

Oracle Datenbanken mit Oracle Solaris virtualisiert

Heiko Stein
etomer GmbH
Berlin

Detlef Drewanz
Oracle Deutschland BV & Co. KG
Berlin

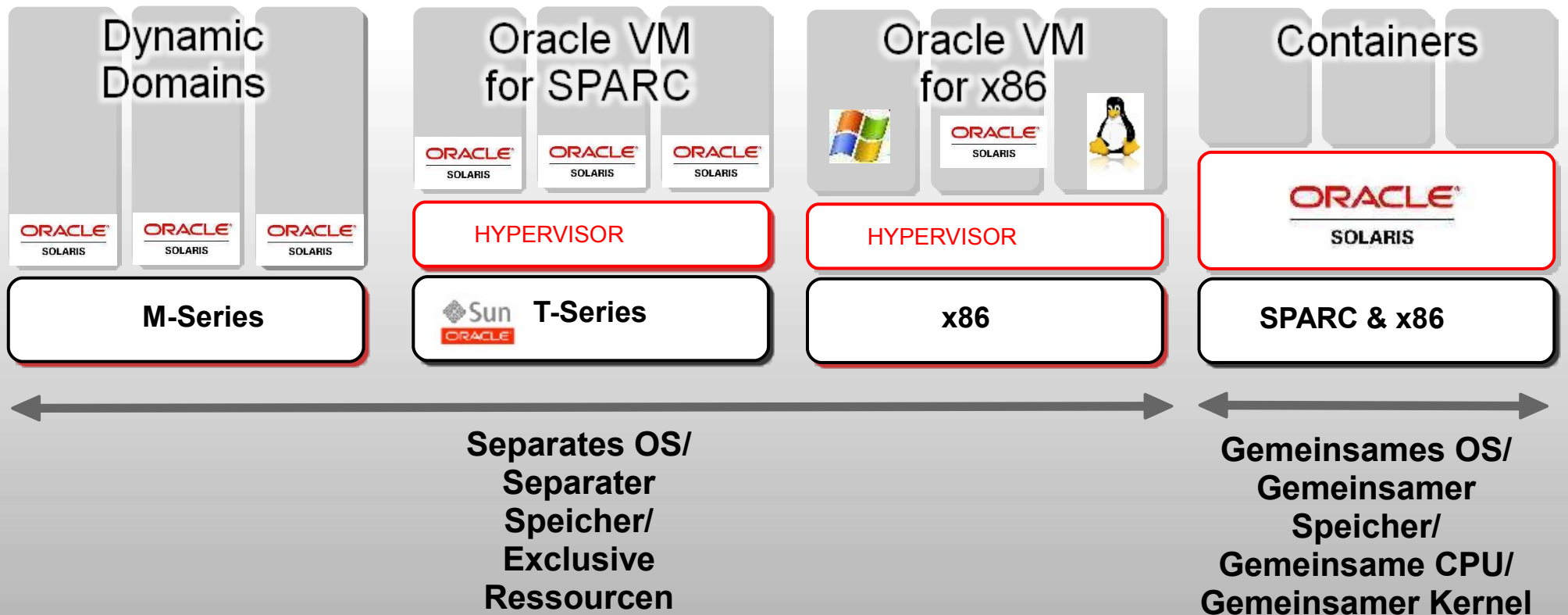
etomer GmbH – das ganzheitliche Systemhaus aus Berlin

- **Beratungsorientiert.**
 - Fokus auf Technik, Prozesse und Menschen.
- **Hochspezialisiert.**
 - Kernkompetenz auf Technologien und Prozesse im Umfeld produktionskritischer und hochverfügbarer IT-Infrastrukturen.
- **Projekterfahren.**
 - Mehr als 12 Jahre Erfahrung in Projekten bei Enterprise-Kunden verschiedenster Branchen.
- **Verbindlich.**
 - Leitbild - Zentrum unserer Aktivitäten. Wir übernehmen Verantwortung für unsere Beratung, unser Tun und Handeln.
- **Hersteller neutral.**
 - Technologie- und herstellerübergreifende Beratung zur lösungsorientierten Arbeit mit unseren Kunden.

