Recent Improvements in Gluster for VM image storage

Pranith Kumar Karampuri
20th Aug 2015

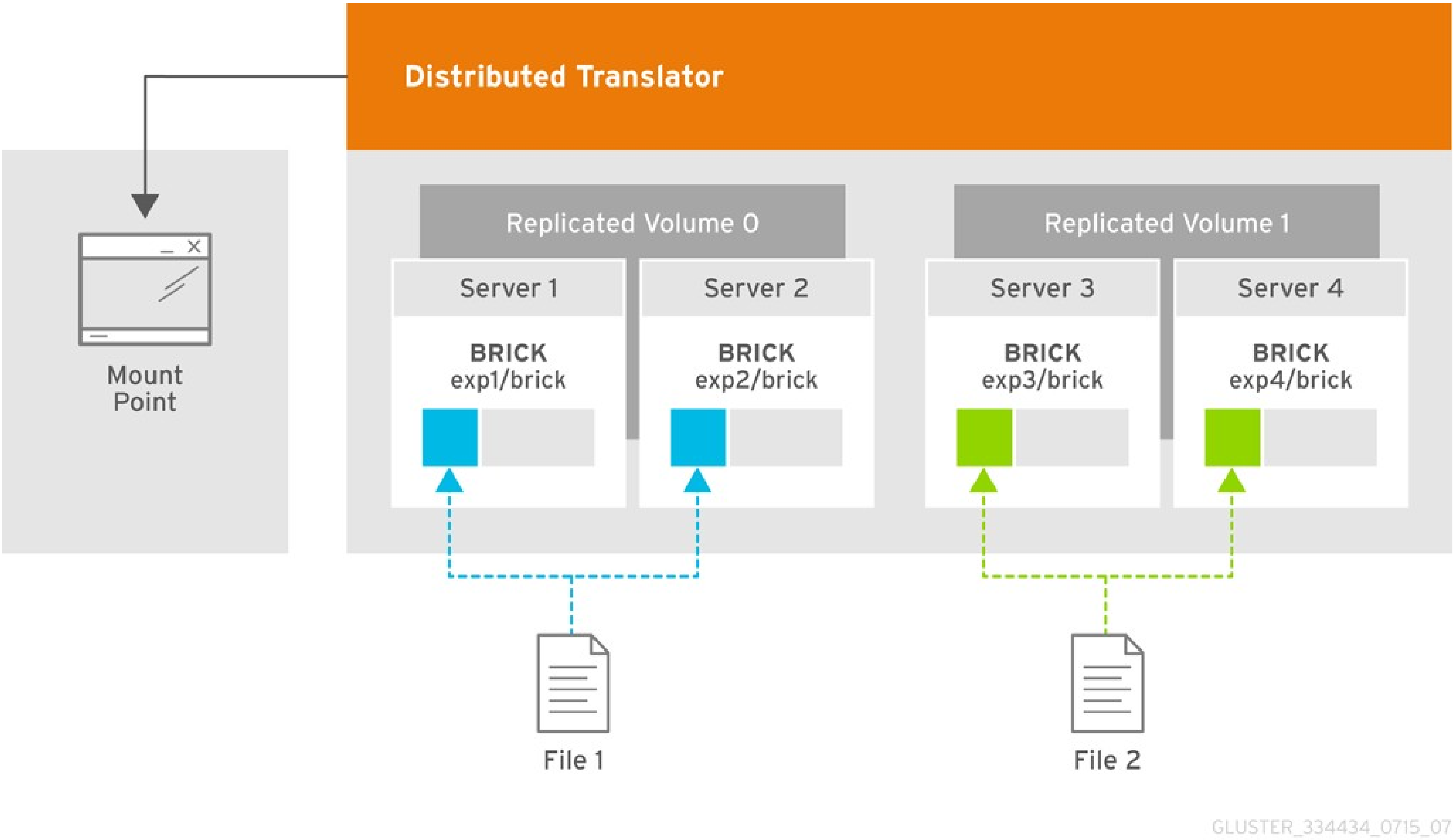
Agenda

- What is GlusterFS
- VM Store Usecase
- High Availability, self-heal, split-brain
- Improvements

