

Kostenoptimierte Cloud-Administration mit Solaris 10 Container Technologie

Eine TCO Berechnungsmodell für virtualisierte Infrastrukturen

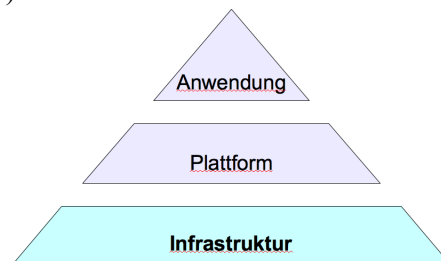
Joachim Dietsch
ORACLE Deutschland B.V. & Co. KG
Frankfurt

Schlüsselworte:

Virtualisierung, Solaris Container Technologie, Cloud Computing, Cloud Infrastruktur, Cloud Technologie, Platform as a Service (PaaS), Infrastructure as a Service (IaaS), Hypervisor, VMware, Layer-2 Hypervisor, VirtualBox, Domains, Architekturen, Administration, Betriebskosten, Verrechnungsmodell, TCO, ROI.

Einleitung

Virtualisierung ist die entscheidende Basis-Technologie für alle Cloud-Services und in der Ebene Infrastructure as a Service (IaaS) beheimatet.



Bei der Gesamtkostenbetrachtung von IT-Services wird in zwei Kategorien unterschieden, welche insbesondere bei größeren, börsennotierten Unternehmen bilanztechnisch sehr wichtig sind:

- CAPEX: CAPital EXpenditure = Investitionsausgaben für längerfristige Anlagegüter.
- OPEX: OPERational EXpenditure = Betriebskosten für den operativen Geschäftsbetrieb.

Mit der neuesten Generation an extrem leistungsfähigen Servern liegt das Verhältnis CAPEX zu OPEX bei 30 zu 70, d.h. 70% der Gesamtkosten liegen heute im operativen Bereich.

Bei dem zunehmenden Kostendruck auf die IT-Bereiche in Unternehmen oder auch gegenüber externen Dienstleistern gilt es, diesen Kostenblock der Administration zu optimieren.

Insbesondere im x86 Server-Umfeld hat sich die Hypervisor Technologie von unterschiedlichen Anbietern etabliert. Mit der Implementierung von Solaris 10 Container wurde eine neue Form der Virtualisierung erzielt, welches innerhalb eines einzelnen Betriebssystems mit Hilfe von s.g. Resource-Gruppen arbeitet.

Im nachfolgenden werden die unterschiedlichen Virtualisierungs-Technologien beschrieben, deren Vor- und Nachteile gegenüber gestellt sowie die daraus resultierenden Administrationsverfahren beschrieben und hierfür passende Berechnungsmodelle abgeleitet.

Übersicht Virtualisierungs-Technologien

Domains/physikalische Partitionen

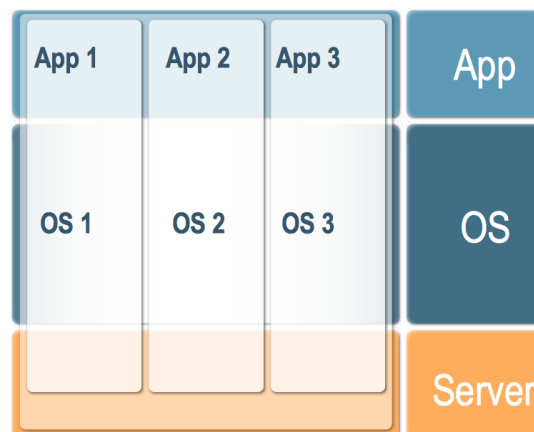
Bei dieser Form der Virtualisierung kann ein Rechner durch Konfiguration in Teil-Rechner (Domain, Partition) zerlegt werden. Die Domains sind fast vollständig physikalisch getrennt, da die elektrischen Verbindungen abgeschaltet werden. Gemeinsam benutzte Teile sind entweder sehr ausfallsicher (Gehäuse) oder redundant aufgebaut (Service-Prozessor, Netzteile).