Agenda

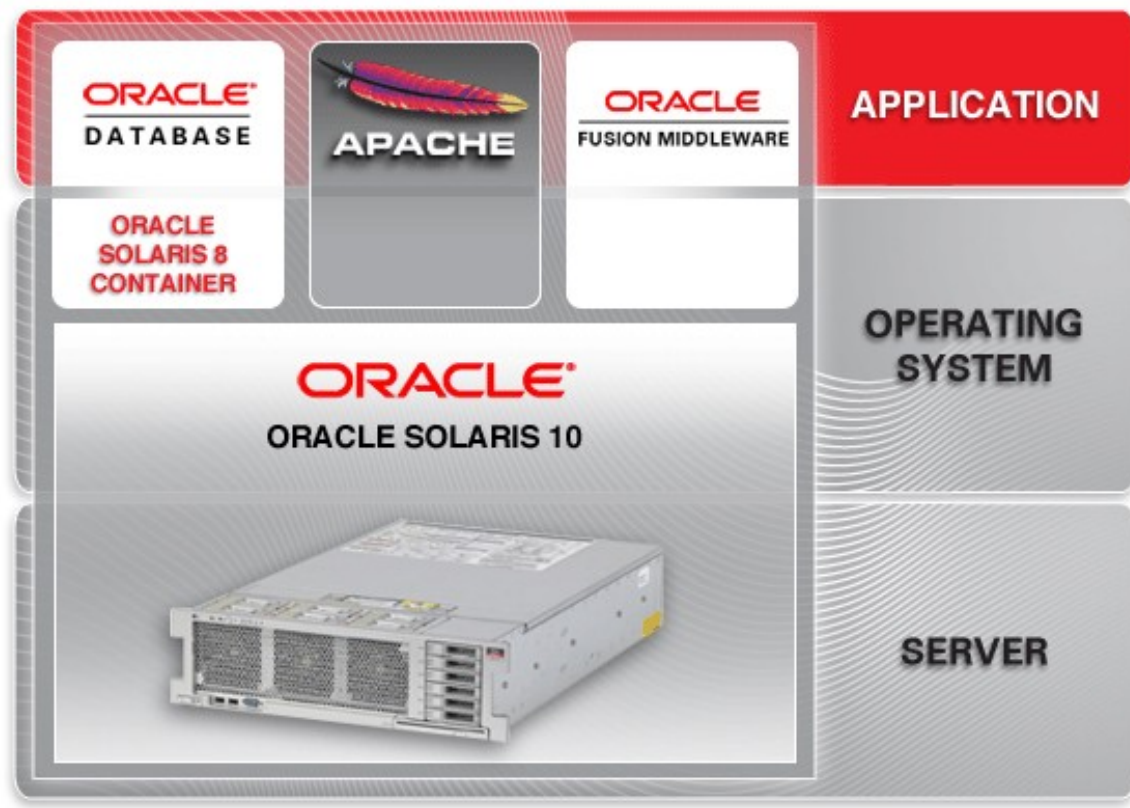
- Virtualisierung mit Oracle Solaris
- Oracle Solaris Zonen
- Ressourcen Management mit Oracle Solaris Zonen
- Best Practices
- Praktische Umsetzung

Arten der Server-Virtualisierung mit Oracle Solaris



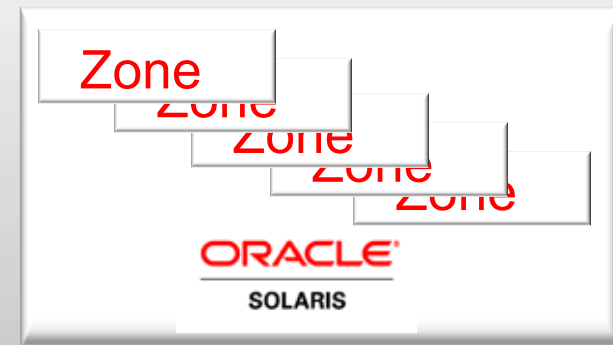
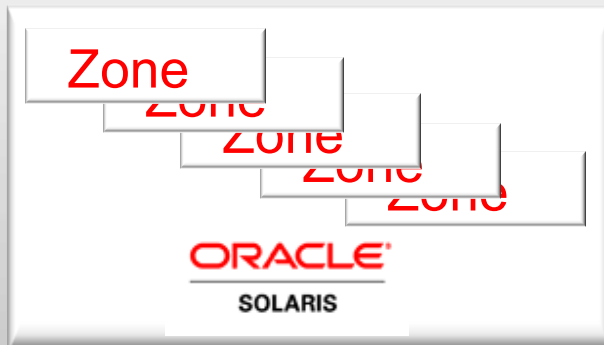
Oracle Solaris Zones: OS Virtualisierung

- Oracle Solaris Container aka Oracle Solaris Zones
- Eine globale Zone und mehrere non-globale Zonen
- Partitionierte Ablaufumgebungen mit Ressourcenmanagement für Anwendungen



Oracle Solaris Zonen

- Verfügbar auf jedem Oracle Solaris System
 - Auf SPARC und x86
 - Keine extra Kosten
 - Sehr weit verbreitet im Einsatz
- Branded Zones
 - Solaris 8 und Solaris 9 Branded Zones auf Oracle Solaris 10(Nur SPARC)
 - Solaris 10 Branded Zones auf Oracle Solaris 11



Oracle Solaris Zonen

- In jedem Oracle Solaris enthalten

- Einfach, mit geringstem Overhead bei höchster Performance
- Schnelle Bereitstellung von Zonen nach Bedarf
 - ZFS für ZFS Snapshot, Clone, Migration nutzbar
- Mission-critical Einsatz mit Oracle Solaris Cluster
 - Failover Zonen Clustern
 - Zonen als virtuelle Cluster Knoten
- Funktionalitäten zur Redundanz auch in den Zonen nutzbar
 - MPXIO
 - IPMP
- Isolation
 - Security
 - Fehler auf Anwendungsebene
 - Delegation von Administrationsrechten
- Dynamisches Ressourcen Management

Datenbanken in Oracle Solaris Zonen - Einsatzgebiete

•Konsolidierung

- Lastoptimierung von Datenbanken
- Zusammenfassung von Einzeldatenbanken
- Betrieb von Test- und Schulungsdatenbanken
- Effektive Nutzung von Hardware und Lizenzen

•Isolation

- Trennung zwischen Datenbanken auf einem System
- Trennung von Entwicklung/Test/Qualitätskontrolle/Produktion auf einem System

