

# Systemverwaltung Solaris 10

Rolf M Dietze

Aug-2013

## Simple Tools

LAB: Simple Tools

Loopback File Driver

Device Filesystem

LAB: Device Filesystem

Benutzerverwaltung, lokal

Network Filesystem, NFS

Automounter, autoFS

zfs

Systemboot x86/x64-Solaris

Service Management Facility

SMF Admin

SMF Zustände

Milestones

SMF Konfiguration

SMF Manifeste

Graphiken, Tabellen und Übersichten, mit freundlicher Genehmigung des Springer-Verlages aus: OpenSolaris für Anwender Administratoren und Rechenzentren, Dietze, Heuser, Schillig, März 2006

# Simple Tools

## Einfache Tools zur Übersicht

## Systembelastung, prstat(1M)

PID	USERNAME	SIZE	RSS	STATE	PRI	NICE	TIME	CPU	PROCESS/M
1285	root	3912K	3176K	cpu1	59	0	0:00:00	0.1%	prstat/1
442	root	7632K	5976K	sleep	59	0	0:00:19	0.0%	intrd/1
133	root	7248K	4040K	sleep	59	0	0:00:03	0.0%	nscd/32
145	root	5432K	4152K	sleep	59	0	0:00:07	0.0%	picld/9
209	root	2056K	1376K	sleep	59	0	0:00:00	0.0%	cron/1
230	root	4072K	2040K	sleep	59	0	0:00:00	0.0%	hald-run
268	daemon	2976K	1904K	sleep	60	-20	0:00:00	0.0%	nfs4cbd/2
124	daemon	7776K	4696K	sleep	59	0	0:00:00	0.0%	kcfcd/3
300	root	1744K	1024K	sleep	59	0	0:00:00	0.0%	utmpd/1
216	root	3440K	1736K	sleep	59	0	0:00:00	0.0%	dbus-daem
290	root	2464K	1344K	sleep	59	0	0:00:00	0.0%	sac/1
136	root	2264K	1656K	sleep	59	0	0:00:00	0.0%	powerd/3
Total: 42 processes, 212 lwps, load averages: 0.00, 0.00, 0.00									

# CPU Belastung

```
# uptime
```

```
12:21pm up 22:46, 1 user, load average: 0.00, 0.00, 0.00
```

```
# mpstat 1 2
```

CPU	minf	mjf	xcal	intr	ithr	csw	icsw	migr	smtx	srw	syscl	usr
0	1	0	2	268	65	99	0	5	1	0	52	0
1	2	0	2	12	7	111	0	5	0	0	37	0
CPU	minf	mjf	xcal	intr	ithr	csw	icsw	migr	smtx	srw	syscl	usr
0	0	0	1	261	58	104	0	4	0	0	0	0
1	7	0	1	12	8	89	0	5	0	0	49	0

# I/O Belastung

```
# iostat -xM 1
```

## extended device statistics

device	r/s	w/s	Mr/s	Mw/s	wait	actv	svc_t	%w	%b
md0	0.0	0.4	0.0	0.0	0.0	0.2	596.3	0	0
md1	0.0	0.5	0.0	0.1	0.0	0.1	128.3	0	0
ramdisk1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0
sd21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0

## extended device statistics

device	r/s	w/s	Mr/s	Mw/s	wait	actv	svc_t	%w	%b
md0	0.0	0.4	0.0	0.0	0.0	0.2	596.3	0	0
md1	0.0	0.5	0.0	0.1	0.0	0.1	128.3	0	0
ramdisk1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0
sd21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0

# User

```

0 1 rmd@mirage pts/20 ~ 1> w
12:30pm up 4 day(s), 12:51, 7 users, load average: 0.18, 0.18,
User      tty          login@  idle   JCPU   PCPU   what
theuser  dtlocal     Fri10am 8:58   35    35    /usr/lib/ospm/os
theuser  pts/5       Fri10am 25:19  1:07  1:05  tcsh
rrother  dtlocal     Fri12pm 8:58   12    12    /usr/lib/ospm/os
rmd      dtlocal     Fri12pm 8:58           rlogin arietis
rmd      pts/7       Fri12pm 4days           rlogin arietis
rmd      pts/8       Fri12pm 4days    1    -tcsh
rmd      pts/9       Fri12pm 4days           rlogin arietis
rmd      pts/10      Fri12pm 3days           tcsh

```

# User

```
0 1 rmd@mirage pts/20 ~ 2> who
theuser      dtlocal      Aug  8 10:04      (:2)
theuser      pts/5        Aug  8 10:11      (:2.0)
rrother      dtlocal      Aug  8 12:05      (:4)
rmd          dtlocal      Aug  8 12:36      (:5)
rmd          pts/7        Aug  8 12:36      (:5.0)
rmd          pts/8        Aug  8 12:36      (:5.0)
rmd          pts/9        Aug  8 12:37      (:5.0)
rmd          pts/10       Aug  8 12:37      (:5.0)
rrother      pts/11       Aug  8 14:03      (:4.0)
theuser      pts/14       Aug 11 10:39      (:2.0)
rmd          pts/15       Aug 11 16:23      (arietis)
```



## tty, Prozesse

```
0 1 rmd@mirage pts/20 ~ 3> tty  
/dev/pts/20
```

```
0 1 rmd@mirage pts/20 ~ 3> ps  
  PID TTY          TIME CMD  
11660 pts/20        0:00 tcsh  
11667 pts/20        0:00 ps
```

## Prozesse

```
0 1 rmd@mirage pts/20 ~ 5> ps -ef|more
  UID  PID  PPID  C   STIME TTY  TIME CMD
  root   0    0    0   Aug 07 ?    0:02 sched
  root   1    0    0   Aug 07 ?    0:01 /sbin/init
  root   2    0    0   Aug 07 ?    0:00 pageout
  root   3    0    1   Aug 07 ?   15:27 fsflush
  root  381    7    0   Aug 07 ?    0:00 /usr/lib/saf/sac -t 300
  root   7    1    0   Aug 07 ?    0:08 /lib/svc/bin/svc.startd
  root   9    1    0   Aug 07 ?    0:27 /lib/svc/bin/svc.configd
daemon 345    1    0   Aug 07 ?    0:01 /usr/lib/nfs/nfsmapid
daemon 351    1    0   Aug 07 ?    0:00 /usr/lib/nfs/nfs4cbd
   root  440  384   0      - ?          0:00 <defunct>
   root  133    1    0   Aug 07 ?    0:00 /usr/lib/power/powerd
daemon 125    1    0   Aug 07 ?    0:01 /usr/lib/crypto/kcfd
....
```

# Prozesse

```
0 2 rmd@vega pts/8 ~ 1% ptree 7
7    /lib/svc/bin/svc.startd
409  /usr/lib/saf/sac -t 300
429  /usr/lib/saf/ttymon
413  /usr/lib/saf/ttymon -g -d /dev/console -l console -
```

## Prozesse

```
0 2 rmd@vega pts/8 ~ 2# pargs 413
413:    /usr/lib/saf/ttymon -g -d /dev/console -l console -
argv[0]: /usr/lib/saf/ttymon
argv[1]: -g
argv[2]: -d
argv[3]: /dev/console
argv[4]: -l
argv[5]: console
argv[6]: -m
argv[7]: ldterm,ttcompat
argv[8]: -h
argv[9]: -p
argv[10]: vega console login:
```

## LAB: Simple Tools

Wieviele Prozesse laufen aktuell?

Welche ProzessID hat der svc.configd?

Welche ProcessID hat der svc.startd?

Welcher Prozess ist der Parentprozess der aktuellen Shell?

Wieviele Benutzer sind aktuell angemeldet?

Von woher kommen die aktuellen Logins?

Welche Argumente wurden beim Aufruf des gdm angegeben?

Welche Argumente wurden dem Mixer-Applet mitgegeben?

lofi

# Loopback File Driver

# lofi

```
nx1# lofiadm -a /export/cdromimage  
/dev/lofi/1
```

```
nx1# lofiadm  
Block Device      File  
/dev/lofi/1       /export/cdromimage
```

```
nx1# mount -F hsfs -o ro /dev/lofi/1 /mnt
```

```
nx1# umount /mnt
```

```
nx1# lofiadm -d /dev/lofi/1
```

# Devicefilesystem

## Devicefilesystem



# devFS

```
nx1# ls -lL /dev/term/*
crw-rw-rw-  1 root      sys          20,  0 Sep 12  2003 /dev/term/a
crw-rw-rw-  1 root      sys          20,  1 Oct 20 23:31 /dev/term/b
```

```
nx1> ls -l /dev/dsk/c0t0d0s7
lrwxrwxrwx 1 root  root  42 Feb 29 23:33 /dev/dsk/c0t0d0s7 ->
          ../../devices/pci@4,4000/scsi@6,1/sd@0,0 :h
```

