



## Solaris Virtualisierung (Zones) in der Praxis

Die Solaris Zones bieten sehr viele Möglichkeiten, wie man eine Solaris Umgebung virtualisieren kann. Jedoch sollte die Virtualisierung letzten Endes auch einen wirtschaftlichen Nutzen haben und keinen Mehraufwand bringen. Auf Grund der verschiedenen Möglichkeiten Zones zu konfigurieren und Ressourcen zu allozieren, kann man sich sehr schnell in technischen Details verlieren. Alle diese Einstellungen können am Schluss zu einem grossen Administrationsaufwand führen, welcher so nicht geplant war. Daher geht es in diesem Vortrag nicht darum alle technischen Möglichkeiten aus zu leuchten, sondern auf zu zeigen, was mit einem vernünftigen Aufwand machbar ist und wie man eine solche Virtualisierung in einem Framework einbinden könnte.

## Solaris Virtualisierung in der Praxis

Mit der Virtualisierung kann man  
Kosten reduzieren, wenn sie  
richtig eingesetzt wird.

## Solaris Virtualisierung in der Praxis

### Inhalt

- Einführung, Vor- und Nachteile der Virtualisierung.
- Technischer Teil, Typen von Zones, Installation, Konfiguration.
- Beispiel einer Einbindung der Zones in ein Framework.

## Einführung

# Was sind Solaris Zones?

- Bieten eine isolierte Umgebung zur Anwendungsausführung  
Prozesse können auf keinerlei Ressourcen ausserhalb ihrer Zone zugreifen  
Vergleichbar mit BSD Jails / Xen
- Ab Solaris 10 verfügbar  
Zones benötigen keine besondere Hardware
- Flexibel  
Zones sind schnell erstellt, kein Reboot nötig, kleiner Overhead



# DIGICOMP

Einführung

Situation ohne Zones



Applikationen

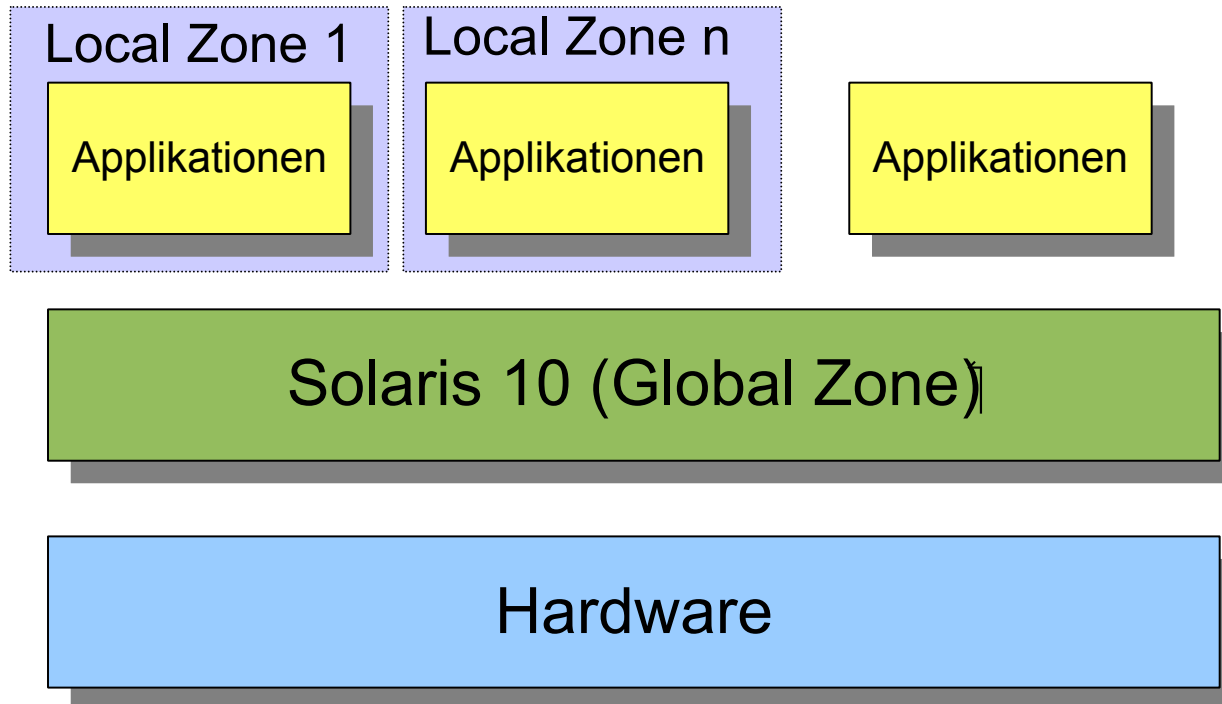
Ungenutzte Kapazität

Solaris

Hardware

## Einführung

### Situation mit Zones



## Einführung

# Vorteile einer Virtualisierung

- Reduktion der Kosten  
Flexibilität, Konsolidierung
- Steigerung des Service Levels  
Die Zones können auf andere Server verschoben werden (Framework)
- Reduktion der Datacenter Ressourcen  
Weniger Strom, weniger Kühlung, weniger Platz

