

Sun Cluster und Solaris 10 Zonen

Heiko Stein
Senior Engineer/Architect PSD

Agenda

- Motivation
- Übersicht Varianten SunCluster + Zonen
 - > Keine Betrachtung des Zonen Cluster (ab SC 3.2u2)
- Zonen Node Cluster vs. Failover Zonen
- Wann welche Variante ?
- Trapdoors bzw. "... vorher daran denken ..."
- Einfache Setupbeispiele
- Informationsquellen & Q/A

Motivation

- Vorteile Solaris 10 Zonen

- > Einfache Handhabung
- > Lastneutral
- > Isolation/Security
- > Resource Controls
- > Kostenlos :)

+

- Vorteile Sun Cluster

- > Techn. ausgereifte HA-Lösung
- > Umfangreiches Agenten-repository
- > Integration aktueller Solaris-Features (ZFS, Zonen, RCTL, Ldom)
- > ...

=

- Weitere Vorteile

- > HA für Zonen
- > HA für "spezielle" Applikationen
- > Automatisierter Umgang/Administration von Zonen
- > Hochverfügbare Virtualisierungsarchitekturen
- > ...

Übersicht Varianten SunCluster + Zonen

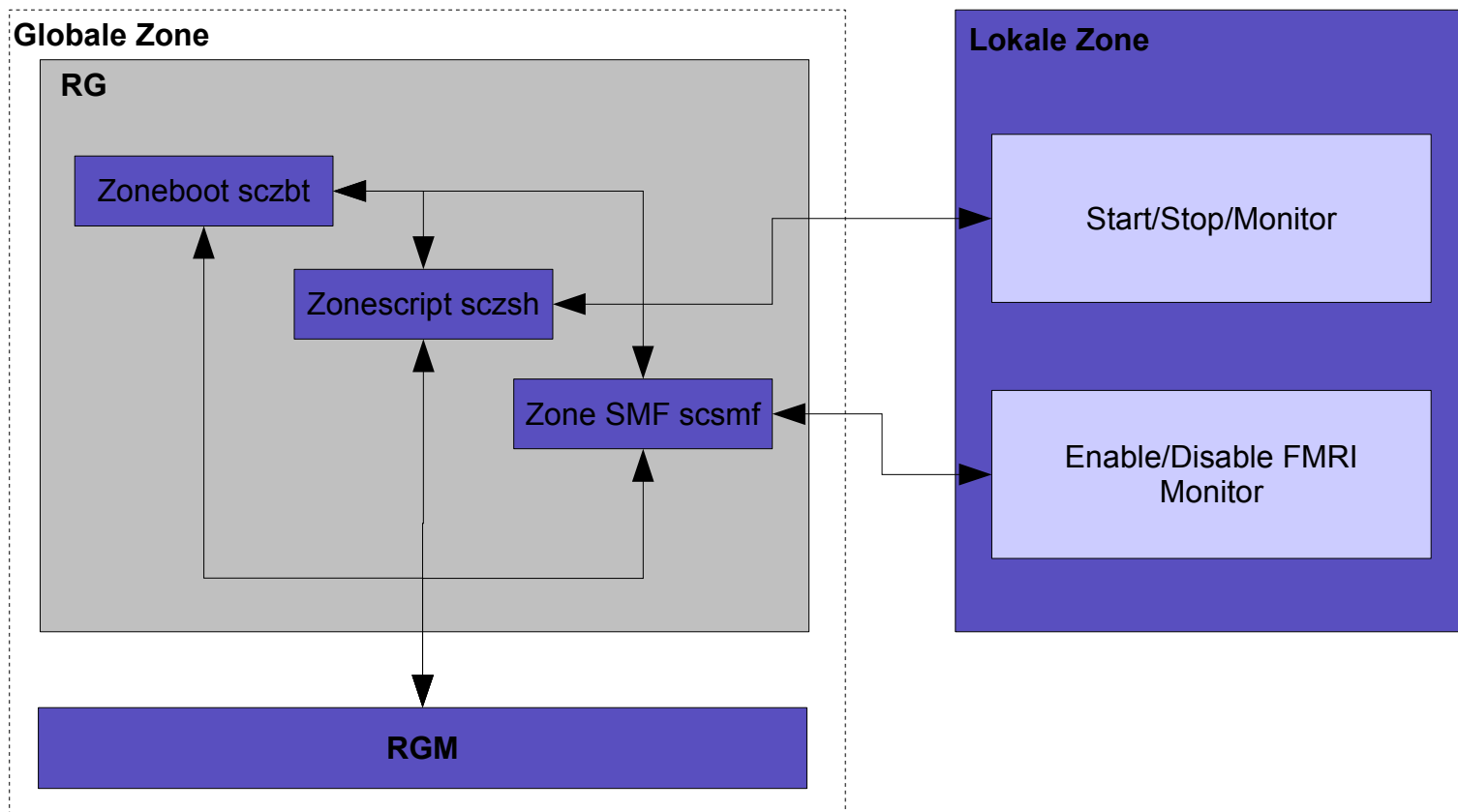
- HA-Zonen Agent (1)
 - > Verfügbar seit Sun Cluster 3.1 8/05 Sparc/X86 (in GY entwickelt)
 - > Agent basiert auf SUNW.gds
 - > Zonen werden als Clusterresource behandelt
 - > Unterstützte Konfigurationen sind:
 - > Failover
 - > Multiple Master
 - > Single Node
 - > Funktionalitäten sind Start/Stop/Monitor
 - > Script- und SMF-Schnittstelle für die lokale Zone
 - > Der Agent läuft in der globalen Zone