GlusterFS

- Distributed Network File System
- Commodity hardware
- Free and open source

Volume, brick, translators, distribution, replication



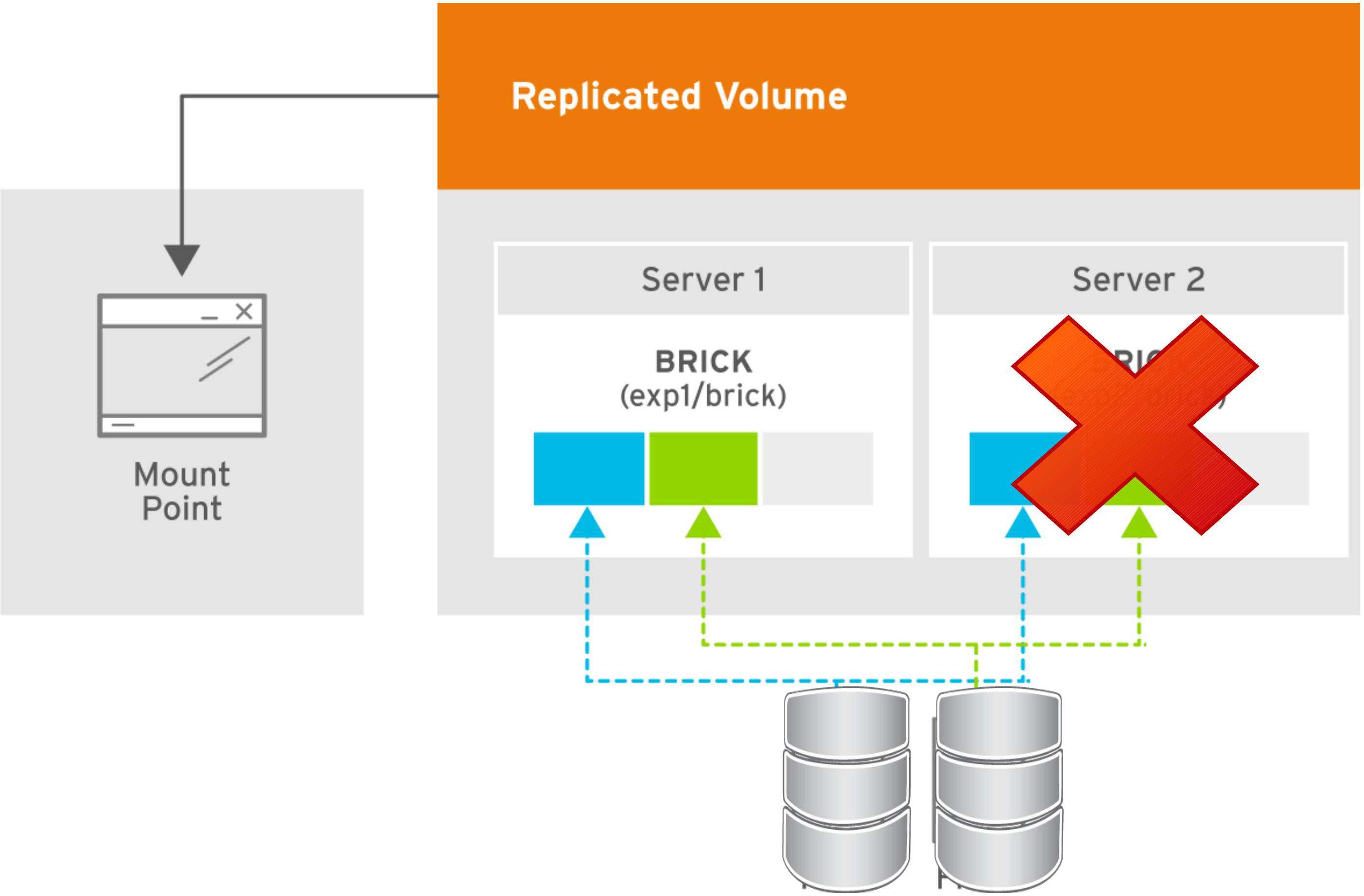
VM Store

- VM image files are stored on the volume
- Can be accessed through mount-point/qemu-gfapi
- VM Store optimization profile
  - http://www.gluster.org/community/documentation/index.php/Libgfapi_with_qemu_libvirt

VM Store – High Availability

- When one of the bricks goes offline, VM operations are served from the remaining brick(s)
- The brick that is serving data is updated with information that the other brick needs repair/heal

