



# **New LVM RAID Features GUUG FFG Kölle 2016**

**Heinz Mauelshagen  
Consulting Development Engineer**

# TOP

- Current LVM RAID features and missing ones
- New LVM RAID features being developed
- Beyond MD...
- Use cases and example
- New (and existing) RAID features demo
- Conclusions
- Q&A

# Current LVM RAID Features

- Creation of raid1/4,raid5(\_ls,\_la,\_rs,\_ra),raid6(\_zr,\_nr,\_nc) and one limited raid10(\_near) (2 data copies, even number of stripes) layouts
- Resizing of raid1 LVs
- Only growing of raid4/5/6/10 LVs
- RAID LVs limited to maximum of 8 images
- Conversion of linear to raid1 and back
- Conversion of mirror to raid1 but not back
- Repair of RAID LVs (i.e. replacement of dead RAID images)
- Replacement of RAID images (i.e. intact ones)
- Conversion of thinpools and cachepools to RAID
- pvmove of RAID images

## ...and missing LVM RAID Features

- Shrinking of raid4/5/6/10 Lvs
- More than 8 images per RAID LV
- Split mirrors of mirrors
- raid0 support
- Takeover → switch between RAID levels
- Reshaping → layout change on a certain RAID level
  - number of stripes
  - stripe size
  - allocation algorithm
  - number of data copies
- Conversions from native dm targets to RAID  
(but aforementioned linear <-> raid1 and mirror → raid1)
- Layered RAID like raid61 with stripe and/or stripe size variations

# New LVM RAID features being developed

- All MD supported RAID levels supported
  - raid0(\_meta)
  - raid5\_n
  - raid6\_ls\_6, raid6\_la\_6, raid6\_rs\_6, raid6\_ra\_6, raid6\_n\_6  
to be able to takeover raid5\_\* ↔ raid6\_\*
  - raid10\_near, raid10\_offset, raid10\_far  
with any number of stripes/data copies
- Conversion between all raid levels (aka MD takeover) and more
  - raid0 ↔ raid4/raid5/raid6/raid10
  - raid4 ↔ raid5
  - raid5 ↔ raid6
  - raid1 ↔ raid5

# New LVM RAID features being developed...

- Conversion of RAID layouts (aka MD reshaping):
  - number of stripes
  - stripe size
  - region size (but raid0\*)
  - allocation algorithm
- Change of region size on existing RAID LVs to allow for upscale or optimization of region contention
  - `lvconvert --regionsize N $lv`

# Beyond MD...

- MD reshaping causes size changes when adding/removing image components to/from RAID LVs → introduce RAID LV duplication (i.e. raid1 top-level LV with N sub LVs of arbitrary layout)
  - allows to use (almost) any arbitrary LV type (e.g. linear, thin, striped, raid\*)
  - “lvconvert –duplicate –type ... \$lv” converts given LV into a duplicating one with another sub LV defined by “--type ...” and sync to it; sub LVs go by name infix “\_dup\_”
  - each additional “lvconvert –duplicate ...” adds another sub LV and syncs to it
  - lvconvert –unduplicate –name ... \$lv” removes the top-level raid1 LV keeping the named sub LV whilst removing the rest of the sub LVs
  - “lvconvert –splitmirror 1 –name ... \$lv” splits the named sub LV off
  - “lvconvert –splitmirror 1 –trackchanges –name ... \$lv” splits the named sub LV off r/o, allowing it to be merged back into the duplicating LV

## ...Beyond MD

- Introduce raid01, not being supported by MD
  - “lvcreate –type raid01 –stripes N –mirrors M ... \$vg”
- raid10\_far can't be reshaped at all in MD
  - enhanced “lvconvert –mirrors N \$lv” to change number of data copies
  - “lvconvert –type striped/raid10\_far \$lv” to convert to/from raid10\_far
- raid10\_{near,offset} can only be reshaped into each other, change stripe size and add disks in MD; duplication removes constraints
  - “lvconvert –duplicate –type raid10\_{near,offset} –mirrors N ... \$lv” allows to change the number of data copies and/or reduce stripes



# Use Cases and Examples

- (Changing) throughput requirements can be optimized online by reshaping to different layouts (e.g. `raid5_ls` → `raid5_ra`), change number of stripes or stripe size
- Resilience levels can be adjusted online, e.g.:
  - `raid5` → `raid6` to enhance from single image to double image failures
  - `raid4/5/6` → `raid10*` to even stand multiple image failures (i.e. `raid10_offset` with 5 data copies on minimum 5 stripes standing a maximum of 4 image failures; more disks recommended to take advantage of striped throughput advantage)
- Performance enhancements by online restriping
  - Add images to existing striped RAID LVs growing them in size (e.g. `raid6` with 4 data stripes/6 stripes total → 7 data stripes/9 stripes total)
  - Direct conversion of a (say linear) LV to a striped LV by duplication
- Set up enhanced, distributed RAID LVs
  - Duplication across fire areas, e.g. a 2-legged `raid61` LV (use PV tags to distinguish disks)
  - Duplication to cope with varying storage properties, e.g. a 3-way striped sub LV on fast and a 5-way striped on slow PVs

# New (and existing) RAID features demo

- ...

# Conclusions...

- The new LVM/DM RAID features provide full MD RAID functionality
- They go beyond MD conversion constraints with the new duplication feature or the possibility to change the number of data copies in raid10\_far LVs
- Thus, they provide the user with a broad range of powerful possibilities to address performance, resilience and TCO storage requirements and redefine them online thus coping with varying demands

## Q&A

- Which are your LVM RAID use cases?
- What RAID functionality do you request?
- ...