

Re: Alignment of disk-I/O from userland.

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/arch/2003-10/0034.html>

From: Harti Brandt (brandt_at_fokus.fraunhofer.de)

Date: 10/07/03

Date: Tue, 7 Oct 2003 09:25:16 +0200 (CEST)

To: Scott Long <scottl@freebsd.org>

On Mon, 6 Oct 2003, Scott Long wrote:

SL>On Mon, 6 Oct 2003, Poul-Henning Kamp wrote:

SL>> In message <200310062146.h96Lkpx0093486@khavrinen.lcs.mit.edu>, Garrett Wollman

SL>> writes:

SL>>

SL>> >I think that gives us plenary authority to require appropriate

SL>> >alignment of data buffers used to access disk devices directly.

SL>>

SL>> Well, apart from us wedging or botching the request rather than

SL>> return a consistent error we're standards compliant then.

SL>>

SL>> It has been mentioned on #thatchannel that busdma should take care

SL>> of this by copying the request as necessary. Faced with the prospect

SL>> of a request several megabytes in size, I don't like the prospect

SL>> of malloc/bcopy too much.

SL>

SL>Working around hardware requirements from the block layer gets messy.

SL>You obviously don't want to manually align buffers that are destined

SL>for hardware that doesn't care about the alignment. How do you

SL>communicate this property upwards from the driver? Callbacks? Extra

SL>flags in disk_create()? It's all rather messy that way. We already

SL>have the busdma interface whose sole purpose is to take system

SL>buffers and prepare them for transfer to/from hardware. It already

SL>understands the concept of alignment, though it only applies it to

SL>buffers that it allocates, not buffers that are handed to it. Fixing

This seems to be not true. The alignment parameter for the busdma tag is more or less ignored in all backends. With the old malloc code you could simply make sure that the buffer you request is at least the size of alignment you need. With UMA this doesn't work anymore. I bumped into this problem while writing several ATM drivers which need alignment of 16 or 32 byte. At the moment the driver does this alignment itself by allocating a large enough dmamem buffer and fiddling with the pointers.

harti

freebsd-arch: Re: Alignment of disk-I/O from userland.

SL>that would be easy, and would allow for optimizations based on the
SL>bus (GART and IOMMU remapping). It also keeps the property down at
SL>the device driver where it belongs.

SL>

SL>As for returning an error code for a buffer that we (arbitrarily) believe
SL>to be too big to align, I'm not sure that it's a valid thing to worry
SL>about. People expect their hardware to Just Work, regardless of how cheap
SL>it is. If someone buys a cheap card with a cheap DMA engine, their cost
SL>comes in from higher system time per I/O. If they don't like that, they
SL>can complain to the manufacturer and/or buy a better card. I'm not aware
SL>of Linux nor OSX rejecting I/O based on arbitrary size limits.

SL>

SL>Scott

SL>

SL>

freebsd-arch@freebsd.org mailing list

SL><http://lists.freebsd.org/mailman/listinfo/freebsd-arch>

SL>To unsubscribe, send any mail to "freebsd-arch-unsubscribe@freebsd.org"

SL>

--

harti brandt,

<http://www.fokus.fraunhofer.de/research/cc/cats/employees/hartmut.brandt/private>
brandt@fokus.fraunhofer.de, harti@freebsd.org

freebsd-arch@freebsd.org mailing list

<http://lists.freebsd.org/mailman/listinfo/freebsd-arch>

To unsubscribe, send any mail to "freebsd-arch-unsubscribe@freebsd.org"