# devFS

```
nx1> ls -lL /dev/dsk/c4t0d0s?
```

```
brw-r----- 1 root sys 118,488 May 25 10:58 /dev/dsk/c4t0d0s0
brw-r----- 1 root sys 118,489 May 25 10:58 /dev/dsk/c4t0d0s1
brw-r----- 1 root sys 118,490 May 25 10:58 /dev/dsk/c4t0d0s2
brw-r----- 1 root sys 118,491 May 25 10:58 /dev/dsk/c4t0d0s3
brw-r----- 1 root sys 118,492 May 25 10:58 /dev/dsk/c4t0d0s4
brw-r----- 1 root sys 118,493 May 25 10:58 /dev/dsk/c4t0d0s5
brw-r----- 1 root sys 118,494 May 25 10:58 /dev/dsk/c4t0d0s6
brw-r----- 1 root sys 118,495 May 25 10:58 /dev/dsk/c4t0d0s7
```

# devFS

```
nx1> ls -lL
total 0
crw-rw-rw-  1 root    sys      33,777 May 35 20:30 0
crw-rw-rw-  1 root    sys      33,841 May 31 19:01 0b
crw-rw-rw-  1 root    sys      33,845 May 31 19:01 0bn
crw-rw-rw-  1 root    sys      33,793 May 31 19:01 0c
crw-rw-rw-  1 root    sys      33,857 May 31 19:01 0cb
crw-rw-rw-  1 root    sys      33,861 May 31 19:01 0cbn
crw-rw-rw-  1 root    sys      33,797 May 31 19:01 0cn
crw-rw-rw-  1 root    sys      33,785 May 31 19:01 0h
crw-rw-rw-  1 root    sys      33,849 May 31 19:01 0hb
crw-rw-rw-  1 root    sys      33,853 May 31 19:01 0hbn
crw-rw-rw-  1 root    sys      33,789 May 31 19:01 0hn
...
```

## devFS, path\_to\_inst

```
extended device statistics
device      r/s      w/s      kr/s      kw/s  wait  actv   svc_t   %w   %b
sd0         0.0      0.0      0.0      0.0   0.0   0.0    0.0    0    0
sd1         0.0      0.0      0.0      0.0   0.0   0.0    0.0    0    0
sd2         0.0      0.0      0.0      0.0   0.0   0.0    0.0    0    0
sd3         0.0      5.0      0.0     30.5   0.0   0.0    8.7    0    3
sd4         0.0      0.0      0.0      0.0   0.0   0.0    0.0    0    0
....
nfs1        0.0      0.0      0.0      0.0   0.0   0.0    0.0    0    0
....
```

## devFS, path\_to\_inst

*“OBP-Devicepath“ Instanznummer “Treibername“*

```
"/pci@4,4000/scsi@6,1/sd@1,0" 16 "sd"
```

## devFS, path\_to\_inst

```

/options" 0 "options"
"/pseudo" 0 "pseudo"
"/pci@1f,4000" 0 "pcipsy"
"/pci@1f,4000/network@01,1" 0 "hme"
"/pci@1f,4000/ebus@1" 0 "ebus"
"/pci@1f,4000/ebus@1/su_pnp@14,3803f8" 0 "su_pnp"
"/pci@1f,4000/ebus@1/su_pnp@14,3602f8" 1 "su_pnp"
"/pci@1f,4000/ebus@1/se@14,400000" 0 "se"
"/pci@1f,4000/ebus@1/fdthree@14,3203f0" 0 "fd"
"/pci@1f,4000/ebus@1/ecpp@14,340278" 0 "ecpp"
"/pci@1f,4000/ebus@1/power@14,724000" 0 "power"
"/pci@1f,4000/ebus@1/i2c@14,600000" 0 "i2c"
"/pci@1f,4000/ebus@1/i2c@14,600000/adc@0,9e" 0 "i2cadc"
"/pci@1f,4000/ebus@1/i2c@14,600000/adc@0,9c" 1 "i2cadc"
"/pci@1f,4000/ebus@1/i2c@14,600000/adc@0,9a" 2 "i2cadc"
"/pci@1f,4000/ebus@1/i2c@14,600000/gpio@0,78" 0 "i2cgpio"
"/pci@1f,4000/ebus@1/i2c@14,600000/gpio@0,70" 1 "i2cgpio"
"/pci@1f,4000/pci@3" 0 "pci_pci"
"/pci@1f,4000/pci@3/SUNW,qlc@4" 0 "qlc"
"/pci@1f,4000/pci@3/SUNW,qlc@4/fp@0,0" 0 "fp"
"/pci@1f,4000/pci@3/SUNW,qlc@4/fp@0,0/ses@w5080020000048bab,0" 32 "ses"
"/pci@1f,4000/pci@3/SUNW,qlc@4/fp@0,0/ses@w5080020000048bac,0" 33 "ses"
....
"/pci@1f,4000/pci@3/SUNW,qlc@5" 1 "qlc"
"/pci@1f,4000/pci@3/SUNW,qlc@5/fp@0,0" 1 "fp"
"/pci@1f,4000/pci@3/SUNW,qlc@5/fp@0,0/ses@w5080020000048bab,0" 34 "ses"
"/pci@1f,4000/pci@3/SUNW,qlc@5/fp@0,0/ses@w5080020000048bac,0" 35 "ses"
...
"/pci@4,4000/scsi@6,1" 1 "glm"
"/pci@4,4000/scsi@6,1/sd@0,0" 15 "sd"
"/pci@4,4000/scsi@6,1/sd@1,0" 16 "sd"

```

## devFS, Management

Bei Laufender Solaris-Domain:

- ▶ `devfsadm`
- ▶ `devfsadm -C`

Bei einem Boot/Reboot:

- ▶ Aus OBP:

```
{128} ok boot -r
```

- ▶ Aus laufendem OS durch Reboot:

```
nx1# reboot -- -r
```

- ▶ Alternativ aus laufendem OS durch Reboot:

```
nx1# touch /reconfigure
```

```
nx1# reboot
```

## LAB: Devicefilesystem

Wieviele serielle Schnittstellen hat die aktuelle Maschine?

Auf welcher Adresse ist die Graphikkarte zu finden?

Wie lautet die Instanz der Bootplatte?

Wieviele Prozessoren hat das aktuelle System?

Wieviele voninander unabhanegige Busse hat das System?

Was für ein Controller bedient die die Bootplatte?

Wird eine USB- oder eine PS2maus auf dem Zielsystem verwendet?



# Benutzerverwaltung

## Lokale Benutzerverwaltung

# Benutzerverwaltung

- ▶ `/etc/passwd`  
User-ID, Name, Heimatverzeichnis, Startprogramm
- ▶ `/etc/shadow`  
Passwortfile
- ▶ `/etc/group`  
Gruppenzugehörigkeit
- ▶ `Homedirectory`  
Verzeichnis, in dem der User Arbeiten kann, auf per NFS

# Benutzerverwaltung, Einrichtung

- ▶ Per Kommando: useradd/userdel/usermod
- ▶ Manuell:
  1. Eintrag in /etc/passwd erstellen
  2. Eintrag in /etc/group erstellen
  3. /etc/shadow-Eintrag mit dem Kommando pwconv(1M) erstellen
  4. Homedirectory erstellen und per chmod übereignen
- ▶ Testen!!

## Benutzerverwaltung, /etc/passwd

```
guest:x:200:200:Gast Benutzer:/home/guest:/bin/sh
  1   2   3   4           5           6           7
```

- 1 Benutzername (username)
- 2 Passwort (x: Referenz auf */etc/shadow*)
- 3 Benutzernummer (UserID, uid)
- 4 Gruppennummer (GroupID, gid)
- 5 Gecos-Field,<sup>1</sup> Beschreibung von Name, Telefon, Raum, ...
- 6 Pfad zum Homedirectory
- 7 Erstes auszuführendes Programm nach erfolgreichem Login

---

<sup>1</sup>General Electric Comprehensive Operating System

## Benutzerverwaltung, /etc/shadow

```
user:password:lastchg: min:max:warn: inactive:expire:flag
1         2         3         4     5     6         7         8         9
```

- 1 Benutzername (aus der *passwd*, Kolumne 1)
- 2 Verschlüsseltes Passwort oder Status
- 3 Letzte Änderung in der Epoche
- 4 Minimaler Zeitraum in Tagen zwischen einer Passwortänderung
- 5 Maximale Anzahl in Tagen für die Gültigkeit eines Passworts
- 6 Warnperiode: Angabe in Tagen vor Ungültigkeit des Passworts
- 7 Anzahl der Tage, die ein Useraccount inaktiv sein kann
- 8 Tag, ab dem ein Benutzerzugang gesperrt wird
- 9 Nicht benutzt

## Benutzerverwaltung, /etc/shadow

Status	Feld 2	Bedeutung	Kommando
LK	*LK*	gesperrt	<code>passwd -l &lt;username&gt;</code>
NL	NP	Für interaktiven login gesperrt, <i>cron/at/batch</i> jobs freigegeben	<code>passwd -n &lt;username&gt;</code>
NP	leer	no password	<code>passwd -d &lt;username&gt;</code>
UN	ungültig	no login	editor
PS	<Passwort>	Gültiges ver- schlüsseltes Passwort	<code>passwd &lt;username&gt;</code>
PS	Feld 3 0	Benutzer muß das Passwort beim ersten login ändern.	<code>passwd -f &lt;username&gt;</code>

## Benutzerverwaltung, /etc/group

```
daemon::12:root,daemon
```

```
1 2 3 4
```

- 1 Gruppenname
- 2 Gruppenpasswort
- 3 Gruppennummer (GroupID, gid).
- 4 Benutzer, die dieser Gruppe zugeordnet sind (kommaseparierte Liste)

# Network Filesystem, NFS

## Network Filesystem, NFS <sup>2</sup>

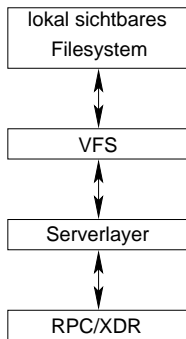
---

<sup>2</sup>Nitemare Filesystem?

