

Re: [2.6 patch] ieee1394_core.c: remove unneeded EXPORT_SYMBOL's

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2004-12/4698.html>

From: Arne Caspari (arnem_at_informatik.uni-bremen.de)

Date: 12/21/04

Date: Tue, 21 Dec 2004 09:33:45 +0100

To: Alan Cox <alan@lxorguk.ukuu.org.uk>

Alan Cox wrote:

> *On Llu, 2004-12-20 at 15:46, Ben Collins wrote:*

>

>>> *You might as well remove the ifdef if you do that since vendors will
>>> have to guess what the right answer is an will probably uniformly say
>>> "Y". At that point its basically a non-option. Far better to submit the
>>> driver*

>>

>> *You are missing the point though. Lots of these are part of our API, and*

>

>

> *I think you missed my point. Any vendor faced with that Config option*

> *will say Y so almost every tree will always have it – so why ask as*

> *opposed to keeping the status quo.*

My concerns with this option is actually that some mainstream distribution will say "N" here accidentally. So every customer with this distribution will call us and require intense support. If Linux will cause intense support, it will just not be supported at all because Windows support is almost zero in this regard and we make almost zero profits with Linux sales – and 99.99 % of our income with Windows sales.

> *Sure but if Adrian was trying to just tidy licensing issues he'd submit
> a switch to EXPORT_SYMBOL_GPL. (Admittedly for anything as closely tied
> as the innards of the ieee1394 layer its probably implied anyway).*

I do not see the licensing issue of a stable kernel API where vendors can rely on. Our driver is GPL so there should no be a licensing issue.

The reason why I have not submitted this driver is as follows:

If I submit a driver and there is a firmware change in our device that breaks compatibility to older drivers (as there once was), customers which get the new devices will have a driver that is not working. So each customer asks us for support and we need to guide him to replace

the kernel driver with the new one. This situation will remain until mainstream distributions update their kernel.

In the current situation we just have a website which says that you have to get the driver from sourceforge and compile it yourself. This works rather good: Customers that bought a device automatically got the right driver. And if I need to make some changes in the driver (ie. fix bugs or add functionality), I do not need to wait until the patches go into mainstream distributions (you can not wait for that) but just update the SF site and let the customer go through the same steps he already had gone through in the beginning.

It is all about avoiding (expensive) support. I can not stress this point enough: If supporting Linux becomes a cost factor it will just not be supported. There is virtually no profit for us in this market.

> *There are two conflicting goals here – to have clean complete API's and*
> *to stamp out the large number of unused, historic and at times bogus*
> *exports. If these API's are needed and used then they should stay just*
> *as some others elsewhere in the kernel have.*

On Windows I can rely on a stable kernel API for many years now. We have single drivers that work on the WindowsXP of today and also work on Windows 2000 which is almost 5 years old. They would most likely even work on Windows98 if we would support that platform. Windows circumvents the symbol export problem by using their IO Request Blocks etc. which makes things more complicated but at least stable.

Linux does not have such a model. So in my eyes, special care has to be taken to generate an API that is valid today and will remain valid for some years. Vendors need to be able to rely on a stable API.

I would take it like in a library: The API should not change between minor versions – likewise it should be stable in the kernel among all 2.6.x versions. If it changes to version 2.7.x or 2.8.x it would be OK since we could release a driver for a 2.8.x tree then.

/Arne

–

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

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

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