Übersicht Varianten SunCluster + Zonen

- HA-Zonen Agent (2)
 - > Package SUNWsczone aus 3 Komponenten
 - > Zonen-Boot Agent sczbt
 - > Zonen-Script Agent sczsh
 - > Zonen-SMF Agent sczsmf
 - > Callback via RGM/GDS
 - > Schnittstelle zur Verwaltung der IP-Adresse der Zone mittels SUNW.LogicalHostname
 - > Unterstützung von SC-Features, wie Inter-RG/Resource Dependencies, RG_Affinities, Failover capabilities

Übersicht Varianten SunCluster + Zonen

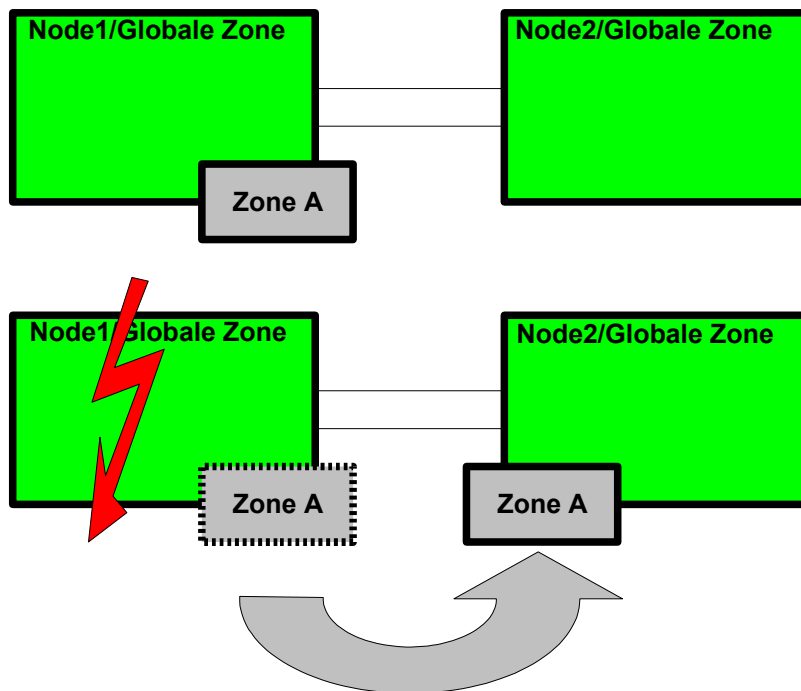
- HA-Zonen Agent (3)



