Re: Unexpected splice "always copy" behavior observed

From: Linus Torvalds
Date: Wed May 19 2010 - 20:12:58 EST

- **Next message:** Peter Hutterer: "Re: [PATCH 2/2] input: mt: Document the MT event slot protocol (rev2)"
- **Previous message:** H. Peter Anvin: "Re: [PATCH] Prevent reserving RAM in the region already reservedby BIOS"
- **In reply to:** Mathieu Desnoyers: "Re: Unexpected splice "always copy" behavior observed"
- **Next in thread:** Mathieu Desnoyers: "Re: Unexpected splice "always copy" behavior observed"
- **Messages sorted by:** [date] [thread] [subject] [author]

On Wed, 19 May 2010, Mathieu Desnoyers wrote:
>
> *A faced a small counter-intuitive fadvise behavior though.*
>
> *posix_fadvise(fd, 0, 0, POSIX_FADV_DONTNEED);*
>
> *only seems to affect the parts of a file that already exist.*

POSIX_FADV_DONTNEED does not have _any_ long-term behavior. So when you do
a

posix_fadvise(fd, 0, 0, POSIX_FADV_DONTNEED);

it only affects the pages that are there right now, it has no effect on
any future actions.

> *So after each splice() that appends to the file, I have to call fadvise*
> *again. I would have expected the "0" len parameter to tell the kernel to*
> *apply the hint to the whole file, even parts that will be added in the*
> *future.*

It's not a hint about future at all. It's a "throw current pages away".

I would also suggest against doing that kind of thing in a streaming write
situation. The behavior for dirty page writeback is _not_ welldefined, and
if you do POSIX_FADV_DONTNEED, I would suggest you do it as part of that
writeback logic, ie you do it only on ranges that you have just waited on.

IOW, in my example, you'd couple the

sync_file_range(fd, (index-1)*BUFSIZE, BUFSIZE,
SYNC_FILE_RANGE_WAIT_BEFORE|SYNC_FILE_RANGE_WRITE|SYNC_FILE_RANGE_WAIT_AFTER);

with a

posix_fadvise(fd, (index-1)*BUFSIZE, BUFSIZE, POSIX_FADV_DONTNEED);

afterwards to throw out the pages that you just waited for.

Linus
--

- **Next message:** Peter Hutterer: "Re: [PATCH 2/2] input: mt: Document the MT event slot protocol (rev2)"
- **Previous message:** H. Peter Anvin: "Re: [PATCH] Prevent reserving RAM in the region already reservedby BIOS"
- **In reply to:** Mathieu Desnoyers: "Re: Unexpected splice "always copy" behavior observed"
- **Next in thread:** Mathieu Desnoyers: "Re: Unexpected splice "always copy" behavior observed"
- **Messages sorted by:** [date] [thread] [subject] [author]