UNICOS Accounting Futures and Generic Report Generation

Charles R. Smith, Cray Research Inc.

ABSTRACT: *This paper discusses the UNICOS 9.0 feature, "generic front end formatting," and some of the future changes which UNICOS Accounting will undergo as it is implemented on a UNICOS/mk system.*

UNICOS Accounting allows a site to do production style accounting in a manner that is not possible on any other UNIX system today. With UNICOS 9.0, UNICOS Accounting will be extended to include the new "generic front end formatting programs," csagcon and csagfef. This paper begins by discussing the background on why these programs are needed and the way in which they fit into UNICOS Accounting.

Following the discussion of report generation, some of the key differences between the UNICOS Accounting package as it exists today and the initial UNICOS/mk Accounting package will be discussed. This is a very brief overview, but should aid you by pointing out some of the key differences.

Why Generic Report Generation

Under UNIX and UNICOS today there are a number of different programs which can be used to generate accounting reports (See Slide 1). While these reports have met many customer needs, there are a number of key features which are missing from the standard (UNIX) accounting package and from Cray System Accounting (CSA).

Both UNIX accounting and CSA read pacct files (CSA also makes use of additional daemon supplied data) and generate either reports or consolidated files from which an additional set of reports can be generated at a later time. However, UNIX accounting has very limited options for formatting, and while CSA offers a number of extensions, neither accounting package allows sufficient customization for all sites.

The difficulty of creating custom reports is compounded by the limits imposed by source licensing requirements. Both UNIX and CSA accounting have requirements for source licensing. As a result, a binary site is limited to the existing functionality of the accounting systems. Even CSA's template commands are unavailable to binary sites.

To address these issues we developed the CSA "Generic Front End Formatting Package." This package consists of the two programs, csagcon and csagfef. These programs allow a user to select the data to be consolidated and select the format of the final report.

How Generic Formatting Works

The generic formatting package works in two phases. The first phase is data consolidation, which is followed by the report generation phase. The csagfef and csagcon commands are easily customized to alter the end results.

Slide 2 shows the flow of data through the csagcon and csagfef programs.

Csagcon reads accounting data from either a pacct file or from a session record file. Based on the contents of the data request file, csagcon constructs a file containing the requested data, after consolidating that data based on options (for job, id, or class).

The output file from csagcon is used as input to csagfef. Csagfef reads the consolidated file and the script file. Based on the contents of the script file, csagfef formats a report containing the data in the consolidated file.

An Example Report

At the end of this paper is the script for a simple csagfef example that duplicates the functionality of the csacon and csacrep programs. This example and a few others are contained in the UNICOS release package.

The example contains a very basic structure which consists of a BEGIN, and a BODY section. For more information on the language used by csagfef, you can see the administration ma, you can examine the documentation for the tsar program. The csagfef program uses a report generation language very similar to the language used by tsar.

The example also demonstrates the use of built in functions to calculate unit conversions, id conversions, and time conversions. Csagfef contains a large number of built-in functions which will aid the programmer in designing reports.

Csagfef also allows writing of binary data fields. Binary fields must be properly aligned by the user and must be 64 bit fields. Post processing is necessary if the fields need to be converted to a different word size. However, this is much easier than processing raw accounting data since the user has control over the format and the conversion process. A simple example of this is given as part of the UNICOS 9.0 release package.

Additional Materials on Csagcon and Csagfef

A detailed description of csagcon and csagfef can be found in the UNICOS 9.0 System Administrator's Guide section on Accounting.

UNICOS Accounting Futures on UNICOS/mk

The next major evolutionary change for UNICOS will be the change to a "serverized" system architecture. This change is described in two other papers presented at this conference. Briefly, the major subsystems of UNICOS will be changed into "servers" which are separate pieces of code. This provides a number of benefits. However, from an accounting and billing standpoint, this operating system change will also require changes to the accounting system. This section of the paper describes differences which may be visible in the earliest releases of UNICOS/mk,. Please note that as UNICOS/mk becomes more mature, some of these differences will disappear, as the functionality is implemented in UNICOS/mk.

UNICOS/mk Accounting will be very similar to UNICOS Accounting in it's initial release. However, there will be a few differences of which you should be aware. The purpose of this paper is to help define some of the changes at a high level, and explain what the longer term plan for accounting looks like.

