Subject: bug in systime(1) on IDL 5.2 Linux/Intel?
Posted by woodford on Tue, 23 Feb 1999 08:00:00 GMT
View Forum Message <> Reply to Message

systime(1) on IDL 5.2 Linux/Intel seems to always yield whole numbers.
For instance,

print, systime(1) - long(systime(1))

always returns 0.0000... In contrast, this statement returns various
fractions in IDL 5.2 WinNT/Intel and IDL 5.0 Mac. Has anyone else
observed this behavior? It wreaks havoc with trying to use time_test3 and
graphics_times3 for benchmarking.

--
Paul Woodford
woodford@essexcorp.com

Subject: Re: bug in systime(1) on IDL 5.2 Linux/Intel?
Posted by J.D. Smith on Mon, 01 Mar 1999 08:00:00 GMT
View Forum Message <> Reply to Message

Ivan Zimine wrote:
>
> I have the same behavior...
>
> IDL> print, !version
> { x86 linux unix 5.2 Oct 30 1998}
> IDL> t=systime(1) & f=dialog_pickfile() & print, systime(1)-t
> 3.0000000
> IDL> t=systime(1) & f=dialog_pickfile() & print, systime(1)-t
> 1.0000000
> etc 10 times
>
> Couldn't get any fractional part. I doubt that my internal timing is so
> perfect.
>
> "Woodford, Paul" wrote:
>>
>> systime(1) on IDL 5.2 Linux/Intel seems to always yield whole numbers.
>> For instance,
>>
>> print, systime(1) - long(systime(1))
>>
>> always returns 0.0000... In contrast, this statement returns various
>> fractions in IDL 5.2 WinNT/Intel and IDL 5.0 Mac. Has anyone else
>> observed this behavior? It wreaks havoc with trying to use time_test3 and

>> graphics_times3 for benchmarking.
>>
>> --
>> Paul Woodford
>> woodford@essexcorp.com
>
> --
> Ivan Zimine
> Dpt. of Radiology (MRI), Geneva University Hospitals
> email: ivan.zimine@physics.unige.ch
> tel. : (+41 22) 372 70 70

This is a known bug, and has been reported. A consequence of this I must delay a 2nd generation idl speed survey. Without accurate timing information, this is useless on affected architechtures. I guess we'll just have to wait for 5.2.1.... (and even then I'll have to code around this brokenness :{).

JD

--
J.D. Smith |*| WORK: (607) 255-5842
Cornell University Dept. of Astronomy |*| (607) 255-6263
304 Space Sciences Bldg. |*| FAX: (607) 255-5875
Ithaca, NY 14853 |*|

Subject: Re: bug in systime(1) on IDL 5.2 Linux/Intel?
Posted by Ivan Zimine on Mon, 01 Mar 1999 08:00:00 GMT
View Forum Message <> Reply to Message

I have the same behavior...

```
IDL> print, !version
{ x86 linux unix 5.2 Oct 30 1998}
IDL> t=systime(1) & f=dialog_pickfile() & print, systime(1)-t
      3.0000000
IDL> t=systime(1) & f=dialog_pickfile() & print, systime(1)-t
      1.0000000
etc 10 times
```

Couldn't get any fractional part. I doubt that my internal timing is so perfect.

"Woodford, Paul" wrote:

>
> systime(1) on IDL 5.2 Linux/Intel seems to always yield whole numbers.
> For instance,
>
> print, systime(1) - long(systime(1))
>
> always returns 0.0000... In contrast, this statement returns various
> fractions in IDL 5.2 WinNT/Intel and IDL 5.0 Mac. Has anyone else
> observed this behavior? It wreaks havoc with trying to use time_test3 and
> graphics_times3 for benchmarking.
>
> --
> Paul Woodford
> woodford@essexcorp.com

--
Ivan Zimine
Dpt. of Radiology (MRI), Geneva University Hospitals
email: ivan.zimine@physics.unige.ch
tel. : (+41 22) 372 70 70

Subject: Re: bug in systime(1) on IDL 5.2 Linux/Intel?
Posted by Kevin P. Reardon on Tue, 02 Mar 1999 08:00:00 GMT
View Forum Message <> Reply to Message

"Woodford, Paul" wrote:
>
> systime(1) on IDL 5.2 Linux/Intel seems to always yield whole
> numbers.

I just ran into this problem while installing a IDL 5.2 on a new Linux machine. Desperate to benchmark the beast, I coded up a quick workaround using the information contained in /proc/uptime (which contains the uptime in hundredths of a second). A simple program to read and format this information in the same way as systime is given below. Then in $IDL_DIR/lib/time_test.pro (or rather, a copy thereof), I substituted all occurences of "systime" with "uptime". Millisecond accuracy it ain't, but for running time_test[123], it seems sufficient (the fastest of the individual tests on my machine takes 0.07 seconds). Of course, a real fix for systime would be better.

Of course, this only works if you have the /proc filesystem enabled and you machine doesn't reboot during the time_test.

I hope it helps,
kevin reardon
Osservatorio Astronomico di Capodimonte

```
; a program to read the /proc/uptime "file" in a systime(1)-like
; format

function uptime,return_type

   ; select the first of the two fields in the file
   spawn, 'cat /proc/uptime | cut -f 1 -d " "' , uptime

   uptime=double(uptime[0])

   return,uptime

end
```