



CCAPRINT

A Newsletter Excerpt for Model 204 Users

July 2007

USE OF AND ACCESS TO PRODUCTS AND FEATURES ARE IN ACCORDANCE WITH THE TERMS AND CONDITIONS OF THE USER'S SOFTWARE LICENSE. THE PRESENTATION OF MATERIAL HEREIN DOES NOT, IN ANY MANNER, MODIFY SUCH TERMS AND CONDITIONS.

High Performance REGENERATE Using ONEPASS

By James Damon

Sometimes you need to recover a file by means other than roll back/roll forward recovery. Media failures such as head crashes and other DASD errors, though very uncommon these days, can render a file unreadable. Backups offer some hope of recovery, but much updating activity may be lost if the backup is not current. Accidentally deleted datasets or roll back/roll forward recovery failures, which cannot be resolved for one or more files, are other examples.

The media recovery feature built into BATCH204 in the form of the REGENERATE command provides a solution to these problems. Using it you can restore a Model 204 file from a backup either taken with the Model 204 DUMP command or taken using a third party backup utility such as IEBGENER, FDR, DFDSS and others. Following the restoration process, the file can then be rolled forward using all CCAJRNLS supplied from runs which updated the file since the backup was taken. For example, if you took a backup of a file on Monday, then updated it on Monday, Tuesday, Wednesday and Thursday with a mixture of Online and Batch204 jobs and the file is lost somehow on Thursday, you can recover the file to its state as of the last update on Thursday-- **provided you saved all CCAJRNLS from those jobs.**

The REGENERATE Command Revised

The REGENERATE command has been available for quite a few releases and was enhanced in V5R1.0 to include the ONEPASS option. The ONEPASS option specifies to make only one pass through the journal data supplied in the CCAGEN DD statement. This may be a collection of concatenated or merged journals (see the MERGEJ utility in the *Model 204 System Manager's Guide*). Figure 1 shows the syntax of the REGENERATE command.

Figure 1.

```
REGENERATE [ONEPASS] FILE filename [FROM dumpname]
           [TO {last update
              | UPDATE nn
              | CHECKPOINT yy.kkk hh.h:mm:ss.th}]
```

The following fuzzy stopping points are not supported with the ONEPASS option.

- TO LAST UPDATE BEFORE ...
- TO LAST CHECKPOINT BEFORE ...

They require two passes through CCAGEN.

Benefits of ONEPASS Processing

Input to the REGENERATE command is supplied via the DD statement CCAGEN. This collection of CCAJRNL data may be quite large. It could be the CCAJRNL from one day of Online production (hundreds or thousands of cylinders) or much more, if CCAGEN represents multiple days of production Online and Batch204 runs. The ONEPASS option reduces the number of physical I/Os to CCAGEN by roughly 50% in all cases.

The results in Figure 2 were obtained from two identical REGENERATE runs, one using ONEPASS, the other not.

Figure 2.

	JOBNAME	STEPNAME	RC	EXCP	CONN
Without ONEPASS:	V61REGEN	NOONEPASS	04	20964	13708
With ONEPASS:	V61REGEN	ONEPASS	04	10793	7232
Without ONEPASS:					
17.21.46	JOB03210	IEF403I V61REGEN	-	STARTED	- TIME=17.21.46
17.22.09	JOB03210	IEF404I V61REGEN	-	ENDED	- TIME=17.22.09
(elapsed time 23 secs)					
With ONEPASS:					
17.06.17	JOB03155	IEF403I V61REGEN	-	STARTED	- TIME=17.25.17
17.06.39	JOB03155	IEF404I V61REGEN	-	ENDED	- TIME=17.25.30
(elapsed time 13 secs)					

With ONEPASS, the EXCP count was reduced by almost half, as was device connect time in milliseconds (CONN). The job's elapsed time, indicated by the IEF403I and IEF404I messages, was also reduced by almost half. The elapsed time reduction will vary according to system load, job priority and channel and device contention but the reduction in physical I/Os should always be around 50%.

In Summary

If you make any use of the REGENERATE command, definitely consider using the ONEPASS option. Unless you require the use of fuzzy stopping points, there is really no reason not to use ONEPASS. You will save elapsed time, EXCPs, and get your files back into production without unnecessary delay.