Our current plan is to port the functionality of UNICOS Accounting to UNICOS/mk in stages. Initially, we will port the most commonly used functions of UNICOS to UNICOS/mk. These functions include UNIX style accounting commands, UNICOS extensions for disk accounting, project or accounting ids, the ja command, and the CSA programs.

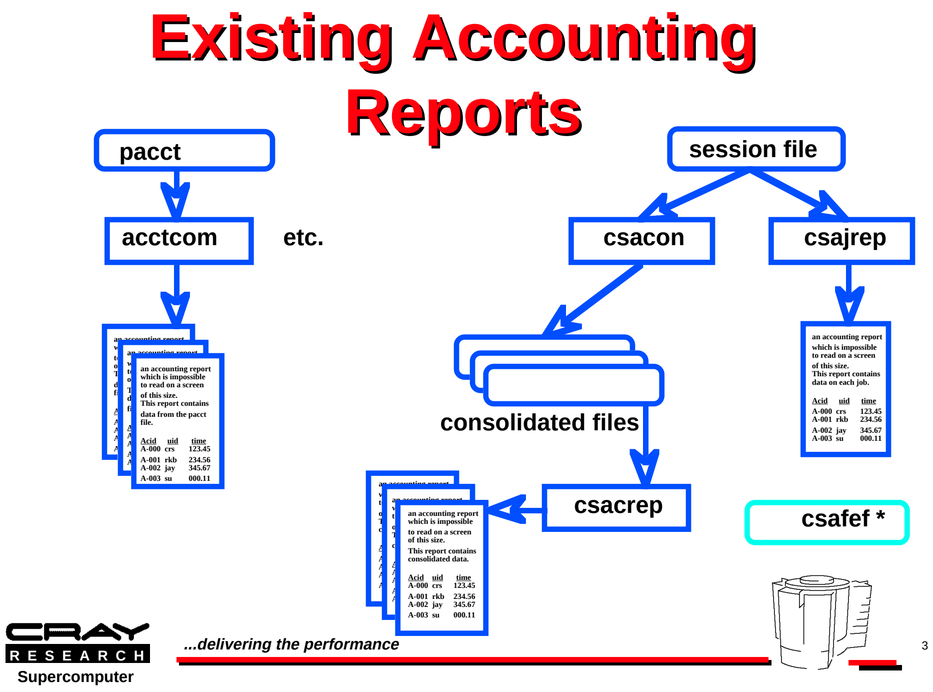
While the list above indicates that most of UNICOS Accounting will be present in UNICOS/mk, there will be some differences which you need to be aware of. The differences include:

- Device accounting records will be removed. Device accounting will be replaced by system wide device data in the SA package.
- Performance accounting records will be removed.
- Hardware device records (T3D and at least initially SDS) records will be removed.
- Accounting record formats will change.
- Accounting will record data in machine independent units.
- Multi-tasking record removed (initially)
- New record types to aid in accounting resiliency and problem diagnosis will be added.
- USCP records removed.

The reason for these changes is that some of the existing UNICOS functionality no longer is supported, or no longer makes sense in the UNICOS/mk environment. In addition, some of the internal changes above are being made to address long requested changes to accounting (additional resiliency). The remaining changes are being made because the existing UNICOS functionality makes little sense or is very difficult to implement properly in the UNICOS/mk environment. We expect to implement some of the items above in later releases, as UNICOS/mk comes to maturity.

In addition, to these changes, several commands will have altered behavior (new output formats, due to new data). In order to accomplish this with minimal disruption, the commands will all have a compatibility mode which will allow the user to select the new or old modes of operation. The administrator will also be able to configure the default mode of operation.

Finally, it is worth mentioning that we are designing to accommodate additional accounting functionality in the future. Papers in the future will detail these changes. At this time we are looking at extensions for MPP systems, application accounting, and incremental accounting as the primary areas in which we may extend the current accounting functionality.



Existing Accounting Reports

pacct → acctcom → etc.

an accounting report which is impossible to read on a screen of this size. This report contains data from the pacct file.

Acid	uid	time
A-000	crs	123.45
A-001	rkb	234.56
A-002	jay	345.67
A-003	su	000.11

session file → csacon, csajrep

csacon → consolidated files → csacrep

an accounting report which is impossible to read on a screen of this size. This report contains consolidated data.

