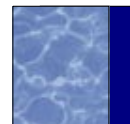
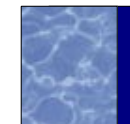
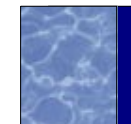


RAID, LVM und Co. Was steckt dahinter?

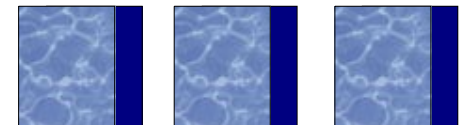


10.03.2001 – 3. Chemnitzer Linux-Tag



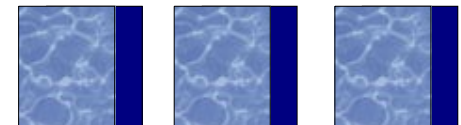
Agenda

- ❑ Aufgabenstellungen im Serverbetrieb
 - ❑ Gewährleistung von Datensicherheit (nicht Datenschutz!)
 - ❑ flexibler Umgang mit Ressourcen, Dynamik
- ❑ RAID
- ❑ Logical Volume Manager (LVM)
- ❑ RAID + LVM



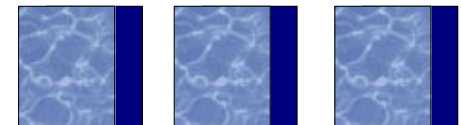
Problemstellung #1

- ❑ Absicherung gegen Ausfall von Festplatten
 - ❑ Wie lange brauche ich, um das System wieder flott zu bekommen?!
 - ❑ Wo ist das Backup-Band?! ;–)
 - ❑ ...
- ❑ Performance-Gewinn?!



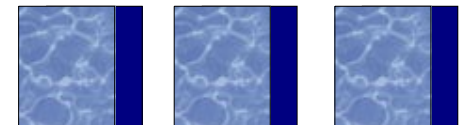
RAID – was ist das?!

- ❑ RAID = „Redundant Array of Inexpensive / Independent Disks“
- ❑ weitere Infos, siehe HowTo's:
<http://www.linuxdoc.org/HOWTO/Software-RAID-HOWTO.html>



Linear mode

- ❑ 2 oder mehr Disks unterschiedlicher Grösse
- ❑ Prinzip: „Aneinanderhängen“
- ❑ keine Redundance
- ❑ keine Performancevorteile



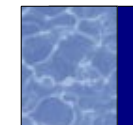
RAID0

- ❑ 2 oder mehr Disks möglichst gleicher Grösse
- ❑ Prinzip: „mal hier – mal da“
- ❑ keine Redundance
- ❑ Vorteile durch paralleles Schreiben und Lesen auf unterschiedliche Disks
- ❑ auch als „Stripe Mode“ bekannt



RAID1

- ❑ 2 oder mehr (N) Disks gleicher Grösse
- ❑ Prinzip: „Spiegelung“
- ❑ Redundance durch Kopien
Crashs von (N-1) Disks unkritisch
- ❑ Read-Performance „N-mal“ besser;
Write-Performance u.U. schlecht, da
N-faches Schreiben notwendig
- ❑ $\frac{1}{2}$ des Diskspace von 2 Disks nutzbar



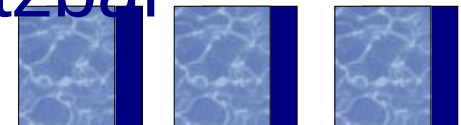
RAID4

- ❑ 3 oder mehr Disks gleicher Grösse
- ❑ Prinzip: „mal hier – mal da“ + „parity“-Disk
- ❑ Redundance: Recovery durch parity-Infos möglich; Crash einer Disk unkritisch
- ❑ Performance: naja:) – parity-Disk muss schnell sein, sonst Flaschenhals
- ❑ relativ ungebräuchlicher Mode



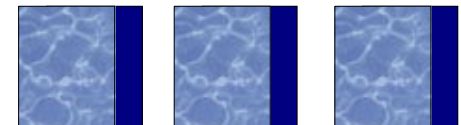
RAID5

- ❑ 3 oder mehr Disks gleicher Grösse
- ❑ Prinzip: Chaos pur ;)...
auch parity-Infos werden verteilt
- ❑ Redundance: Recovery durch parity-Infos möglich; Crash einer Disk unkritisch
- ❑ Performance: besser als Einzeldisk; schwer abschätzbar
- ❑ 2/3 des Diskspace von 3 Disks nutzbar



