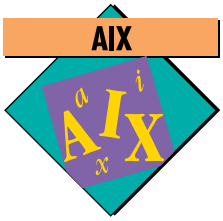
Using Accounting Checkpoints in AIX/ESA



By Bob Gensler

System administrators can now charge users for resources used by their applications during the normal accounting cycle, even if the applications have not yet terminated. This article discusses accounting checkpoint concepts as they relate to the Accounting Checkpoints Enhancement for AIX/ESA Version 2 Release 2. It also provides instructions for installing this package on an AIX/ESA system and for effectively using the new functions.

Accounting Checkpoints improves AIX/ESA accounting facilities by enabling the system to capture accounting information for both running and completed applications. A history of accounting information can now be obtained for long-running applications. System administrators can charge running applications for resource usage before the applications complete. Even if the operating system should fail, administrators can still charge incomplete applications for resource usage before the system failure. The package also gives administrators the flexibility to charge differing rates based on differing levels of CPU priorities used by these applications.

This package gives the AIX/ESA accounting system greater accuracy, historical accounting for running applications, and greater billing flexibility.

Overview

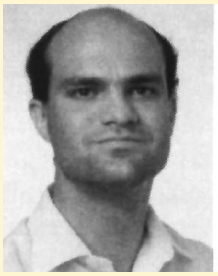
The AIX/ESA Accounting Checkpoints package creates interim process accounting records, which can be obtained for any user process. Using the package, system administrators can request that records be created at the following times:

- **At specific intervals**: For example, they can request that the operating system create interim accounting records 30 minutes after the process begins.
- **Upon demand**: The system administrator can request interim records by name for any process or user.
- **Whenever the `nice` command or system call changes the priority of a process**: A record is created containing the resources used under the previous priority level.

Interim process accounting records report changes in resources used by a process. The accounting information in each record shows the additional resource used by a process after the previous interim accounting record, or from the beginning of the process if no previous record exists. These records are placed in the process accounting output file, usually named `/var/adm/pacct`. The total record of resources used by a process can be obtained by adding the accounting information from all interim accounting records for that process.

The package also provides a new total accounting record format. These records, which report the complete accounting statistics for each user, now contain an additional field for process `nice` value. Using the proper commands, CPU-based resources can be separated by process `nice` value into different total accounting records for the same user. Since these records divide the CPU-based resources into different priority levels, system administrators can charge different rates for CPU resources used at different priorities.

Accounting Checkpoints adds four commands and two new C library system calls to AIX/ESA. In addition, to take advantage of new features in the package, seven existing commands have been extended. The package also provides complete man pages for the new and enhanced commands. Figure 1 shows highlights of these modifications.



Bob Gensler

Name	Type	Description
acctint()	New system call	Enables or disables the Accounting Checkpoints function; sets the default accounting checkpoint interval.
acctint	New command	Represents a command-line interface to the acctint system call.
acctch()	New system call	Creates interim process accounting call checkpoint records for one or more processes or users.
acctch	New command	Represents a command-line interface to the acctch system call.
ckptmerg	New command, migration aid	Summarizes all interim process accounting records for completed processes; imitates the behavior of the traditional AIX accounting package after Accounting Checkpoints has been enabled.
tacctconv	New command, installation tool	Converts existing files in the total accounting (tacct) record format to the extended total accounting record format; must be executed after installing the Accounting Checkpoints package.
acctcms	Enhanced command	Represents two new command-line options, -m and -w, which have been added to load and save the command's working storage between executions.
acctprc1	Enhanced command	Adds a reason code and a process nice value to the command's output.
acctprc2	Enhanced command	Handles the new, 12-column command input from acctprc1; a new option, -n, sorts output by uid/logname/nice.
acctcom	Enhanced command	Summarizes, by default, all interim process accounting records into summary records before processing the input; a new -x option processes input (interim records) instead of summarizing the raw input; a new -p option reports information specific to interim accounting records.
lastcomm	Enhanced command	Summarizes all interim process accounting records for completed processes before generating its output; a new -s option skips this summation step, generating only a list of completed commands.
acctmerg	Enhanced command	Represents a new option, -n, that summarizes output by process nice value, in addition to the summation technique selected by other options to this command; a nice value column has been added to the command's output.
runacct	Modified shell script	Adds an option -n to sort the daily accounting file by uid/logname/nice; by default, sorting is by uid/logname.