## Einführung

# Nachteile einer Virtualisierung

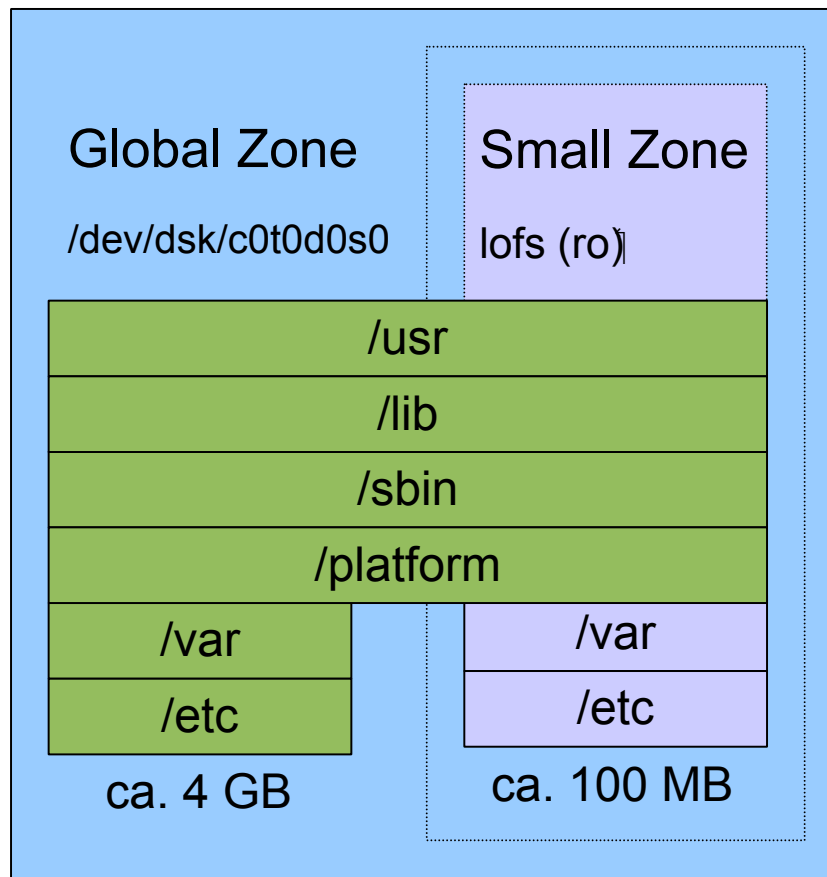
- Nicht für alles geeignet
  - Eignet sich nicht für „Turnkey“-Lösungen
- Abhängigkeiten
  - Ein Kernel, ein IP-Stack
- Konzentration auf wenig Hardware
  - Bei einem Ausfall des physikalischen Servers sind alle virtuelle Server Weg

## Installation, Konfiguration Erstellen von Zones

- Es gibt zwei Typen von Zones
  - Big Zones (Whole Zones) und Small Zones (Sparse Zones)
- Die CPU und Memory-Ressourcen einer Zone kann limitiert werden
  - Maximaler Konsum unter Last

## Installation, Konfiguration

### Die Small Zone



#### Vorteile:

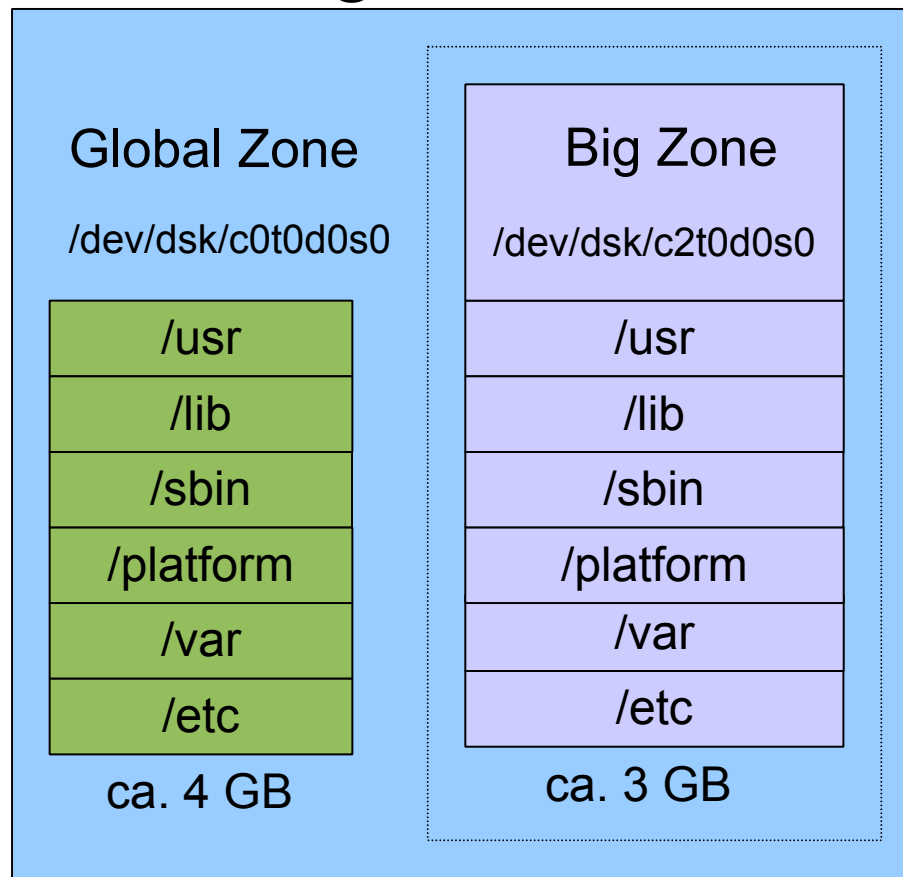
- Alle Packages und Patches einheitlich
- Weniger Platzbedarf
- Einfachere Verwaltung (alles aus der Global Zone)
- Alle Zonen haben den selben Patch-Level

