VEEAM

Volume backup for Linux or how to create a snapshot

Sergey Shtepa

Senior Developer

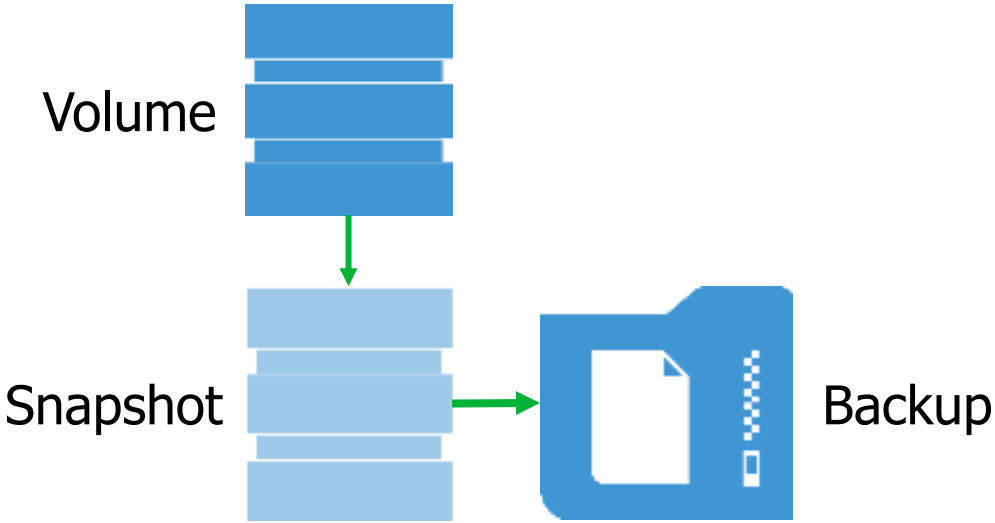
Volume backup or File backup

File backup allows you to restore only specific data files.
OS and installed services will have to be restored separately.

Volume backup allows you to recover the entire machine at once, thus reducing RTO (Restore Time Objective)

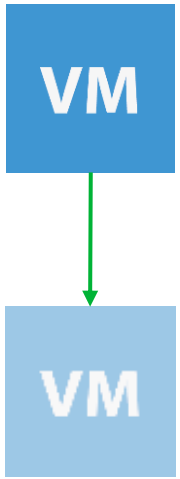
RTO is a key factor for business.