Vorteile:

- Separation: Die Applikationen sind gut voneinander separiert, ein gegenseitiger Einfluss über das OS oder ausgefallene gemeinsame Hardware ist nicht möglich.
- Applikation: Alle Applikationen sind lauffähig, sofern sie im Basis-Betriebssystem lauffähig sind.
- Skalierbarkeit: Die Leistung einer Virtualisierungsinstanz (hier eine Domain) lässt sich bei einigen Implementierungen im laufenden Betrieb ändern (Dynamic Reconfiguration), indem HW-Ressourcen zwischen Domains verschoben werden.
- HW-Maintenance: Bei Ausfall einer Komponente kann bei entsprechender Auslegung der Domain die Applikation trotzdem weiter betrieben werden. Reparaturen sind durch Dynamic Reconfiguration im laufenden Betrieb möglich (bei redundanter Auslegung). Nur um Total-Ausfälle abzufangen, muss noch ein Cluster eingerichtet werden (Stromversorgung, Brand des Gehäuses, Ausfall des RZ, Software-Fehler).
- OS-Versionen: Unterschiedliche Betriebssysteme/Versionen sind möglich.

Nachteile:

- OS-Maintenance: Jede Maschine muss separat administriert werden. OS-Installation, Patches und die Umsetzung Firmen-interner Standards müssen für jede Maschine separat erfolgen.
- Delegation: Die für die Applikation / den Service zuständige Abteilung braucht für einen Teil der Ablaufsteuerung Root-Rechte oder muss mit dem Rechner-Betrieb bezüglich Änderungen kommunizieren. Das kann die Sicherheit beeinträchtigen, aufwändig werden bzw. länger dauern.
- Overhead: Jede Maschine hat den Overhead eines eigenen Betriebssystems.



Oracle bietet Domains in den High-End Servern E20K, E25K, den Midrange Servern E2900, E4900, E6900 und den SPARC Enterprise Servern M4000, M5000, M8000 und M9000 an.

Diese Virtualisierungstechnik wird von einigen Herstellern zur Verfügung gestellt (Sun Dynamic System Domains, Fujitsu Partitions, HP nPars). HW-Support und (wenig) OS-Support sind notwendig.

Logische Partitionen

Auf der Hardware eines Rechners läuft ein minimales Betriebssystem, der Hypervisor, welcher die Schnittstelle zwischen Hardware und OS des Rechners virtualisiert. Auf den entstehenden sogenannten virtuellen Maschinen lässt sich jeweils ein eigenes Betriebssystem (Gast-Betriebssystem) installieren. Bei einigen Implementierungen läuft der Hypervisor auch als normales Applikationsprogramm, was jedoch erhöhten Overhead bedeutet (sog. Layer-2 Hypervisor).

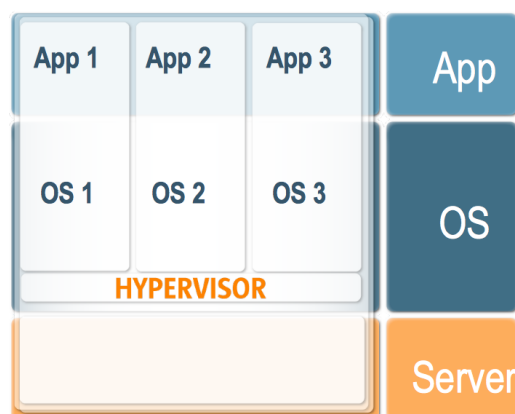
In der Regel werden durch Emulation aus den realen Devices virtuelle Devices erzeugt; reale und virtuelle Devices werden den logischen Partitionen durch Konfiguration zugeteilt.

Vorteile:

- Applikation: Alle Applikationen des Gast-Betriebssystems sind lauffähig.
- Skalierbarkeit: Die Leistung einer logischen Partition lässt sich teilweise im laufenden Betrieb ändern, sofern das OS und der Hypervisor es zulassen.
- Separation: Die Applikationen sind voneinander separiert, ein direkter gegenseitiger Einfluss über das OS ist nicht möglich.
- OS-Versionen: Unterschiedliche Betriebssysteme/Versionen sind möglich.