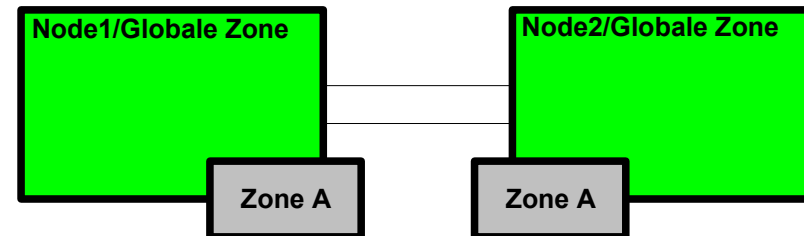
Übersicht Varianten SunCluster + Zonen

- HA-Zonen Agent (4)
 - > Architekturbeispiele

Failover-Zone



Multimaster-Zone



Übersicht Varianten SunCluster + Zonen

- SunCluster Zonen Nodes (1)
 - > Verfügbar seit Sun Cluster 3.2 Sparc/X86
 - > Lokale Zonen werden als virtuelle Nodes behandelt
 - > Ab SC 3.2u2 neues Feature “virtuelle Nodes eines Zonen Node Cluster”
 - > Zonen werden als Nodes in der Hostlist der RG geführt (z.B. Node1:zoneA,Node2:zoneB)
 - > Unterstützte Konfigurationen sind:
 - > Failover
 - > Multiple Master
 - > Scalable RG
 - > Integration in den RGM
 - > Bestandteile des SunCluster Frameworks laufen in den lokalen Zonen
 - > Ein Großteil der Standardagents laufen in Zone Node-Konfigurationen

Übersicht Varianten SunCluster + Zonen

- SunCluster Zonen Nodes (2)
 - > Verschiedene Ressourcen können in der gleichen Zone laufen und unabhängig voneinander auf andere Zonen umgeschaltet werden
 - > Koexistenz mit Failover-Zonen
 - > Kontrolle und Administration der Zonen mit Standard Solaris-Bordmitteln
 - > Failover-Services können im Single Node-Cluster konfiguriert werden
 - > Servicevirtualisierung
 - > Workloadmanagement mit RCTL (siehe auch SC Telemetry ...)

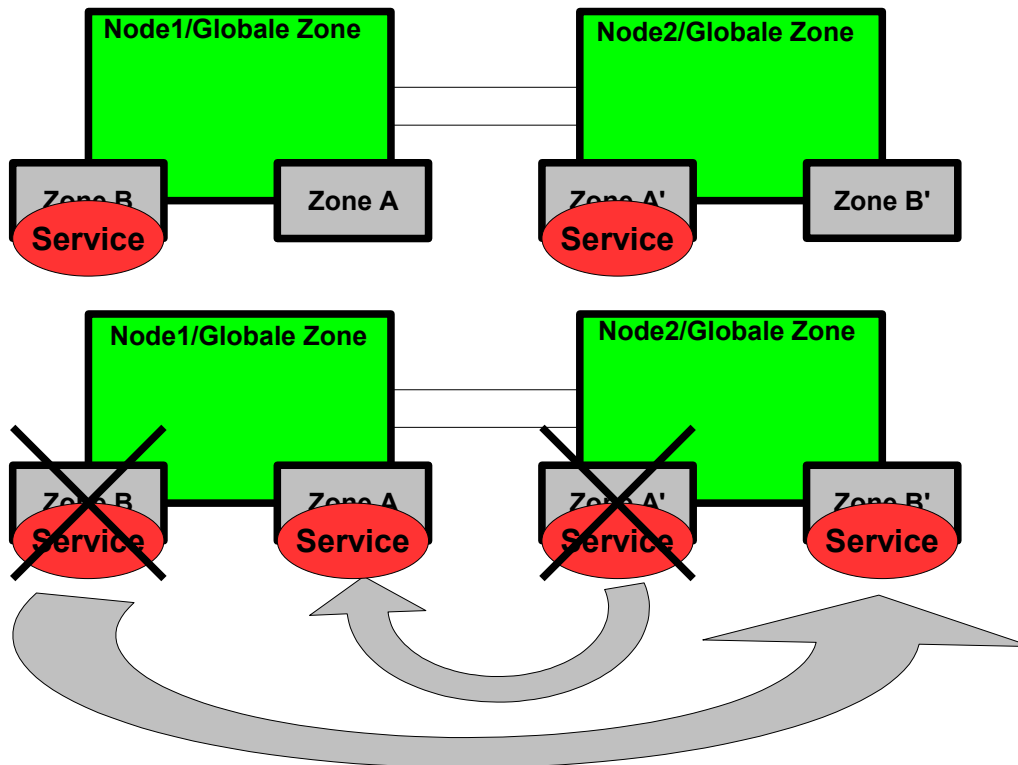
Übersicht Varianten SunCluster + Zonen

- SunCluster Zonen Nodes (3)
 - > Zonen/Prozesse werden nach Startup registriert/Shutdown deregistriert
 - > Nutzung von Solaris Contracts (ctrun ...)
- Modifizierte Membership-Engine pro Zonen-Node
 - > Alle Events werden hier empfangen
 - > Weiterleitung der Events zum globalen Membership-Manager