Backup a volume using its snapshot

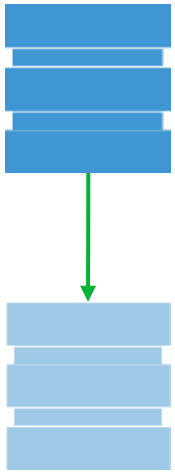


Snapshots for Linux

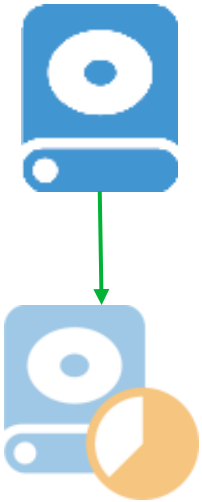
Virtual machine



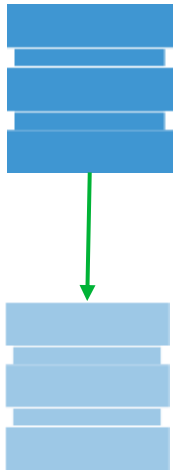
LVM & Thin Provisioning



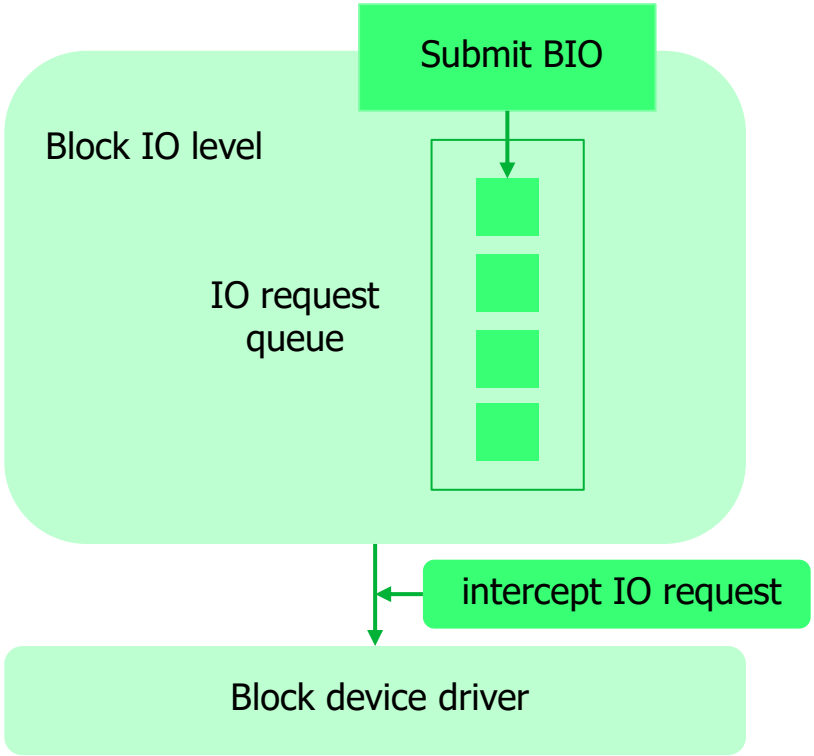
BTRFS & ZFS

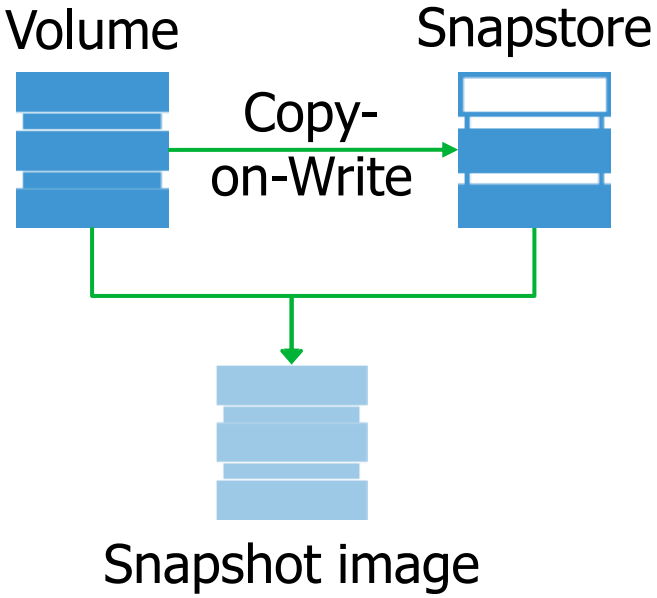


Pure block device



How it works





Consistent state

Consistent state ensures the integrity of data structures and completeness of initiated transactions.

A consistent state is required for:

- File system metadata
- Database files
- Files of other programs

File system consistent state

The Linux kernel provides the following functions:

```
    freeze_bdev – lock a filesystem and force it into a
consistent state
    thaw_bdev – unlock filesystem
```

A file system has to support them:

```
struct super_operations {
      ...
      int (*freeze_fs) (struct super_block *);
      int (*unfreeze_fs) (struct super_block *);
      ...
}
```

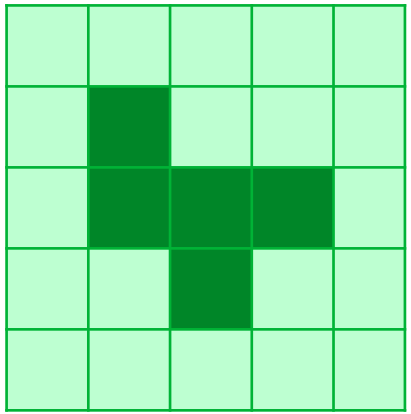
Applications consistent state

Calling pre-freeze and post-thaw scripts allows to prepare applications for backup.

