

# FILE STRIPING

The text below describes how files from Ceph file system clients are stored across objects stored in RADOS.

## CEPH\_FILE\_LAYOUT

Ceph distributes (stripes) the data for a given file across a number of underlying objects. The way file data is mapped to those objects is defined by the `ceph_file_layout` structure. The data distribution is a modified RAID 0, where data is striped across a set of objects up to a (per-file) fixed size, at which point another set of objects holds the file's data. The second set also holds no more than the fixed amount of data, and then another set is used, and so on.

Defining some terminology will go a long way toward explaining the way file data is laid out across Ceph objects.

- file  
A collection of contiguous data, named from the perspective of the Ceph client (i.e., a file on a Linux system using Ceph storage). The data for a file is divided into fixed-size "stripe units," which are stored in ceph "objects."
- stripe unit  
The size (in bytes) of a block of data used in the RAID 0 distribution of a file. All stripe units for a file have equal size. The last stripe unit is typically incomplete—i.e. it represents the data at the end of the file as well as unused "space" beyond it up to the end of the fixed stripe unit size.
- stripe count  
The number of consecutive stripe units that constitute a RAID 0 "stripe" of file data.
- stripe  
A contiguous range of file data, RAID 0 striped across "stripe count" objects in fixed-size "stripe unit" blocks.
- object  
A collection of data maintained by Ceph storage. Objects are used to hold portions of Ceph client files.
- object set  
A set of objects that together represent a contiguous portion of a file.

Three fields in the `ceph_file_layout` structure define this mapping:

```
u32 fl_stripe_unit;  
u32 fl_stripe_count;  
u32 fl_object_size;
```

(They are actually maintained in their on-disk format, `__le32`.)

The role of the first two fields should be clear from the definitions above.

The third field is the maximum size (in bytes) of an object used to back file data. The object size is a multiple of the stripe unit.

A file's data is blocked into stripe units, and consecutive stripe units are stored on objects in an object set. The number of objects in a set is the same as the stripe count. No object storing file data will exceed the file's designated object size, so after some fixed number of complete stripes, a new object set is used to store subsequent file data.

Note that by default, Ceph uses a simple striping strategy in which `object_size` equals `stripe_unit` and `stripe_count` is 1. This simply puts one `stripe_unit` in each object.

Here's a more complex example:

```
file size = 1 trillion = 1000000000000 bytes  
  
fl_stripe_unit = 64KB = 65536 bytes  
fl_stripe_count = 5 stripe units per stripe  
fl_object_size = 64GB = 68719476736 bytes
```

This means:

```
file stripe size = 64KB * 5 = 320KB = 327680 bytes
each object holds 64GB / 64KB = 1048576 stripe units
file object set size = 64GB * 5 = 320GB = 343597383680 bytes
    (also 1048576 stripe units * 327680 bytes per stripe unit)
```

So the file's 1 trillion bytes can be divided into complete object sets, then complete stripes, then complete stripe units, and finally a single incomplete stripe unit:

- 1 trillion bytes / 320GB per object set = 2 complete object sets  
(with 312805232640 bytes remaining)
- 312805232640 bytes / 320KB per stripe = 954605 complete stripes  
(with 266240 bytes remaining)
- 266240 bytes / 64KB per stripe unit = 4 complete stripe units  
(with 4096 bytes remaining)
- and the final incomplete stripe unit holds those 4096 bytes.

The ASCII art below attempts to capture this:

/object 0\		/object 1\		/object 2\		/object 3\		/object 4\		
+=====+		+=====+		+=====+		+=====+		+=====+		
object 0	stripe unit 0	stripe unit 1	stripe unit 2	stripe unit 3	stripe unit 4	stripe 0				
	stripe unit 5	stripe unit 6	stripe unit 7	stripe unit 8	stripe unit 9	stripe 1				
	.	.	.	.	.					
	.	.	.	.	.					
0	stripe unit 5242875	stripe unit 5242876	stripe unit 5242877	stripe unit 5242878	stripe unit 5242879	stripe 1048575				
	=====		=====		=====		=====		=====	
/object 5\		/object 6\		/object 7\		/object 8\		/object 9\		
+=====+		+=====+		+=====+		+=====+		+=====+		
object 1	stripe unit 5242880	stripe unit 5242881	stripe unit 5242882	stripe unit 5242883	stripe unit 5242884	stripe 1048576				
	stripe unit 5242885	stripe unit 5242886	stripe unit 5242887	stripe unit 5242888	stripe unit 5242889	stripe 1048577				
	.	.	.	.	.					
	.	.	.	.	.					
1	stripe unit 10485755	stripe unit 10485756	stripe unit 10485757	stripe unit 10485758	stripe unit 10485759	stripe 2097151				
	=====		=====		=====		=====		=====	
/object 10\		/object 11\		/object 12\		/object 13\		/object 14\		
+=====+		+=====+		+=====+		+=====+		+=====+		
object 2	stripe unit 10485760	stripe unit 10485761	stripe unit 10485762	stripe unit 10485763	stripe unit 10485764	stripe 2097152				
	stripe unit 10485765	stripe unit 10485766	stripe unit 10485767	stripe unit 10485768	stripe unit 10485769	stripe 2097153				
	.	.	.	.	.					
	.	.	.	.	.					

t	.	.	.	.	.	
2	stripe unit 15258780	stripe unit 15258781	stripe unit 15258782	stripe unit 15258783	stripe unit 15258784	stripe 3051756
	stripe unit 15258785	stripe unit 15258786	stripe unit 15258787	stripe unit 15258788	(partial stripe unit)	(partial stripe 3051757)
	\=====/ 	\=====/ 	\=====/ 	\=====/ 	\=====/ 	