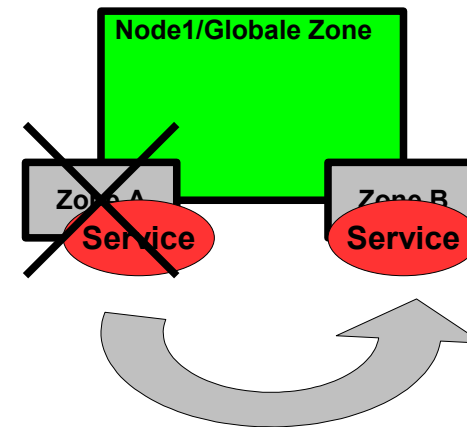
Übersicht Varianten SunCluster + Zonen

- SunCluster Zonen Nodes (4)
 - > Architekturbeispiele

Service Failover Zonen Nodes



Single Zonen Node



Zonen Node Cluster vs. Failover Zonen

| | Zone Node Cluster | Failover Zonen |
|-----------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| Resource/Resource Group | Zonen sind Member der RG-Nodelist | Zonen sind Ressourcen |
| Scalable/Multiple Masters | Support für Multiple Masters and Scalable Resources | Support für Multiple Masters Ressourcen |
| Redundante Zonen Konfiguration/Installation | Zonen müssen auf jedem Node vorhanden sein | Die Zonenkonfiguration muß auf jedem Node vorhanden sein, die Zone ist nur einmal zu installieren. (siehe Ablage Zonenroot) |
| Ablage Zonenroot | Lokal verfügbarer Storage | Shared Storage |
| Startup/Boot der Zonen | Die Zonen werden über das OS des Nodes gestartet (autoboot=true;svc:/system/zones:default) | Die Zonen werden durch die sczbt-Methode des Agenten gestartet (autoboot=false) |
| Administration/Pflege des Zoneninhaltes | Jede Zone aus der Nodelist der RG. | Nur die Failover-Zone. |
| Affinitäten zwischen RGs running in der globalen bzw. lokalen Zoneglobal and non-global zones | ab SC 3.2u2 | Ja |
| Custom Agents | Ja | sczsh/sczsmf |
| SMF services | Ja via proxy resource type. SMF-Services müssen pro Zone im Sync gehalten werden | sczsmf |

Wann welche Variante ?

- HA-Zonen Agent
 - > Applikation ist in Solaris Zonen unterstützt
 - > Blackbox-Applikationen
 - > Failover-Zeiten sind kein Kriterium
 - > Monitoring der Applikation via SMF/sh ist ausreichend
 - > Service Downtimes der Zonen ist akzeptabel
 - > Applikation bindet sich gegen HostID
 - > Applikation bindet sich gegen phys. Hostname bzw. Nutzt `gethostbyname()`, `gethostbyaddr()`
 - > zentrale/einmalige Administration nur einer Zone
 - > Keine zwingende Nutzung von Standard-Applikationsagenten

Wann welche Variante ?

- Zonen Node Cluster
 - > Applikation ist in Solaris Zonen unterstützt
 - > Failover-Zeiten sind ein Kriterium
 - > Hohe Granularität der RG
 - > Servicevirtualisierung
 - > Unterschiedlich leistungsfähige Clusternodes
 - > Nutzung von Standardagenten
 - > Migration von herkömmlichen HA-Cluster in Zonen Node Cluster
 - > Keine Verzahnung der Applikation mit der Zone

Trapdoors bzw. “... vorher daran denken ...”

- Planung Patchmanagement/Upgrade OS
 - > LU mit Zonen (ab Sol. 10u5)
- Planung RCTL (FSS, Capping, SHM, Swap, ...)
 - > Siehe # rctladm | grep zone
- Setup Zonen mit per default registrierten RCTL's bzw. relevanten Limitierungen/Einstellungen
 - > zone.max-lwps/zone.max-swap
 - > Privilegien zur Nutzung dtrace
- Planung einer zentralen Softwareverteilung/ Administration bei Zonen Node Cluster
- Standardisiertes Setup der Zonen
- Trennung Applikation/Konfigdaten von der Zonenroot

Trapdoors bzw. “... vorher daran denken ...”