The Oracle Application Processing is implemented starting with version 3.0

Change Block Tracking

Block device with changed blocks



Only changed blocks in a backup

CBT in other products

- VMware vSphere CBT is available since vSphere 4 (2009)
- Microsoft Hyper-V CBT is available since Windows Server 2016
- Veeam File Change Tracking driver for OS Windows is available since VBR 6.0 (2012)
- Veeam Volume Change Tracking driver for OS Windows is available since Veeam Agent for Windows 2.1 (2017)

The principle of operation of CBT

Snapshot #3

3	0	3	0	0	0
0	3	3	3	3	0
0	0	3	3	1	1
0	1	1	3	3	1
1	1	0	1	0	3
0	0	1	0	1	0
0	0	0	0	0	1

Snapshot #5

5	0	4	0	0	0
0	4	5	4	3	0
0	0	5	5	5	1
0	1	1	5	3	1
1	1	0	4	4	3
0	0	1	0	1	0
0	0	0	0	0	1

Difference

5		4			
	4	5	4		
		5	5	5	
			5		
			4	4	

The snapshot is a block device

That allows you to mount the file system that resides on the snapshot in order to perform the following actions:

- Get a map of free blocks
- Perform indexing

COW vs ROW

Redirect-on-Write

+ Better performance compared to COW
– The original device has to undergo the merge operation

Implemented in: BTRFS, Thin provisioning, vSphere, Hyper-V.

Copy-on-Write

+ The original device does not need to be merged
– Lower performance due to additional reads and writes

Implemented in: LVM, VSS.

The snapstore location

Challenges:

- Cannot allocate a large enough snapstore in RAM
- Snapstore cannot be a file on a file system
- Remote snapstore location will reduce performance
- Usually there is no unallocated space available

Solution:

- Allocate free disk space on the existing file system

Allocating the snapstore on FS

```
NAME
    fallocate - manipulate file space
SYNOPSIS
    int fallocate(int fd, int mode, off_t offset, off_t len);
DESCRIPTION
    The default operation of fallocate() allocates the disk space within the
    range specified by offset and len.
```

http://man7.org/linux/man-pages/man2/fallocate.2.html

```
============
Fiemap Ioctl
============

The fiemap ioctl is an efficient method for userspace to get fileextent mappings.
Instead of block-by-block mapping (such as bmap), fiemap returns a list of extents.
```

https://www.kernel.org/doc/Documentation/filesystems/fiemap.txt

Snapstore allocation algorithm: «Generic»

«Generic» snapstore allocation algorithm does not depend on a file system type. Algorithm:

- A specific pattern is written to the file
- The module intercepts write requests, searches for a pattern
- Pattern-containing blocks belong to the snapstore

Disadvantages:

- Poor performance
- Relatively high complexity

Snapshot overflow

The size of the snapstore depends on:

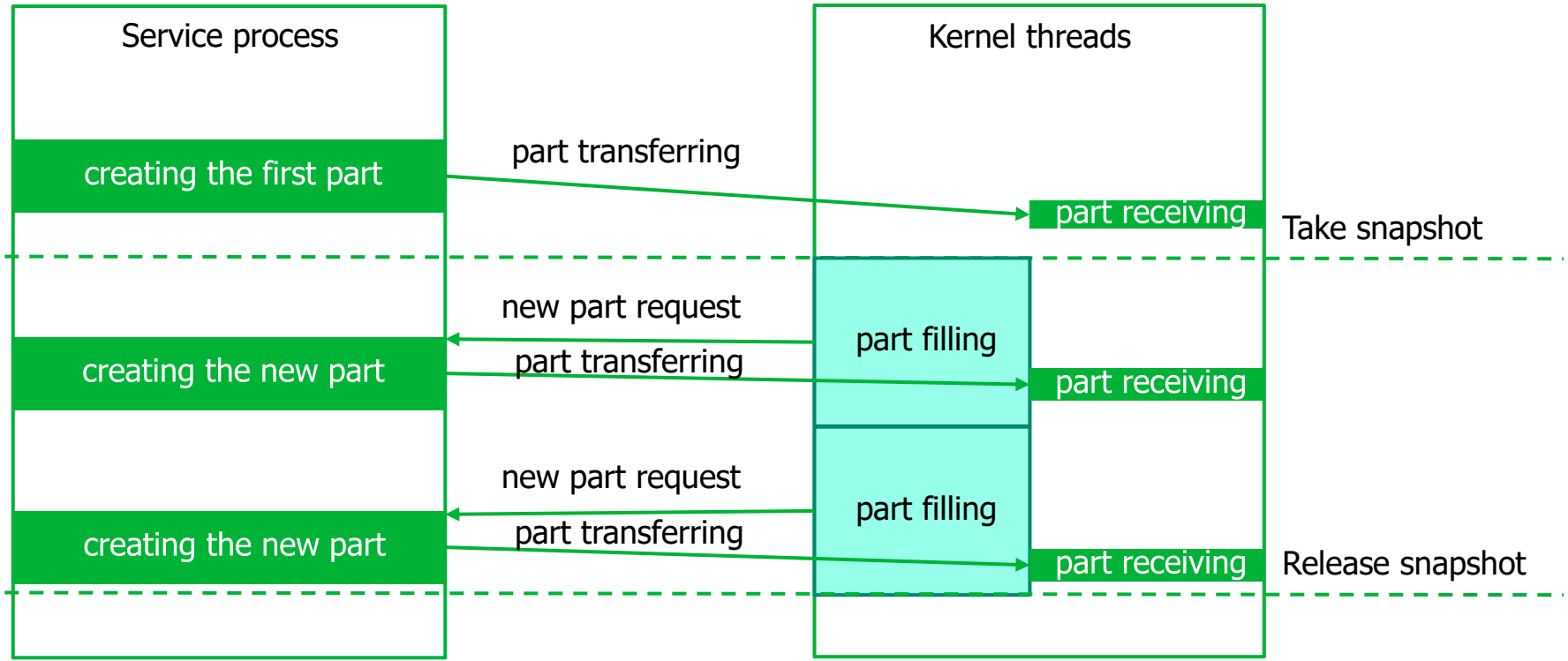
- The server workload during backup
- The backup duration

These parameters are unique for each server and it is difficult to predict them.

Solution:

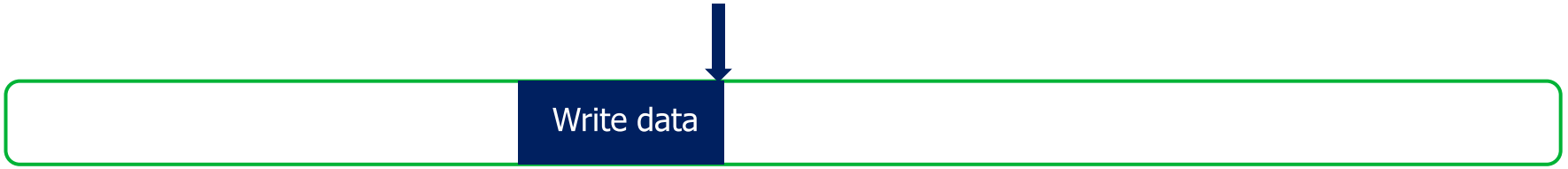
- Create stretch snapstore that can grow as you fill it.