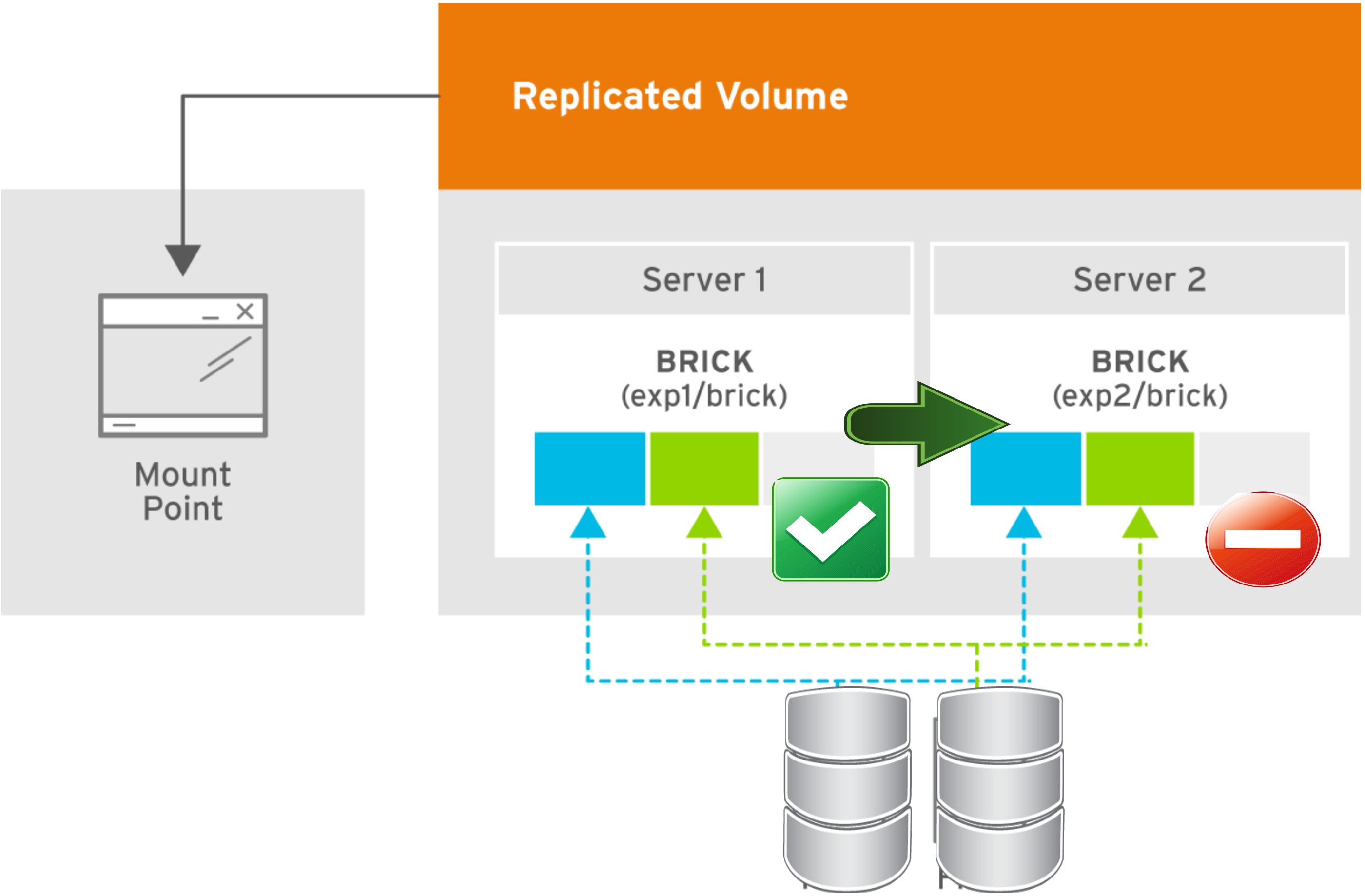
VM Store – High Availability



VM Store – Self-heal

- When the brick comes online, all the VM images that need repair are healed while the read operations continue to happen from the good copy of the brick
- Once the VM image is completely healed the bricks are updated with the info that all copies of the image are in sync with each other