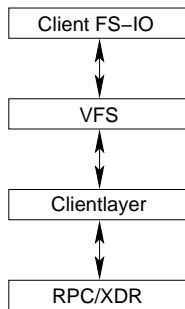


# NFS

Server



Client

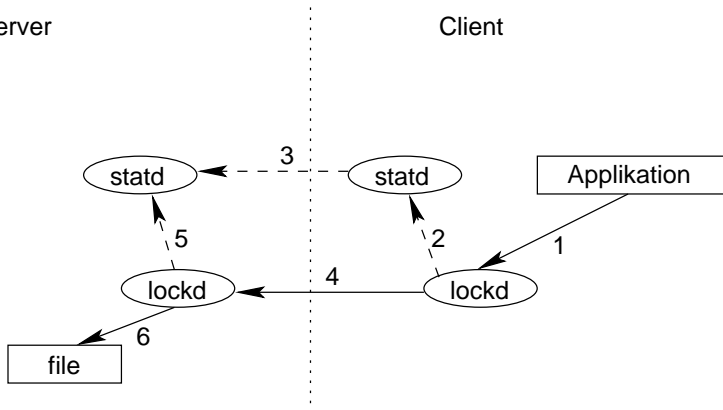


Netzwerk

## NFS (3)

Server

Client



## NFS (Server)

```
root@sirius # svcs -a |grep nfs
disabled      2:31:25 svc:/network/nfs/client:default
disabled      2:31:27 svc:/network/nfs/server:default
disabled      2:32:42 svc:/network/nfs/rquota:default
online        6:37:47 svc:/network/nfs/cbd:default
online        6:37:48 svc:/network/nfs/mapid:default
online        6:37:48 svc:/network/nfs/status:default
online        6:37:48 svc:/network/nfs/nlockmgr:default
```

## NFS (Server)

/etc/dfs/dfstab:

```
root@sirius # more /etc/dfs/dfstab
share -F nfs -o anon=0,sec=sys,ro -d "Installserver" /export/
share -F nfs -d "Homedirectories" /export/home/fub
root@sirius # share
- /export/install anon=0,sec=sys,ro "Installserver"
- /export/home/fub rw "Homedirectories"
```

## NFS (Client)

Was exportiert der Server?

```
root@sirius 1# dfshares mirage
```

RESOURCE	SERVER	ACCESS	TRANSPORT
mirage:/export/isrv	mirage	-	-
mirage:/export/home/fub	mirage	-	-

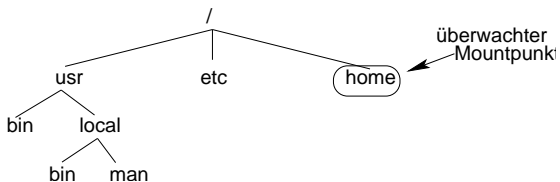
Mount: mount <host>:<path> <targetpath>

Unmount: umount <targetpath>

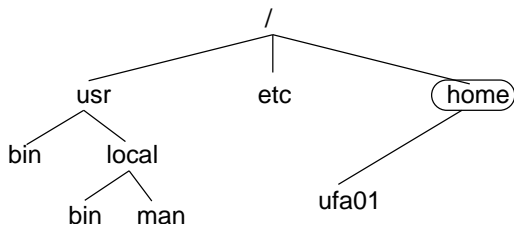
autoFS

# Automounter

# autoFS

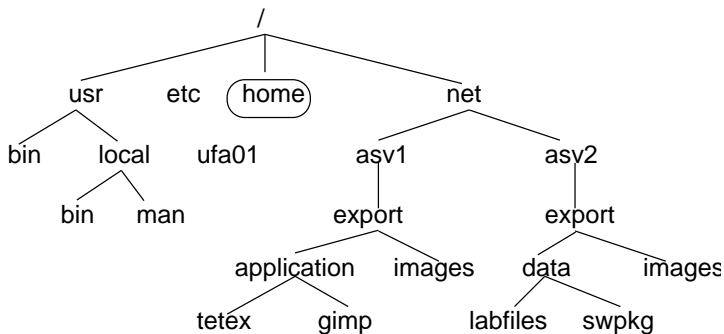


# autoFS





# autoFS



# autoFS

## smit automount

### Start Automounter

Type or select values in entry fields.  
Press Enter AFTER making all desired changes.

[Entry Fields]

PARAMETERS to be used for the automount daemon []

F1=Help	F2=Refresh	F3=Cancel	F4=List
F5=Reset	F6=Command	F7=Edit	F8=Image
F9=Shell	F10=Exit	Enter=Do	

## autoFS, Setup AIX 6.1++

- ▶ Kopieren der Files unter `/usr/samples/nfs/auto*` nach `/etc`
- ▶ Anpassen der Automountermaps
- ▶ Mountpunkt `/home` freischalten (mount, `/etc/filesystems`)
- ▶ AutoFS starten: `/usr/sbin/automount`  
alternativ `startsrc -s automountd`
- ▶ Check:

```
root@hlp10:/$ lssrc -s automountd
```

Subsystem	Group	PID	Status
automountd	autofs	159956	active

- ▶ AutoFS stoppen: `stopsrc -s automountd`

## autoFS, /etc/auto\_master

Allgemeines Format einer Automountermap:

```
key    -Mountoptions    nfs-Server:Serverpath  Options
```

Master-Map:

```
#
# Master map for automounter
#
+auto_master
/net          -hosts          -nosuid,nobrowse
/home        auto_home       -nobrowse
```

## autoFS, /etc/auto\_home

```
#
# Home directory map for automounter
#
+auto_home
svw01  mirage:/export2/sysb/home/svw01
svw02  mirage:/export2/sysb/home/svw02
svw03  mirage:/export2/sysb/home/svw03
svw04  mirage:/export2/sysb/home/svw04
svw05  mirage:/export2/sysb/home/svw05
```

Vereinfachung:

```
*      mirage:/export2/sysb/home/&
```

zfs

# Zettabyte Filesystem

# zfs

- ▶ 128Bit-Filesystem
- ▶ RAID-Software und Filesystem in Einem
- ▶ Copy-on-Write-FS
- ▶ Transaktionsloggesichert (2PC)
- ▶ Optimiert auch auf sequentiellen I/O
- ▶ Windows-ACLs, zusammen mit NFSv4 exportierbar
- ▶ Datenabsicherung durch Prüfsummen
- ▶ Selfrecovery bei Blockfehlern
- ▶ Einfache Administration (jedoch wenig Eingriff)

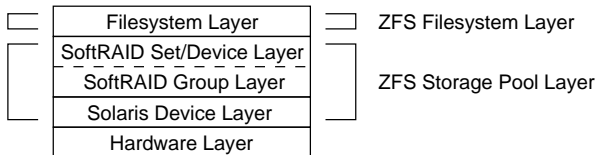
# zfs

## Attribute per Filesystem setzbar:

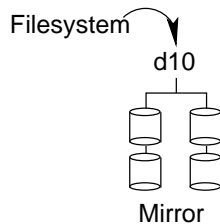
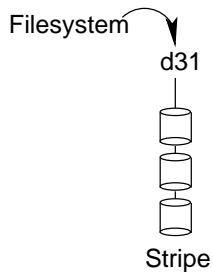
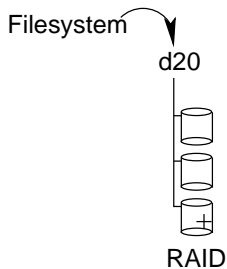
- ▶ Quotas, Reservations
- ▶ Blocksize/Recordsize
- ▶ NFS-Options
- ▶ iSCSI-Targetmode
- ▶ Checksumming
- ▶ exec + suid Rechte beschränkbar
- ▶ Compression
- ▶ Geplant: Cryptomechanismus
- ▶ ....