Snapshot algorithm: «Stretch»

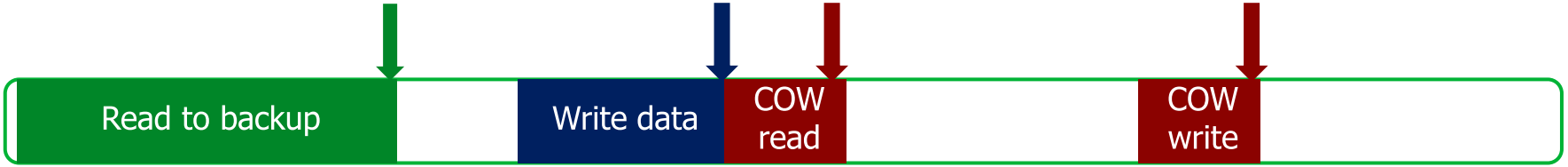


Performance problem

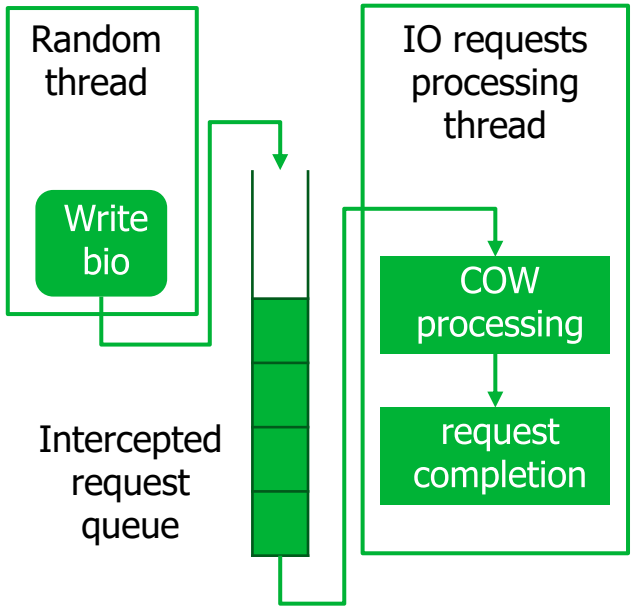
Linear data writing

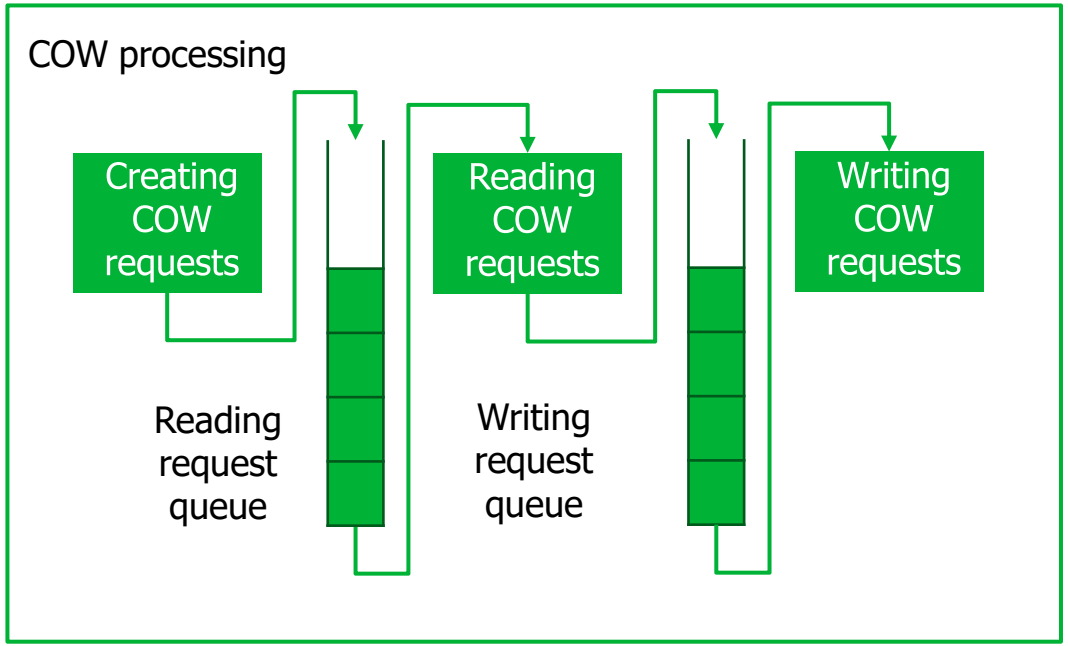