Acid	uid	time
A-000	crs	123.45
A-001	rkb	234.56
A-002	jay	345.67
A-003	su	000.11

csajrep

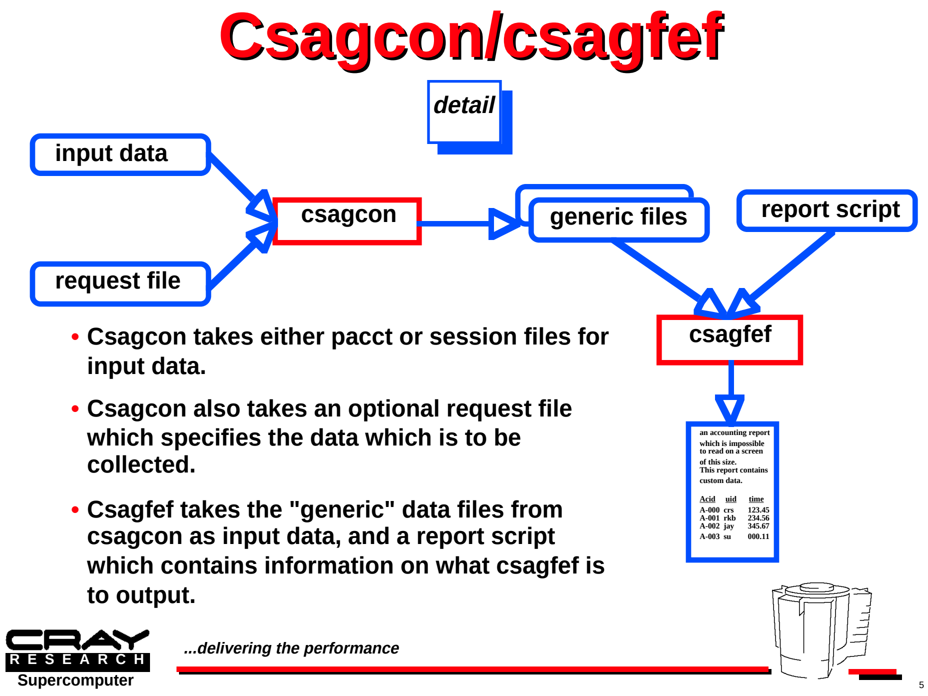
an accounting report which is impossible to read on a screen of this size. This report contains data on each job.

Acid	uid	time
A-000	crs	123.45
A-001	rkb	234.56
A-002	jay	345.67
A-003	su	000.11

csafef *

CRAY RESEARCH Supercomputer ...delivering the performance 3

Slide 1



Csagcon/csagfef

detail

input data, request file → csagcon → generic files

generic files, report script → csagfef

- Csagcon takes either pacct or session files for input data.
- Csagcon also takes an optional request file which specifies the data which is to be collected.
- Csagfef takes the "generic" data files from csagcon as input data, and a report script which contains information on what csagfef is to output.

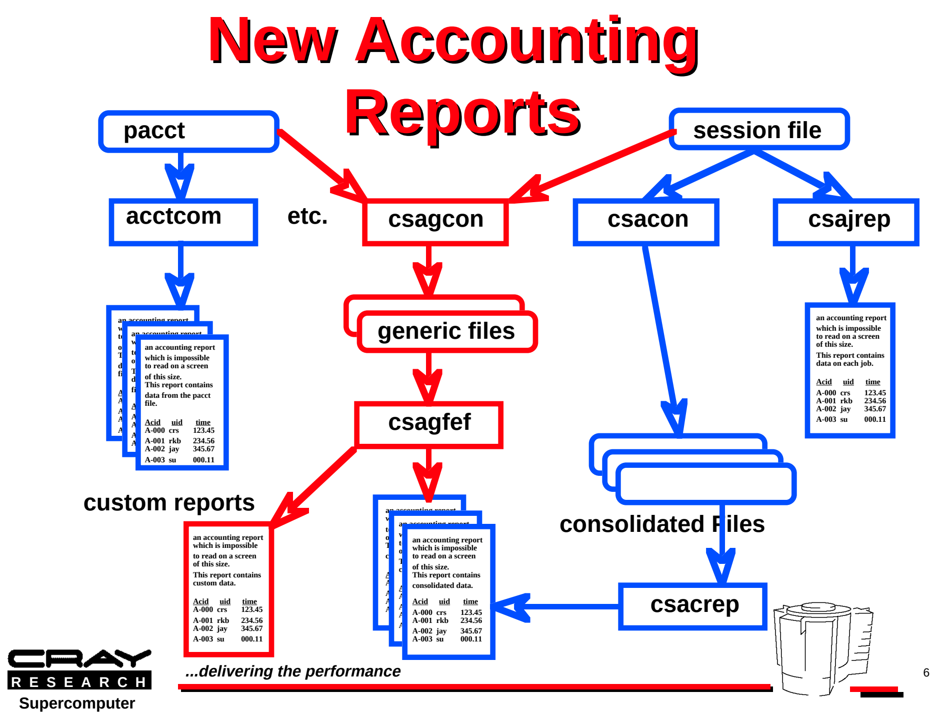
an accounting report which is impossible to read on a screen of this size. This report contains custom data.