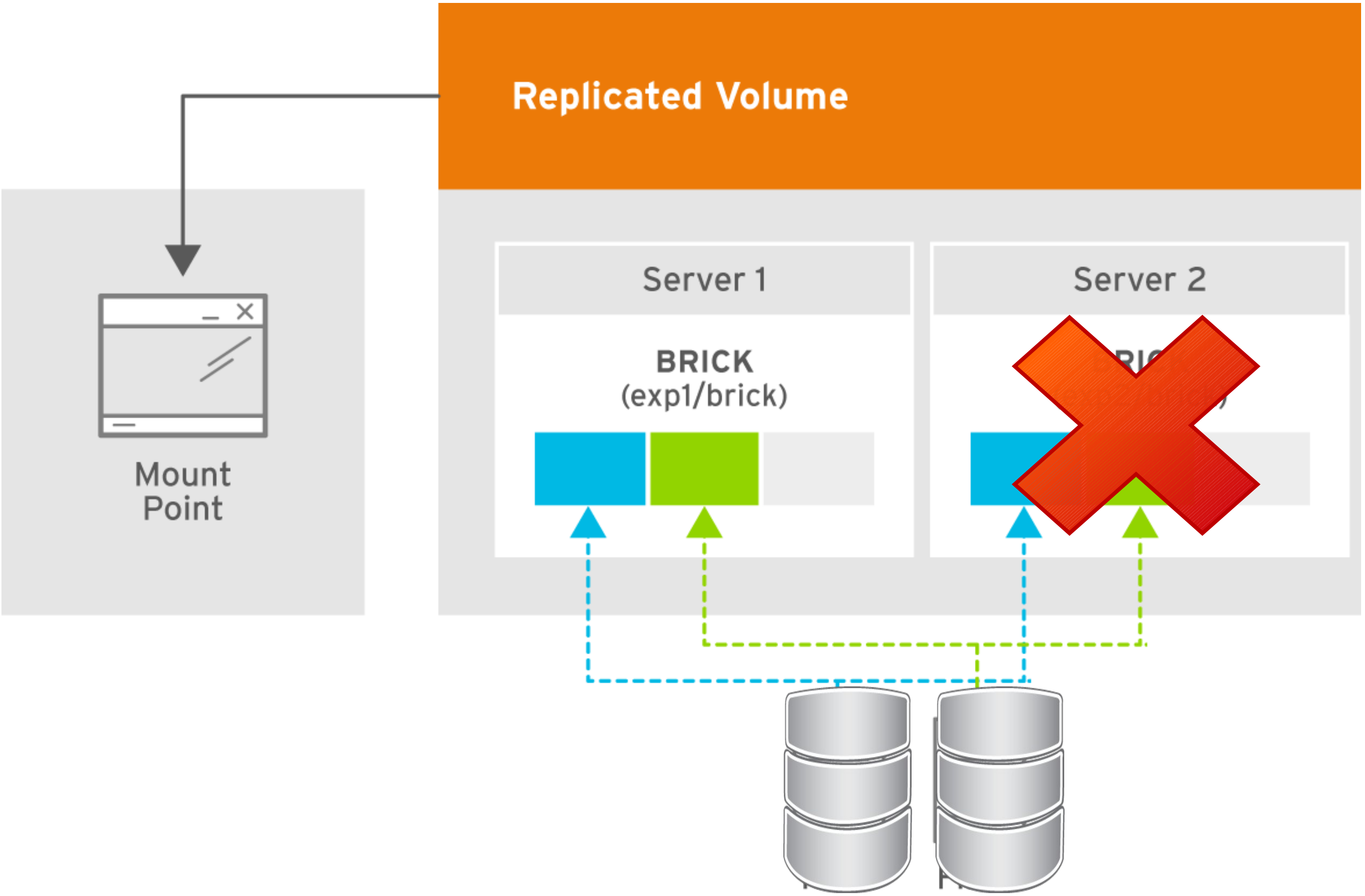
VM Store – Selfheal



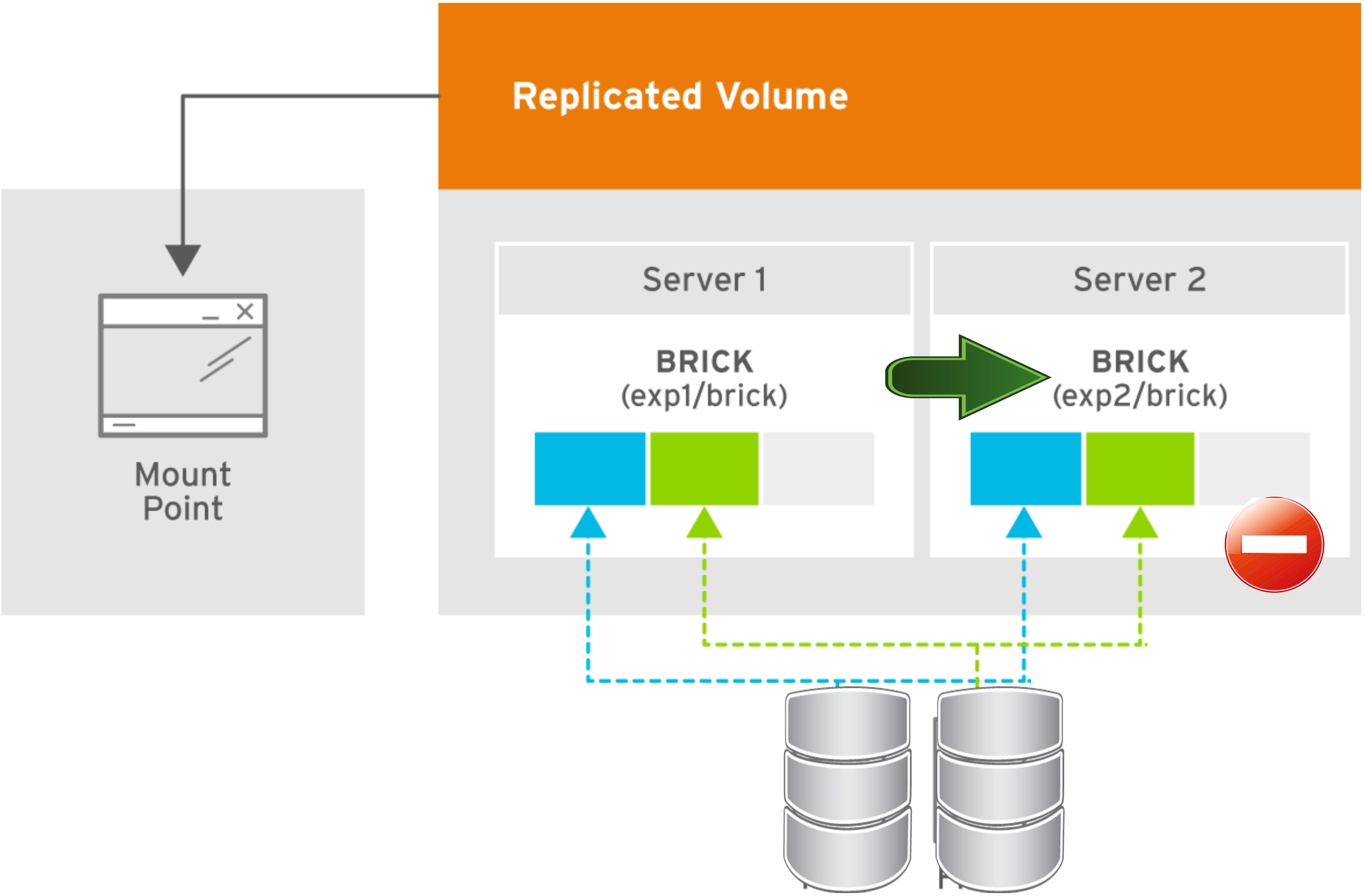
VM Store – Split-brain

- Split-brain is a state when VM images on all the bricks are marked to mean that it is a good copy and the other ones are bad copies.
- Vms go into paused state when this happens. This may happen even when sanlock file goes into split-brain.