During backup, queries to the disk are no longer linear

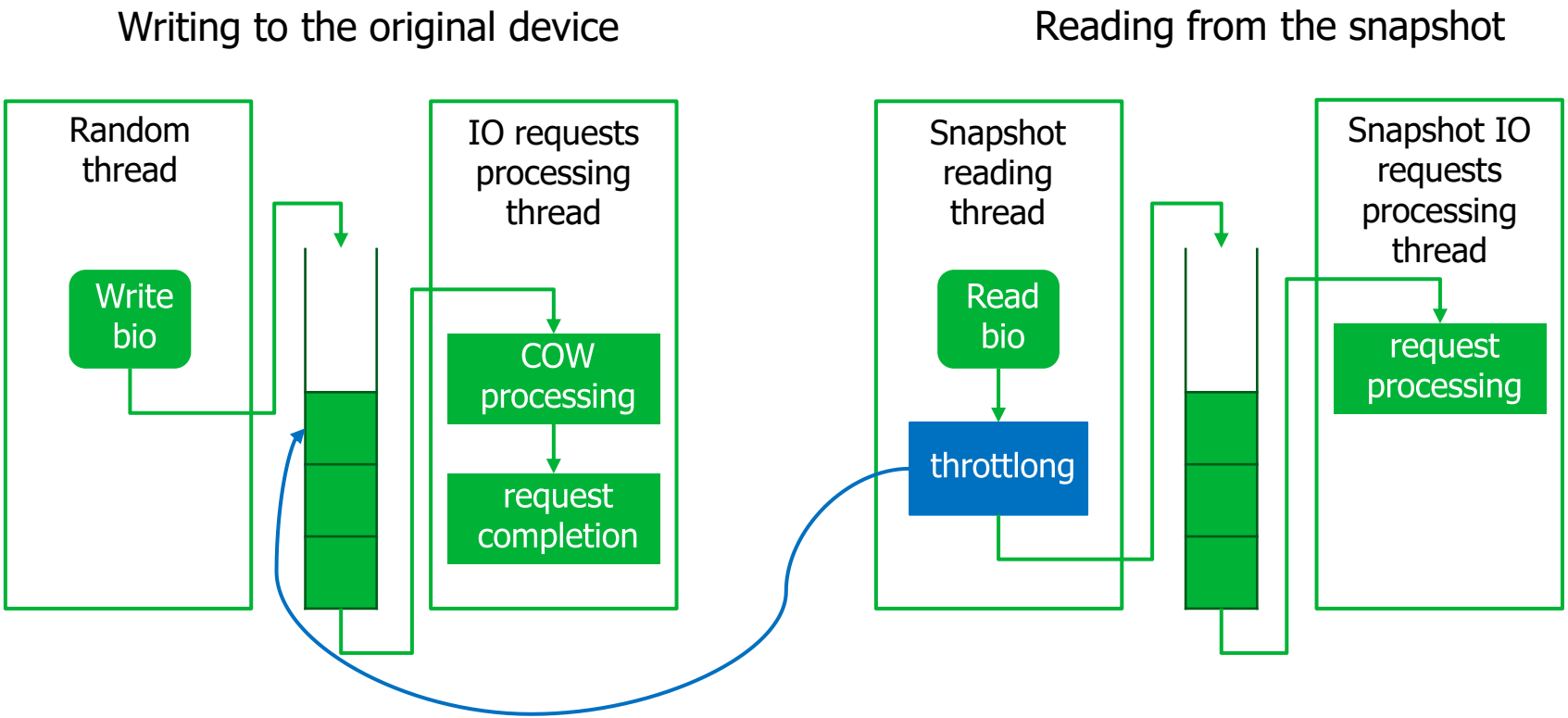


Requests processing

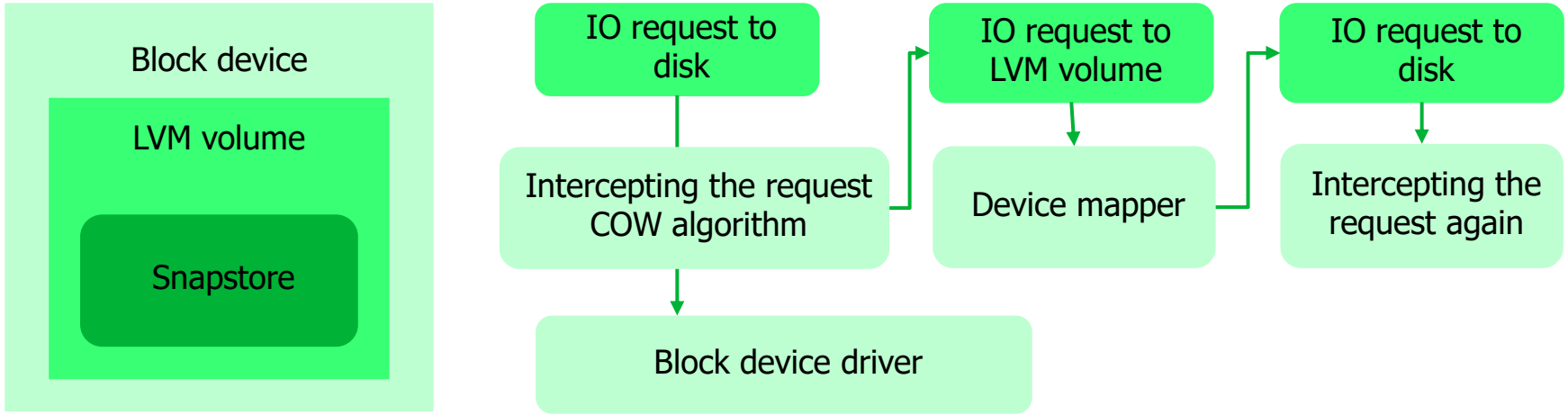