Nachteile:

- HW-Maintenance: Bei Ausfall einer gemeinsam genutzten Komponente sind ggf. viele oder alle logischen Partitionen betroffen. Es wird jedoch versucht, durch vorbeugende Analyse Anzeichen eines zukünftigen Ausfalls zu erkennen, um vorab die Fehler auszugrenzen.
- Separation: Die Applikationen können sich über gemeinsam genutzte Hardware gegenseitig beeinflussen. Ein Beispiel ist hier das virtuelle Netzwerk, da der Hypervisor einen Switch emulieren muss. Auch die virtuellen Platten, die zusammen auf einer realen Platte liegen und dort gegenseitig um den Schreib-/Lese-Kopf „konkurrieren“, sind ein Beispiel für dieses Verhalten. Um dies zu vermeiden, kann man reale Netzwerk-Interfaces und/oder dedizierte Platten verwenden, was den Aufwand zur Nutzung von logischen Partitionen allerdings erhöht.
- OS-Maintenance: Jede Partition muss separat administriert werden. OS-Installation, Patches und die Umsetzung Firmen-interner Standards müssen für jede Partition separat erfolgen.
- Delegation: Wenn die für die Applikation / den Service zuständige Abteilung Root-Rechte benötigt, dann sind alle Aspekte des Betriebssystems in der logischen Partition administrierbar. Das kann die Sicherheit beeinträchtigen.
- Overhead: Jede logische Partition hat den Overhead eines eigenen Betriebssystems, insbesondere wird der Hauptspeicherbedarf der einzelnen Systeme beibehalten.



Zu logischer Partitionierung gehören das IBM VM-Betriebssystem, IBM LPARs auf z/OS und AIX, HP vPars, sowie VMware und XEN. Oracle bietet die Logical Domains an (SPARC: seit Solaris 10 11/06), Oracle VM Server sowie Oracle VM VirtualBox (x86- und x64-Architekturen).

Container (Solaris Zonen) in einem OS

In einer Betriebssystem-Installation werden hier voneinander unabhängige Ausführungsumgebungen für Anwendungen und Dienste geschaffen. Der Kernel wird mandantenfähig: er existiert nur einmal, erscheint aber jeder Zone als wäre er exklusiv zugeordnet.

Die Separation wird implementiert, indem der Zugriff auf Ressourcen eingeschränkt wird, wie z.B. die Sichtbarkeit der Prozesse (modifiziertes `procfs`), Nutzbarkeit der Devices (modifiziertes `devfs`), die Sichtbarkeit des Dateibaumes (wie bei `chroot`).

Vorteile:

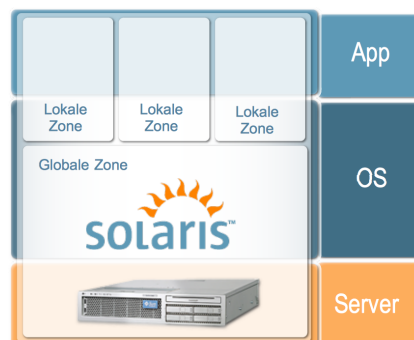
- Applikation: Alle Applikationen sind lauffähig, sofern sie keine eigenen Treiber benutzen oder sonst systemnahe Features nutzen. Eigene Treiber sind jedoch über Installation in der globalen Zone nutzbar.
- Skalierbarkeit: Die Leistung eines Containers ist konfigurierbar (mit Ressource Management, Prozessorsets und CPU-Caps).
- Separation: Die Applikationen sind voneinander separiert, ein direkter gegenseitiger Einfluss über das OS ist nicht möglich.
- OS-Maintenance: Nur an zentraler Stelle (in der globalen Zone) muss die OS-Installation, Patches und Umsetzung Firmen-interner Standards erfolgen.
- Delegation: Die für die Applikation / den Service zuständige Abteilung braucht für einen Teil der Ablaufsteuerung Root-Rechte. Sie kann hierzu die Root-Rechte in der Zone erhalten, ohne die anderen lokalen Zonen oder die globale Zone beeinträchtigen zu können. Die Zuweisung von Ressourcen ist allein der globalen Zone vorbehalten.
- Overhead: Alle Prozesse lokaler Zonen sind aus Sicht der globalen Zone nur normale Applikationsprozesse. Der OS-Overhead (Memory Management, Scheduling, Kernel) und der Speicherbedarf für shared Objekte (Dateien, Programme, Libraries) entsteht nur einmal. Jede Zone hat zusätzlich nur eine kleine Anzahl von Systemprozessen. Daher sind hunderte von Zonen auf einem 1-Prozessor-System möglich.

Nachteile:

HW-Maintenance: Bei Ausfall einer gemeinsam genutzten Komponente sind ggf. viele oder alle Zonen betroffen. Mit FMA (Fault Management Architecture) erkennt Solaris 10 Anzeichen eines zukünftigen Ausfalls und kann die betroffenen Komponenten (CPU, Memory, Bus-Systeme) im laufenden Betrieb deaktivieren bzw. benutzt stattdessen alternativ vorhandene Komponenten.