•Architekturen

- Aufbau flexibler Installationen (schnelles on-/offlining)
- Kapselung und Modularisierung
- Bereitstellung von Solaris 8/9 Umgebungen (für ältere Datenbankversionen)

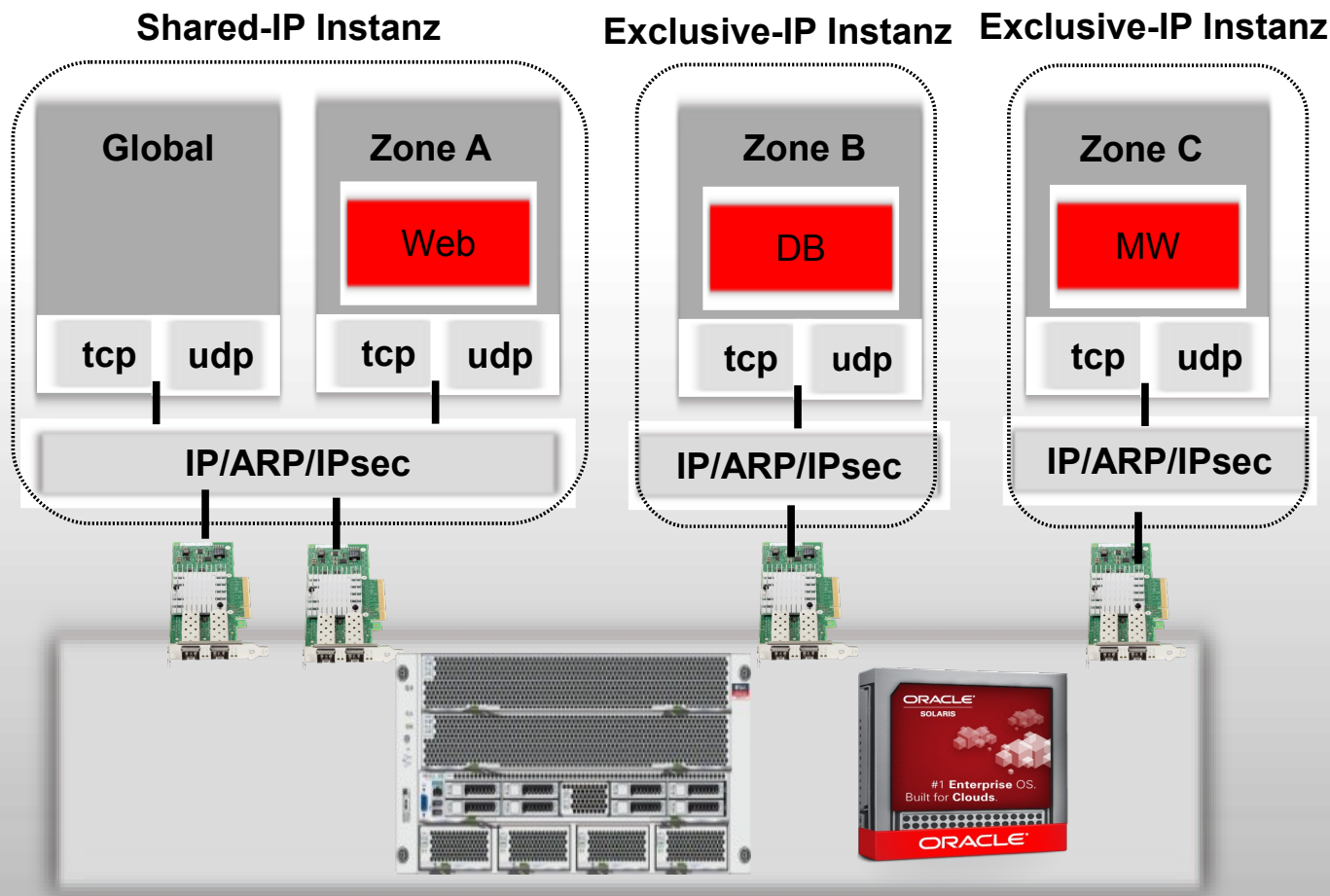
Die Wichtigkeit des Ressourcen Managements

- Solaris sorgt für eine faire Verteilung der Ressourcen
- Konsolidierte Umgebungen vereinen Arbeitslast mit unterschiedlichen Anforderungen
 - Durchsatz
 - Antwortzeit
 - Verfügbarkeit
 - SLA
- Sicherung der Besonderheiten und „Bevorzugungen“ durch Ressource Management
- Ressourcen Management muss dynamisch, d.h. zur Laufzeit änderbar, sein

Formen des Ressourcenmanagement

- Ausbalancierung
 - Faire Verteilung der Ressourcen nach Regeln
 - Bei Vollast Verteilung nach vorher zugesicherten Anteilen an der Gesamtlast
- Limitierung (Capping)
 - Kappung des Ressourcenverbrauches an einem Limit
- Partitionierung
 - Exklusive Zuordnung und Benutzung von Ressourcen

Shared-IP und Exclusive IP-Instanzen - Gemeinsamer oder exklusiver Netzwerkstack



Resource Management mit Zonen - Hilft SLA's zu erfüllen

· CPU

- CPU Fair-Sharing
- CPU Capping
- CPU Partitioning (Prozessorsets, Ressourcepools)