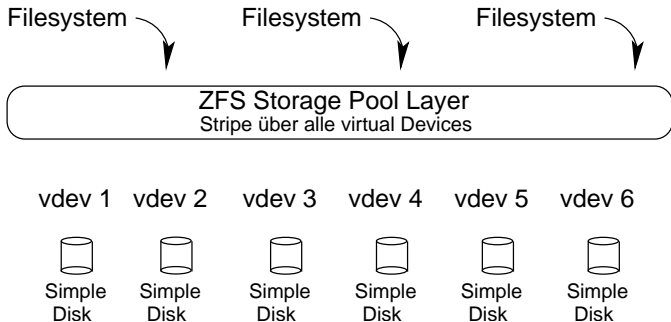
## zfs



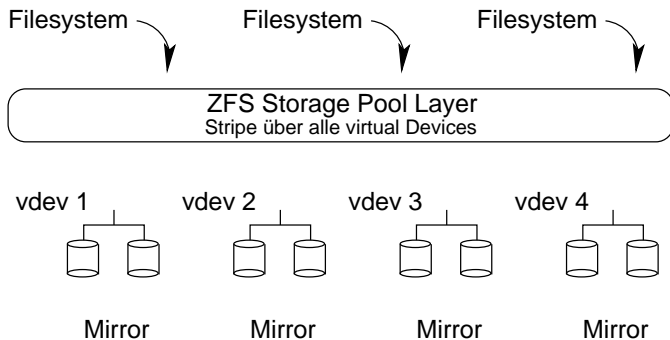
## zfs



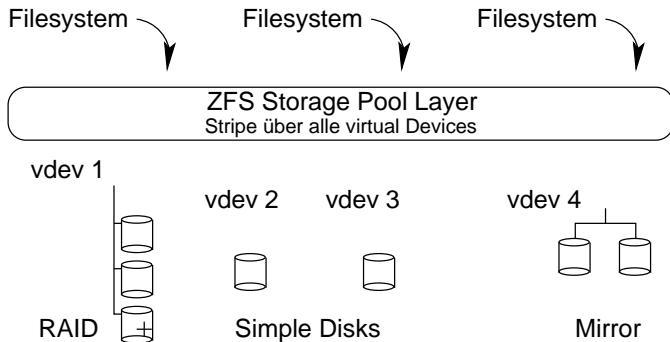
## zfs



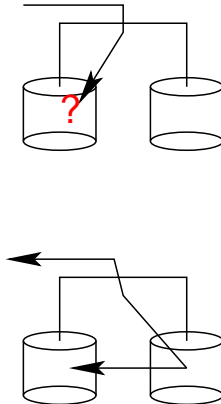
## zfs



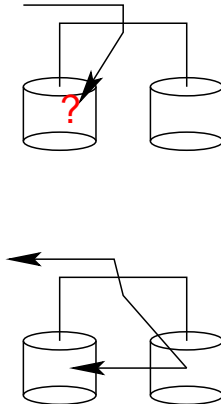
## zfs



## zfs



## zfs



# zfs

Unterstützte RAID-Level:

**Simple Disk** Grundtyp, einzelnes vdev

**Simple Container** Grundtyp, Containerfile (mkfile ...)

**Mirror** Konstruierter Typ, Spiegel mehrerer Grundtypen

**RAID-Z** Konstruierter Typ, RAID-5 naher RAID-Level

**RAID-Z2** Konstruierter Typ, Doubleparity-RAID-5 (RAID-6)

Wird ein zpool erweitert, so ist darauf zu achten die Redundanzqualität zu erhalten.



## zfs

## zpool Subkommandos:

```

create [-fn] [-R root] [-m mountpoint] <pool> <vdev> ...
destroy [-f] <pool>
add [-fn] <pool> <vdev> ...
remove <pool> <device>
list [-H] [-o field[,field]*] [pool] ...
iostat [-v] [pool] ... [interval [count]]
status [-vx] [pool] ...
online <pool> <device> ...
offline [-t] <pool> <device> ...
clear <pool> [device]
attach [-f] <pool> <device> <new_device>
detach <pool> <device>
replace [-f] <pool> <device> [new_device]
scrub [-s] <pool> ...
import [-d dir] [-D]
import [-d dir] [-D] [-f] [-o opts] [-R root] -a
import [-d dir] [-D] [-f] [-o opts] [-R root ] <pool | id> [newpool]
export [-f] <pool> ...
upgrade
upgrade -v
upgrade <-a | pool>

```

## zfs Hands on

```
endeavour# zpool list
NAME                SIZE  USED  AVAIL  CAP  HEALTH  ALTROOT
pm1                  16.8G 45.5K 16.7G  0%  FAULTED  -
pm2                  16.8G 48.5K 16.7G  0%  ONLINE   -
```

```
endeavour# zpool iostat
          capacity      operations      bandwidth
pool      used  avail  read  write  read  write
-----  -
pm1       45.5K 16.7G    0    0    638   432
pm2       48.5K 16.7G    0    0    712   424
-----  -
```

## zfs Hands on

```
endeavour# zpool iostat -v
```

pool	capacity		operations		bandwidth	
	used	avail	read	write	read	write
pm1	45.5K	16.7G	0	0	637	432
c6t2d0	0	8.38G	0	0	253	183
c7t2d0	45.5K	8.37G	0	0	383	248
pm2	48.5K	16.7G	0	0	711	423
c6t1d0	0	8.38G	0	0	261	191
c7t1d0	48.5K	8.37G	0	0	449	232

# zfs Hands on

## Erzeugung eines Stripess über 4 Festplatten

```
endeavour# zpool create p1 c6t3d0 c6t4d0 c6t5d0 c6t6d0
endeavour# zpool iostat -v
```

pool	capacity		operations		bandwidth	
	used	avail	read	write	read	write
p1	33.0K	33.5G	0	0	0	2.29K
c6t3d0	0	8.38G	0	0	0	576
c6t4d0	0	8.38G	0	0	0	578
c6t5d0	0	8.38G	0	0	0	574
c6t6d0	33.0k	8.37G	0	0	0	611

# zfs Hands on

## Aufsetzen eines Zweifachspiegels

```
endeavour# zpool create -f p3 mirror c6t7d0 c7t7d0
endeavour# zpool iostat -v
```

pool	capacity		operations		bandwidth	
	used	avail	read	write	read	write
p3	33.0K	8.37G	0	0	0	35
mirror	33.0K	8.37G	0	0	0	35
c6t7d0	-	-	0	0	50	687
c7t7d0	-	-	0	0	50	687

## zfs Hands on

```
endeavour# zpool create -f p4 mirror c6t8d0 c7t8d0 mirror c6t9d0 c7t9d0
```

```
endeavour# zpool iostat -v
```

pool	capacity		operations		bandwidth	
	used	avail	read	write	read	write
p3	33.0K	16.7G	0	0	0	35
mirror	33.0K	8.37G	0	0	0	35
c6t8d0	-	-	0	0	10	687
c7t8d0	-	-	0	0	12	687
mirror	33.0K	8.37G	0	0	0	35
c6t9d0	-	-	0	0	50	737
c7t9d0	-	-	0	0	50	737

## zfs Hands on

```
df -k
Filesystem          kbytes    used  avail capacity  Mounted on
...
pm3                  17426432    8 17426386    1%    /pm3
pm3/demo             17426432    8 17426386    1%    /pm3/demo
...
endeavour# zpool export pm3
endeavour# zpool import pm3 new3
endeavour# df -k
Filesystem          kbytes    used  avail capacity  Mounted on
...
new3                 17426432    8 17426386    1%    /new3
new3/demo            17426432    8 17426386    1%    /new3/demo
...
```

## zfs Hands on

```
endeavour# zpool status -x
pool: new2
state: FAULTED
status: One or more devices could not be opened.  There are
        insufficient replicas for the pool to continue functioning.
action: Attach the missing device and online it using 'zpool online'.
        see: http://www.sun.com/msg/ZFS-8000-3C
scrub: none requested
config:
```

NAME	STATE	READ	WRITE	CKSUM	
new2	FAULTED	0	4	0	
c6t2000002037260FE2d0s0	FAULTED	0	4	0	cannot open
c6t200000203708CE21d0s0	ONLINE	0	0	0	



## zfs Hands on

```
endeavour# zpool replace new2 c6t2000002037260FE2d0s0
endeavour# zpool status -v
zpool status -v
  pool: new2
  state: ONLINE
  scrub: resilver completed with 0 errors on Sat Dec 31 20:13:38 2005
config:
  NAME                                STATE      READ WRITE CKSUM
  new2                                  ONLINE    0     0     0
    c6t2000002037228331d0s0           ONLINE    0     0     0
    c6t200000203708CE21d0s0           ONLINE    0     0     0
```

## zfs Hands on

### zfs Subkommandos

```

create <filesystem>
create [-s] [-b blocksize] -V <size> <volume>
destroy [-rRf] <filesystem|volume|snapshot>
snapshot [-r] <filesystem@name|volume@name>
rollback [-rRf] <snapshot>
clone <snapshot> <filesystem|volume>
promote <clone filesystem>
rename <filesystem|volume|snapshot> <filesystem|volume|snapshot>
list [-rH] [-o property[,property]...] [-t type[,type]...]
    [-s property [-s property]...] [-S property [-S property]...]
    [filesystem|volume|snapshot] ...
set <property=value> <filesystem|volume> ...
get [-rHp] [-o field[,field]...] [-s source[,source]...]
    <all | property[,property]...> <filesystem|volume|snapshot> ...
inherit [-r] <property> <filesystem|volume> ...
mount
mount [-o opts] [-O] -a
mount [-o opts] [-O] <filesystem>
umount [-f] -a
umount [-f] <filesystem|mountpoint>
share -a
share <filesystem>
unshare [-f] -a
unshare [-f] <filesystem|mountpoint>
send [-i <snapshot>] <snapshot>
receive [-vn] <filesystem|volume|snapshot>
receive [-vn] -d <filesystem>