Durch Einsatz von Cluster Software kann die Verfügbarkeit der Applikation in der Zone verbessert werden (Solaris Container Cluster/ Solaris Container).

- Separation: Die Applikationen können sich über gemeinsam genutzte Hardware gegenseitig beeinflussen. Der Einfluss kann bei Solaris mit Ressource Management und Netzwerk-Bandbreitenkontrolle minimiert werden.
- OS-Versionen: Unterschiedliche Betriebssysteme/Versionen sind nur mit Branded Zones möglich. Hierbei wird in einer Zone eine virtuelle Ablaufumgebung für ein anderes Betriebssystem erzeugt, der Kern der globalen Zone wird aber von den Branded Zones mitbenutzt.



Implementierungen von Containern in einem OS sind im freien BSD Betriebssystem die Jails, in Solaris die Zonen und im Linux das vserver Projekt.
 HW Voraussetzungen sind nicht erforderlich.

Gegenüberstellung der Virtualisierungs-Technologien

Die vorgestellten Virtualisierungs-Technologien lassen sich in der folgenden Tabelle -im Vergleich zur Installation auf einem separaten Rechner - zusammenfassen.

	Separate Rechner	Domains/ Physikalische Partitionen	Logische Partitionen	Container (Solaris Zonen) in einem OS
Separation	+	O	+ (Software) - (Hardware)	+ (Software) - (Hardware)
Applikation	+	+	+	+
SW-Maintenance	-	-	-	+
HW-Maintenance	+	+	O	O
Delegation	-	-	-	+
Skalierbarkeit	O	+	O	+
Overhead	-	-	-	+
OS-Versionen	je eine	mehrere	mehrere	eine

Legende zur Bedeutung der Symbole:

- + gut
- O neutral
- hat Nachteile

Betriebs- und Verrechnungsmodell von virtualisierten Umgebungen

Viele Anwender und Betreiber haben eigene Betriebs- und Verrechnungsmodelle entwickelt oder vorhandene adaptiert. Die vorgehend genannten Virtualisierungs-Technologien unterscheiden sich jedoch gravierend voneinander, sowohl technologisch als auch prozess- und verrechnungstechnisch. Eine Berücksichtigung der technologischen Features auf das Verrechnungsmodell erfolgt i.d.R. nur selten. Um dies genauer zu beleuchten, müssen hier zunächst die typischen administrativen Tätigkeiten bei der Installation sowie beim anschließenden Betrieb inkl. Wartung betrachtet werden. Wie in der Gegenüberstellung zuvor beschrieben, besteht die Virtualisierung von Solaris 10 mit Containern aus einem einzigen Betriebssystem pro Server, lediglich einige wenige Prozesse werden zusätzlich pro Zone aufgestartet und verwaltet. Dem gegenüber wird sowohl bei der HW-Partitionierung als auch bei dem Einsatz eines Hypervisors eine entsprechende Anzahl von vollwertigen Betriebssystem-Instanzen komplett installiert. Diese Installationen erfordern dementsprechend auch eine individuelle Administration. Zusätzlich erfordert bei dem Einsatz eines Hypervisors dieser ebenfalls kontinuierliche Administration und verbraucht HW-Ressourcen wie CPU und RAM.

Gesamtüberblick: Typische Verteilung der Administrationstätigkeiten: Single Instanz vs. Solaris Zones

Das nachfolgende Schaubild gibt einen Überblick über die typischen Administrationstätigkeiten und eine Zuordnung, wo diese Tätigkeit angewendet wird:

- Auf der ganz linken Spalte die typischen Administrations-Tätigkeiten.
- Die gelbe Spalte entspricht einer typischen virtualisierten Instanz auf Basis eines Hypervisors (z.B. LPAR, LDOM, Vmware, XEN). Da hier pro Instanz eine komplette Betriebssystem-Umgebung installiert wird, sind hierfür auch alle Administrativen Tätigkeiten notwendig. Solaris 10 mit der Zonen/Container Technologie unterscheidet sich hier grundlegend:
- Die hellblaue Spalte entspricht der globalen Zone von Solaris 10. Einige Zeilen wie z.B. die Benutzerverwaltung erfolgen nicht in der Zone 0, sodass der administrative Aufwand für diese Zone in der Summe geringer ausfällt.
- Die fliederfarbige Spalte repräsentiert eine typische standardisierte Folge-Zone (Sparse), welche größtenteils lediglich auf die Zone 0 referenziert ist, und somit einen wesentlich geringeren administrativen Aufwand bedarf.
- Die hellgrüne Spalte stellt (zur Vollständigkeit der unterschiedlichen Möglichkeiten der Zonen mit Solaris 10) eine eigenständige Full Zone dar. Hier erfolgt z.B. das Patch-Management ebenfalls pro Zone und generiert hierdurch auch einen entsprechend höheren administrativen Aufwand.