Figure 1. New and enhanced commands in Accounting Checkpoints

Starting It Up

If your AIX/ESA system has been configured to enable Process Accounting during system startup, the Accounting Checkpoints can also be activated at that time. The Accounting Checkpoints package installs a new file called `/etc/acct_interval` on your system that indicates whether Accounting Checkpoints should be automatically activated. This ASCII file contains a single-integer value that is used as the argument to the `acctint` command during system initialization.

If the `/etc/acct_interval` file contains 0, Accounting Checkpoints will not be activated during system initialization; Process Accounting will behave in the traditional manner, creating process accounting records only if a process terminates.

To activate Accounting Checkpoints during system initialization, change the value in the `/etc/acct_interval` file to the desired interval in minutes. This interval specifies the time between automatic creations of interim accounting records for an application. During system initialization, the accounting checkpoint interval will be set to match this value.

If your system does not automatically enable Process Accounting during system initialization, you can activate Accounting Checkpoints after system initialization. Simply use the `acctint` command and specify the accounting checkpoint interval as the argument. For example, to create interim accounting records for each process every 30 minutes, enter the following:

```
/usr/lbin/acct/acctint 30
```

The `acctch` command can create interim process accounting records even if Accounting Checkpoints has not been activated. If you do not

want the system to automatically create interim records at frequent intervals, you can choose not to activate Accounting Checkpoints and still execute the `acctch` command to create interim accounting records for processes or users. Accounting Checkpoints must be active for records to be created whenever a process changes its own `nice` value. Process Accounting must be activated for any interim records, either automatic or manual, to be created.

Selecting the Checkpoint Interval

Accounting Checkpoints was designed to capture accounting information at regular intervals for long-running applications. Although the accounting checkpoint interval can be any positive, non-zero integer, this value should be selected with care. If the interval is too small, medium-sized applications may create records more frequently, causing the accounting output files to fill rapidly. If the interval is too large, you may lack up-to-date information when performing accounting duties for applications that are still running.

The choice of an accounting checkpoint interval depends on your definition of a long-running application and when you perform accounting and chargeback duties. Select an interval that will cause the system to create at least one interim process accounting record for every long-running application by the time you perform your accounting duties.

An initial interval of 60 minutes is recommended. This interval will avoid creating interim accounting records for short applications, such as the `ls` command, while creating records for applications running longer than one hour. After you become comfortable with the Accounting Checkpoints concept and have a history of application runtimes for your system, you can then tailor the accounting checkpoint interval to your specific needs.

The `acctint` command adjusts the accounting checkpoint interval. To change the accounting checkpoint interval from 30 minutes to one hour, enter the following:

```
/usr/lbin/acct/acctint 60
```

The changed accounting checkpoint interval takes effect on all user applications that begin after the `acctint` command has been issued. Applications that were already running will continue to use the previous accounting checkpoint interval until that interval expires. Once the interval expires, the revised accounting checkpoint interval will be used to schedule the creation of the next interim record.

Using Interim Records

Interim process accounting records capture resource information for running applications. AIX/ESA then collects these interim records during the accounting chargeback cycle and uses them to create total accounting records for each user. By collecting interim records during chargeback cycles, total accounting records allow system administrators to charge running applications for system resources used during previous accounting periods as well. In addition, if an operating system should fail while applications are still running, this process enables system administrators to charge any uncompleted processes for all resources used before the system failure.

To create these records automatically, system administrators must set the accounting checkpoint interval using either the `/etc/acct_interval` file or the `acctint` command. AIX/ESA will then create an interim accounting record for all user processes begun after this interval is set, either when the application's `nice` value changes or when its interval expires.

Interim records can be created upon demand with the `acctch` command. System administrators can request interim accounting records in the following categories: a single process, a list of processes, a process group, a user, a list of users, or all current user processes.

Using interim process accounting records is not difficult; all accounting commands and scripts have been modified to handle these new records and to take advantage of the record's new features. To begin using interim process accounting records, simply activate the Accounting Checkpoints feature. The accounting system will then take over, requiring very little intervention from system administrators.

When creating interim accounting records automatically, the accounting checkpoint interval and the start time of the process determine the exact time that an interim record will be created.

By adding the accounting checkpoint interval to the start time of the process, the AIX/ESA kernel calculates the time at which the first record will be created. When this time expires, the kernel adds it to the accounting checkpoint interval to determine when the next record will be created. The cycle repeats until the process completes.

Occasionally, system administrators may not want to wait for these records to be automatically

Accounting Checkpoints was designed to capture accounting information at regular intervals for long-running applications.

created, preferring instead to create interim accounting records for one or more processes. They can use the `acctch` command to target processes, process groups, or users. The kernel will schedule interim process accounting records to be created for all processes named by the command, or for all processes owned by the user named by the command.