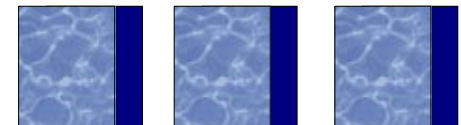
Hardware-RAID-Systeme

- ❑ spezielle Hardware-Controller notwendig
- ❑ Konfiguration der Arrays via Firmware / BIOS oder Spezialsoftware
- ❑ Prinzip:
Betriebssystem sieht nur das virtuelle Array
(Logical Disk)



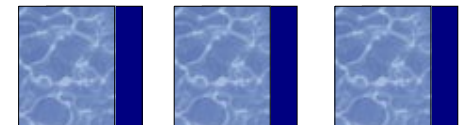
Software-RAID im Linux

- ❑ RAID-Unterstützung im Kernel für Modes linear, raid0, raid1 und raid5
 - ❑ Infos:
cat /proc/mdstat
- ❑ notwendige Utilities:
 - ❑ md-utils (Kernel 2.0.x)
 - ❑ raidtools (Kernel 2.2.x ggf. mit Patch und 2.4.x)



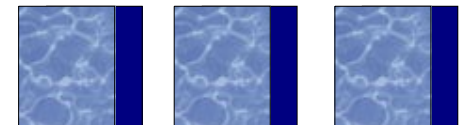
RAID ist toll; aber trotzdem...

- ❑ REGELMÄSSIGES BACKUP DER DATEN
- ❑ Unterbrechnungsfreie StromVersorgung (USV) im Server-Betrieb
- ❑ Hints
 - ❑ kritische Daten?! ... dann Hardware-RAID
 - ❑ ggf. Spare-Disks für den Notfall vorhalten
 - ❑ SWAP nicht auf RAID!



RaidTools live

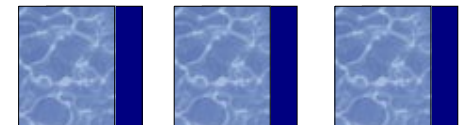
- ❑ Beispiele in „grüner Box“ unter `/usr/share/doc/packages/raidtools/`
- ❑ Vier Schritte zum Ziel...
 - ❑ Kopieren & Anpassen, Erstellen von `/etc/raidtab`
 - ❑ `# mkraid /dev/md0`
 - ❑ `# mke2fs /dev/md0`
`# mkreiserfs /dev/md0`
 - ❑ `# mount /dev/md0 ...`
`... and it works :*)`



/etc/raidtab für RAID5

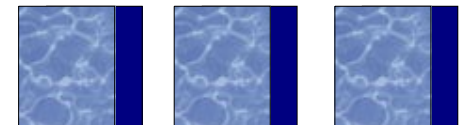
```
peacock:~ # cat /etc/raidtab
#
# 'persistent' RAID5 setup, with no spare disks:
#
raiddev /dev/md0
    raid-level          5
    nr-raid-disks       3
    nr-spare-disks      0
    persistent-superblock 1
    chunk-size          4

    device              /dev/hdb7
    raid-disk           0
    device              /dev/hdb8
    raid-disk           1
    device              /dev/hdb9
    raid-disk           2
```



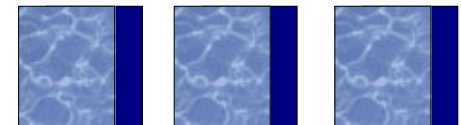
Problemstellung #2

- ❑ Filesystem voll :(
- ❑ Partition zu klein ?!
- ❑ Filesystem soll grösser als die Kapazität einer Festplatte sein (z.B. 100GB)