#### Nachteile:

- Weniger flexibel
- Alle Zonen haben den selben Patch-Level
- Individuelle Packages und Patches nur sehr bedingt möglich
- Beim Patchen werden alle Zonen gestoppt

## Installation, Konfiguration

### Die Big Zone



#### Vorteile:

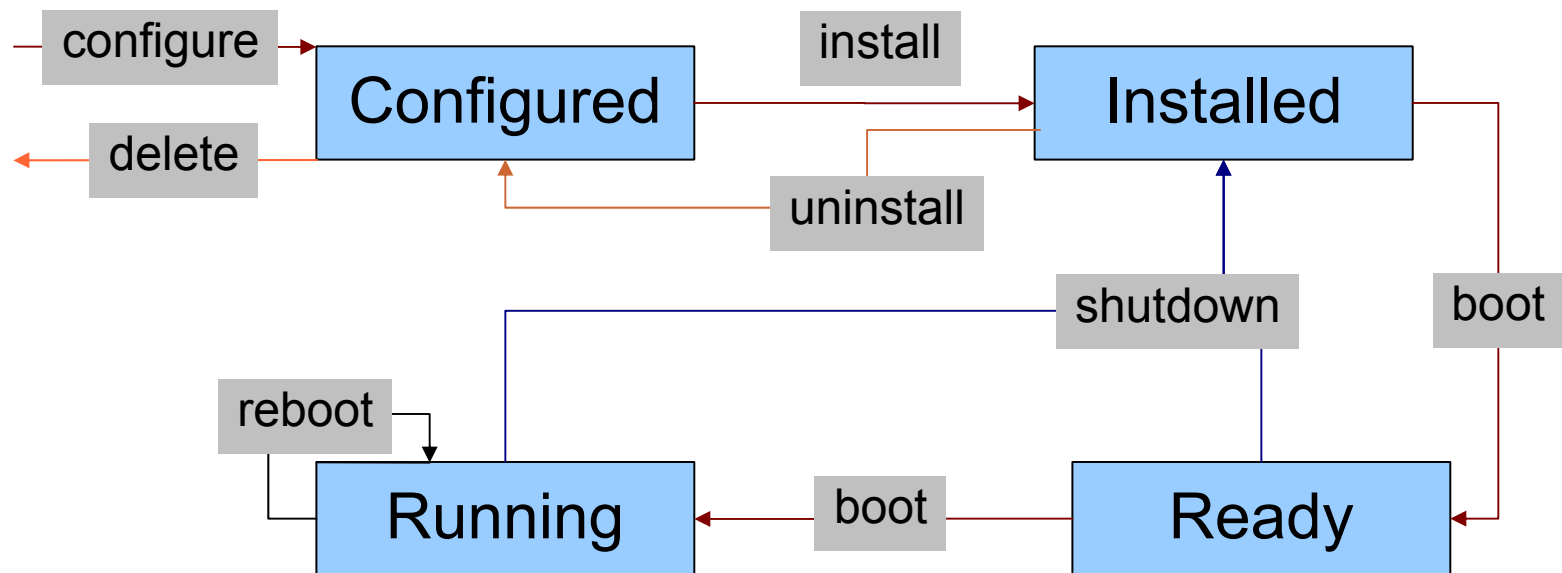
- Alle Packages und Patches können einheitlich sein, müssen aber nicht.
- Flexibler, da unabhängiger
- Eigenständige Installation von Patches und Packages möglich.  
(`pkgadd -G`, `patchadd -G`)

#### Nachteile:

- Braucht mehr Platz
- Mehr Aufwand für Verwaltung

# DIGICOMP

## Installation, Konfiguration Zustände einer Zone





# DIGICOMP

## Installation, Konfiguration Erstellen einer Zone

- Zuerst muss das root-Verzeichnis erstellt werden

```
mkdir /export/small-zone1  
chmod 700 /export/small-zone1
```

## Installation, Konfiguration

## Erstellen einer Zone

- Grundkonfiguration

```
zonecfg -z small-zone1
```

```
zonecfg:small-zone1> create
```

```
zonecfg:small-zone1> set autoboot=true
```

```
zonecfg:small-zone1> set zonepath=/export/small-zone1
```

## Installation, Konfiguration

# Erstellen einer Zone

- Netzwerk Konfiguration

```
zonecfg:small-zone1> add net
```

```
zonecfg:small-zone1:net> set address=192.168.1.10
```

```
zonecfg:small-zone1:net> set physical=ce0
```

```
zonecfg:small-zone1:net> end
```

- Die Netzmaske wird in der Global Zone gesetzt

```
/etc/netmasks
```

## Installation, Konfiguration

### Erstellen einer Zone

- Ressource Control (Details in Anhang: Ressource Control)

```
zonecfg:small-zone1> add rctl
```

```
zonecfg:small-zone1:rctl> set name=zone.cpu-shares
```

```
zonecfg:small-zone1:rctl> add value (priv=privileged,limit=15,action=none)
```

```
zonecfg:small-zone1:rctl> end
```

- priv: basic, privileged, system
- action: deny, signal, none

## Installation, Konfiguration

### Erstellen einer Zone

- Attribute hinzufügen

```
zonecfg:small-zone1> add attr
```

```
zonecfg:small-zone1:attr> set name=comment
```

```
zonecfg:small-zone1:attr> set type=string
```

```
zonecfg:small-zone1:attr> set value="Small Zone with 15 Shares"
```

```
zonecfg:small-zone1:attr> end
```