· Memory

- Benutzter physischer Hauptspeicher (RSS)
- Virtueller Hauptspeicher (Swap)
- Non-Pageable Memory (Locked)

· Prozesse

- LWP's
- Prozesse (Oracle Solaris 11)

zone.max-swap					
system	16.0EB	max	deny	-	
zone.max-locked-memory					
system	16.0EB	max	deny	-	
zone.max-shm-memory					
system	16.0EB	max	deny	-	
zone.max-shm-ids					
system	16.8M	max	deny	-	
zone.max-sem-ids					
system	16.8M	max	deny	-	
zone.max-msg-ids					
system	16.8M	max	deny	-	
zone.max-lwps					
system	2.15G	max	deny	-	
zone.cpu-cap					
system	4.29G	inf	deny	-	
zone.cpu-shares					
privileged	1	-	none	-	
system	65.5K	max	none	-	

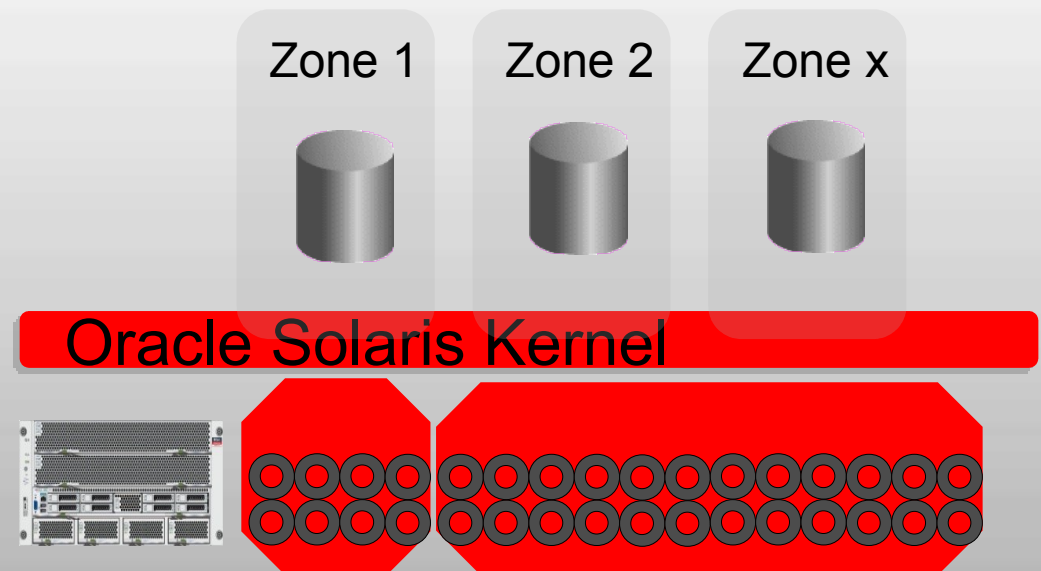
Resource Management mit Zonen - Hilft SLA's zu erfüllen

- Netzwerk (Solaris 11)
 - CPU Nutzung für Netzwerkverarbeitung
 - Capping der Netzwerkbandbreite
- Interprozess Kommunikation
 - Shared Memory
 - Semaphore
 - Message Queues
- Gemeinsam genutzter Kernel
 - Zentrales Monitoring
 - Dynamische Änderung der Einstellungen an zentraler Stelle
 - poolcfg
 - prctl
 - rcapadm



Capped Container zur Optimierung von Oracle Lizenzen

- Capped Container als Hard Partitions
 - Bildung eines Prozessorset durch feste Zuweisung von Prozessoren
 - Zuweisung des Prozessorset zu einem Ressourcenpool
 - Bindung der Zone an den Ressourcenpool
- Lizenzierung der Prozessoren im Ressourcenpool
- Sinnvoll, wenn einige Prozessor einer separaten Zone für andere Anwendungen genutzt werden sollen
- **Nicht zu verwechseln mit CPU Capping !**
- Z.B. 8 CPU und 24 CPU vs. 32 CPU



Best Practice Nutzung von Oracle Solaris Container (1)