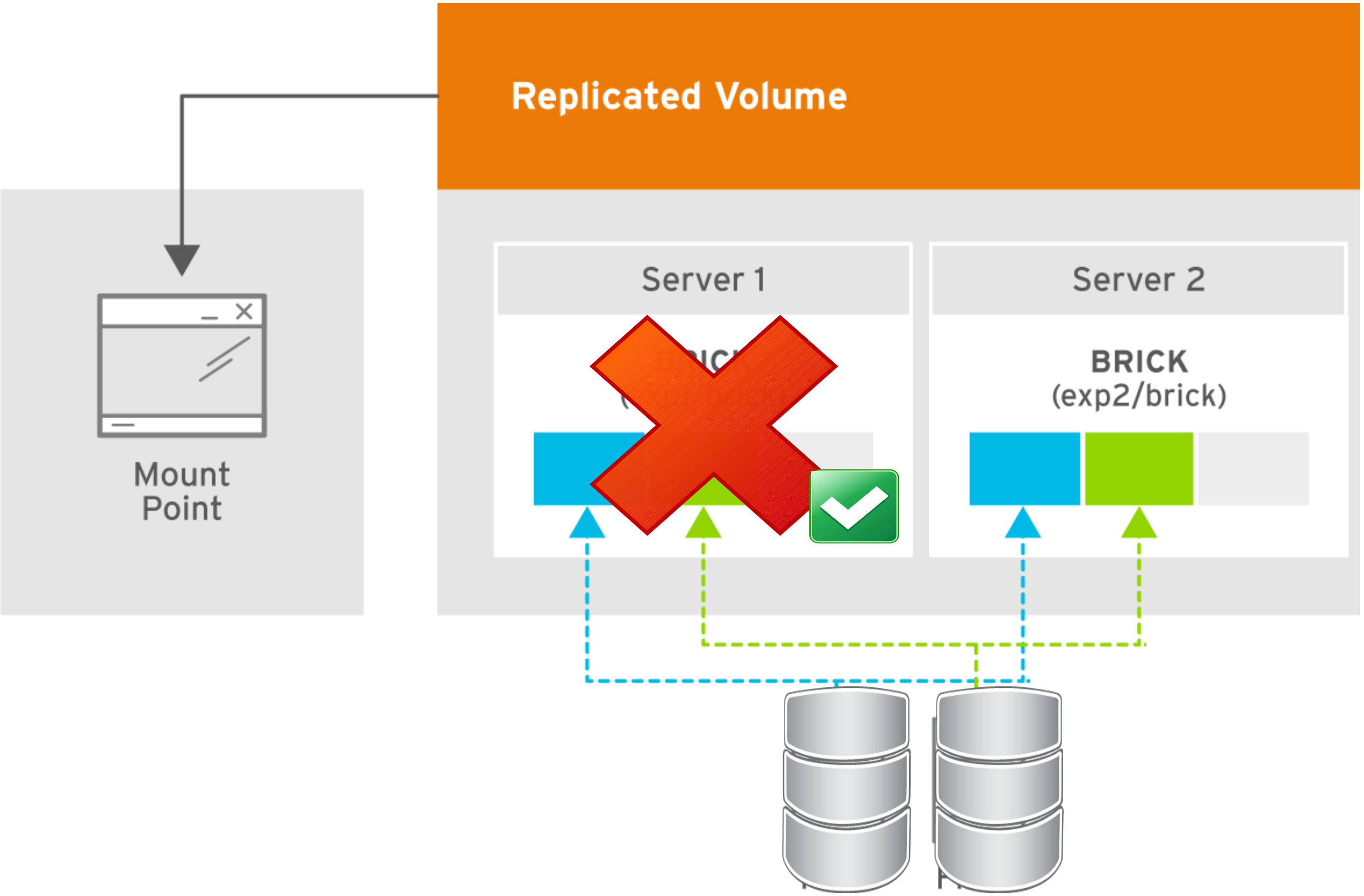
VM Store – Split-brain



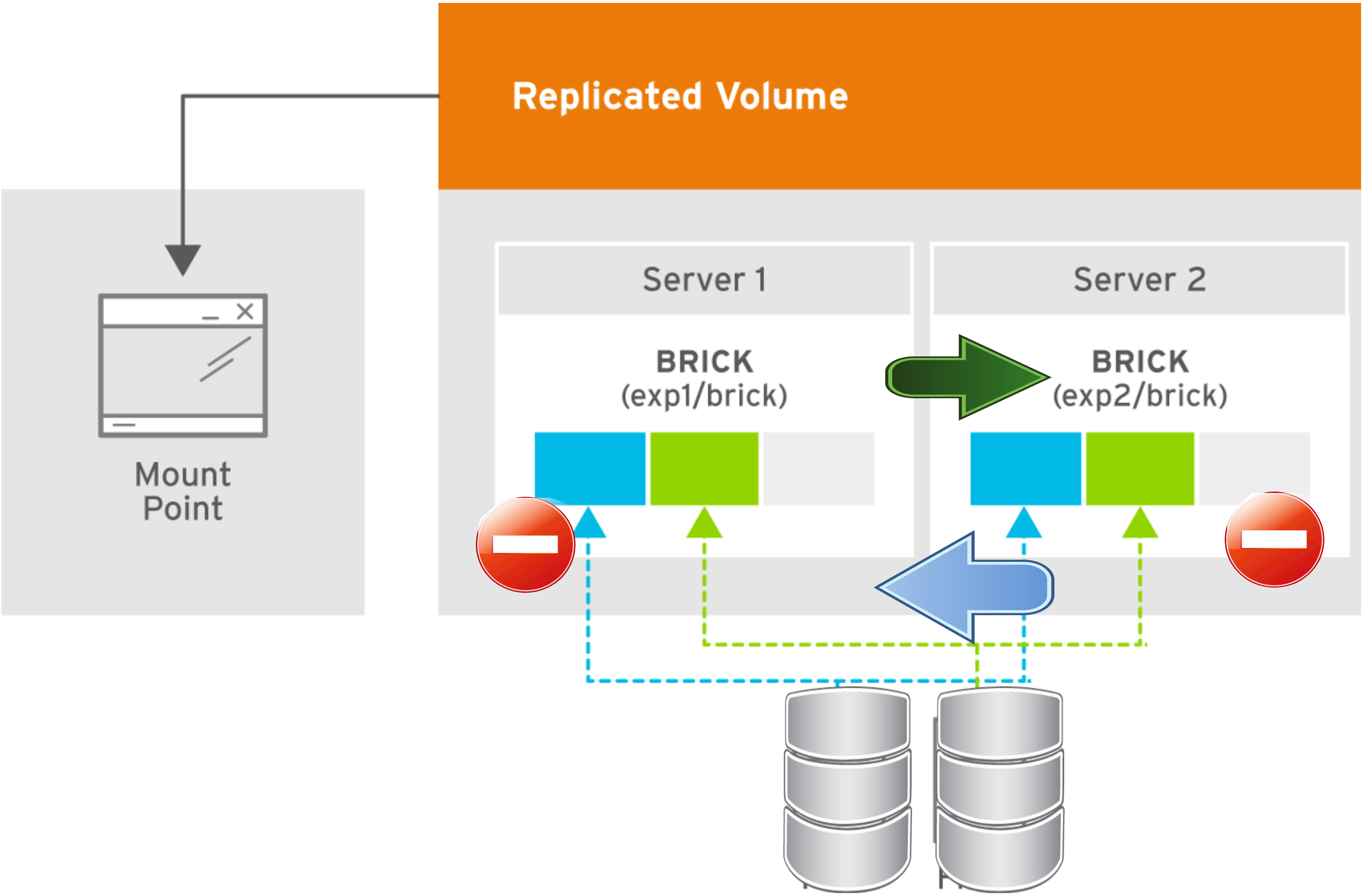
VM Store – Split-brain



VM Store – Split-brain



VM Store – Split-brain



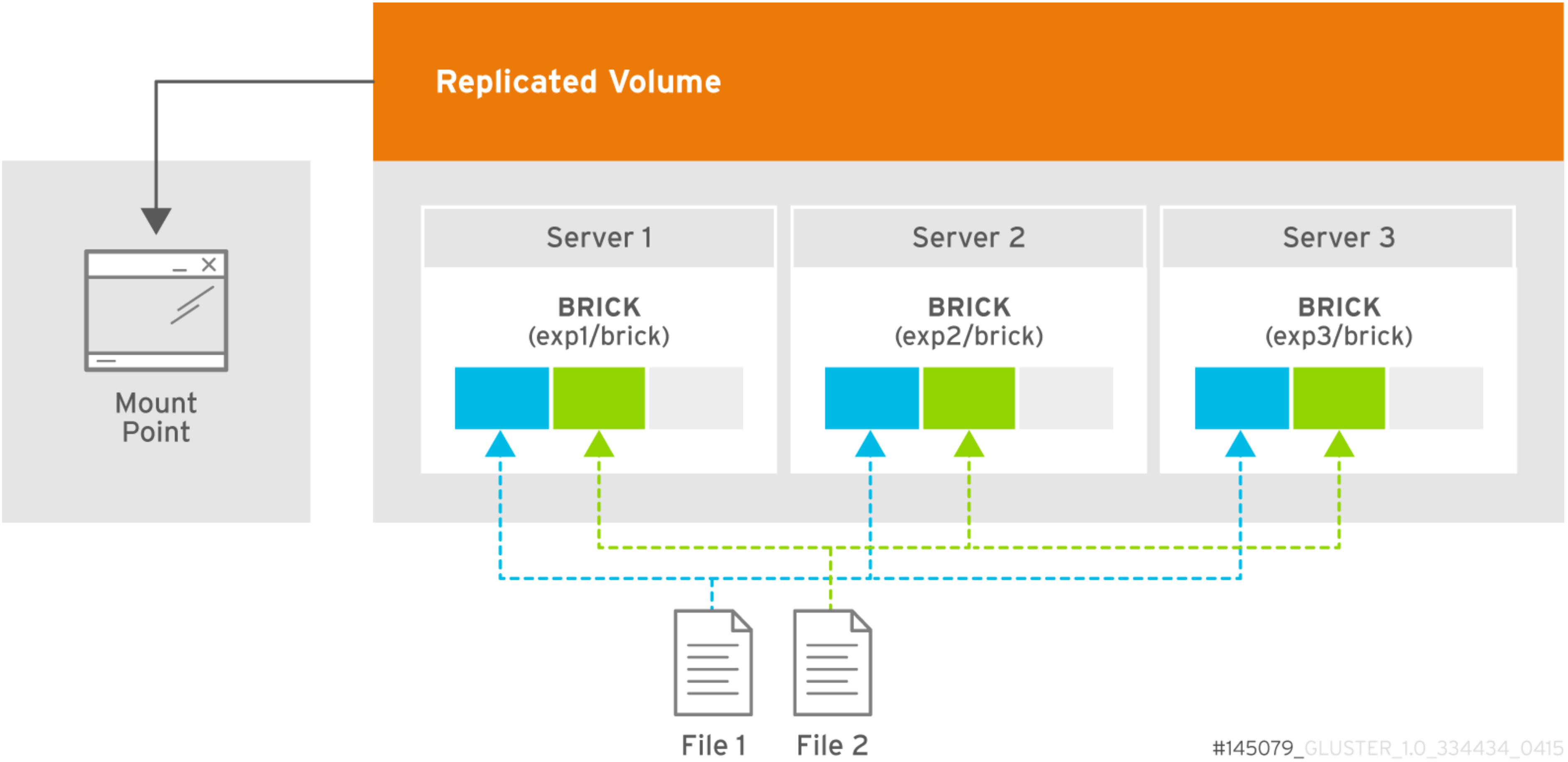
Improvements

- Split-brain Prevention
- Healing time for VM images

Split-brain Prevention

- Main problem with two way replication is there is no useful quorum we can apply to prevent split-brains.
- We need at least 3 bricks in replication, and we can prevent split-brains by failing operations if the operation doesn't succeed on majority of the bricks.
- 3 way replication prevents split-brain, but costly
- arbiter brick instead of third full replica

