

Flying Container mit Oracle Solaris Container und Oracle Solaris Cluster

**Hartmut Streppel
ORACLE Deutschland B.V. & Co. KG
München**

Schlüsselworte:

Oracle Solaris Zones, Oracle Solaris Container, Failover Zones, Flying Container, Oracle Solaris Cluster

Einleitung

Solaris Container sind die Standardmethode, um Anwendungen in einer Solaris Umgebung zu virtualisieren und zu konsolidieren. Die Möglichkeit, auf einfache Weise Ressourcen gemeinsam zu nutzen, zu kontrollieren und administrative Rechte zu delegieren und damit sichere und gekapselte Ablaufumgebungen zu implementieren, haben zu einer breiten Akzeptanz bei Solaris Nutzern geführt. Einer der wesentlichen Vorteile von Containern ist die Möglichkeit, sie zwischen Solaris-Instanzen hin und her schieben zu können.

Werden Container unter der Kontrolle von Oracle Solaris Cluster genutzt und damit hochverfügbar gemacht, sind keinerlei Änderungen im Container und an den dort laufenden Anwendungen notwendig. Der Anwender und der Administrator eines Containers merken keinen Unterschied. Ein Cluster-Agent (HA Container) überwacht die Funktionsfähigkeit des Containers und ist im Fehlerfall in der Lage, ihn entweder neu zu starten oder auf einen anderen Clusterknoten zu schwenken.

Im Folgenden werden einige Grundkonzepte von Containern, ihre Integration in Oracle Solaris Cluster und Anwendungsszenarien bei Kunden erläutert.

Oracle Solaris Container als Virtualisierungsmethode

Virtualisierung wird heutzutage meist mit virtuellen Maschinen (VM) gleichgesetzt. Ein Hypervisor stellt eine virtuelle Hardware zur Verfügung, auf der dann klassische Betriebssysteme aufsetzen. Diese Methode hat wie alle anderen auch Nachteile und Vorteile. Der wesentliche Vorteil ist, dass die als VM installierten Betriebssystemumgebungen fast genauso betrieben werden können wie "normale" Betriebssysteme, die direkt auf der Hardware aufsetzen. Der wesentliche Nachteil ist, dass die Anzahl der zu administrierenden Betriebssysteminstanzen trotz Konsolidierung nicht schrumpft, sondern sogar noch um den Hypervisor wächst (für den meist auch noch zusätzliche Lizenzgebühren zu zahlen sind). Eng verbunden mit diesem Aspekt ist die Tatsache, dass es in einer solchen mit virtuellen Maschinen virtualisierten Umgebung keinerlei Anreiz gibt, mehrere Anwendungen auf einer Betriebssysteminstanz zu konsolidieren.

Oracle Solaris Container sind keine virtuellen Maschinen, sondern stellen eine Art virtuelle Betriebssystemumgebung zur Verfügung. Container „gaukeln“ also einer Anwendung die Existenz eines vollständigen Betriebssystems vor. Ebenso fühlt sich ein Administrator in einem Container so, als beherrsche er/sie die gesamte Maschine, was mitnichten der Fall ist. Die Rechte des Administrators in einem Container sind so weit eingeschränkt, dass sie nur innerhalb des Containers gelten. Kommunikation über die Grenzen des Containers hinweg sind nur möglich über das Netz, das mit einer Firewall abgesichert werden kann.

Ressourcen können soweit eingeschränkt werden, dass dem Container genau die zur Verfügung stehen, die die Anwendungen, die in ihm laufen, brauchen.

Auf Grund der - verglichen mit einem Hypervisor - minimalen Virtualisierungsschicht ist der Overhead beim Betrieb von Containern (es werden offiziell bis zu 8192 unterstützt) minimal. Upgrades und Patches macht man einmal pro Solaris Instanz; dabei werden alle Container mit hochgezogen oder gepatcht. Wo das ein Problem ist, z.B. in hoch konsolidierten Umgebungen, in denen man sich auf keine gemeinsame Betriebsunterbrechung einigen kann, gibt es andere Verfahren, die ein individuelles Patchen ermöglichen.