## Installation, Konfiguration Erstellen einer Zone

- Konfiguration abschliessen

```
zonecfg:small-zone1> verify
```

```
zonecfg:small-zone1> commit
```

```
zonecfg:small-zone1> exit
```

- Die Zone ist nun im „Configured“-Mode

## Installation, Konfiguration

## Erstellen einer Zone

- Zone installieren

```
zonecfg -z small-zone1 info  
zoneadm -z small-zone1 verify  
zoneadm -z small-zone1 install  
zoneadm -z small-zone1 boot
```

- Die Zone ist nun im „Running“-Mode

## Installation, Konfiguration Erstellen einer Zone

- Zones anzeigen

```
zoneadm list -iv
```

ID	NAME	STATUS	PATH
0	global	running	/
-	small-zone1	running	/export/small-zone1

- Diese Zones sind am Laufen



# DIGICOMP

## Installation, Konfiguration

### Erstellen einer Zone

- In die Zone einloggen

```
zlogin -C small-zone1
```

- Diese Zone ist nun bereit

## Installation, Konfiguration

### Pool Konfiguration

- Zur besseren Aufteilung der Ressourcen können Pools erstellt werden
- Die Zones können dann Pools zugewiesen werden
- Die Pools werden durch den Prozess „poold“ kontrolliert.
- Konfigurations-File für poold ist „/etc/pooladm.conf“

## Installation, Konfiguration

### Pool Konfiguration

- Der Pool Daemon wird gestartet mit ***pooladm -e*** (stoppen ***-d***)
- Mit dem Kommando ***pooladm*** werden danach alle CPUs angezeigt:

```
pooladm
```

```
.....
```

```
cpu
```

```
int    cpu.sys_id 17  
string cpu.comment  
string cpu.status on-line
```

```
.....
```

## Installation, Konfiguration

### Pool Konfiguration

- Erstellen von „create\_pool\_fix“ mit folgendem Inhalt

```
create pset work1-pset ( uint pset.min = 2; uint pset.max = 6 )
```

```
create pset work2-pset ( uint pset.min = 1; uint pset.max = 4 )
```

```
create pool work1-pool
```

```
create pool work2-pool
```

```
create pool work3-pool
```

```
associate pool work1-pool ( pset work1-pset )
```

```
associate pool work2-pool ( pset work2-pset )
```

## Installation, Konfiguration

### Pool Konfiguration

- Erstellen eines leeren Konfigurations-File: *pooladm -s* (löschen *-x*)
- Laden der neuen Konfiguration: *poolcfg -f create\_pool\_fix*
- Anschauen der CPU-Aufteilung

*psrset*

user processor set 1: processors 0 1 2 3

user processor set 2: processors 8 9 10

## Installation, Konfiguration

### Pool Konfiguration

- Nun können die Zones den Pools zugeteilt werden:

```
zonecfg -z small-zone1
```

```
zonecfg:small-zone1> set pool=work1-pool
```

```
zonecfg:small-zone1> verify
```

```
zonecfg:small-zone1> commit
```

```
zonecfg:small-zone1> exit
```

- Anschliessend Zone rebooten

## Installation, Konfiguration

# Resource Control mit FSS

- Mit dem Fair Share Scheduler (FSS) kann die Last feiner verteilt werden
- Damit FSS aktiv wird muss Solaris rebootet werden.
- Der FSS tritt erst in Aktion, wenn die CPU-Last 100% erreicht
- Die Last wird in „Shares“ vergeben. Default ist 1 Share.

## Installation, Konfiguration

# Resource Control mit FSS

- Pools müssen mit dem FSS kreiert werden
- Erstellen von „create\_pool\_fss“ mit folgendem Inhalt

```
create pset work1-pset ( uint pset.min = 2; uint pset.max = 4 )  
create pset work2-pset ( uint pset.min = 2; uint pset.max = 6 )  
create pool work1-pool ( string pool.scheduler = "FSS" )  
create pool work2-pool ( string pool.scheduler = "FSS" )  
create pool work3-pool ( string pool.scheduler = "FSS" )  
associate pool work1-pool ( pset work1-pset )  
associate pool work2-pool ( pset work2-pset )
```

## Installation, Konfiguration

# Resource Control mit FSS

- Nach dem Aktivieren von FSS, muss Solaris rebootet werden

```
dispadmin -d FSS  
init 6
```