```

# zfs Hands on

```
endeavour# zfs create pm/demo
endeavour# df -k
...
pm                34852864      8 34852820    1%  /pm
pm/demo          34852864      8 34852820    1%  /pm/demo
...
```

## zfs Hands on

```
endeavour# zpool create export mirror c2t1d0 c3t1d0
endeavour# zfs create export/home
endeavour# zfs create export/home/ufa00
endeavour# zfs create export/home/ufa01
endeavour# zfs create export/home/ufa02
endeavour# zfs create export/applications
endeavour# zfs list
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
export	92.0K	16.6G	9.5K	/export
export/applications	8K	16.6G	8K	/export/applications
export/home	34.0K	16.6G	10.0K	/export/home
export/home/ufa00	8K	16.6G	8K	/export/home/ufa00
export/home/ufa01	8K	16.6G	8K	/export/home/ufa01
export/home/ufa02	8K	16.6G	8K	/export/home/ufa02
pm	5.47M	33.2G	8.50K	/pm
pm/demo	5.42M	33.2G	5.41M	/pm/demo
pm/demo/subdemo	8K	33.2G	8K	/pm/demo/subdemo

## zfs Hands on

zfs-Filesystem löschen: *zfs destroy <yourzfsfs>*

Rekursiv löschen: *zfs destroy -r <yourzfsfs>*

## zfs Hands on

### Erzeugen und Auflisten von Snapshots:

```
endeavour# zfs snapshot export/home/ufa00@globus
endeavour# ls -l /export/home/ufa00/.zfs/snapshot
total 2
drwxr-xr-x  3 root    sys           3 Jan  1 15:22 globus/
endeavour# zfs list -t snapshot
NAME                                USED  AVAIL  REFER  MOUNTPOINT
export/applications@newyearsday    0     -      2.19G  -
export/home/ufa00@1.1.06           23.0K -      41.1M  -
export/home/ufa00@sunday            15.5K -      41.1M  -
pm/demo@grits_tests                 0     -      5.41M  /mnt
```

# zfs Hands on

## Snapshot Rollback:

```
endeavour# zfs rollback export/home/ufa00@1.1.06
endeavour# zfs list -t snapshot
NAME                                USED  AVAIL  REFER  MOUNTPOINT
export/applications@newyearsday    0     -      2.19G  -
export/home/ufa00@sunday            15.5K -      41.1M  -
pm/demo@grits_tests                0     -      5.41M  /mnt
```

## zfs Hands on

```

endeavour> zfs get all pm
NAME                PROPERTY          VALUE                SOURCE
pm                  type              filesystem            -
pm                  creation          Sat Dec 31 21:05 2005 -
pm                  used              5.47M                -
pm                  available         33.2G                -
pm                  referenced        8.50K                -
pm                  compressratio     1.00x                -
pm                  mounted           yes                   -
pm                  quota             10G                  local
pm                  reservation       none                  default
pm                  recordsize        128K                  default
pm                  mountpoint        /pm                   default
pm                  sharenfs          off                   default
pm                  checksum          on                    default
pm                  compression       on                    local
pm                  atime             on                    default
pm                  devices           on                    default
pm                  exec              on                    default
pm                  setuid            on                    default
pm                  readonly          off                   default
pm                  zoned             off                   default
pm                  snapdir           visible               default
pm                  aclmode           groupmask             default
pm                  aclinherit        secure                default

```



# zfs Hands on

```
endeavour# zfs set reservation=10G export/home/ufa01
endeavour# df -h
Filesystem      size  used  avail capacity  Mounted on
...
export/home/ufa01  17G   8K   14G    1%   /export/home/ufa01
...
endeavour# zfs set quota=15G export/home/ufa01
```

## zfs Hands on

```

endeavour# zfs get all export/home/ufa01
NAME                PROPERTY           VALUE                SOURCE
export/home/ufa01  type              filesystem          -
export/home/ufa01  creation          Sun Jan  1 14:44 2006 -
export/home/ufa01  used             8K                 -
export/home/ufa01  available        14.4G              -
export/home/ufa01  referenced       8K                 -
export/home/ufa01  compressratio    1.00x              -
export/home/ufa01  mounted         yes                 -
export/home/ufa01  quota           15.0G              local
export/home/ufa01  reservation     10.0G              local
export/home/ufa01  recordsize      128K               default
export/home/ufa01  mountpoint      /export/home/ufa01 default
export/home/ufa01  sharenfs       off                 default
export/home/ufa01  checksum       on                  default
export/home/ufa01  compression    off                 default
export/home/ufa01  atime          on                  default
export/home/ufa01  devices        on                  default
export/home/ufa01  exec           on                  default
export/home/ufa01  setuid         on                  default
export/home/ufa01  readonly       off                 default
export/home/ufa01  zoned          off                 default
export/home/ufa01  snapdir        visible             default
export/home/ufa01  aclmode        groupmask           default
export/home/ufa01  aclinherit     secure              default

```

## zfs Hands on

Das Spiel beenden, ohne Warnung:

```
endeavour# zpool destroy pm
endeavour# zpool iostat -v
no pools available
```

# Systemboot x86/x64-Solaris

## x86/x64 SOLARIS System Boot

## Systemboot x86/x64-Solaris

- ▶ x86 Systemboot ist komplexer als Sparc Systemboot
- ▶ Seit Build 14 GRUB-basierter Systemboot  
keine eigenen Gerätetreiber  
nur INT-13 Calls
- ▶ Festplattenboot unterscheidet sich von CD-Boot  
CD-Boot: El-Torito Standard aus Image im NO-Emulation  
Modus Festplattenboot: Laden eines komprimierten FS-Image  
für /
- ▶ keine FDISK-Unterstützung durch das BIOS  
FDISK-Support im 1ten Sektor der Bootplatte
- ▶ MS-ähnlicher Boot
- ▶ Erweiterung der Bootfunktionalität durch Laden einer  
Imagedatei

# Systemboot x86/x64-Solaris

## 1 GRUB Load, Phase 1

- (a) Laden des Ersten Sektors der Bootplatte  
FDISK-Informationen  
Übergabe der Kontrolle an den geladenen Code
- (b) Laden des Codes des 1ten Sektors der aktiven Partition
- (c) Laden der zweiten GRUB-Boot Stufe

# Systemboot x86/x64-Solaris

## 2 GRUB Load, Phase 2

Enthält Filesystemtreiber und kann fortan Festplatten lesen

- (a) Boot-Menue wird aufgelistet, Timeout
- (b) Laden und Ausführen eines GRUB-Multiboot kompatiblen Programms

Notwendig zur Trennung des GPL-lizenzierten GRUB-Systems vom CDDL-lizenzierten Solaris Systemkern.

- (c) Multibootprogramm lädt gzipped RAMDiskimage aus  
`/platform/<uname -i>/boot_archive`  
`/boot/solaris/bootenv.rc` entsprch Sparc-NVRAM
- (d) Erstellen der RAMDisk
- (e) Multiboot lädt aus der RAMDisk:  
`/platform/<uname -i>/kernel/unix` bzw. `/kernel/amd64/unix`

## Systemboot x86/x64-Solaris

- ▶
  - (f) Kernel lädt Runtime-Linker nach
  - (g) Nachladen von Treibern aus Filesystem der RAMDisk  
(krltd entlädt fs-Treiber bei x86/x64)
  - (h) Übermounten des Root-fs entsprechend *bootpath* Variable  
(Wenn nicht gesetzt: RAMDisk-fs)
- ▶ Es folgt der Systemladevorgang analog zu Sparc-Solaris



# Systemhalt x86/x64-Solaris

**RAMDiskimage** wird erstellt durch das Kommando *bootadm*

- ▶ Legt *bootenv.rc* etc. fest
- ▶ Erzeugt mit *mkisofs* /-fs Image als hfsfs-Filesystem
- ▶ Komprimiert das Imagefile

**Modifikation** Wenn das System modifiziert wurde ist das Image neu zu erzeugen

- ▶ Ändern der Systemkonfiguration  
USB-Stick etc.

# Systemhalt x86/x64-Solaris

**Imageerstellung** Beim Systemstop wird überprüft ob die Systemkonfiguration vom alten RAMDiskimage variiert.

- ▶ Sicherung der *bootenv.rc* Parameter
- ▶ Erstellen eines RAMDiskimage per *mkisofs*
- ▶ Komprimierung des Images

Wird der Vorgang unterbrochen, kann u.U. der Neustart nicht mehr erfolgen!

## LAB: Systemboot

Führen Sie einen Systemboot von Netz aus. Was ist zu tun?

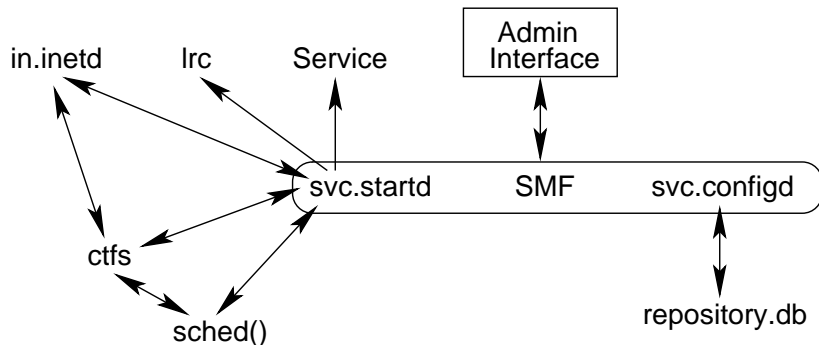
Führen Sie einen klassischen Systemboot (*boot -sw*) in den Single-user Mode aus.

In Welchem Zustand ist Ihre LAB-Maschine? Multiuser/Singleuser?

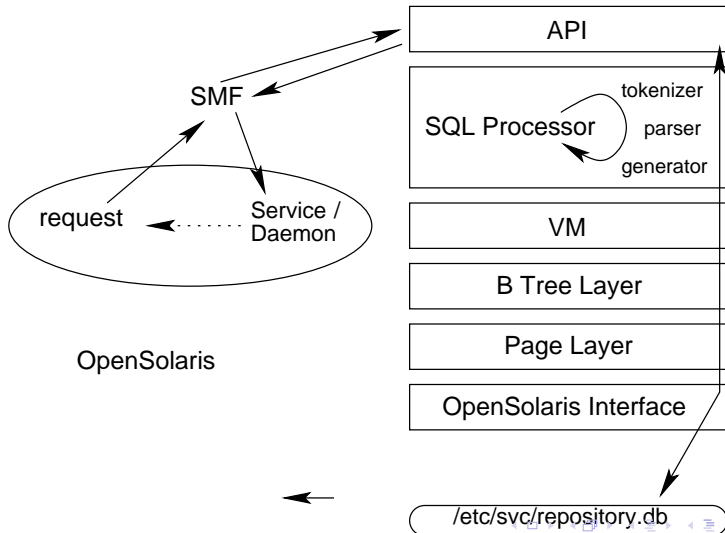
# Service Management Facility

## Service Management Facility

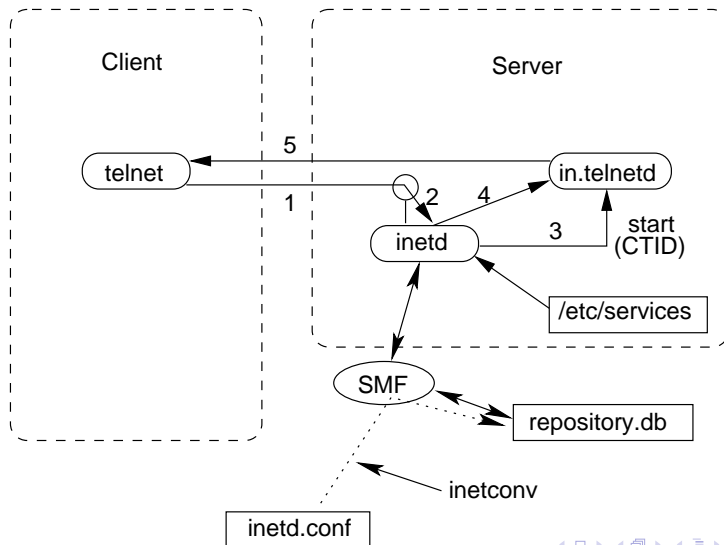
# Service Management Facility



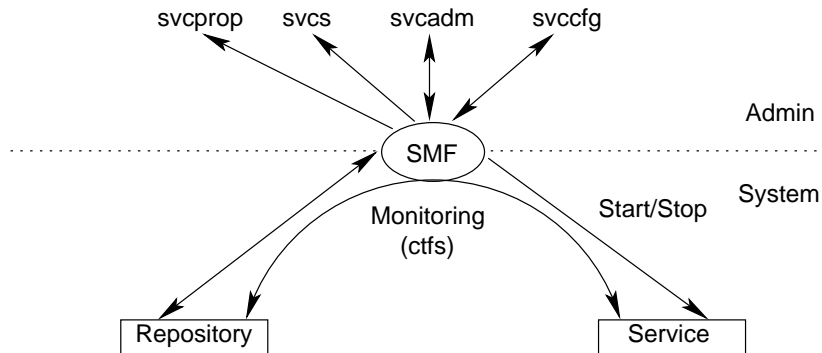
# SMF/Repository



# SMF/inetd



# Service Management Facility





# SMF, Files & Directories

`/lib/svc/bin` SMF Serviceprogramme

`lsvcrun` Legacy Run

`mfstscan` Manifestparser

`prophist` Propertyhistory Managementprogramm

`svc.startd` SMF Holderprozess

`svc.configd` Configdaemon

`restore_repository` Maintenanceprogramm zur  
Reparatur des Repositories

`sqlite` DB-Managementprogramm für das  
Repository

`/lib/svc/method` Start/Stop-Methoden der Services

`/lib/svc/seed` Seed für Neuaufbau eines Repositories

## SMF, Files & Directories

Manifeste:

Applikationen in */var/svc/manifest/application*,

Devices in */var/svc/manifest/device*,

Milestones in */var/svc/manifest/milestone*,

Netzwerk in */var/svc/manifest/network*,

Plattformspezifisch in */var/svc/manifest/platform*,

Sitelokal in */var/svc/manifest/site* und

System in */var/svc/manifest/system*.

## SMF, Files & Directories

Profiles in `/var/svc/profile`:

`generic_open.xml` Profile für *net service open*

`generic_limited_net.xml` Profile für *net service limited*

`platform_*` Platformsspezifische Profiles

`ns_*` Nameservice Profiles

**Defaulteinstellung** Per Softlink auf das  
entsprechende Profile

Logfiles in `/var/svc/log`:

`*.log` für jeden Service, hier stehen die  
Logfiles der Methoden!!

## SMF, Files & Directories

Repository in `/etc/svc`

`repository.db` Aktives und aktuelles Repository von SMF

`repository-boot-<Datum-Stamp>`

Bootrepositoryarchiv

`repository-boot` Softlink auf letztes Bootrepository

`repository-manifest_import-*` Repository aus Manifestimport

`repository-manifest` Link auf letztes Repository aus Manifestimport

`volatile` Unterverzeichnis für Logfiles der Services

# SMF Administration

## SMF Administration

## SMF Administration

Administration einzelner Services: *svcadm(1M)*

**Start** Persistenter Start eines Services:  
*svcadm enable <FMRI>*

**Stop** Persistenter Stop eines Services:  
*svcadm disable <FMRI>*

**Reread** Neu Einlesen der Parameterfiles eines Services:  
*svcadm refresh <FMRI>*

**Restart** Stop-Start eines Services:  
*svcadm restart <FMRI>*

**Clear** Zurücksetzen des Fehlerstatus eines Services:  
*svcadm clear <FMRI>*

**Mark** Markieren für Status von Folgediensten eines Services:  
*svcadm mark <Status><FMRI>*

# SMF Service States

## SMF Service States

# SMF Service States

Administrationsinterface:

*svcs(1M)* Statusanzeige der Services unter SMF-Kontrolle

*svcadm(1M)* Administration der Services

*svccprop(1M)* Anzeige von Properties von Services

*svccfg(1M)* Konfiguration der Services



# SMF Service States, Anzeige

`svcs` Anzeige aller Services mit Status **enabled**

`svcs -a` Anzeige aller Services

```
nova# svcs
STATE      STIME      FMRI
...
legacy_run 12:59:43 lrc:/etc/rc2_d/S31utsyscfg
legacy_run 12:59:43 lrc:/etc/rc2_d/S401lc2
legacy_run 12:59:43 lrc:/etc/rc2_d/S42ncakmod
legacy_run 12:59:44 lrc:/etc/rc2_d/S51utacleanup
legacy_run 12:59:44 lrc:/etc/rc2_d/S72autoinstall
...
online     12:59:19 svc:/system/svc/restarter:default
online     12:59:19 svc:/network/pfil:default
online     12:59:20 svc:/network/loopback:default
online     12:59:20 svc:/system/installupdates:default
online     12:59:21 svc:/milestone/name-services:default
...
```

Gültige Zustände s.g. Fault Managed Resource IDs (FMRI):

- legacy\_run** Service über rc-Scripten gestartet
- online** Service aktiviert, laufend
- disabled** Service deaktiviert
- offline** Service aktiviert, läuft nicht
- degraded** Service aktiviert, läuft nicht vollständig
- maintenance** Fehler bei Service-Start/Betrieb, manuell auflösen
- uninitialized** Konfiguration wurde noch nicht gelesen

# SMF Service States, Anzeige

**svcs -p <...>** Anzeige der zu einem Service gehörenden Prozesse

```
nova# svcs -p cron
STATE      STIME      FMRI
online     12:59:32   svc:/system/cron:default
           12:59:32   224 cron
```

**svcs -d <...>** Anzeige der Services, von dem ein Service abhängt

```
nova# svcs -d cron
STATE      STIME      FMRI
online     12:59:21   svc:/milestone/name-services:default
online     12:59:32   svc:/system/filesystem/local:default
```

**svcs -D <...>** Anzeige der abhängigen Services

```
nova# svcs -D cron
STATE      STIME      FMRI
online     12:59:46   svc:/milestone/multi-user:default
```

## SMF Service States, Anzeige

`svcs -l <... >` Detaillierte Anzeige der Instanzen, Abhängigkeiten, Zustände etc. . .