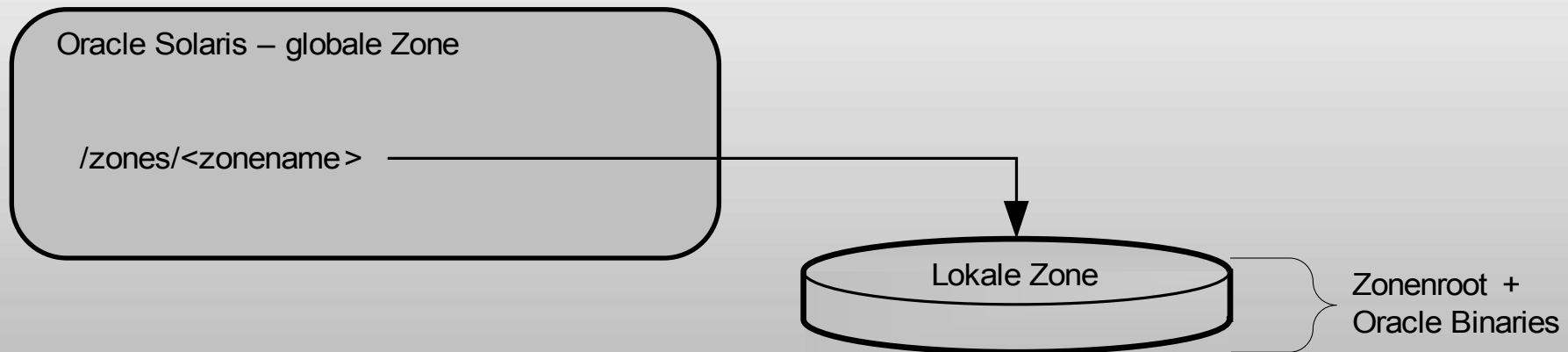
- Im Folgenden eine Zusammenfassung von Erfahrungen bei der Nutzung von Oracle Solaris Container mit Oracle Datenbanken
- Zur universellen Nutzung von Oracle Solaris Containern empfiehlt es sich, ein standardisiertes Setup zu definieren, welches ein Mindestmaß einer Containerkonfiguration abdeckt.
- Viele dieser Einstellung unterstützen sicherheitsrelevante Anforderungen an ein virtualisiertes System und eine stärkere Kapselung und Abtrennung der nicht-globalen Zone von der globalen Zone.
- Umfang der Konfiguration
 - Containerinstallation auf einem Oracle Solaris System mit aktuellen Patchlevel
 - Oracle RDBMS Installation auf aktuellen Patchlevel im Container
 - Setup konform zu OFA/Installguide
 - Vorbereitung für späteres Setup der Oracle Instance(s)
 - Strikte Trennung Solaris/Oracle RDBMS Binaries von den DB-Daten

Best Practice Nutzung von Oracle Solaris Container Basislayout

· 1. Schritt: Basis Containerinstallation

· Zonenarchitektur

- Installationsumfang Oracle Solaris OS (whole root)
- Disklayout
- Netzwerkkonfiguration



Best Practice Nutzung von Oracle Solaris Container

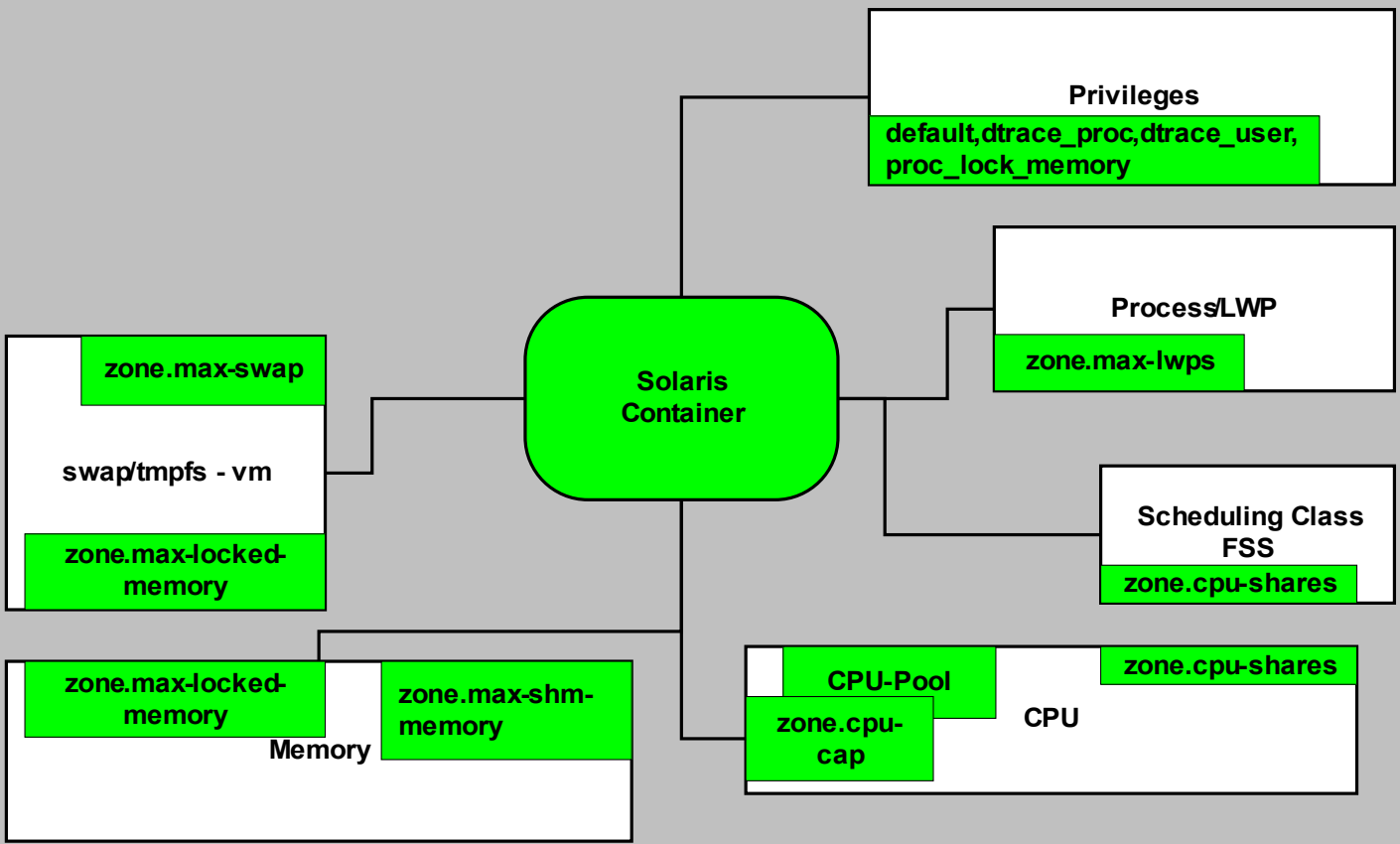
Nutzung welcher Resourcecontrols ?