For example, consider the following scenario. As system administrator, you want to perform billing for the user bobgens at 1 A.M. Since you want the billing to be up-to-date, you need the most recent information on any running applications for this user. However, some applications used by bobgens may not yet have created interim process accounting records. Instead of waiting for these records to be created automatically, you could issue the following command to gain the latest accounting information for this user:

```
/usr/lbin/acct/acctch -u bobgens
```

The total accounting record contains resource usage information for all users that ran applications on the system during the accounting period.

Within a few seconds, all running processes owned by user bobgens will have created interim process accounting records in the process accounting output file.

Although the command schedules the creation of interim accounting records, not all records may be created by the time the `acctch` command completes. To have its interim record created, an application must be using the CPU at that time; suspended or waiting applications will not create interim accounting records until they begin running again. You should allow a small amount of time—usually just a few seconds—after the completion of the `acctch` command before expecting interim process accounting records to be available.

Using the Extended Total Accounting Record

The total accounting record provides information for charging customers for system usage. These records contain resource usage information for all users that ran applications on the system during the accounting period.

With Accounting Checkpoints installed, the operating system can create multiple total accounting records for a single user. One total accounting record can be created for each `nice` value selected by the user during the previous accounting period. CPU-related resources, such as primetime and non-primetime shift work, are split according to the `nice` value used by these resources and reported in separate records for each `nice` value. All non-CPU related resources, such as printer or disk resource usage figures, are reported in the total accounting record for the zero `nice` value. The total CPU-related resources for a user can be calculated by adding the CPU-resource usage figures from all the user's total accounting records.

By dividing the CPU-based resources into separate total accounting records, Accounting Checkpoints enables system administrators to charge different rates for CPU resources used at different levels of priority. Charging less expensive rates for CPU-based resources used at a lower priority can encourage users to lower the priority of their applications more frequently.

Installing the Package

The Accounting Checkpoints package can be installed only on AIX/ESA Version 2 Release 2 operating systems. There are no special hardware requirements for this package, but a complete list of software requirements and requisites is provided in the Accounting Checkpoints installation documentation.

You should schedule some system downtime to install the Accounting Checkpoints package. Installing the package requires a rebuild of the kernel, and some existing accounting data files must be converted to a new format. Single-user mode is recommended to perform the installation and file conversions. The package is installed using the `installp` command.

Accounting Checkpoints will change the format of the total accounting (`tacct`) record. If the system has been operational before the installation of Accounting Checkpoints, you will probably have several files residing on the system that use the older total accounting record format. These files, created by the `runacct` shell script and the `acctprc2` command, are used as input to the `acctmerg`, `prdaily`, and `monacct` commands. Each of these commands has been modified in the Accounting Checkpoints package. For these commands to function properly, the existing data files must be converted to the new total accounting record format after the Accounting Checkpoints package has been installed, but before bringing the system back into multi-user mode.

If applications generate total accounting record files, these data files may be in some nonstandard locations. These files must be manually located and converted to the new `tacct` format. By default, the AIX/ESA accounting commands generate the following total accounting record files:

```
/var/adm/acct/fiscal/tacct.mm
/var/adm/acct/sum/tacct
/var/adm/acct/sum/tacct.mmdd
/var/adm/acct/sum/tacct.prev
/var/adm/acct/sum/ptacct.mmdd
/var/adm/acct/nite/ctacct.mmdd
/var/adm/acct/nite/daytacct
/var/adm/acct/nite/ptacctn.mmdd
```

Convert these files to the new total accounting record format with the `tacctconv` command, before bringing the system back to multi-user mode and before using any accounting commands. Figure 2 shows an example of how to do this. Once these data files are converted, the system can be brought to multi-user mode, and the accounting commands can be used.

Migration and Compatibility

Until now, one process accounting record represented the entire execution of an application, because such records were created only when an application terminated. With the Accounting Checkpoints enhancement, an application can now create one or more process accounting records during its life. The accounting commands have been changed to remove the basic assumption of "one record, one execution," but your system may contain local applications that were designed using this assumption. If such applications exist on your system, they need to use the migration aids included in the Accounting Checkpoints package to work properly.

Accounting Checkpoints also implements a new process accounting record format called the *acct structure*, described in the `<sys/acct.h>` header file. Your system may have applications that use this record; any applications that examine the process accounting output file `/var/adm/pacct` use this record format.