Aktivitäten	Single Instanz	Global Zone	Sparse Zone	Full Zone
Installation	X	X	X	X
Updates/Patches	X	X		X (1)
User Management	X		X	X
Disk Management	X	X	(2)	(2)
Network Management	X	X		
System Backup	X	X		
Warm Backup (Applikation)	X		X	X
Cold Backup (Applikation)	X	X		
System Management	X	X		

(1) Ausgenommen Kernel Patches, nur für dedizierte Applikations-bezogene Patches

(2) Gleiche Administrations Aktivitäten unabhängig von Zonen-Modell

↑ ↑ ↑
Hilfreich zur Ermittlung der gesamten administrativen Aufwände für eine single OS Instanz im Vergleich zu einer sogenannten „sparse Zone“ sowie zu einer „full Zone“.

Detailüberblick: Typische Verteilung der Administrationstätigkeiten

Aus dem vorangehenden Schaubild geht hervor, dass für die Administration einer kompletten Betriebssystem-Instanz in Verbindung mit einem Hypervisor jeweils der volle Aufwand von 100% notwendig ist. Bei Solaris 10 Container erfolgt eine Trennung des Betriebssystems in die:

- globale Zone mit allen gemeinsamen Funktionalitäten sowie die
- Zonen mit individuellen Funktionalitäten

Im nachfolgenden eine Aufstellung der Administrationstätigkeiten, aufgeteilt in die einzelnen Spalten:

1. **Kategorie: Detail**
 - Kategorie-Überbegriff sowie Details
2. **Betriebssystem: komplette Installation**
 - Administrationsaufwand pro Betriebssystem-Instanz z.B. (Auf single Server, virtualisiert via Hypervisor, Solaris 10 Zone 0 (Root-Zone))
3. **Solaris 10 Sparse-Zone**
 - Administrationsaufwand pro Sparse-Zone, d.h. Aufwände die außerhalb der Zone 0 pro Instanz durchgeführt werden (in Prozent, relativ zu Spalte 2 (Zone 0))
4. **Synergien/Vorteile**
 - Kurze Beschreibung der Synergien mit den entsprechenden Technologien
5. **Gewichtung Faktor**
 - Faktor um die Häufigkeit der Tätigkeit zu gewichten

Kategorie: Detail	Solaris 10 Instanz - single Instanz	Solaris 10 Container - Sparse Zone vs. single Inst.	Synergien / Vorteile	Gewichtung Faktor [1-10]
Server Betriebssystem Dienst: - Change Management - Problem Management - Release Management	100%	10%	Durch Nutzung gemeinsamer Komponenten wie z.B.: Kernel, Device-Driver, Patch-Stände, System-Parameter, TCP/IP Stack. Inkl. User-Verwaltung (lokal) und ssh-keys. Hinweis: „Live-Upgrade“ für Zonen ab Updt. 4 (07/07)	6
Server Betriebssystem Dienst: - Tuning	100%	0%	Durch Nutzung gemeinsamer Komponenten wie z.B.: Kernel, Device-Driver, Patch-Stände, System-Parameter, TCP/IP Stack. Hinweis: „Live-Upgrade“ für Zonen ab Updt. 4 (07/07)	6
Server Betriebssystem Dienst: - OS Backup	100%	< 5%	Einmaliger Eintrag notwendig. Backup über Zone 0, dadurch auch Backup von <u>nicht</u> gestarteten Zonen möglich. Einsatz von „sparse Zones“: (3,5GB vs. 100MB)	10
Security Management: Prüfen von Security Patches und SW-Ständen (Betriebssystem bezogene Stände)	100%	0% Admin. 5% Zeitl.	Zentrales Patch-Management insbesondere von Security Patches ausschließlich in Zone 0. Zentrale Security Konfiguration (Hardening und Minimization) in Zone 0. Hinweis: „Live-Upgrade“ für Zonen ab Updt. 4 (07/07)	10
Configuration Management: Pflege der Konfiguration	100%	5%	Nur Eintrag von Name und IP-Adresse(n) pro Zone	2
Server Netzwerk Komponenten: Administration der Netzwerk Komponenten	100%	< 10%	Zentrales Management aller physikalischen und logischen Netzwerk-Interfaces. Lokale Verwaltung z.B. der Netzwerk-Portnummern pro Container spezifisch bezogen auf Applikation.	2
Dokumentations-Dienst: Pflege der Betriebs-Dokumentation	100%	15%	Betriebshandbuch. Durch Nutzung gemeinsamer Komponenten wie z.B.: Kernel, Device-Driver, Patch-Stände, System-Parameter, TCP/IP Stack. Lokale Konfiguration von IPQoS wie z.B. IP-Filter.	2
Betriebssystem-Wechsel (OS-Version)	100%	<5%	Sehr selten! Durch Cloning von vorkonfigurierten Referenz-Containern gemäß vordefiniertem Installations-Standard z.B. für Web-Server, Appl.-Server, Oracle DB-Server. Ansonsten identische Tools zur initialen Installation (Zone 0) z.B. mit Flash-Archive.	2