· 2. Schritt: Anpassung RCTL's

- Resourcecontrols - generisch
 - Einstellung der Scheduling-Class
 - Einrichtung der CPU-Shares
 - Einrichtung des CPU-Capping
 - Limitierung der Anzahl der LWP's
 - Einrichtung erweiterter Berechtigungen zur Nutzung von dtrace in der Zone
 - Limitierung des tmpfs
 - Aktivierung des Logging von RCTL-Events in der globalen Zone
- Resourcecontrols – Oracle Kontext
 - Limitierung shared Memory
 - Limitierung locked Memory
 - Einrichtung erweiterter Berechtigungen zur Nutzung von DISM in der lokalen Zone
 - Ggf. Einrichtung eines statischen CPU-Pool
 - Oracle Lizenzmodell/„Capped Container“

Best Practice Nutzung von Oracle Solaris Container Übersicht der RCTL's

Oracle Solaris – globale Zone



Limitierung der Nutzung von Ressourcen der globalen Zone durch RCTL's.

- zone.max-swaps
- zone.max-locked-memory
- zone.max-shm-memory
- zone.max-lwps
- zone.cpu-cap
- zone.cpu-shares
- Resourcepool

Best Practice Nutzung von Oracle Solaris Container

Oracle Binaries

- 3. Schritt: Vorbereitung der Installation der Oracle Binaries
 - Anpassungen innerhalb des Container
 - Vorbereitung Oracle RDBMS-Installation gemäß Oracle Installguide
 - Verzeichnisse
 - User/Gruppen
 - RCTL 's im Container
 - /etc/project
 - Installation der Oracle RDBMS Binaries gemäß Oracle Installguide

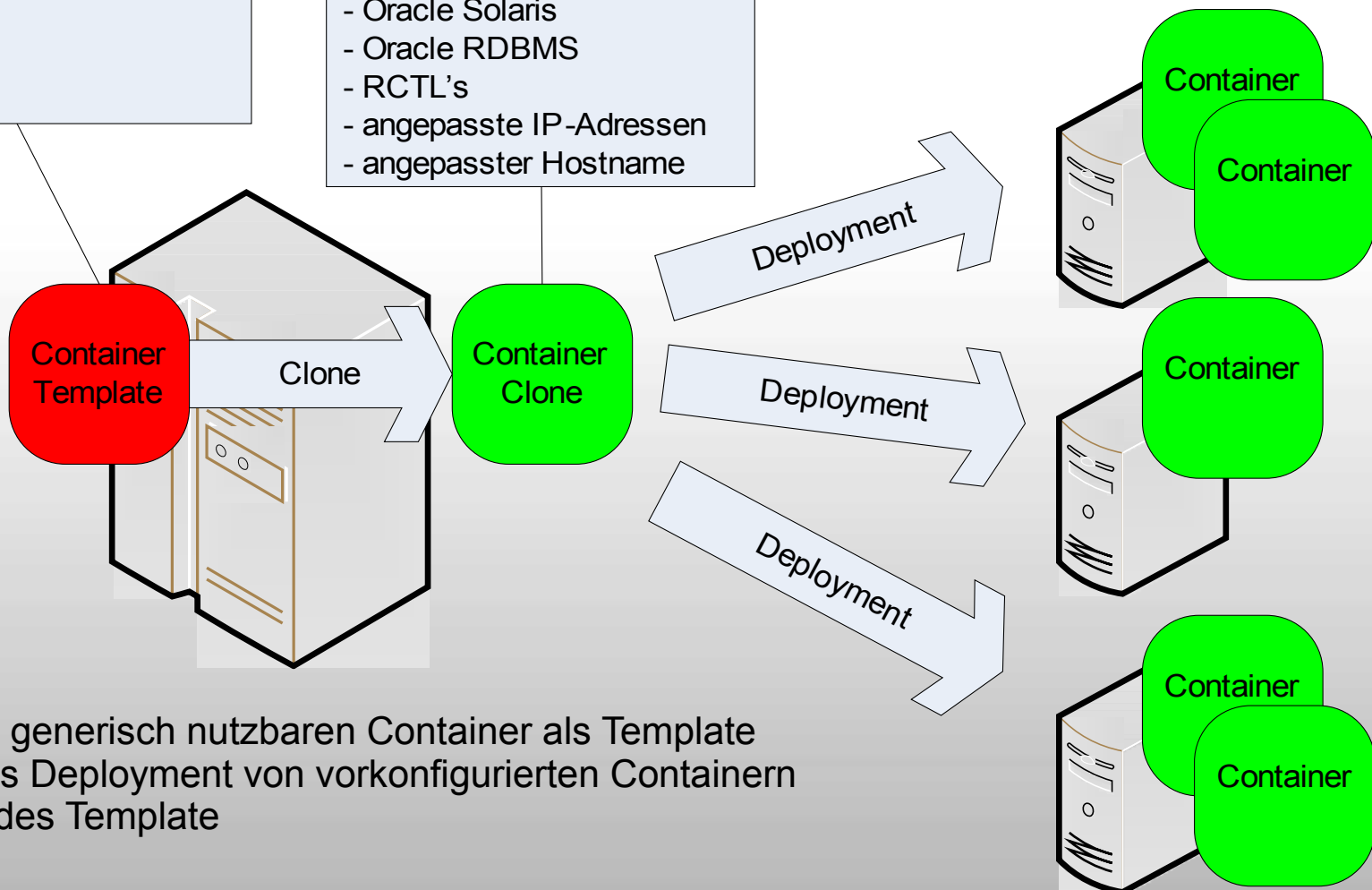
Best Practice Nutzung von Oracle Solaris Container Vorkonfigurierte Containertemplates ?