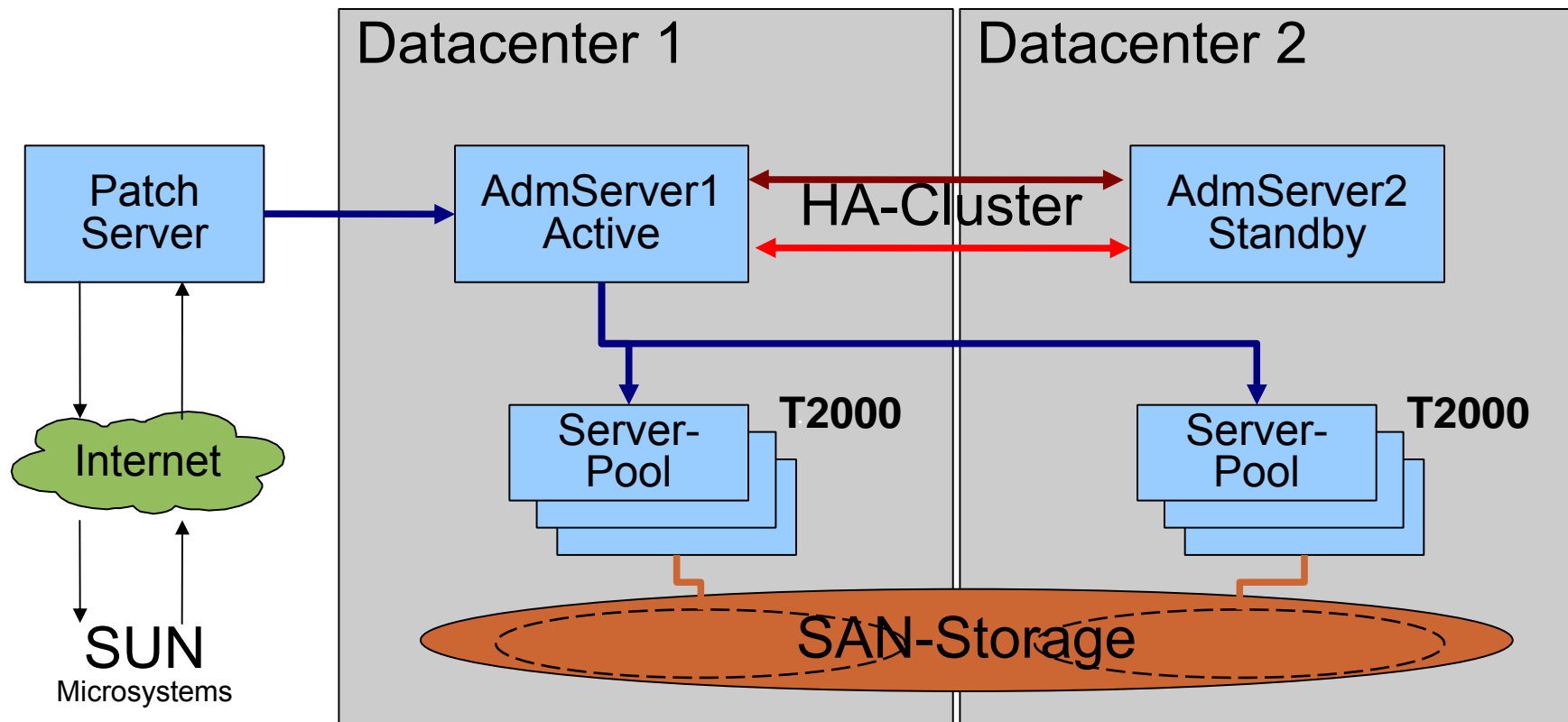
## Framework

# Einbindung von Zones in ein Framework

- Es wird ein Compute-Pool mit n Servern gebildet
- Über einen Management Server können die Zones verwaltet werden.
- Mehr Flexibilität
- Höhere Verfügbarkeit der Services