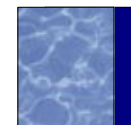
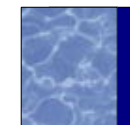
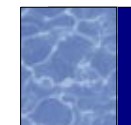
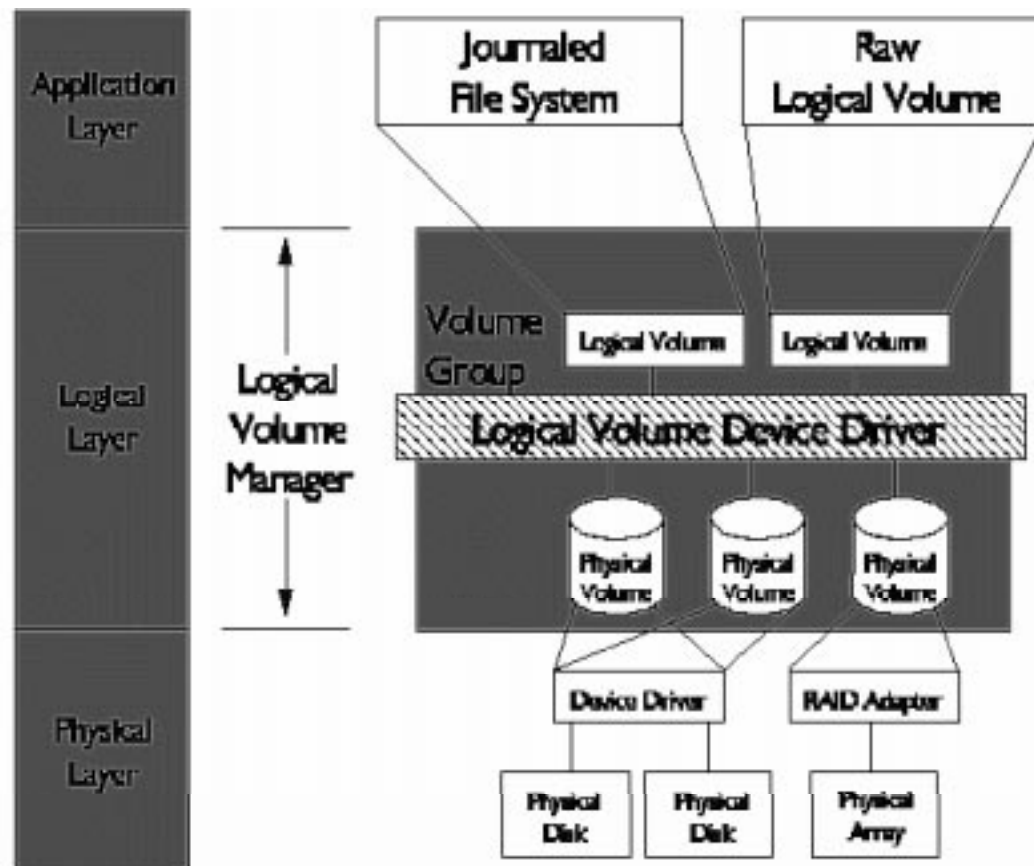


Logical Volume Manager (LVM)

- ❑ Schicht zwischen Hardware / Device-Driver und Filesystemen
- ❑ Virtualisierung der physikalischen Devices (Disks) auf logische Devices (Logical Volumes)
- ❑ Verweis auf LVM-HowTo, AIX-Doku, ...
<http://www.linuxdoc.org/HOWTO/LVM-HOWTO.html>

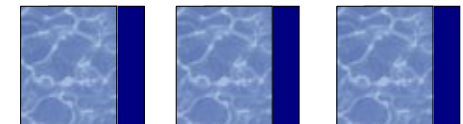


Überblick



LVM-Glossar

- ❑ PV = Physical Volume (Harddisk, Typ8e-Partition)
- ❑ PE = Physical Extents (Datenblöcke auf PV)
- ❑ VG = Volume Group (Sammlung von PV's)
- ❑ LV = Logical Volume ('logische Partition')
- ❑ FS = Filesystem as usual



Step 1: Anlegen von PV's

```
peacock:~ # fdisk -l /dev/hdb
Disk /dev/hdb: 255 heads, 63 sectors, 523 cylinders
Units = cylinders of 16065 * 512 bytes
```