Vorkonfiguriertes Containertemplate

- Oracle Solaris
- Oracle RDBMS
- RCTL's

Clone

- Oracle Solaris
- Oracle RDBMS
- RCTL's
- angepasste IP-Adressen
- angepasster Hostname

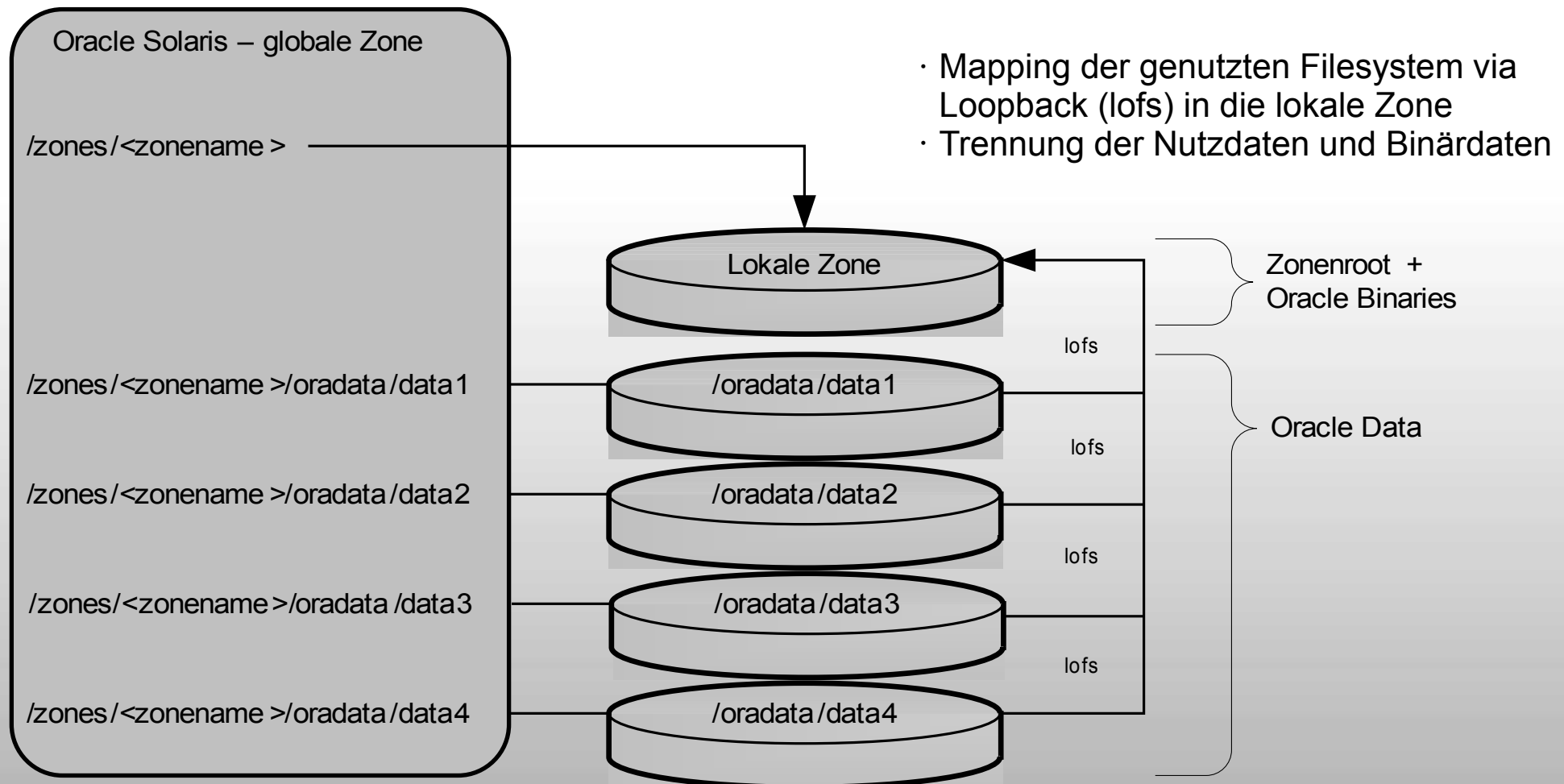


- Nutzung eines generisch nutzbaren Container als Template
- Möglichkeit des Deployment von vorkonfigurierten Containern durch Clonen des Template

Best Practice Nutzung von Oracle Solaris Container

Wie kommt die DB zu ihren Daten ?

