Subject: [PATCH 0/4] fuse: optimize scatter-gather direct IO
Posted by Maxim Patlasov on Fri, 20 Jul 2012 11:50:07 GMT
View Forum Message <> Reply to Message

Hi,

Existing fuse implementation processes scatter-gather direct IO in suboptimal way: fuse_direct_IO passes iovec[] to fuse_loop_dio and the latter calls fuse_direct_read/write for each iovec from iovec[] array. Thus we have as many submitted fuse-requests as the number of elements in iovec[] array. This is pure waste of resources and affects performance negatively especially for the case of many small chunks (e.g. page-size) packed in one iovec[] array.

The patch-set amends situation in a natural way: let's simply pack as many iovec[] segments to every fuse-request as possible.

To estimate performance improvement I used slightly modified fusexmp over tmpfs (clearing O_DIRECT bit from fi->flags in xmp_open). The test opened a file with O_DIRECT, then called readv/writev in a loop. An iovec[] for readv/writev consisted of 32 segments of 4K each. The throughput on some commodity (rather feeble) server was (in MB/sec):

	original	/	patched
writev:	~107	/	~480
readv:	~114	/	~569

We're exploring possiblity to use fuse for our own distributed storage implementation and big iovec[] arrays of many page-size chunks is typical use-case for device virtualization thread performing i/o on behalf of virtual-machine it serves.

Thanks,
Maxim

Maxim Patlasov (4):
    fuse: add basic support of iovec[] to fuse_req
    fuse: re-work fuse_get_user_pages() to operate on iovec[]
    fuse: re-work fuse_direct_io() to operate on iovec[]
    fuse: re-work fuse_direct_IO()

 fs/fuse/dev.c | 52 ++++++++++++++++++-
 fs/fuse/file.c | 145 ++++++++++++++++++++++++++++++++++++++----------------
 fs/fuse/fuse_i.h | 12 ++++
 3 files changed, 140 insertions(+), 69 deletions(-)