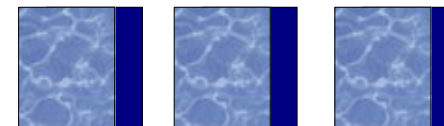
Device	Boot	Start	End	Blocks	Id	System
...						
/dev/hdb7		22	28	56196	8e	Linux LVM
/dev/hdb8		29	35	56196	8e	Linux LVM
/dev/hdb9		36	42	56196	8e	Linux LVM
...						

```
peacock:~ # pvcreate /dev/hdb7
pvcreate -- physical volume "/dev/hdb7" successfully created
```

```
peacock:~ # pvcreate /dev/hdb8
pvcreate -- physical volume "/dev/hdb8" successfully created
```

```
peacock:~ # pvcreate /dev/hdb9
pvcreate -- physical volume "/dev/hdb9" successfully created
```

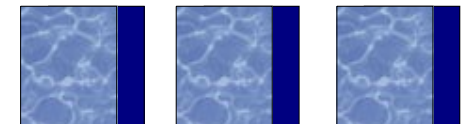
```
peacock:~ # pvdisplay /dev/hdb7
pvdisplay -- "/dev/hdb7" is a new physical volume of 54.88 MB
```



Step 2: Anlegen von VG's

```
peacock:~ # vgcreate /dev/testvg /dev/hdb7 /dev/hdb8 /dev/hdb9
vgcreate -- INFO: using default physical extent size 4 MB
vgcreate -- INFO: maximum logical volume size is 255.99 Gigabyte
vgcreate -- doing automatic backup of volume group "testvg"
vgcreate -- volume group "testvg" successfully created and activated
```

```
peacock:~ # vgdisplay /dev/testvg
--- Volume group ---
VG Name                testvg
VG Access              read/write
VG Status              available/resizable
VG #                  1
MAX LV                 256
Cur LV               0
Open LV               0
MAX LV Size           255.99 GB
Max PV                256
Cur PV               3
Act PV               3
VG Size               156 MB
PE Size               4 MB
Total PE              39
Alloc PE / Size       0 / 0
Free PE / Size      39 / 156 MB
VG UUID               oqqAZp-QCcy-HIbi-MMc3-006U-0CYw-x9fMB1
```



Step 3: Erstellen von LV's

```
peacock:~ # lvcreate -L 10M -n datalv -i 2 testvg
lvcreate -- INFO: using default stripe size 16 KB
lvcreate -- rounding up size to physical extent boundary "12 MB"
lvcreate -- rounding 12288 KB to stripe boundary size 16384 KB / 4 PE
lvcreate -- doing automatic backup of "testvg"
lvcreate -- logical volume "/dev/testvg/datalv" successfully created
```

```
peacock:~ # lvsdisplay -v /dev/testvg/datalv
```

--- Logical volume ---

LV Name	/dev/testvg/datalv
VG Name	testvg

...

LV Size	16 MB
Current LE	4
Allocated LE	4
Stripes	2
Stripe size (KByte)	16

...

--- Distribution of logical volume on 2 physical volumes ---

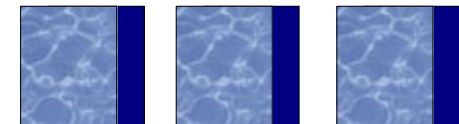
PV Name	PE on PV	reads	writes
/dev/hdb7	2	0	2
/dev/hdb8	2	0	0

--- logical volume i/o statistic ---

0 reads 2 writes

--- Logical extents ---

LE	PV	PE	reads	writes
00000	/dev/hdb7	00000	0	2
00001	/dev/hdb7	00001	0	0
00002	/dev/hdb8	00000	0	0
00003	/dev/hdb8	00001	0	0