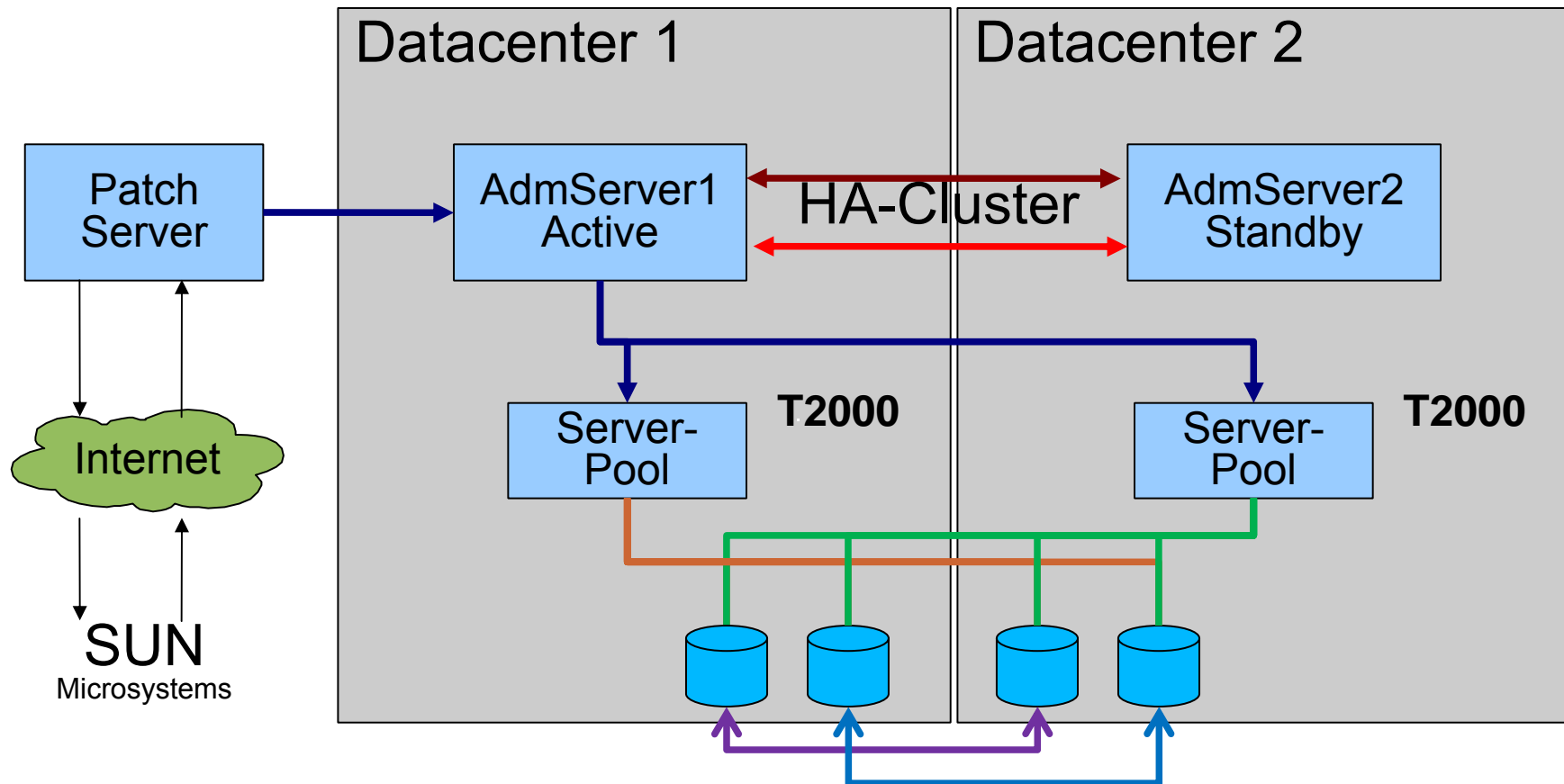
## Framework

### Beispiel eines Framework



## Framework

### Beispiel eines Framework



## Framework

### Vorteile eines Framework

- Die Zones können innerhalb eines Compute-Pools verschoben werden
- Höhere Verfügbarkeit, Switchen der Zones über zwei Sites
- Mehr Flexibilität, da schnell neue Zones erstellt werden können
- Zentrale Verwaltung der Zones (Patches etc.)

## Framework

### Nachteile eines Framework

- Alle Server im Pool müssen die selbe CPU-Architektur haben
- Alle Server im Pool müssen die selbe Kernel Version haben
- Es braucht einen zentralen Storage
- Alle LUNs im Pool müssen auf allen Servern sichtbar sein.
- Beim Patchen müssen alle Zones auf einem Server gestoppt werden



**DIGICOMP**

**Drive your life.**



## Anhang Solaris Zonen

In diesem Anhang werden einige Beispiele im Zusammenhang mit dem erstellen von Solaris Zonen erleutert.

## Erstellen einer Zone

Folgende Syntax wird in diesem Dokument verwendet:

Ausführen eines Kommandos in der Shell

Beispiel: ***mkdir /export/small-zone1***

Ausgabe eines Kommandos:

Beispiel: ***psrset***

user processor set 1: processors 0 1 2 3

user processor set 2: processors 8 9 10

user processor set 3: processors 11 16

Der Kommandoprompt wird nicht speziell gekennzeichnet, ausser er entspricht nicht dem Standard Unix Prompt

Beispiel: ***zonecfg -z small-zone1***

*zonecfg:small-zone1>* create

## Erstellen einer Zone

### Erstellen einer Zone ohne Limiten

Auf den nachfolgenden Seiten wird das erstellen einer Zone anhand der Zone small-zone1 Schritt um Schritt dargestellt. Nachfolgende Kommandos müssen mit dem root-User ausgeführt werden.

### Erstellen des root-Verzeichnis für die Zone

```
mkdir /export/small-zone1  
chmod 700 /export/small-zone1
```

### Zone erstellen:

```
zonecfg -z small-zone1  
zonecfg:small-zone1> create  
zonecfg:small-zone1> set autoboot=true  
zonecfg:small-zone1> set zonepath=/export/small-zone1  
zonecfg:small-zone1> add net  
zonecfg:small-zone1:net> set address=192.168.1.10  
zonecfg:small-zone1:net> set physical=ce0  
zonecfg:small-zone1:net> end  
zonecfg:small-zone1> add attr  
zonecfg:small-zone1:attr> set name=comment  
zonecfg:small-zone1:attr> set type=string  
zonecfg:small-zone1:attr> set value="Small Zone1"  
zonecfg:small-zone1:attr> end  
zonecfg:small-zone1> verify  
zonecfg:small-zone1> commit  
zonecfg:small-zone1> exit
```

## Erstellen einer Zone

Zonen Info abfragen:

```
zonecfg -z small-zone1 info
```

Zone installieren und starten:

```
zoneadm -z small-zone1 verify  
zoneadm -z small-zone1 install  
zoneadm -z small-zone1 boot
```

Status der Zonen auflisten

```
zoneadm list -iv
```

```
ID NAME      STATUS  PATH  
0 global    running /  
- small-zone1 running /export/small-zone1
```

Mit der Console auf die Zone einloggen

```
zlogin -C small-zone1
```

Eine Zone wieder entfernen

```
zoneadm -z small-zone1 uninstall  
zonecfg -z small-zone1 delete
```

## Erstellen einer Zone

### 1. Fenster

Select a Language

- 0. English
- 1. es
- 2. fr

Please make a choice (0 - 2), or press h or ? for help: 0

---

### Fenster2:

Select a Locale

- 0. English (C - 7-bit ASCII)
- 1. Canada-English (ISO8859-1)
- 2. U.S.A. (en\_US.ISO8859-1)
- 3. U.S.A. (en\_US.ISO8859-15)
- 4. Go Back to Previous Screen

Please make a choice (0 - 4), or press h or ? for help: 3

## Erstellen einer Zone

### Fenster3:

What type of terminal are you using?

