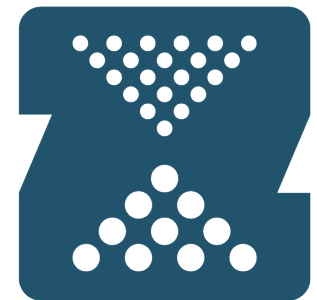# Pool Performance

George Wilson

gwilson@delphix.com

@zfsdude

DELPHIX

OpenZFS

Imbalanced LUNs enhancements

- Adding space to full pools does not improve the performance of the pool
  - Ideally add space to pool long before it's needed
- ZFS will try to adjust the amount of data it writes to full vdevs vs. empty vdevs
  - Not aggressive enough when the disparity of free space is large
- Force ZFS to avoid writing to vdevs that are mostly full for a period of time
  - zfs_mg_noalloc_threshold - percentage of free space that makes a device eligible for allocations

Imbalanced LUNs
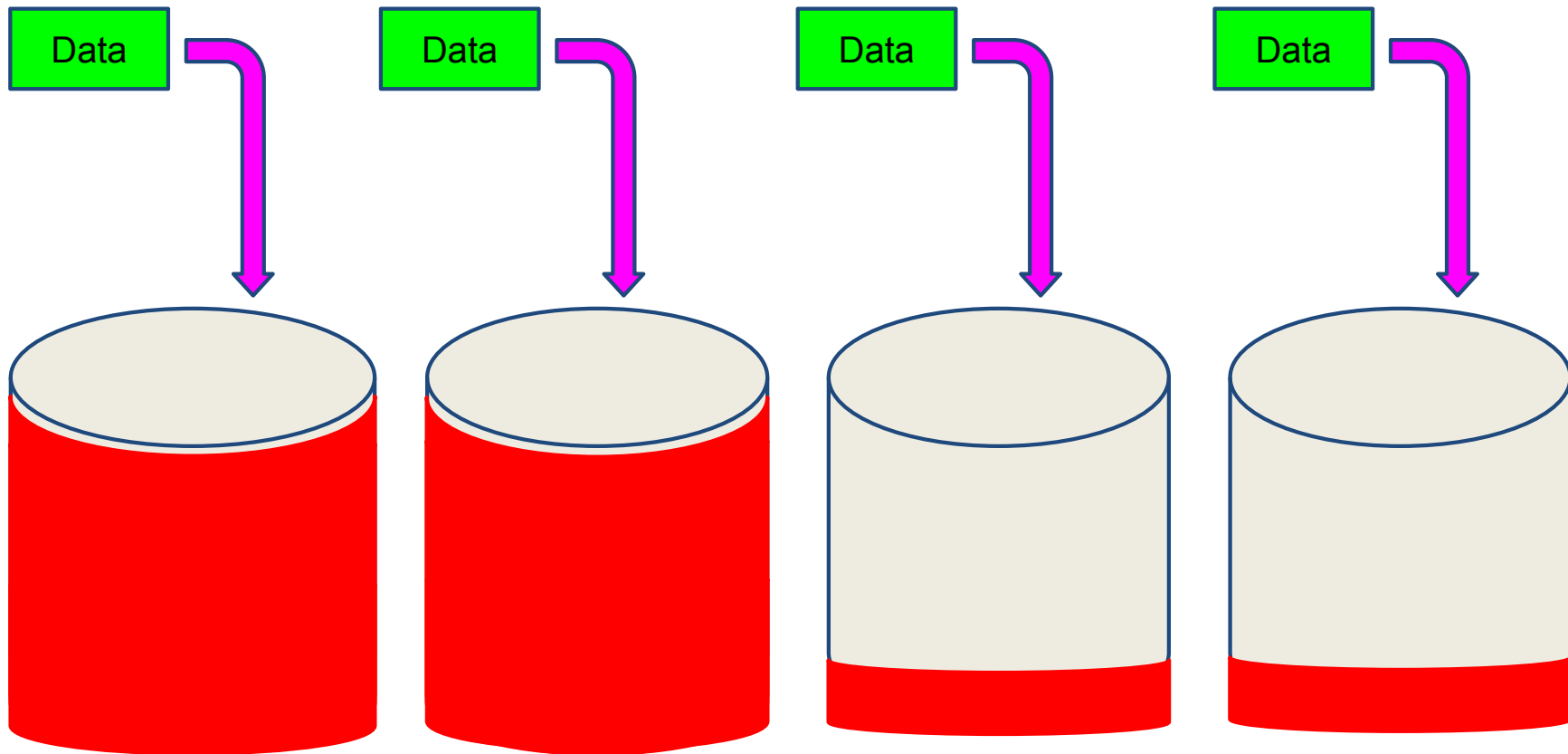
Imbalanced LUN enhancement
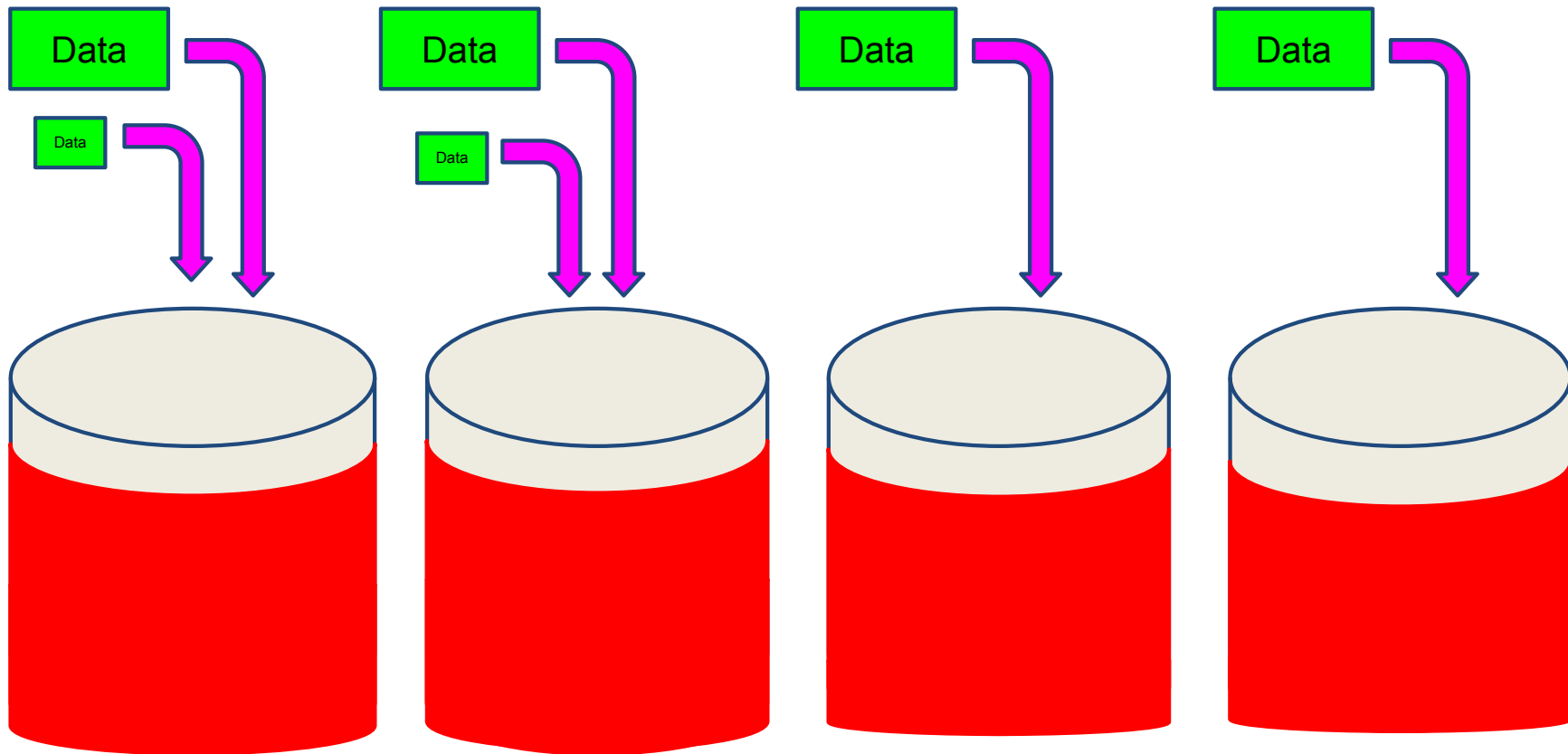
Understanding Free space