· 4. Schritt: Bereitstellung der Filesysteme für Oracle DB



Praktische Umsetzung (1)

· ZFS Pool bereitstellen

```
#zpool create -O mountpoint=/zones zonen-zp c2t2d0
#zfs list -H
zonen-zp          82.5K    19.6G    21K      /zones
```

· CPU Pool ora-z.pool erzeugen

```
# pooladm -e
# pooladm -s
# poolcfg -c 'create pset ora-z.pset (uint pset.min=1; uint pset.max = 1) '
# poolcfg -c 'create pool ora-z.pool'
# poolcfg -c 'associate pool ora-z.pool (pset ora-z.pset) '
# pooladm -c /etc/pooladm.conf
# poolstat -p ora-z.pool

                pset
id pool          size used load
 1 ora-z.pool    1 0.00 0.00
```


Praktische Umsetzung (2)

- Konfigurationsdatei ora-z.cfg für Zone ora-z erzeugen

```
#vi ora-t.cfg
create -b
  set zonepath=/zones/ora-z
  set autoboot=false
  set pool=ora-z.pool
  set limitpriv=default,dtrace_proc,dtrace_user,proc_lock_memory
  set scheduling-class=FSS
  set ip-type=shared
add net
  set address=192.168.1.2/24
  set physical=e1000g1
end
add net
  set address=1.1.1.2/8
  set physical=e1000g0
end
add rctl
  set name=zone.max-lwps
  add value (priv=privileged,limit=4000,action=deny)
end
```

Praktische Umsetzung (3)

- Konfigurationsdatei ora-z.cfg für Zone ora-z erzeugen

```

add rctl
  set name=zone.max-shm-memory
  add value (priv=privileged,limit=1073741824,action=deny)
end
add rctl
  set name=zone.max-swap
  add value (priv=privileged,limit=1073741824,action=deny)
end
add rctl
  set name=zone.max-locked-memory
  add value (priv=privileged,limit=1073741824,action=deny)
end
add rctl
  set name=zone.cpu-shares
  add value (priv=privileged,limit=1,action=none)
end
add rctl
  set name=zone.cpu-cap
  add value (priv=privileged,limit=75,action=deny)
end
  
```

Praktische Umsetzung (4)

- Zone ora-z konfigurieren/installieren/starten und final konfigurieren

```
# zonecfg -z ora-z -f ora-z.cfg && zoneadm -z ora-z install && zoneadm -z ora-z
boot; zlogin -C ora-z

# zoneadm list -icp
0:global:running://::native:shared
3:ora-z:running:/zones/ora-z:23e398f1-157a-6171-8d9faf0ee06aa7b1:native:shared
```

- Installation Oracle Binaries in der lokale Zone

- “Business as usual” :)
- Installationsablauf entspricht einer Installation in der globalen Zone

- Bereitstellung der Filesysteme für Oracle DB

- Mountpoints in Zone ora-z erzeugen

```
# mkdir -p /oradata/data1 /oradata/data2 /oradata/data3 /oradata/data4
# chown -R oracle:oinstall /oradata
```

- ZFS Pools erzeugen/mounten

```
# zpool create -O mountpoint=/zones/ora-z/oradata/data1 data1-zp c2t1d0
# zpool create -O mountpoint=/zones/ora-z/oradata/data2 data2-zp c2t2d0
# zpool create -O mountpoint=/zones/ora-z/oradata/data3 data3-zp c2t3d0
# chown -R oracle:oinstall /zones/ora-z/oradata
```

Praktische Umsetzung (5)

- Filesysteme in Zone ora-z einbinden (Hierbei als Beispiel via loopback)

```
# vi add_fs.cfg
add fs
  set type=lofs
  set special=/zones/ora-z/oradata/data1
  set dir=/oradata/data1
end
add fs
  set type=lofs
  set special=/zones/ora-z/oradata/data2
  set dir=/oradata/data2
end
add fs
  set type=lofs
  set special=/zones/ora-z/oradata/data3
  set dir=/oradata/data3
end
```

- Konfiguration aktivieren

```
# zonecfg -z ora-z -f add_fs.cfg
# zoneadm -z ora-z boot
```

Praktische Umsetzung (6)

· Kontrolle

```
# df -h|grep oradata
data1-zp          100.0G    21K    100.0G    1% /zones/ora-z/oradata/data1
data2-zp          100.0G    21K    100.0G    1% /zones/ora-z/oradata/data2
data3-zp          100.0G    21K    100.0G    1% /zones/ora-z/oradata/data3

# zlogin ora-z "df -h|grep oradata"
/oradata/data1    100.0G    21K    100.0G    1% /oradata/data1
/oradata/data2    100.0G    21K    100.0G    1% /oradata/data2
/oradata/data3    100.0G    21K    100.0G    1% /oradata/data3
```

· Erzeugung DB-Instance in der lokalen Zone

- “Business as usual” :) (zum 2.Mal ...)
- Installationsablauf entspricht einer Installation in der globalen Zone (siehe zum 2.Mal ...)

Zusammenfassung

- Oracle Datenbanken in Oracle Solaris Zonen zu benutzen, ist nicht kompliziert
- Vorteile durch
 - Reproduzierbares Deployment
 - Clonen von konfigurierten Containern
 - Kombination mit anderen Technologien wie ZFS, Oracle Solaris Cluster
 - Einfache Migration zwischen verschiedenen Nodes

Fragen? Gern.

·Danke für Ihre Aufmerksamkeit !

- heiko.stein@etomer.com
- detlef.drewanz@oracle.com