

Re: Linux in a binary world... a doomsday scenario

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2005-12/msg05159.html>

- *From:* Bodo Eggert <harvested.in.lkml@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Fri, 16 Dec 2005 07:57:21 +0100
-

Al Boldi <a1426z@xxxxxxxx> wrote:

>> Disadvantages of a stable API:

- >> * It encourages binary-only drivers, while we prefer source drivers.
- >> Changing the API often and without warning is one way of
- >> hampering binary-only driver development without harming
- >> open-source drivers.
- >
- > You are really shooting yourself in the foot here.

If binary drivers were a nice idea, it would be so, but unfortunately they aren't. E.g.:

Nvidia stuffs a userspace library into the kernel. BAD.

Ati can't work correctly if there are other graphic cards in the system, because they include functions from a different layer in their driver, which off cause can't interface correctly. The WXP drivers are affected, too. Doubleplusungood.

Both of these drivers are about 7 times larger than the whole kernel image. Other binary drivers are similar. That's a waste of resources.

DOS didn't have an API for non-FAT filesystems. In order to support ISO9660, the network support was abused instead of changing the internals. Obviously this defeated reasonable caching, and the drivers needed to add their own cache implementation, which had to be maintained seperately. Double work, and a waste of resources.

(The DOS internals usurally can't be changed because they are stable. The stable API was too cumbersome for the programmer's needs, so they started using the internal structures directly. Therefore there was no way to increase the max. path length beyond 64 bytes.)

- >> Do a stable API save us work? No, because whoever changes the API
- >> also fixes all in-kernel users of said API.
- >
- > That's very inefficient.

Re: Linux in a binary world... a doomsday scenario

More inefficient than adding a compatibility layer?

If each change has to add another compatibility layer, the calls will have to traverse several functions without doing anything usefull. Multiply this with the number of users, and you'll see that one millisecond of CPU time will accumulate to a quarter of an hour for each million users. Multiply this with thousands of calls each user will do within each hour, and you'll see that within each hour, you'd waste more than eleven days of CPU time. Within one day and a half, you have wasted a whole year. Within that time, you can easily change the API several times. That is, unless you hold meetings about how it should look like.

Ich danke GMX dafür, die Verwendung meiner Adressen mittels per SPF verbreiteten Lügen zu sabotieren.

-

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@xxxxxxxxxxxxxxxxxx

More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Please read the FAQ at <http://www.tux.org/lkml/>

-
- Prev by Date: [*\[PATCH\] mv64x600 wdt: convert to the new platform device interface*](#)
 - Next by Date: [*campanha de oracao*](#)
 - Previous by thread: [*Re: Linux in a binary world... a doomsday scenario*](#)
 - Next by thread: [*\[PATCH\] Fix bug in RCU torture test*](#)
 - Index(es):
 - ◆ [*Date*](#)
 - ◆ [*Thread*](#)