- Backupmethode Zonen
- Security/Hardening Zonen
 - > SST (aka JASS)
- Nutzung von Bart
- Netzwerkanbindung Zonen
 - > Shared IP
 - > Exklusiv IP
 - > ...
- Bei der Nutzung von ZFS, Planung Snapshots der Zonenroot etc.
- Berechnung Speicherauslastung/Limitierungen im “worst case”

Trapdoors bzw. “... vorher daran denken ...”

- Bei Nutzung rcapd - Planung/Berücksichtigung des Umganges mit Memory-Capping
 - > rcapd → Userland

Einfache Setupbeispiele

- Failover Zone Annahme:
 - > Sol.10/SC 3.2x
 - > SUNWsczone installiert
 - > Node1/Node2
 - > Zone appl, Zonenpath=/zonen/appl
 - > Shared Zpool zone_zp, Mountpoint=/zonen
 - > IP-Adresse wird in der Zone verwaltet (/etc/zones/appl.xml)
 - > RG zone_rg

Einfache Setupbeispiele

- Failover Zone (1)

Registrierung benötigter Typen

```
# clrt register SUNW.HAStoragePlus
```

```
# clrt register SUNW.gds
```

Setup RG

```
#clrg create -n node1,node2 zone_rg
```

```
# clrg manage zone_rg
```

Setup Storage-RS

```
# clrs create -g zone_rg -t SUNW.HAStoragePlus -x Zpools=zone_zp zone_stor_rs
```

```
# clrs enable zone_stor_rs
```

Einfache Setupbeispiele

- Failover Zone (2)

Setup Zonen-Agent

```
# cd /opt/SUNWsczone/sczbt/util
```

```
# cp sczbt_config appl_config
```

```
# cp sczbt_register appl_register
```

```
# vi appl_config
```

...

```
RS=appl_rs
```

```
RG=zone_rg
```

```
PARAMETERDIR=/zonen/appl
```

```
HAS_RS=zone_stor_rs
```

```
Zonename=zone1
```

...

Einfache Setupbeispiele

- Failover Zone (3)

Setup Zonen-Agent

```
# vi appl_register
```

```
. `dirname $0`/appl_config
```

```
...
```

```
# ./appl_register
```

```
# clrs enable appl_rs
```

Einfache Setupbeispiele

- Zonen Node Cluster Annahme:
 - > Sol.10/SC 3.2x
 - > Clusternodes node1/node2
 - > Zone appl, Zonenpath=/zonen/appl
 - > Lokaler Zpool zone_zp, Mountpoint=/zonen
 - > Shared Zpool appl_zp, Mountpoint(in Zone)=/appl
 - > log.Host appl_ip, IPMP-Gruppe appl_ipmp auf beiden Nodes
 - > RG appl_rg

Einfache Setupbeispiele

- Zonen Node Cluster (1)

Registrierung benötigter Types

```
# clrt register SUNW.HAStoragePlus
```

```
# clrt register SUNW.gds
```

Setup RG

```
#clrg create -n node1:appl,node2:appl appl_rg
```

```
# clrg manage appl_rg
```

Setup Storage-RS

```
# clrs create -g appl_rg -t SUNW.HAStoragePlus -x Zpools=appl_zp appl_stor_rs
```

```
# clrs enable appl_stor_rs
```

Einfache Setupbeispiele

- Zonen Node Cluster (2)

Setup logischer Host

```
# clreslogicalhostname create -g appl_rg -h appl_ip -N  
appl_ipmp@node1,appl_ipmp@node2 appl-ip-rs
```

Informationsquellen & Q/A

- <http://docs.sun.com/app/docs/coll/1531.2?l=en&q=whats+new>
- <http://docs.sun.com/app/docs/coll/1124.7?l=en&q=sun+cluster+3.2>
- <http://opensolaris.org/os/community/ha-clusters/>
- <http://blogs.sun.com/main/tags/cluster>

Fragen ?

Heiko Stein
Senior Engineer/Architect PSD
heiko.stein@sun.com