Flying Container - auch Failover Container genannt

Der Begriff „Fliegender Container“ entstand im Feld, nicht in der Entwicklung. Kunden erkannten sofort die Möglichkeit, Container als flexible Einheiten zu nutzen, in denen man komplette Anwendungsumgebungen betreiben und zwischen Rechnern hin- und herschieben konnte. Und obwohl die ersten Solaris 10 Updates ein Verschieben offiziell gar nicht vorsahen, konnte man mit den Mitteln des Basisbetriebssystems Container „fliegen“ lassen. In der gleichen Zeit entstand die initiale Sun Cluster Unterstützung für verschiebbare Container, der sog. HA Container Agent. Dieser nutzte die in Solaris vorhandenen Möglichkeiten, um einen Container als eine „black box“ zu verwalten, eine Einheit, die durch eine Cluster Ressource implementiert wurde, und bei der nur die „box“ von Interesse ist, aber nicht der Inhalt.

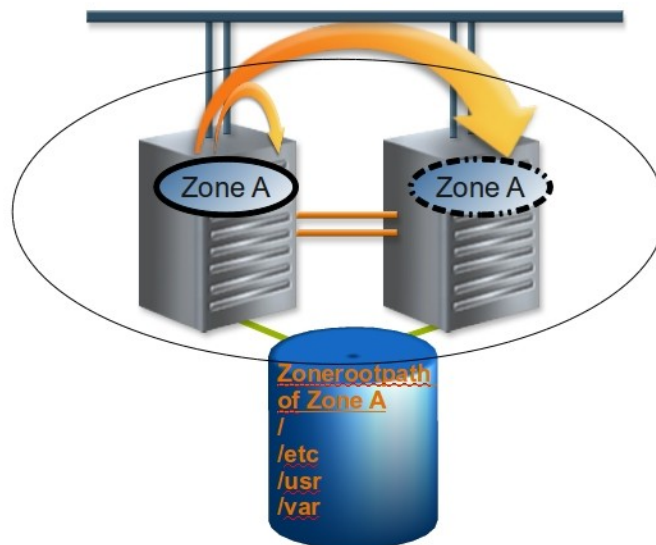


Abb. 1: Flying Container im Oracle Solaris Cluster

Auf dem Bild sieht man deutlich, dass der Zonenpfad, der das Root-Dateisystem der Zone enthält auf „shared Storage“ konfiguriert wurde, so dass beide Clusterknoten auf ihn zugreifen können.

HA Container Agent

Ein Flying Container ist also ein Oracle Solaris Container, der auf shared Storage konfiguriert wurde und manuell oder automatisch durch Oracle Solaris Cluster zwischen Solaris Systemen hin- und

hergeschoben werden kann. Der sog. Zonenpfad (zonepath) der Containerkonfiguration muss im SAN liegen, damit alle Clusterknoten in der Lage sind, diesen Container auch zu booten.

Es gibt zwei alternative Modelle, die hier nicht weiter erklärt werden sollen:

- multi Master Container, bei denen mehrere identische Container aktiv sind und
- mehrere lokale Container, die alternativ gestartet werden.

Der HA Container Agent setzt die Existenz der Containerkonfiguration auf allen Clusterknoten, die potentiell diesen Container betreiben können, voraus. Die Containerkonfiguration muss den Container im Status „installed“ zeigen. Das Starten und Stoppen eines Container geschieht mittels ausgefeilter Start- und Stopp-Methoden, die Fehlersituationen erkennen und in den meisten Fällen auch beheben können. Die Überwachung eines Containers geschieht durch das Monitoring des Schedulerprozesses, der für den Container zuständig ist und zusätzliche Maßnahmen.

In Fehlerfällen wird der Agent in Abhängigkeit bestimmter Cluster-Parameter zunächst einen Neustart eines Containers versuchen. Danach und in den Fällen, in denen ein Neustart sinnlos oder unmöglich ist, wird ein Container geschwenkt. Dazu wird, falls noch möglich, der Container angehalten, die IP-Adresse gestoppt, und der zum Container gehörende Storage freigegeben. Auf dem Zielknoten wird der Storage importiert, die IP-Adresse gestartet und dann der Container gestartet. Die Container-Verwaltung wird mit den Standard Zonen-Kommandos durchgeführt.

Mit einem Solaris 10 Update wurde ein Verfahren implementiert, Zonen von einer Solaris-Instanz abzuhängen und an eine andere Solaris-Instanz wieder anzuhängen. Dies wurde mit den „zoneadm“ Unterkommandos „detach“ und „attach“ realisiert. Dieser Prozess stellte sicher, dass Quell- und Zielsystem auf demselben Softwarestand waren. War dies nicht der Fall, was beim „attach“ festgestellt wurde, konnte ein Container auch nicht angehängt werden.