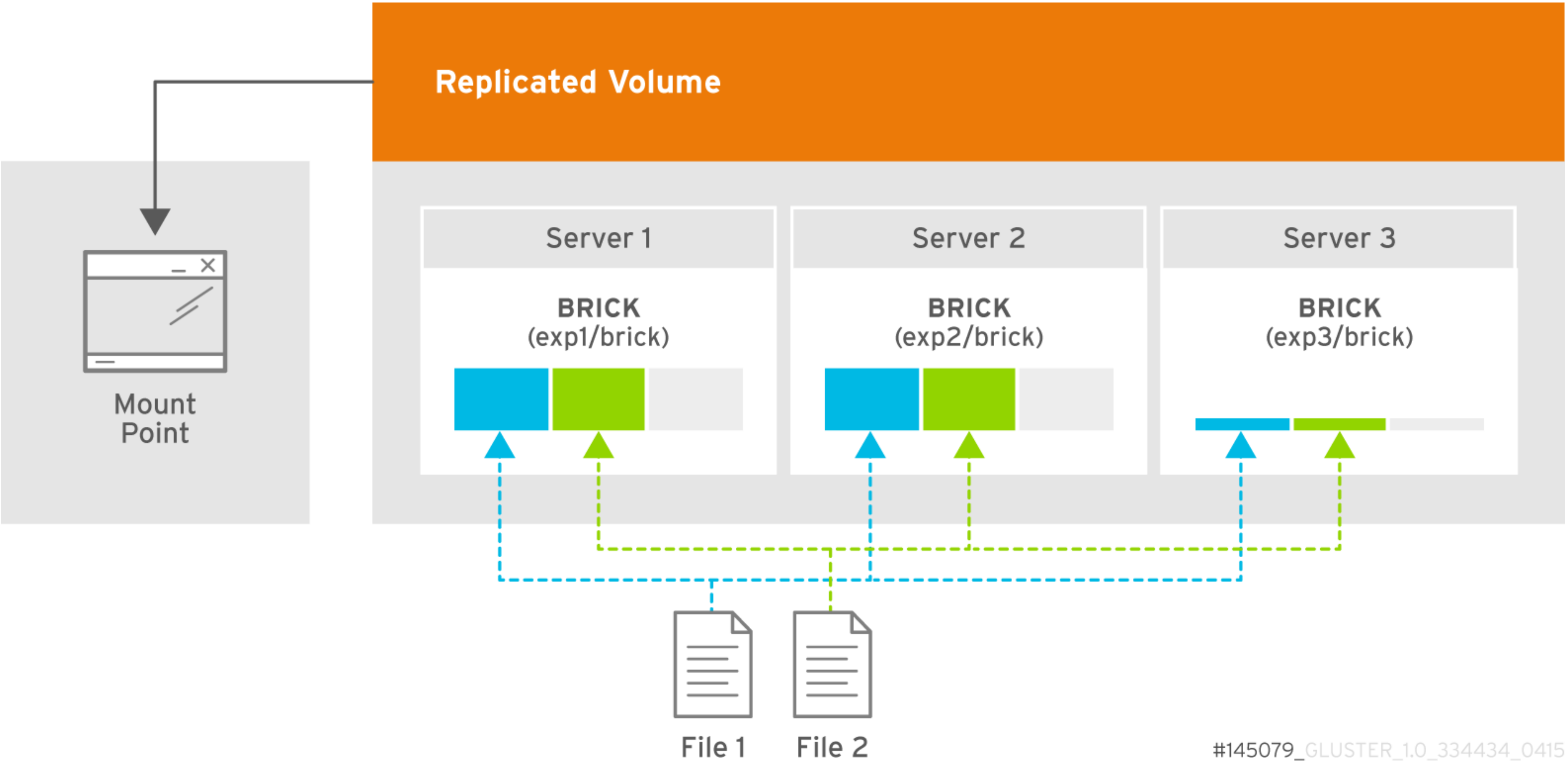
Split-brain Prevention 3 replicas & quorum



Split-brain Prevention - arbiter

- Create the files on all the bricks including arbiter brick.
- Only perform entry, metadata operations on the arbiter along with marking of good/bad status of the files
- Ignore data operations both read/write on arbiter brick
- Allow data operation only when at least 2 bricks are available and at least one data-brick is good copy.
- Healing arbiter brick only involves creating of files and updating metadata and markers.

Split-brain Prevention



Reducing Heal time

- Limitation of the replication design is that it remembers the information of good/bad status at a file level. Even if there is a byte difference, the full file needs to be healed by comparing on all the bricks.
- There are two ways to mitigate this problem
  - Break the VM image into small files i.e. shards
  - Increase granularity of the good/bad status to exact portions in file which need healing

Reducing heal time - sharding

- File is divided into shards with pre-defined shard size
- Default shard size is 4MB
- Heals only shards that need healing
- Better utilization of the disk space, as the shards can be created wherever there is space in the volume.
- Individual shards will never be shown to the user. Only the main files are shown to user.

What we are working on now

- Improve I/O latency with sharding with caching
- Since arbiter brick is anyway going to ignore write operation, send a dummy write to improve bandwidth usage.
- Designing granular change log feature as a parallel solution. In the best case scenario, only the parts that need healing in individual shards will be healed if everything pans out well.

Thanks. Q/A

- Please attend “oVirt and Gluster, hyper-converged! - Martin Sivak” tomorrow – 11:15 AM
- Arbiter (Main contributor: Ravishankar N)

  https://github.com/gluster/glusterfs-specs/blob/master/Feature%20Planning/GlusterFS%203.7/arbiter.md
- Sharding (Main contributor: Krutika Dhananjay)
- https://github.com/gluster/glusterfs-specs/blob/master/Feature%20Planning/GlusterFS%203.7/Sharding%20xlator.md