
Subject: [PATCH 05/11] fuse: rework fuse_perform_write()
Posted by [Maxim Patlasov](#) on Wed, 19 Sep 2012 16:32:25 GMT
[View Forum Message](#) <> [Reply to Message](#)

The patch allocates as many page pointers in fuse_req as needed to cover interval [pos .. pos+len-1]. Inline helper fuse_wr_pages() is introduced to hide this cumbersome arithmetic.

Signed-off-by: Maxim Patlasov <mpatlasov@parallels.com>

fs/fuse/file.c | 13 ++++++++
1 files changed, 11 insertions(+), 2 deletions(-)

diff --git a/fs/fuse/file.c b/fs/fuse/file.c

index a618371..d72d638 100644

--- a/fs/fuse/file.c

+++ b/fs/fuse/file.c

```
@@ -881,11 +881,19 @@ static ssize_t fuse_fill_write_pages(struct fuse_req *req,
    if (!fc->big_writes)
        break;
    } while (iov_iter_count(ii) && count < fc->max_write &&
-   req->num_pages < FUSE_MAX_PAGES_PER_REQ && offset == 0);
+   req->num_pages < req->max_pages && offset == 0);
```

```
    return count > 0 ? count : err;
}
```

```
+static inline unsigned fuse_wr_pages(loff_t pos, size_t len)
```

```
+{
+   return min_t(unsigned,
+       ((pos + len - 1) >> PAGE_CACHE_SHIFT) -
+       (pos >> PAGE_CACHE_SHIFT) + 1,
+       FUSE_MAX_PAGES_PER_REQ);
+}
```

```
+static ssize_t fuse_perform_write(struct file *file,
+    struct address_space *mapping,
+    struct iov_iter *ii, loff_t pos)
```

```
@@ -901,8 +909,9 @@ static ssize_t fuse_perform_write(struct file *file,
    do {
        struct fuse_req *req;
        ssize_t count;
+       unsigned nr_pages = fuse_wr_pages(pos, iov_iter_count(ii));

-       req = fuse_get_req(fc, FUSE_MAX_PAGES_PER_REQ);
+       req = fuse_get_req(fc, nr_pages);
        if (IS_ERR(req)) {
            err = PTR_ERR(req);
```

break;
