On Thu, 18 Oct 2007 21:58:15 +0000, Karl Schultz wrote:

> wlandsman <wlandsman@gmail.com> wrote:
>> On Oct 18, 2:30 pm, Karl Schultz <k...@io.frii.com> wrote:
>>> Because it is not a *stable* sort. Stable sorting algorithms preserve
>>> the order of equal keys.
>>>
>>>> IDL just uses the sort algorithm of the underlying OS.   As far as I
>>>> am aware, the SORT function on Linux boxes *does* preserve the order
>>>> of equal values, but that on Mac and Windows machines does not.   I
>>>> would be interested to hear if anyone finds any exceptions to this
>>>> rule.
>>>
>>> Are you using this SORT function from the command line? If so, you
>>> are using a shell function or a sort program in your PATH. Someone
>>> probably decided that a stable sort made more sense for people sorting
>>> things from the command line or from shell scripts. Reasonable.
>>>
>>
>> I don't understand this paragraph.   I am just using the IDL intrinisc
>> SORT command.   On every Linux box I have ever been on, it appears
>> that the C lib sort algorithm used by IDL SORT() *is* stable, whereas
>> it is *not* stable on Windows or MacOS.
>>
>
> When you said "SORT function on Linux boxes", I thought you meant from the
> Linux command line. My bad.
>
> So it looks like the qsort() implementation on the Linux distros you tried
> happens to be stable. That's all.

I side with Wayne: this platform difference has a real impact on many
SORT-based algorithms. I understand the goal of re-using a tuned system
QSORT, but going the extra step to get it to function the same on all
IDL-supported systems would seem a no-brainer.

JD