- Free space histogram
  - New upgraded space_map objects (feature flag)
  - Histograms store free space in power of 2 buckets
  - Histogram available at metalsab, device, and pool level

New Block Allocator

- New cursor-based block allocator
  - Allocates contiguous blocks in the largest region within a metaslab
  - Maintains a cursor to last allocated block

New metaslab handling

- Use metaslab fragmentation metric to determine effective free space
  - Metaslab's weighting is adjusted down
- Metaslab are preloaded
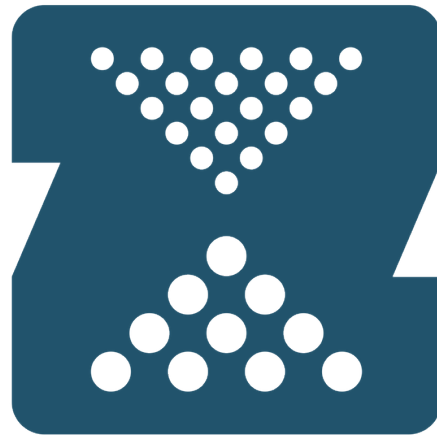 - Preload the next N metaslabs at the end of each txg sync

Pool fragmentation

- Ability to see the fragmentation at a pool and device level
- Utilizes free space histograms stored on-disk

NAME	SIZE	ALLOC	FREE	FRAG	EXPANDSZ	CAP	DEDUP	HEALTH	ALTROOT
tank	47.7G	24.6G	23.0G	23%	-	51%	1.00x	ONLINE	-
c1t3d0	7.94G	7.48G	467M	50%	-				
c1t4d0	39.8G	17.2G	22.6G	19%	-				

Device selection

- Select device based on fragmentation metric
- Devices that are less fragmented receive more allocations
- Allocation queue limits number of allocations per device
  - Allocations are ordered

NAME	SIZE	ALLOC	FREE	FRAG	EXPANDSZ	CAP	DEDUP	HEALTH	ALTROOT
tank	47.7G	24.6G	23.0G	23%	-	51%	1.00x	ONLINE	-
c1t3d0	7.94G	7.48G	467M	50%	-				
c1t4d0	39.8G	17.2G	22.6G	19%	-				

http://open-zfs.org

George Wilson

gwilson@delphix.com

@zfsdude