Es ist wichtig zu verstehen, dass der HA Container Agent diesen Mechanismus nicht nutzt. Da es eine Voraussetzung der Nutzung dieses Agenten ist, dass die Clusterknoten auf demselben Softwarestand sind, ist diese zusätzliche Überprüfung auch nicht notwendig.

Es ist offensichtlich, dass, sollte ein automatischer Start und Stopp von Anwendungen im Container gewünscht sein, dies über eine Integration in das SMF Subsystem implementiert werden sollte.

Erweiterungen des HA Container Agenten

Der „black box“ Ansatz hat neben vielen Vorteilen auch einen Nachteil: eine Anwendungsüberwachung durch die Standard Agenten von Oracle Solaris Cluster ist nicht möglich. Um nun doch eine Integration über eine einfache Shellskript-, bzw. SMF-Schnittstelle zu ermöglichen, wurden zwei weitere Agenten entwickelt, die allerdings beide abhängig vom „Haupt“-Container-Agenten sind. Der eine dieser Agenten implementiert eine Schnittstelle, die es ermöglicht, in einem Container Shell-Skripte zum Starten, Stoppen und Überwachen von Anwendungen laufen zu lassen, und auf die Rückgabewerte zu reagieren, und der andere, um gleiches mit SMF-Methoden zu tun.

Alternativen zu Fliegenden Containern

Nicht immer steht ein SAN zur Verfügung, auf dem Container angelegt und innerhalb eines Clusterverbands verschoben werden können. In diesem Fall kann ein kompletter Container auch über das Netz zu einem anderen Rechner kopiert, dort die Zone und die notwendigen Devices angelegt, und der Container gestartet werden. Dazu gibt es ein sehr gut dokumentiertes Papier (s.a. Literaturverzeichnis).

Die Alternative zum komfortablen Verschieben von Containern mit Hilfe von Oracle Solaris Cluster ist ein manuelles Verschieben. Dazu sind aber mehrere manuelle Schritte notwendig, die das Verfahren etwas aufwändiger machen. Es ist aber offensichtlich, dass es bei diesem Verfahren keinerlei Automatismen und damit auch keine Hochverfügbarkeit gibt.

Anwendungsfälle

Es ist heute, nachdem Solaris 10 mit seinen Containern viele Jahre am Markt ist, eine anerkannte und empfohlene Praxis, Anwendungen grundsätzlich in Containern zu installieren. Die Vorteile sind:

- sichere Kapselung der Anwendung und ihrer Anwender
- weitere Sicherheitsbeschränkungen einfach implementierbar
- vollständiges Ressource Management möglich
- Konsolidierung
- Hochverfügbarkeit.

Für typische Anwendungen ändert sich beim Ausrollen in einen Container nichts. Administratoren merken keinen Unterschied, auch nicht, wenn ihr Container unter Clusterkontrolle läuft.

Einige große Oracle Solaris Anwender haben ihre Oracle Solaris Umgebungen fast vollständig auf Container umgestellt. Dort bekommen interne Kunden für ihre Anwendungen standardisierte Container mit den notwendigen Ressourcen zur Verfügung gestellt. Die Administratoren dieser Container haben alle administrativen Rechte innerhalb eines Containers. Nur für den Fall, dass Änderungen an der Container-Konfiguration durchgeführt werden müssen, z.B. Hinzufügen neuer Ressourcen, ist eine Eingriff der Infrastruktur Administration notwendig.

Die Anwendungscontainer laufen zunächst noch nicht in der Produktionsumgebung. Ist die Installation und Konfiguration der Anwendung getestet und damit freigegeben, kann der Container durch Umkonfiguration im SAN oder auch durch Kopieren in einen Container auf einem Produktionssystem einfach in die Produktion eingehängt werden. Dort wird er mit wenigen Kommandos auch in die Oracle Solaris Cluster Umgebung integriert. Die Produktionsumgebungen sind Oracle Solaris Cluster, die typischerweise aus mehr als 4 Knoten bestehen und mit einem ausgefeilten Management überwacht werden. Dabei wird das Anwendungsmanagement strikt vom Infrastrukturmanagement getrennt, was häufig auch der Organisations-Struktur der Systemverwaltung entspricht.