Berechnungsmodelle

Im nachfolgenden werden für die vorher genannten, unterschiedlichen Virtualisierungs-Technologien Berechnungsmodelle zur Ermittlung und Verrechnung der Administrationsaufwände dargestellt.

1. Berechnungsmodell Domains/physikalische Partitionen mit kompletten OS-Instanzen

Parameter zur Berechnung:

- 1 physikalischer Server = $n * \text{Domain}$
- 1 Domain = 1 OS Instanz
- Kosten pro OS Instanz: Festgelegter mtl. Verrechnungssatz für Administration eines single OS

$$\text{Betriebskosten}_{ges} = \text{Kosten pro OS Instanz} * \text{Anz. Domains } (n)$$

2. Berechnungsmodell Hypervisor mit kompletten OS-Instanzen

Parameter zur Berechnung:

- 1 physikalischer Server = $n * \text{VM} + \text{Hypervisor}$
- 1 VM = 1 OS Instanz
- Kosten pro OS Instanz: Festgelegter mtl. Verrechnungssatz für Administration eines single OS
- Eine zusätzliche SW-Schicht: 'Hypervisor'(proprietary OS)

$$\text{Betriebskosten}_{ges} = \text{Kosten pro OS Instanz} * \text{Anz. VM } (n) + \text{Hypervisor Admin Kosten}$$

2. Berechnungsmodell Solaris 10 Container mit Sparse-Zones

Parameter zur Berechnung:

- 1 physikalischer Server = $1 * \text{Zone 0} + n * \text{Sparse-Zones}$
- n = Anzahl lokaler Zonen (Spares Zones)
- Kosten pro OS Instanz: Festgelegter mtl. Verrechnungssatz für Administration eines single OS

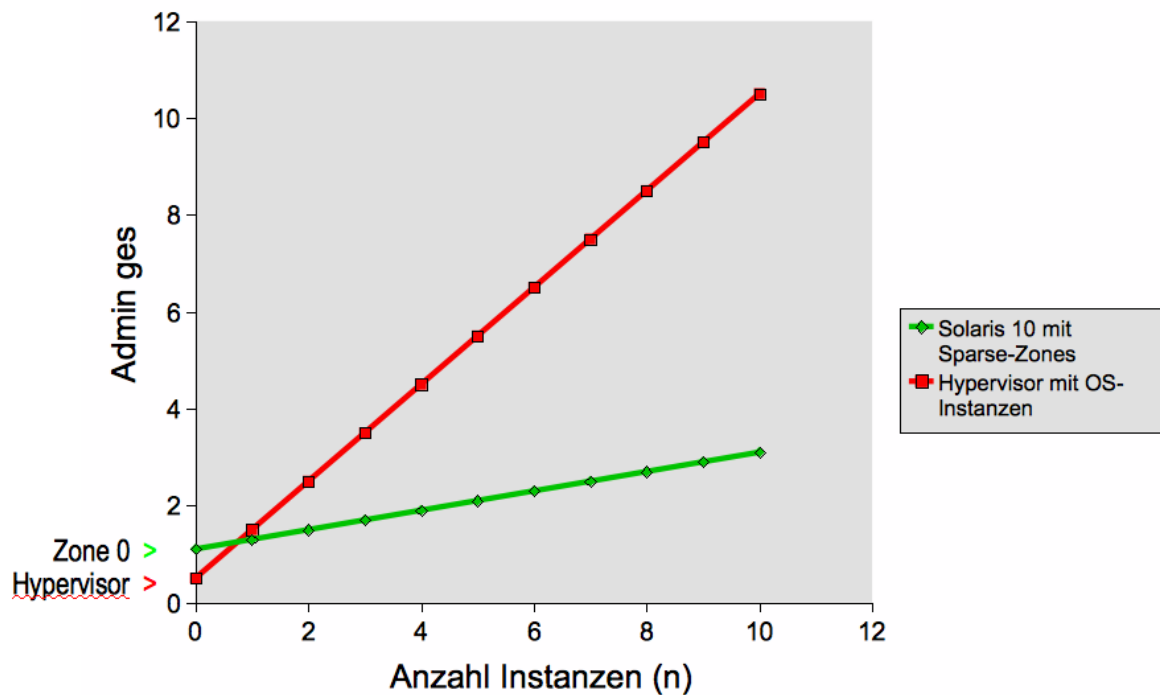
$$\text{Betriebskosten}_{ges} = 1,1 * \text{Kosten pro OS Instanz} + (n * 0,2 * \text{Kosten pro OS Instanz})$$