- 1) ANSI Standard CRT
- 2) DEC VT52
- 3) DEC VT100
- 4) Heathkit 19
- 5) Lear Siegler ADM31
- 6) PC Console
- 7) Sun Command Tool
- 8) Sun Workstation
- 9) Televideo 910
- 10) Televideo 925
- 11) Wyse Model 50
- 12) X Terminal Emulator (xterms)
- 13) CDE Terminal Emulator (dtterm)
- 14) Other

Type the number of your choice and press Return: **3**

## Erstellen einer Zone

### Fenster 4:

– Host Name for ce0:3 \_\_\_\_\_

Enter the host name which identifies this system on the network. The name must be unique within your domain; creating a duplicate host name will cause problems on the network after you install Solaris.

A host name must have at least one character; it can contain letters, digits, and minus signs (-).

Host name for ce0:3 small-zone1

\_\_\_\_\_

F2\_Continue F6\_Help

---

### Fenster 5:

– Confirm Information for ce0:3 \_\_\_\_\_

> Confirm the following information. If it is correct, press F2;  
to change any information, press F4.

Host name: **small-zone1**

\_\_\_\_\_

F2\_Continue F4\_Change F6\_Help

## Erstellen einer Zone

### Fenster 6:

– Configure Security Policy: \_\_\_\_\_

.....

Configure Kerberos Security

Yes

No

\_\_\_\_\_

**F2\_Continue** F6\_Help

\_\_\_\_\_

### Fenster 7:

– Confirm Information \_\_\_\_\_

> Confirm the following information. If it is correct, press F2;  
to change any information, press F4.

Configure Kerberos Security: No

\_\_\_\_\_

**F2\_Continue** F4\_Change F6\_Help

## Erstellen einer Zone

### Fenster 8:

– Name Service \_\_\_\_\_

.....

Name service  
\_\_\_\_\_

- NIS+
- NIS
- DNS
- LDAP
- None

---

**F2\_Continue** F6\_Help

### Fenster 9:

– Confirm Information \_\_\_\_\_

> Confirm the following information. If it is correct, press F2;  
to change any information, press F4.

Name service: None

---

**F2\_Continue** F4\_Change F6\_Help



# DIGICOMP

## Erstellen einer Zone

**Fenster 10:**

– Time Zone \_\_\_\_\_

.....

Continents and Oceans

---

^  Americas

.....

|  Europe

.....

v  Pacific Ocean

---

**F2\_Continue** **F6\_Help**

---

**Fenster 11:**

– Country or Region \_\_\_\_\_

.....

Countries and Regions

---

^  Portugal

.....

|  Switzerland

.....

---

**F2\_Continue** **F6\_Help**

## Erstellen einer Zone

### Fenster 12:

– Confirm Information —————

> Confirm the following information. If it is correct, press F2;  
to change any information, press F4.

Time zone: Europe/Zurich

—————  
F2\_Continue F4\_Change F6\_Help  
—————

### Fenster 13:

– Root Password —————

Please enter the root password for this system.

The root password may contain alphanumeric and special characters. For  
security, the password will not be displayed on the screen as you type it.

> If you do not want a root password, leave both entries blank.

Root password: \*\*\*\*

Root password: \*\*\*\*

—————  
F2\_Continue F6\_Help

## Erstellen einer Zone

### Fenster 14:

System identification is completed.

This system is configured with NFS version 4, which uses a domain name that is automatically derived from the system's name services. The derived domain name is sufficient for most configurations. In a few cases, mounts that cross different domains might cause files to be owned by "nobody" due to the lack of a common domain name.

Do you need to override the system's default NFS version 4 domain name (yes/no) ? [no] : Enter drücken

rebooting system due to change(s) in /etc/default/init

[NOTICE: Zone rebooting]

.....

small-zone1 console login:

## Erstellen einer Zone

### Erstellen einer Zone mit einer CPU Limite

Nachfolgend ein Beispiel für eine Zone mit einer CPU Usage Limite von 30 Shares. Alle anderen Kommandos sind wie im ersten Beispiel beschrieben.

#### *zonecfg -z small\_zone2*

```
zonecfg:small_zone2> create
zonecfg:small_zone2> set zonepath=/export/small_zone2
zonecfg:small_zone2> set autoboot=true
zonecfg:small_zone2> add net
zonecfg:small_zone2:net> set physical=ce0
zonecfg:small_zone2:net> set address=192.168.1.20
zonecfg:small_zone2:net> end
zonecfg:small_zone2> add rctl
zonecfg:small_zone2:rctl> set name=zone.cpu-shares
zonecfg:small_zone2:rctl> add value (priv=privileged,limit=30,action=none)
zonecfg:small_zone2:rctl> end
zonecfg:small_zone2> add attr
zonecfg:small_zone2:attr> set name=comment
zonecfg:small_zone2:attr> set type=string
zonecfg:small_zone2:attr> set value="Small Zone2 with 30 Shares"
zonecfg:small_zone2:attr> end
zonecfg:small_zone2> verify
zonecfg:small_zone2> commit
zonecfg:small_zone2> exit
```

## Erstellen einer Zone

### Erstellen einer Big Zone mit einer CPU Limite

Nachfolgend ein Beispiel für eine Big Zone mit einer CPU Usage Limite von 45 Shares. Alle anderen Kommandos sind wie im ersten Beispiel beschrieben.

