managing job logs](#)

[Cleaning up job log pending](#)

[The QWTRMVJL API](#)

[Best Practices for Managing Job Logs on IBM I - Technote](#)

[Best Practices for Managing IBM i Jobs and Output - Redpaper](#)

[IBM V5R4 Work management summary presentation](#)

You can download the zip file containing all the code for this application [here](#)

**Source URL:** <http://iprodeveloper.com/systems-management/joblogs-explained-cleaning-pending-job-logs-using-rmvpndjobl>