Der Betrieb von Solaris 10 mit Zonen unterscheidet sich somit aus Administrations-Sicht grundlegend von dem bisherigen Modellen und auch von den früheren Solaris Versionen 8 und 9. Mit der Einführung der Zonen-Technologie erfolgt eine Aufteilung der administrativen Tätigkeiten:

- Ein Großteil des Gesamtaufwandes für die Administration erfolgt zentral in der Zone 0, welcher aufgrund der etwas verlängerten Laufzeit bei der Installation von Patches hier auf ca. 110% gegenüber dem vorhergehend beschriebenen Modellen gesetzt wird.
=> Faktor 1,1.
- Lediglich ein kleiner administrativer Anteil von ca. 20% ist pro Folge-Zone (Sparse) für deren Betrieb notwendig.
=> Faktor 0,2.

Graphische Darstellung: Gegenüberstellung Hypervisor mit OS-Instanzen vs. Solaris 10 mit Sparse-Zones

Die vorhergehenden Berechnungsmodelle für einen Hypervisor mit OS-Instanzen (rot) sowie Solaris 10 mit Sparse-Zones (grün) sind hier nochmals graphisch dargestellt am Beispiel von insgesamt 10 Instanzen pro Server. Mit zunehmender Anzahl von virtualisierten Instanzen ergibt sich ein immer größerer Vorteil von Solaris gegenüber Hypervisor-Architekturen, da der administrative Aufwand nur geringfügig steigt.



Fazit

Mit Solaris 10 Container ergeben sich neue, hoch effiziente Administrations-Verfahren, welche mit Hypervisor-Architekturen nicht 1:1 vergleichbar sind.

Diese Technologie ermöglicht neue Virtualisierungs-Architekturen und unterstützt das Bestreben nach standardisierten Umgebungen. Durch eine stark zentralisierte (und lediglich einmal durchzuführende Administration für mehrere Instanzen) resultiert dies in einer enormen Kosteneinsparung im OPEX-Bereich.

Die aufgezeigten Verrechnungsmodelle berücksichtigen diese Unterschiede und tragen somit dazu bei, die Ressourcen-effizienteste und gleichzeitig kostengünstigste Variante zu ermitteln und binnen kürzester Zeit einen hohen ROI beim Umstieg in Richtung Infrastructure as a Service zu erzielen.

Für größere Infrastrukturen sowie für die Layer „Plattform as a Service“ und „Software as a Service“ ergänzt sich hier ideal der Oracle Enterprise Manager OpsCenter, welcher ein komplettes Lifecycle Management mit integriertem Cloud Management zur Verfügung stellt.

Quellen

- Solaris 10 Container Leitfaden
 - Detlef Drewanz, Ulrich Gräf, u.v.a.
 - Sun Microsystems 2009

Kontaktadresse:

Joachim Dietsch

ORACLE Deutschland B.V. & Co. KG

Amperestraße 6

63225 Langen

+49 (0)6103 - 752 0

+49 (0)6103 - 752 299

joachim.dietsch@oracle.com

www.oracle.com/de