```
nova# svcs -l cron
fmri          svc:/system/cron:default
name         clock daemon (cron)
enabled      true
state        online
next_state   none
state_time   Thu Mar 16 12:59:32 2006
logfile      /var/svc/log/system-cron:default.log
restarter    svc:/system/svc/restarter:default
contract_id  34
dependency   require_all/none svc:/system/filesystem/local (online)
dependency   require_all/none svc:/milestone/name-services (online)
```

## SMF Service States, Anzeige

`svcs -vx <... >` Erklärt Servicezustände, `-v` Flag listet abhängige Service mit auf

```
nova# svcs -xv cron
svc:/system/cron:default (clock daemon (cron))
  State: online since Thu May 17 12:59:32 2007
    See: man -M /usr/share/man -s 1M cron
    See: man -M /usr/share/man -s 1 crontab
    See: /var/svc/log/system-cron:default.log
  Impact: None.
```

## SMF Service States, Anzeige

`svcs -vx` Ohne Angabe eines Services listet alle Services auf, die zwar enabled sind, jedoch nicht laufen

```
nova# svcs -xv
svc:/application/print/server:default (LP print server)
  State: disabled since Thu May 17 12:59:20 2007
  Reason: Disabled by an administrator.
    See: http://sun.com/msg/SMF-8000-05
    See: man -M /usr/share/man -s 1M lpsched
    See: /var/svc/log/application-print.log
  Impact: 2 dependent services are not running:
    svc:/application/print/rfc1179:default
    svc:/application/print/ipp-listener:default
```

## SMF Milestones/System State

# Service Management Facility - Milestones -

## SMF Milestones/System State

**Runlevel** Historische Beschreibung des System Status, singleuser/multiuser etc. Der Runlevel bei Solaris 10 ist nahezu immer 3

**Milestone** Milestones beschreiben eine Aggregation gestarteter Services, sie stellen selbst keinen Service dar, Service können Abhängigkeiten zu Milestones definieren.

<i>Milestone</i>	<i>Runlevel</i>
-	0
<i>milestone/single-user</i>	5
<i>milestone/multi-user</i>	2
<i>milestone/multi-user-server</i>	3



## Manipulation der Milestones

**Wechsel** Ist zwischen Milestones zu wechseln:

```
nova# svcadm milestone milestone/single-user
```

**boot** Soll in einen anderen als den Defaultmilestone gebootet werden so ist dies beim boot mitzuteilen

```
boot -m milestone=single-user
```

**verbose boot** Ein Boot, bei dem auf der Systemconsole angezeigt wird, was wann gestartet wird:

```
boot -m verbose
```

## SMF Milestones/System State

```
nova# svcs |grep milestone
online          12:59:21 svc:/milestone/name-services:default
online          12:59:24 svc:/milestone/network:default
online          12:59:27 svc:/milestone/devices:default
online          12:59:30 svc:/milestone/single-user:default
online          12:59:33 svc:/milestone/sysconfig:default
online          12:59:46 svc:/milestone/multi-user:default
online          12:59:59 svc:/milestone/multi-user-server:de
```

## svc:/milestone/name-services

Um den Milestone *name-services* zu erreichen, müssen folgende Service gestartet sein:

**dns** `svc:/network/dns/client`

**ldap** `svc:/network/ldap/client`

**nis\_client** `svc:/network/nis/client`

**nisplus** `svc:/network/rpc/nisplus`

## svc:/milestone/network

Der Milestone *network* ist erreicht, wenn folgende Services gestartet wurden:

- loopback** *svc:/network/loopback*  
Konfiguration des Loopbackinterfaces
- physical** *svc:/network/physical* Konfiguration aller  
Netzwerkinterfaces

## svc:/milestone/devices und sysconfig

Der Milestone *devices* ist erreicht, wenn alle Gerätekonfigurationen abgearbeitet wurden

Der Milestone *sysconfig* beschreibt die Abarbeitung aller Konfigurationsmethoden für Hostnamen, Zeitzonen, Locales etc.  
sysconfig ist abhängig von *svc:/milestone/single-user*

## svc:/milestone/single-user

*sysidtool* Systemidentifikation

*svc:/system/sysidtool:net*

*svc:/system/sysidtool:system*

*nodename* Konfiguration des Nodenames

*svc:/system/identity:node*

*filesystem-minimal* Root-fs etc.

*svc:/system/filesystem/minimal*

*milestone-devices* Erfolgreicher Deviceimport

*svc:/milestone/devices*

*manifests* Manifestimport

*svc:/system/manifest-import*

*loopback-network* Loopbackdevice

*svc:/network/loopback*

*network* Netzwerkkonfiguration

*svc:/milestone/network*

## svc:/milestone/multi-user

Der *multi-user* Milestone entspricht dem klassischen Runlevel 2 und ist erreicht, wenn folgende Services gestartet wurden:

*milestones* Aggregation der Milestones:

*svc:/milestone/single-user*

*svc:/milestone/sysconfig*

*svc:/milestone/name-services*

*fs* Filesysteme

*svc:/system/filesystem/local*

*kdmconfig* Consolekonfiguration bei x86 Systemen

*svc:/platform/i86pc/kdmconfig*

*rpcbind* rpcbind-Services

*svc:/network/rpc/bind*

*syslog* Syslogservices

*svc:/system/system-log*

*/sbin/rc2* Mit grob geschätztem Timeoutwert

## svc:/milestone/multi-user-server

Der *multi-user-server* Milestone entspricht dem klassischen Runlevel 3 und ist erreicht, wenn folgende Services gestartet wurden:

- multi-user* Alles Services des Multiuserbetriebes  
`svc:/milestone/multi-user`
- `/sbin/rc3` Mit grob geschätztem Timeoutwert



# SMF Konfiguration

## SMF Konfiguration

## SMF Konfiguration

Konfiguration von Serviceproperties: *svccfg(1M)*

Subkommandos, die auf Profileebene wirken

**apply Profile** enable/disable Service entsprechend Serviceprofile

**extract >Profile** aktuelles Serviceprofile ->Datei

Subkommandos, die auf Manifestebene wirken

**import File** Lädt Manifest aus File

**export** Exportiert eine FMRI

**archive** aktuelle Servicebeschreibungen in Datei

**inventory** Listet die in einem Manifestfile deklarierten FMRI auf

```
svc:> inventory /tmp/nfsserver
svc:/network/nfs/server:default
svc:/network/nfs/server
```

**validate** Syntaxcheck

```
svc:> validate /tmp/nfsserver
couldn't parse document
```

# Entity-Kommandos

**select** Auswahl einer Selektion oder einer FMRI

```
svc:> select nfs/server  
svc:/network/nfs/server>
```

**unselect** Verlassen einer Selektion oder einer FMRI

```
svc:/network/nfs/server> unselect  
svc:>
```

**list** Auflisten der Entities einer Selektion

```
svc:> list  
...  
network/smtp  
network/ssh  
system/device/fc-fabric  
system/filesystem/autofs  
...
```

**add** Erzeugen einer neuen Entity

**delete** Löschen einer Entity oder einer FMRI

# Property-Gruppenkommandos

## listpg Auflisten einer Gruppe

```
svc:/network/nfs/server> listpg
network          dependency
nlockmgr         dependency
mapid            dependency
...
share-group      dependency
filesystem-local dependency
general          framework
dependents       framework
application      framework
start            method
refresh          method
stop             method
tm_common_name   template
tm_man_nfsd      template
```

**addpg** Hinzufügen einer Propertygruppe

**delpg** Löschen einer Propertygruppe

## Property Kommandos

**listprop** Auflisten aller Properties in der aktuellen Selektion

```
svc:/network/nfs/server> listprop
network                                dependency
network/entities                      fmri      svc:/milestone/network
network/grouping                       astring  require_any
network/restart_on                     astring  error
network/type                           astring  service
nlockmgr                                dependency
nlockmgr/entities                     fmri      svc:/network/nfs/nlockmgr
nlockmgr/grouping                      astring  require_all
nlockmgr/restart_on                    astring  error
nlockmgr/type                          astring  service
mapid                                   dependency
mapid/entities                         fmri      svc:/network/nfs/mapid
mapid/grouping                         astring  optional_all
mapid/restart_on                       astring  error
mapid/type                             astring  service
rpcbind                                dependency
....
```

**setprop** Setzen einer Property

**delprop** Löschen einer Property

**editprop** Ruft einen Editor (\$EDITOR) auf um die aktuellen Properties bzw. deren Einstellung manuell zu ändern

# Property Value Kommandos

**addpropvalue** Einer Property einen Wert zuweisen

**delpropvalue** Wert einer Property löschen

**setenv** Setzen einer Environmentvariablen innerhalb von *svccfg(1M)*

**unsetenv** Löschen einer Environmentvariablen innerhalb von *svccfg(1M)*

# SMF Manifeste, Beispiel: cron I

## SMF Servicemanifeste

# SMF Manifeste, Beispiel: cron I