#### *zonecfg -z big-zone1*

```
zonecfg:big-zone1> create
zonecfg:big-zone1> remove inherit-pkg-dir dir=/sbin
zonecfg:big-zone1> remove inherit-pkg-dir dir=/usr
zonecfg:big-zone1> remove inherit-pkg-dir dir=/platform
zonecfg:big-zone1> remove inherit-pkg-dir dir=/lib
zonecfg:big-zone1> set autoboot=true
zonecfg:big-zone1> set zonepath=/export/big-zone1
zonecfg:big-zone1> add net
zonecfg:big-zone1:net> set physical=ce1
zonecfg:big-zone1:net> set address=192.168.1.30
zonecfg:big-zone1:net> end
zonecfg:big-zone1> add rctl
zonecfg:big-zone1:rctl> set name=zone.cpu-shares
zonecfg:big-zone1:rctl> add value (priv=privileged,limit=45,action=none)
zonecfg:big-zone1:rctl> end
zonecfg:big-zone1> add attr
zonecfg:big-zone1:attr> set name=comment
zonecfg:big-zone1:attr> set type=string
zonecfg:big-zone1:attr> set value="Big Zone1 with 45 Shares"
zonecfg:big-zone1:attr> end
```

## Aufteilung der Last auf Zonen

Die Auslastung der Zonen sieht man am Besten mit folgendem Kommando:

***prstat -Z***

```
PID USERNAME  SIZE  RSS STATE  PRI NICE   TIME  CPU PROCESS/NLWP
7472 root    2944K 1608K run    41  0  0:00:50 8.7% bash/1
7499 root    2944K 1608K run    22  0  0:00:31 8.5% bash/1
7501 root    2944K 1608K run    23  0  0:00:33 8.3% bash/1
7525 root    2944K 1608K run    43  0  0:00:21 7.2% bash/1
7524 root    2944K 1608K run    33  0  0:00:23 7.1% bash/1
7500 root    2944K 1608K run    42  0  0:00:28 7.0% bash/1
7471 root    2944K 1608K run    22  0  0:00:43 7.0% bash/1
6824 root    5040K 4744K cpu0  49  0  0:00:01 0.3% prstat/1
21655 oracle 262M 230M sleep 59  0  1:33:13 0.0% oracle/1
ZONEID  NPROC  SIZE  RSS MEMORY  TIME  CPU ZONE
  6    37 131M  84M  0.9%  0:03:47 62% small-zone1
  7    33 119M  80M  0.8%  0:03:22 31% small-zone2
  0   183 17G 9073M 98% 66:04:43 1.0% global
```

Total: 253 processes, 1564 lwps, load averages: 18.16, 7.98, 3.52

## Ressource Control

- Folgende Ressourcen können konfiguriert werden:

process.max-port-events maximum allowable number of events per event port  
process.crypto-buffer-limit maximum number of bytes allocated for copying  
process.max-crypto-sessions maximum number of entries in the session table  
process.add-crypto-sessions number of entries added when enlarging the session table  
process.min-crypto-sessions minimum number of entries in the session table  
process.max-msg-messages maximum number of messages on a message queue  
process.max-msg-qbytes maximum number of bytes of messages on a message queue  
process.max-sem-ops maximum number of semaphore operations per semop call  
process.max-sem-nsems maximum number of semaphores per semaphore set  
process.max-address-space maximum size of the address space in bytes  
process.max-file-descriptor maximum index in filedescriptor table  
process.max-core-size maximum core file size in bytes  
process.max-stack-size maximum size of the stack segment in bytes  
process.max-data-size maximum size of the data segment in bytes  
process.max-file-size maximum file size in bytes  
process.max-cpu-time maximum CPU time in seconds  
task.max-cpu-time maximum CPU time in seconds  
task.max-lwps maximum number of simultaneously available LWPs  
project.max-port-ids maximum allowable number of event ports  
project.max-shm-memory maximum size of System V shared memory in bytes  
project.max-shm-ids maximum number of System V shared memory segments  
project.max-msg-ids maximum number of System V message queues  
project.max-sem-ids maximum number of System V semaphores  
project.cpu-shares the number of CPU shares  
zones.cpu-shares number of CPU shares per zone

## Literatur

- [1] Jeff Victor, "Solaris Containers Technology Architecture Guide", Sun Blueprint, Mai 2006,  
<http://www.sun.com/blueprints/0506/819-6186.html>
- [2] Jeff Victor, "Zones and Containers FAQ", OpenSolaris FAQ,  
<http://opensolaris.org/os/community/zones/faq/>
- [3] Sun Microsystems Inc., "Solaris Containers Learning Center",  
[http://www.sun.com/software/solaris/containers\\_learning\\_center.jsp](http://www.sun.com/software/solaris/containers_learning_center.jsp)
- [4] Joost Pronk van Hoogeveen, "Working with Solaris Containers and the Solaris Service Manager", Sun Blueprint, Mai 2006,  
<http://www.sun.com/blueprints/0506/819-4328.html>
- [5] Menno Lageman, "Solaris Containers --What They Are and How to Use Them", Sun Blueprint, Mai 2005,  
<http://www.sun.com/blueprints/0505/819-2679.pdf>
- [6] Sun Microsystems Inc., "System Administration Guide: Solaris Containers-Resource Management and Solaris Zones", Solaris 10 Manual, 2006,  
<http://docs.sun.com/app/docs/doc/817-1592>
- [7] OpenSolaris Project: „Crossbow: Network Virtualization and Resource Control“  
<http://opensolaris.org/os/project/crossbow/>
- [8] "Bringing Your Application Into the Zone".  
[http://developers.sun.com/solaris/articles/application\\_in\\_zone.html](http://developers.sun.com/solaris/articles/application_in_zone.html)
- [9] "Virtual Datacenter Control Framework (VDCF)".  
<http://www.jomasoft.ch/products/VDCF/index.html>