The `ckptmerg` command is provided for those who do not wish to update their local applications to handle the new record format or interim process accounting records. This command converts the file to the format expected by existing applications. Summary process accounting records are created for completed applications in either the new extended record format or the Release 2 format. Existing commands can access the accounting output file by using this command as a filter, as shown below (enter as a single command):

```
/usr/lbin/acct/ckptmerg -c
/var/adm/pacct |
old_accounting_appl
```

```
mv /var/adm/acct/sum/tacct /var/adm/acct/sum/tacct.OLD
/usr/lbin/acct/tacctconv /var/adm/acct/sum/tacct.OLD >
    /var/adm/acct/sum/tacct
```

Figure 2. Using tacctconv after installation

When using this command as a filter, you should realize that some long-running applications may have created interim process accounting records that exist in several accounting output files. It is possible for an application to begin while one accounting output file was active, and complete when another output file was active. For the `ckptmerg` command to function properly, it must locate all interim process accounting records for the application. You may need to supply several files as arguments to the command to obtain summary records for certain long-running applications.

The package also changes the format and nature of the total accounting record for users. Users may now have more than one total accounting record generated for them. Applications that manipulate these records must be updated to handle the new format of the total accounting record and to expect multiple records for each user for the same accounting period. The new total accounting record format is specified by the tacct structure in the `<tacct.h>` header file. Applications that use the ASCII version of total accounting records should also be redesigned.

Using the Accounting Checkpoints Package

The following example demonstrates how Accounting Checkpoints can be used in your daily accounting routines. This example is made with the following assumptions:

- The operating system is AIX/ESA Version 2 Release 2 with the Accounting Checkpoints enhancement installed and all necessary file conversions made.
- The system administrator's definition of a long-running job is any application that takes longer than an hour to complete.
- Process accounting is started automatically during system Initial Program Load (IPL).
- The `runacct -n` shell script is scheduled by `cron` to run at 11 P.M. daily.

Before the last IPL of the system, the system administrator changed the contents of the

UID	LOGNAME	PRI_CPU	NPRI_CPU	PRI_MEM	NPRI_MEM	...	NICE
				:			
				:			
101	bobgens	58.7800	10.3460	4255566	890040	...	0
101	bobgens	6.4240	0.0000	0	0	...	2
101	bobgens	12.3570	3.7550	0	0	...	5
				:			
				:			

Figure 4. /var/adm/acct/sum/rprt.1118 file entries for user bobgens (edited)

```
(non-CPU resources used * non-CPU resource usage rate) +
(CPU resources used at default nice value * default CPU usage rate) +
(CPU resources used at nice value 2 * nice value 2 usage rate) +
(CPU resources used at nice value 5 * nice value 5 usage rate)
```

Figure 5. Total charge to user bobgens

etc/acct_interval file from 0 to 60. Since process accounting was automatically started by the operating system during IPL, Accounting Checkpoints was also automatically activated during the last IPL of the operating system. Interim records are now created every 60 minutes for applications that run longer than an hour.

To ensure that the system has the most recent process accounting information at its disposal before the runacct script begins, the system administrator schedules the acctch command to create interim records at 10:55 P.M. for all running applications with the following crontab line:

```
55 22 * * * /usr/lbin/acct/acctch 0
```

With this command, the system will force the creation of interim process accounting records for all running applications at 10:55 P.M. These interim records will be placed in the accounting output file var/adm/pacct and will be processed by the runacct script.

10:55 P.M., November 18: The acctch command forces interim process accounting records to be created for all applications currently running on the system.

11 P.M. that same night: The runacct shell script begins and performs all the accounting chores, creating the /var/adm/acct/sum/rprt.1118 and var/adm/acct/sum/tacct.1118 files. The runacct shell script specified the -n option, resulting in one total accounting record per user, per process nice value, to be created in the output files.

Morning, November 19: The system administrator processes the output file var/adm/acct/sum/tacct.1118 and begins to charge back resource usage to the system's users. The chargeback program finds multiple records for a single user in the ouput file, /var/adm/acct/sum/rprt.1118, shown in Figure 4. The chargeback program assigns a lower usage rate for the CPU resources used by bobgens at nice values of 2 and 5, rewarding that user for his conscientious behavior. Figure 5 shows the total charge made to user bobgens.

Bob Gensler, IBM Corporation, Enterprise Drive, Kingston, NY 12401. Internet: bobgens@minnie.nic.kingston.ibm.com. Mr. Gensler is a software developer for Power Parallel Systems and former lead developer on the AIX/ESA Accounting Checkpoints project team. He has a BS in Computer Science from Central Connecticut State University in New Britain.