```
<?xml version="1.0"?>
<!DOCTYPE service_bundle SYSTEM "/usr/share/lib/xml/dtd/service_bundle.dtd.1">
<!--
  Copyright 2005 Sun Microsystems, Inc. All rights reserved.
  CDDL HEADER START
  ...
-->
<service_bundle type='manifest' name='SUNWcsr:cron'>
<service
  name='system/cron'
  type='service'
  version='1'>
  <single_instance />
  <dependency
    name='usr'
    type='service'
    grouping='require_all'
    restart_on='none'>
    <service_fmri value='svc:/system/filesystem/local' />
  </dependency>
```



## SMF Manifeste, Beispiel: cron II

```
<dependency name='ns'  
  type='service'  
  grouping='require_all'  
  restart_on='none'>  
  <service_fmri value='svc:/milestone/name-services' />  
</dependency>  
<dependent name='cron_multi-user'  
  <propval name='action_authorization' type='astring'  
    value='solaris.smf.manage.cron' />  
</property_group>  
<instance name='default' enabled='false' />  
<stability value='Unstable' />
```

## SMF Manifeste, Beispiel: cron III

```
<template>
  <common_name>
    <loctext xml:lang='C'>
      clock daemon (cron)
    </loctext>
  </common_name>
  <documentation>
    <manpage title='cron' section='1M' manpath='/usr/share/man' />
    <manpage title='crontab' section='1' manpath='/usr/share/man' />
  </documentation>
</template>
</service>

</service_bundle>
```

# LAB: SMF Administration

1. Listen Sie alle Services auf, die aktiviert sind.
2. Laufen alle Services Fehlerfrei?
3. Welche Service sind deaktiviert? (Wie ist dies zu erkennen?)
4. Laufen die NFS-Services?
5. Welchen Prozess startet und monitored der Service `svc:/network/smtp:sendmail?`
6. welchen Service startet und monitored `svc:/system/cron?`
7. Stoppen Sie den `sendmailservice persistent`, wurde dadurch auch der Sendmailprozess gestoppt? rebooten Sie Ihre Maschine. Ist der Sendmailservice nach dem Reboot aktiv?
8. Stoppen Sie die Services `webmin`, `ipp-listener`, `cde-printinfo`, `svc:/network/http:apache2`, `smsserver`, `rfc1179`, `ocfserv`, `rmvolmgr`, `hal`. Rebooten Sie Ihr System und verifizieren Sie das Ergebnis.
9. Deaktivieren Sie die Legacy-Services `S47pppd` und `S70uucp`. Was ist zu tun?
10. Listen Sie alle Services auf, von denen der Milestone `multi-user` abhängt, listen Sie alle Services auf von dem `multi-user` abhängt.
11. Listen Sie die Services auf, von denen `filesystem/autofs` abhängt.
12. Deaktivieren Sie `cron`, `bind`, `system-log` und `filesystem/local`, reboten Sie Ihre Maschine. In welchen Milestone kommt sie?
13. Sorgen Sie dafür, das die Maschine wieder in den `multi-user-server` kommt.

# SMF Manifeste/Profile/Methoden

1. Überprüfen Sie das Servicemanifest zum Multi-User-Server, welche Serviceabhängigkeiten bestehen?
2. Überprüfen Sie das Servicemanifest des Services cron. Welche Serviceabhängigkeiten bestehen? Wie lautet die Start-Methode? Wo steht die Startmethode? Was wird darin wie gestattet?
3. Welcher Prozess führt Startmethoden aus?
4. Verifizieren Sie das Manifest filesystem-local. Welche Abhängigkeiten bestehen. Kann die Maschine ohne diesen Service starten?
5. Verifizieren Sie den Milestone network. Welche Service müssen erfolgreich gestartet sein, damit er erreicht wird?
6. Muss ein Service *online* sein um erfolgreich gestartet zu werden?
7. Wenn ja, was passiert, wenn ein dependant-service auf disabled gesetzt wird und das System rebootet werden soll?
8. Verifizieren Sie hierzu den Milestone svc:/milestone/name-services
9. Verifizieren Sie aus die nis-Services. Ist NIS aktiviert? Wurde der Milestone svc:/milestone/name-services erreicht?

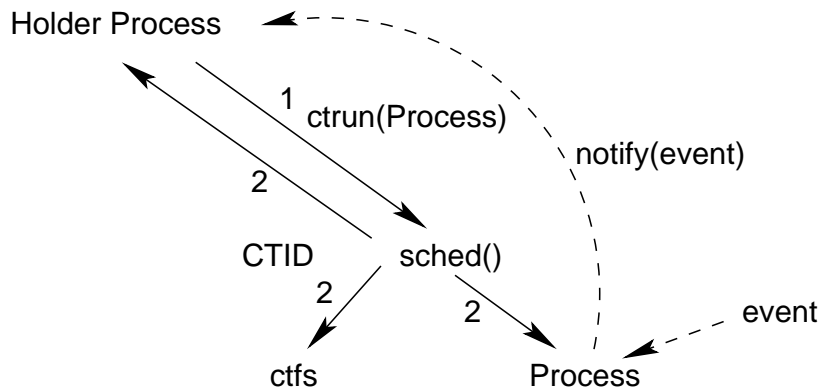
## LAB: SMF Milestones

1. Booten Sie Ihre LAB-Maschine SMF-Konform in den Single-User Status.
2. Verifizieren Sie dies mit dem Kommando *who -r*
3. Was fällt Ihnen auf?
4. Verifizieren Sie den Systemstatus mit dem Kommando *svcs | grep milestone*
5. In welchem Milestone befindet sich das System?
6. Führen Sie das Kommando *svcadm enable multi-user-server*
7. Melden Sie sich erneut an und verifizieren Sie wie oben den Systemstatus.
8. Ist Ihre Maschine in den Multi-User-Server Status gekommen?

# Service Management Facility

## Contract Filesystem

## Contract Filesystem, ctfs



## SMF/ctfs

### ctrun Starten eines Contracts für einen Prozess

```
sunrise# ctrun -i core,empty,fork,hwerr,exit xterm &
```

### ctstat Statusausgabe aller Contracts

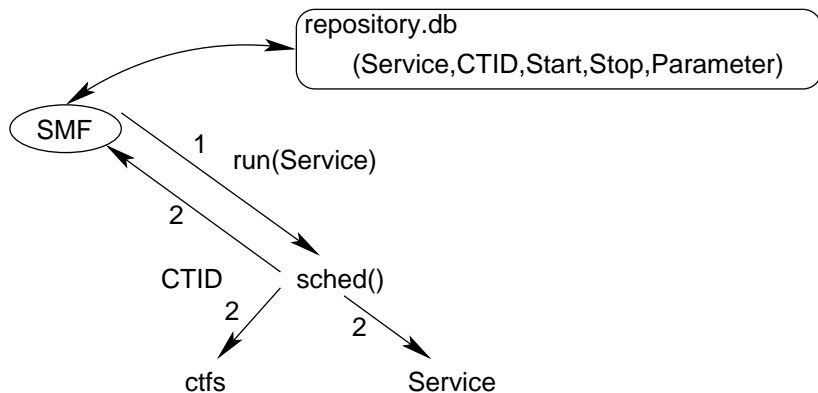
```
sunrise# ctstat
CTID    ZONEID  TYPE      STATE  HOLDER  EVENTS  QTIME  NTIME
1       0       process  owned  0       0       -       -
4       0       process  owned  1       0       -       -
5       0       process  owned  7       0       -       -
17      0       process  owned  7       0       -       -
18      0       process  orphan -       0       -       -
20      0       process  owned  7       0       -       -
24      0       process  owned  7       0       -       -
31      0       process  owned  7       0       -       -
33      0       process  owned  7       0       -       -
38      0       process  owned  7       0       -       -
....
114     0       process  owned  1043   0       -       -
```

### ctwatch Monitoring eines Contracts

```
sunrise# ctwatch 114
CTID      EVID      CRIT  ACK  CTYPE  SUMMARY
```



# SMF/ctfs



# LAB: ctfS

Machen Sie sich mit der Funktionalität des ctfS vertraut.

1. Erlauben Sie den Zugriff auf Ihren X-Server von Ihrer LAB-Maschine aus:

```
rayserver> echo $DISPLAY 4.0  
rayserver> xhost +labclient
```

```
labclient> DISPLAY=rayserver:4.0
```

2. Starten Sie ein `xterm(1)` unter ctfS-Kontrolle mit geeigneten Optionen (z.B.: `fork,exit,core`). Es öffnet sich ein `xterm` auf Ihrem Desktop.
3. Finden Sie die ContractID des gestarteten Prozesses heraus
4. Starten Sie ein Eventmonitoring in einem zweiten Fenster.
5. Starten Sie in dem `xterm`, dass sich auf Ihrem Desktop geöffnet hat einen oder mehrere Prozesse, im Hintergrund.
6. Verifizieren Sie die forks und exits der von Ihnen gestarteten Prozesse in der Ausgabe des `ctwatch`-Prozesses.
7. Lassen Sie sich mit `ps(1)` auch die ContractID anzeigen, verifizieren Sie mit `ptree(1)` die Prozesshierarchie
8. Terminieren Sie nach und nach alle Prozesse innerhalb des gemonitorten Contracts inclusive des zuerst gestarteten `xterms` und Verifizieren sie Ausgabe im `ctwatch`.