Step 4: Anlegen von FS's

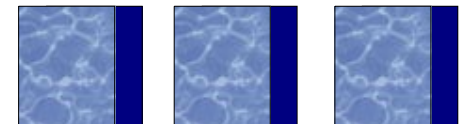
```
peacock:~ # mke2fs /dev/testvg/datalv
mke2fs 1.19, 13-Jul-2000 for EXT2 FS 0.5b, 95/08/09
Filesystem label=
OS type: Linux
Block size=1024 (log=0)
Fragment size=1024 (log=0)
4096 inodes, 16384 blocks
819 blocks (5.00%) reserved for the super user
First data block=1
2 block groups
8192 blocks per group, 8192 fragments per group
2048 inodes per group
Superblock backups stored on blocks:
    8193
```

```
Writing inode tables: done
Writing superblocks and filesystem accounting information: done
```

```
peacock:~ # mount /dev/testvg/datalv /mnt
```

```
peacock:~ # df
```

Filesystem	1k-blocks	Used	Available	Use%	Mounted on
...					
/dev/testvg/datalv	15863	13	15031	1%	/mnt



Filesystem voll?! ... macht nix :)

```
peacock:/ # df
Filesystem            1k-blocks      Used Available Use% Mounted on
/dev/testvg/datalv      15863        15863          0 100% /mnt
```

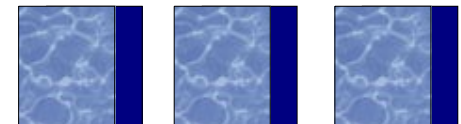
```
peacock:/mnt # lvextend -L +10M /dev/testvg/datalv
lvextend -- rounding size to physical extent boundary
lvextend -- rounding size 28672 KB to stripe boundary size 32768 KB
lvextend -- extending logical volume "/dev/testvg/datalv" to 32 MB
```

```
peacock:/ # umount /mnt
peacock:/ # e2fsck -f /dev/testvg/datalv
...
```

```
peacock:/ # resize2fs /dev/testvg/datalv
resize2fs 1.19 (13-Jul-2000)
The filesystem on /dev/testvg/datalv is now 32768 blocks long.
```

```
peacock:/ # mount /dev/testvg/datalv /mnt
```

```
peacock:/ # df
Filesystem            1k-blocks      Used Available Use% Mounted on
...
/dev/testvg/datalv      31729        15863      14556   53% /mnt
```



LVM misc.

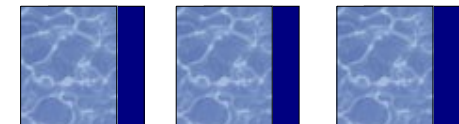
```
peacock:/ # lvdisplay -v /dev/testvg/data1v
```

```
...
--- Logical extents ---
LE      PV                      PE      reads    writes
000000 /dev/hdb7                000000  0         4129
...
000003 /dev/hdb7                000003  0         132
000004 /dev/hdb8                000000  0         4096
...
000007 /dev/hdb8                000003  0         128
```

```
peacock:/ # pvmove -n /dev/testvg/data1v /dev/hdb8 /dev/hdb9
pvmove -- moving physical extents in active volume group "testvg"
pvmove -- WARNING: moving of active logical volumes may cause data loss!
pvmove -- do you want to continue? [y/n] y
pvmove -- doing automatic backup of volume group "testvg"
pvmove -- 4 extents of physical volume "/dev/hdb8" successfully moved
```

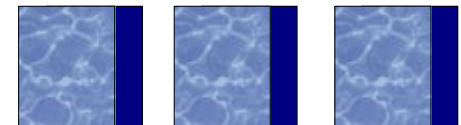
```
peacock:/ # lvdisplay -v /dev/testvg/data1v
```

```
...
--- Logical extents ---
LE      PV                      PE      reads    writes
000000 /dev/hdb7                000000  0         4130
...
000003 /dev/hdb7                000003  0         132
000004 /dev/hdb9                000000  0         4096
...
000007 /dev/hdb9                000003  0         128
```



RAID + LVM

- ❑ viele Kombinationen sinnvoll, z.B.
 - ❑ RAID
 - ❑ Hardware RAID1
(Spiegelung / Read-Performance!)
oder
 - ❑ Hardware RAID5
(hohe Datensicherheit mit wenig Disks)
 - ❑ LVM
 - ❑ Verschmelzen von mehreren RAID-Arrays
zu einem grossen Volume



mwei@linuxvortrag:~ > logout



<http://www.tu-chemnitz.de/~als/images/penguins/>

<http://avalon.tuts.nu/~mwei/archiv/>