Beispiel: Integration eines Oracle Solaris Containers in Oracle Solaris Cluster

Die Integration eines Containers in das Oracle Solaris Cluster Framework erfolgt normalerweise in 6 Schritten. Um ein Gefühl zu vermitteln, wie dieser Prozess auf der Kommandozeile aussieht, sollen die 6 Schritte als Beispiel aufgeführt werden:

1. Anlegen einer neuen Ressource Gruppe (RG)

```
clrg create ora_zone-rg
```

Hier wird eine leere RG mit dem Namen ora_zone-rg angelegt.

2. Anlegen einer Ressource für eine logische IP Adresse mit dem Namen hazone

```
clrslh create -g ora_zone-rg hazone
```

3. Anlegen einer Ressource für eine oder mehrerer Storage-Ressourcen

```
clrs create -t HAStoragePlus -g ora_zone-rg -p \  
zools=ora_rpool,ora_dpool ora_zone-hasp
```

Dieses Kommando erzeugt eine Storage Ressource, die die Zpools ora_rpool und ora_dpool überwacht. Sie sorgt vor allem dafür, dass diese verfügbar sind, wenn abhängige Ressourcen, z.B. die Container Ressource, gestartet werden.

4. Starten der Ressource Gruppe

```
clrg online -M ora_zone-rg
```

Das Starten der Ressourcegruppe bewirkt, dass die IP-Adresse online ist, und die beiden Zpools importiert werden.

5. Setzen der Eigenschaften für die Container Ressource und Anlegen der Container Ressource.

Die Container Ressource benötigt eine Reihe von Informationen, die sinnvollerweise in eine vordefinierte Konfigurationsdatei, die sich unter /opt/SUNWsczone/sczbt/util befindet, eingetragen werden. Danach wird diese Datei von einem Registrierungsskript, das sich im selben Verzeichnis befindet, eingelesen und mit diesen Informationen die Container Ressource angelegt. Einige dieser zu setzenden Parameter sind:

```
RS=ora_zone-rs
RG=ora_zone-rg
PARAMETERDIR=/zones/ora_zone
SC_NETWORK=true
SC_LH=hazone
FAILOVER=true
HAS_RS=ora_zone-hasp
Zonename="ora_zone"
```

Die eigentliche Registrierung geschieht mit dem Skript sczbt_register.

```
./sczbt_register -f ./sczbt_config-ora_zone
```

Dieses Skript überprüft die angegebenen Parameter und legt mit dem „clrs create“ Kommando die Container-Ressource an. Der Parameter Zonename stellt die Verbindung zum existierenden Container her.

6. Starten der Container Ressource, die dann den Container startet

```
clrs enable ora_zone-rs
```

Das enable Unterkommando startet schließlich den Container. Alle diese Kommandos werden in der globalen Zone abgesetzt. Innerhalb des Container ist nicht ersichtlich, dass sich diese Zone unter Clusterkontrolle befindet.

Zusammenfassung

Oracle Solaris Container sind die wichtigste Virtualisierungsmethode in Oracle Solaris. Sehr viele Oracle Solaris Anwender nutzen diese Technologie, um große, bisher getrennt laufende Anwendungsumgebungen unter einer Solaris Instanz zu konsolidieren. Die Integration von Containern mit Oracle Solaris Cluster und dem HA Container Agenten bringt zusätzlich Hochverfügbarkeit für die Container, ohne dass der Container oder an die im Container laufenden Anwendungen modifiziert werden müssen. Die Kombination von Oracle Solaris Cluster mit Oracle Solaris Containern ist die effizienteste Konsolidierungsmethode auf Solaris, die gleichzeitig auch für Hochverfügbarkeit der Anwendungen in den Containern sorgt.

Weiterführende Informationen

- Containerleitfaden - http://blogs.oracle.com/solarium/entry/neue_version_container_leitfaden_3
- System Administration Guide: Oracle Solaris Containers-Resource Management and Oracle Solaris Zones - http://download.oracle.com/docs/cd/E18752_01/html/817-1592/
- Oracle Solaris Cluster Data Service for Solaris Containers Guide - http://download.oracle.com/docs/cd/E18728_01/html/821-2677/

- How to move an Oracle Solaris Container –
http://developers.sun.com/solaris/docs/howto_movesolariscontainer_051410-1.pdf

Kontaktadresse:

Hartmut Streppel
ORACLE Deutschland B.V. & Co. KG
Riesstrasse 25
D-80992 München

Telefon: +49 (0) 89-1430 2588
Fax: +49 (0) 89-1430 1150
E-Mail: Hartmut.Streppel@oracle.com
Internet: www.oracle.com