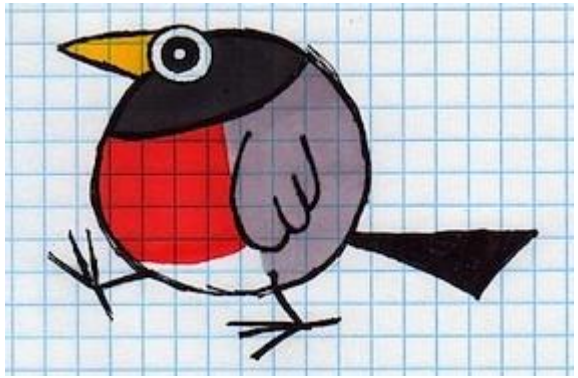
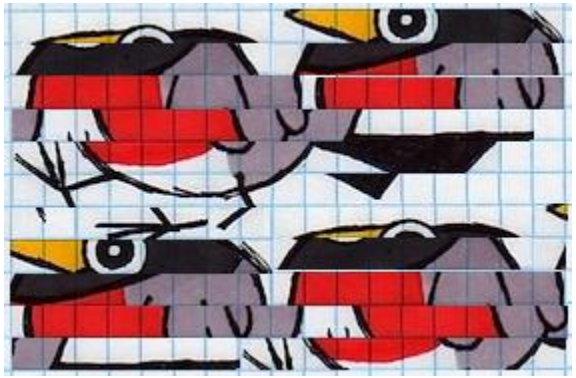
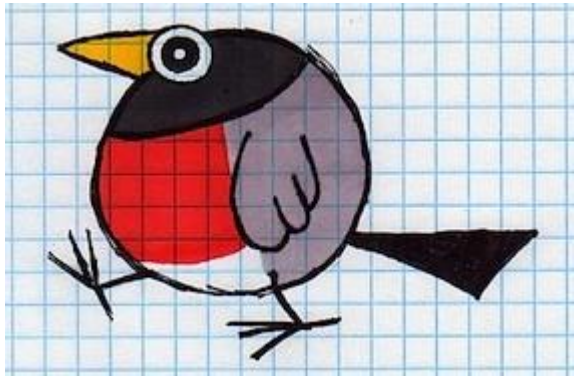
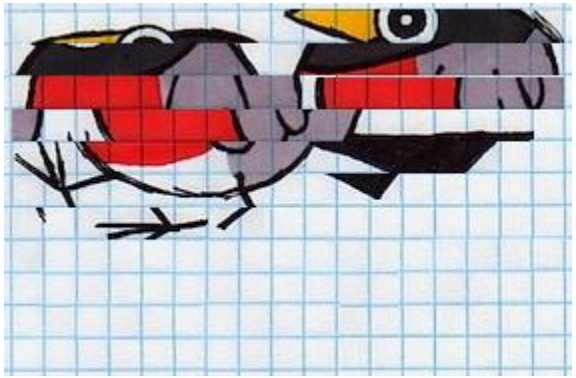
Prioritization problem



