A/UX: Data Structure of sa Manual Correction

Article Created: 22 May 1990

Article Change History

08/31/92 - REVIEWED
- For technical accuracy.

TOPIC -----

We've tried to use the records in the "sardd" files (directory /usr/adm/sa) to monitor some system resources and to get information about freemem, freeswap, and other dynamically changing parameters.

The customer found that the structure of the records written by "sadc" and the description of this structure in "sadc(1M)" did not match. I checked and found that the records written by "sadc" are 228 bytes on our system and the size of "sa" (System Accounting), as described, is 304 bytes.

Please send me the structure of the information contained in the records of /usr/adm/sa/sardd.

DISCUSSION -----

We researched the data structure of "sa", and it seems that the one described in the "sadc(1M)" manual page is wrong; the "sa" data structure defined in the sa.h header file and used by the "sadc" program has the following structure:

```
struct sa {
    struct  sysinfo si;   /* defined in /usr/include/sys/sysinfo.h */
    int     szinode;      /* current size of inode table */
    int     szfile;       /* current size of file table */
    int     sztext;       /* current size of text table */
    int     szproc;       /* current size of proc table */
    int     mszinode;     /* maximum size of inode table */
    int     mszfile;      /* maximum size of file table */
    int     msztext;      /* maximum size of text table */
    int     mszproc;      /* maximum size of proc table */
    long    inodeovf;     /* cumulative overflows of inode table since boot */
    long    fileovf;      /* cumulative overflows of file table since boot */
    long    textovf;      /* cumulative overflows of text table since boot */
```

```
    long    procovf;      /* cumulative overflows of proc table since boot */
    time_t  ts;           /* time stamp */
    long    devio[NDEVSII4];   /* device unit information */

    #define IO_OPS  0    /* number of I/O requests since boot */
    #define IO_BCNT 1    /* number of blocks transferred since boot */
    #define IO_ACT  2    /* cumulative time in ticks when drive is active */
    #define IO_RESP 3 /*cumulative I/O response time in ticks since boot */
};
```

The total size of structure "sa" is 740. However, the record size of "sa" is computed as below:

```
   recsz = sizeof (struct sa) - sizeof d.devio + recsz * sizeof d.devio[0];
```

The result of recsz from the above equation = 740 - 512 + 0 = 228

However, the first record of the data file will have the size of 232, which includes a 4-byte header at the beginning of the data file. The rest of the records have a record size of 228. If you monitor the size of data file, the size will have this order: 232, 460, 688, 916, 1144, and so on.

Concerning the record of the system accounting report file (/usr/adm/sa/sardd), the "sar" program generates different sizes of output files (from the input file "sadd" created with "sadc" program) depending on the options specified in the "sar" command line. By default, without any option, the "sar sadd-file" reports with the CPU utilization:

```
   time-stamp   %user     %sys     %wio    %idle
```

If the -A option (all the options -uqbwcayvm) is used, it generates all supported information, like -b for reporting buffer activity, -y for reporting the tty device activity, and so on.

We reported the inconsistency of the "sa" data structure specified in "sadc(1M)" to A/UX Engineering.

Keywords: <None>

==========

19960215 11:05:19.00
Tech Info Library Article Number: 5581