

## rbd - Feature #13025

### Add scatter/gather support to librbd C/C++ APIs

09/10/2015 02:19 PM - Jason Dillaman

<b>Status:</b>	Resolved	<b>% Done:</b>	0%
<b>Priority:</b>	Normal	<b>Spent time:</b>	0.00 hour
<b>Assignee:</b>	Jason Dillaman		
<b>Category:</b>			
<b>Target version:</b>	v12.0.0		
<b>Source:</b>	other	<b>Reviewed:</b>	
<b>Tags:</b>		<b>Affected Versions:</b>	
<b>Backport:</b>		<b>Pull request ID:</b>	
<b>Description</b>			

#### History

##### #1 - 09/10/2015 02:28 PM - Haomai Wang

Interesting, could you show more about details? I'm also working on something related.

##### #2 - 09/11/2015 01:11 AM - Jason Dillaman

Adding new methods to the C API:

```
rbd_aio_writev(rbd_image_t image, const struct iovec *iov, int iovcnt, uint64_t off, rbd_completion_t c)
rbd_aio_readv(rbd_image_t image, const struct iovec *iov, int iovcnt, uint64_t off, rbd_completion_t c)
```

Internally, all the C API write methods will use a bufferlist created with a static pointer initialized to the provided buffer. From QEMU, this will eliminate the need for the bounce buffer and avoids copying the buffer into a new bufferlist (within librbd::aio\_write in internal.cc). A C API write should only involve one copy to store the provided buffer in the cache (if enabled) and a read should only involve one copy to transfer data into the user buffer.

The C++ API already has basic support for scatter/gather via the bufferlist, so the only change I made there was to ensure it was no longer converting the bufferlist to a C-style array. A C++ API read/write should involve zero copies (assuming it's shareable).

##### #3 - 09/11/2015 05:59 PM - Jason Dillaman

- Status changed from In Progress to Fix Under Review

##### #4 - 09/15/2015 03:43 PM - Josh Durgin

- Target version set to v10.0.1

##### #5 - 08/12/2016 07:36 PM - Jason Dillaman

- Target version changed from v10.0.1 to v11.0.0

**#6 - 02/15/2017 11:12 PM - Jason Dillaman**

- Target version changed from v11.0.0 to v12.0.0

**#7 - 02/15/2017 11:55 PM - Jason Dillaman**

PR: <https://github.com/ceph/ceph/pull/13447>

**#8 - 02/20/2017 08:03 PM - Mykola Golub**

- Status changed from Fix Under Review to Resolved

**#9 - 04/19/2017 06:58 AM - Stefan Priebe**

Is there any chance to get this into jewel?

**#10 - 04/19/2017 12:40 PM - Jason Dillaman**

@Stefan: since this is a new feature, we are not planning to backport it to older versions of Ceph.