Deadlock during request interception

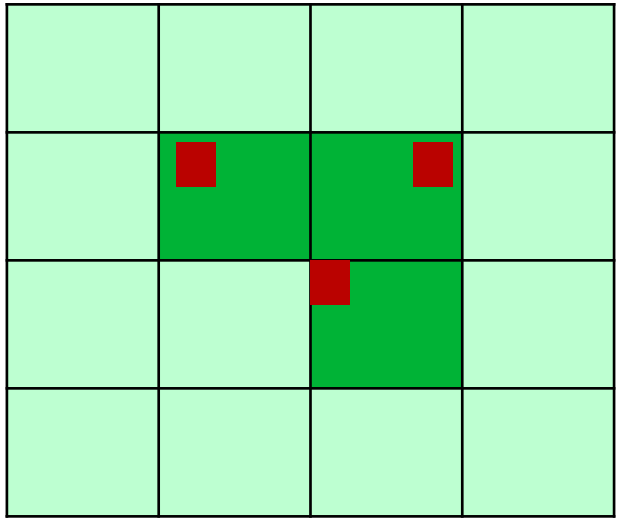


Round Robin Database

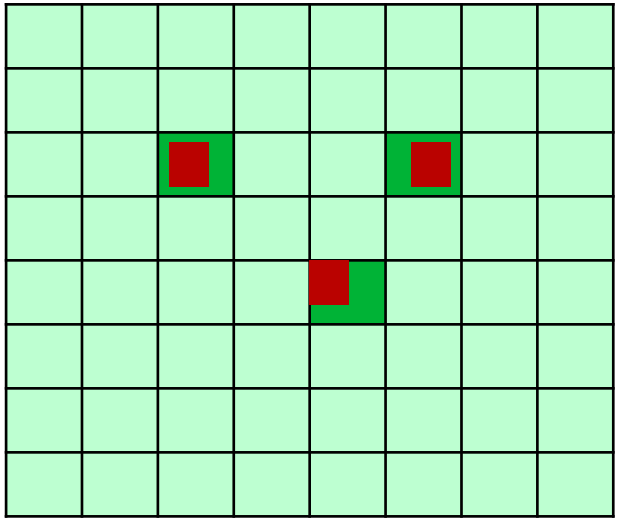
Cow-on-Write	Original block device	Snapstore
Version 1		
Version 2		

Snapstore block size

Large blocks



Small blocks



Conclusion

Veeam Agent for Linux uses the veeamsnap kernel module.
The product is available for free use.

veeamsnap kernel module is available here:
https://github.com/veeam/veeamsnap
GNU General Public License v2.0

Thank you

VEEAM