Acid	uid	time
A-000	crs	123.45
A-001	rkb	234.56
A-002	jay	345.67
A-003	su	000.11

CRAY RESEARCH Supercomputer ...delivering the performance 5

Slide 2

New Accounting Reports



pacct

acctcom

etc.

csagcon

session file

csacon

csajrep

generic files

csagfef

custom reports

consolidated Files

csacrep

an accounting report which is impossible to read on a screen of this size. This report contains data from the pacct file.

Acid	uid	time
A-000	crs	123.45
A-001	rkb	234.56
A-002	jay	345.67
A-003	su	000.11

an accounting report which is impossible to read on a screen of this size. This report contains data on each job.

Acid	uid	time
A-000	crs	123.45
A-001	rkb	234.56
A-002	jay	345.67
A-003	su	000.11

an accounting report which is impossible to read on a screen of this size. This report contains custom data.

Acid	uid	time
A-000	crs	123.45
A-001	rkb	234.56
A-002	jay	345.67
A-003	su	000.11

an accounting report which is impossible to read on a screen of this size. This report contains consolidated data.

Acid	uid	time
A-000	crs	123.45
A-001	rkb	234.56
A-002	jay	345.67
A-003	su	000.11

CRAY RESEARCH

Supercomputer

...delivering the performance

6

Slide 3

Example using csagfef to duplicate csacrep

Commands:

```
        csagcon -au -S input_session_file > generic_output_file
        csagfef
```

```
BEGIN {
#
#       Determine which sessions were consolidated.
#
if (con_key & 0100) {
        CONSOL = "ACTIVE AND COMPLETED SESSIONS"
} else if (con_key & 0200) {
        CONSOL = "ONLY ACTIVE SESSIONS"
} else {
        CONSOL = "ONLY COMPLETED SESSIONS COMPLETED"
}
#
#       Print the headers.
#
printf("%s   DAILY REPORT FOR %s (Rel %s, %s)\n\n",
  strftime("%c", creatime), SYSNAME, RELEASE,
  VERSION)

printf("INCLUDES DATA FOR %s BETWEEN\n", CONSOL)
printf("    %s AND %s\n\n",
  strftime("%c", file_start), strftime("%c", file_end))

printf("ACCOUNT  USER     LOGIN    CPU-TIM  KCORE*   ")
printf("CONNECT IOW-TIM   KCORE*  \n")
printf("NAME     ID       NAME     [SECS]   CPU-MIN  ")
printf("[SECS]   [SECS]   IOW-MIN \n")
printf("======== ======== ======== ======== ======== ")
printf("======== ======== ========\n")
}
#
#       For each data record, extract, convert, and output
#       the desired fields.
#

con_time = ct_con_p + ct_con_n
kcore = pb_p_kcore + pb_n_kcore

#       Convert clocks to seconds.
cpu_time = pb_p_stime + pb_n_stime + pb_p_utime + pb_n_utime
cpu_time = clocks_to(cpu_time, SEC)
iowtime = pb_p_iowtime + pb_n_iowtime
iowtime = clocks_to(iowtime, SEC)

#       Convert click-ticks to kiloword-minutes (kcore-minutes).
iowmem = pb_p_iowmem + pb_n_iowmem
iowmem = ticks_to(clicks_to(iowmem, KW), MIN)

printf("%-8.8s %-8d %-8.8s ", acid2nam(acid), uid, uid2nam(uid))
printf("%8.0f %8.0f %8d %8.0f %8.0f\n",
  cpu_time, kcore, con_time, iowtime, iowmem)
```