

OpenStack mit Oracle

Franz Haberhauer, ORACLE Deutschland B.V. & Co. KG

Im Jahr 2010 initiierten Rackspace und die NASA ein Open-Source-Projekt namens „OpenStack“ für ein Cloud-Framework, zunächst für Infrastructure-as-a-Service: Compute-, Storage- und Netzwerkressourcen werden als Services mit virtualisierten Ressourcen-Pools bereitgestellt. Inzwischen setzen namhafte Unternehmen wie AT&T, eBay, PayPal, Wells Fargo, Sony oder Walt Disney OpenStack teils in großem Maßstab ein.

OpenStack hat eine modulare Architektur, die es Komponenten-Anbietern etwa von Speichersystemen oder Netzwerk-Komponenten erlaubt, sich einfach in das OpenStack-Framework zu integrieren. Diese Möglichkeit haben auch Anbieter alternativer Cloud-Frameworks, was heterogene Cloud-Verbunde ermöglicht. Die OpenStack Foundation wird mittlerweile von gut 300 Unternehmen und Organisationen unterstützt.

Oracle arbeitet seit Dezember 2013 an der OpenStack Foundation als Corporate Sponsor mit und hat OpenStack inzwischen in einige Produkte integriert. Besonders eng ist die Integration in Solaris

11.2, wo einige der neuen Technologien ideal auf die Anforderungen von OpenStack passen. Oracle Linux und Oracle VM unterstützen OpenStack inzwischen ebenso wie einige Storage-Produkte von Oracle – von den ZFS Storage Arrays als Block Storage bis hin zur Oracle Storage Public Cloud als Object Storage.

Die Architektur

OpenStack [1] hat eine modulare Architektur. Die einzelnen Komponenten sind in eigenen Projekten realisiert, die ihre Dienste über REST-Web-Services-APIs bereitstellen. Alternative Virtualisierungstechnologien sowie Geräte-Anpassungen

sind vielfach über Plug-ins realisiert, so dass sie in Implementierungen flexibel ausgetauscht und oft auch parallel eingesetzt werden können. Die Kern-Dienste sind „Nova“ für Compute, „Neutron“ für das Netzwerk, „Cinder“ für Block Storage und „Swift“ für Object Storage. Hinzu kommen als Infrastruktur-Dienste die Identity-Verwaltung „Keystone“ zur Authentifizierung, die Image-Verwaltung „Glance“ zum Ausrollen von Betriebssystem-Images (gegebenenfalls einschließlich Anwendungsstacks) sowie die browserbasierte Verwaltungsoberfläche „Horizon“. Daneben gibt es für die einzelnen Services ein Command Line Interface

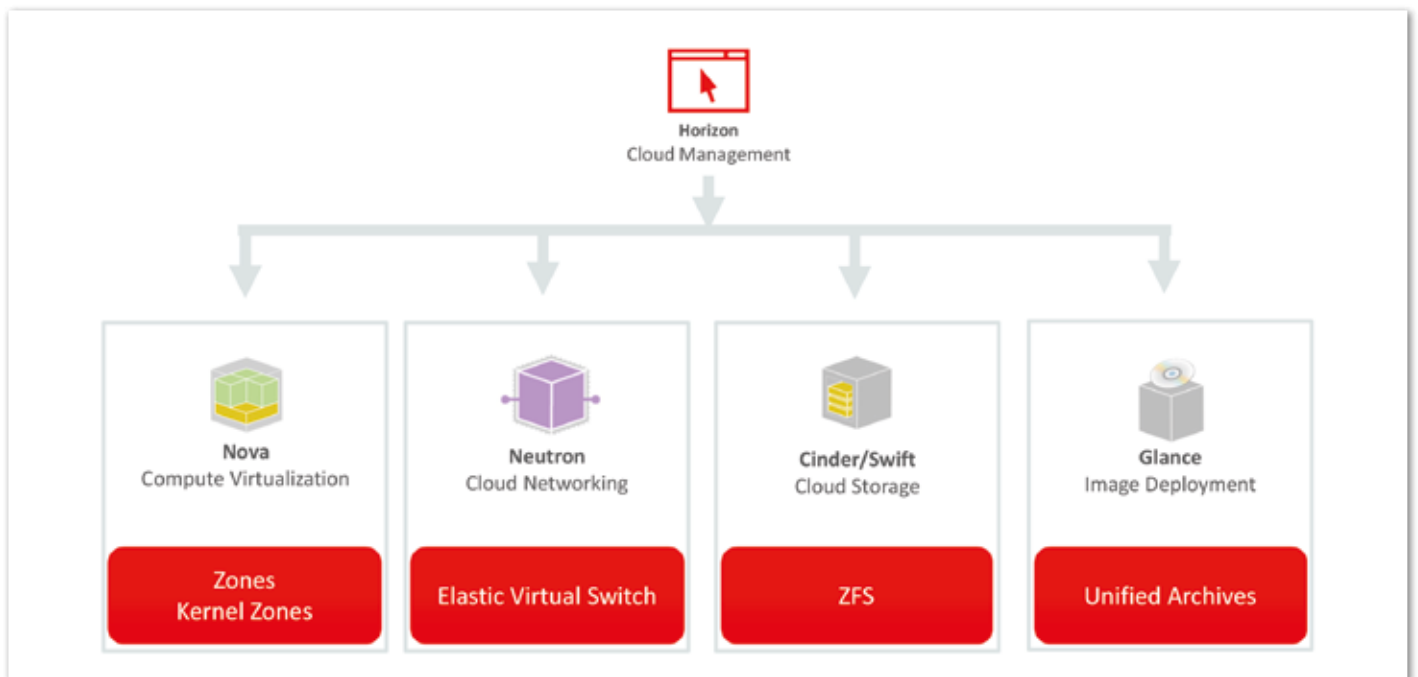


Abbildung 1: Nutzung von Solaris-Technologien in den einzelnen Diensten der Open-Stack-Implementierung in Solaris 11.2

(CLI). Nachdem OpenStack in Python implementiert ist, besteht auch eine direkte Schnittstelle für Python. Nachfolgend sind die einzelnen Dienste (siehe Abbildung 1) am Beispiel der Implementierung in Solaris 11.2 vorgestellt [2].

Nova (Compute)

Das Nova-API erlaubt das Anlegen, Abfragen, Ändern und Löschen von virtuellen Servern. Da diese Funktionen im Service gekapselt sind, ist es der konkreten OpenStack-Implementierung überlassen, welche Virtualisierungstechnologie zum Einsatz kommt. Im Falle von Solaris 11.2 sind dies alternativ die bewährten, besonders effizienten und sehr leichtgewichtigen Zonen oder die neuen Kernel-Zones. Diese sind etwas flexibler, da mit ihnen unterschiedliche Versionsstände der Kernel möglich sind. Zudem können sie suspendiert und auf ein anderes System verschoben werden (Warm Migration).

Solaris-Zonen bieten ein umfassendes Ressourcen-Sharing untereinander – nicht zuletzt durch den gemeinsam genutzten Kern. Sowohl die klassischen Zonen als auch die neuen Kernel-Zones sind im Vergleich zu virtuellen Maschinen auf der Basis virtualisierter Hardware effizienter, da die Virtualisierung rein auf Solaris ausgelegt ist und damit keine Zugeständnisse an die Virtualisierung generischer Hardware-Features nötig sind.

Weitere Implementierungen unterstützen alternative Virtualisierungstechnologien wie Oracle VM, Xen-basierte Lösungen, qemu, KVM, Hyper-V oder VMware ESXi. Beim Anlegen eines Servers wird das zu installierende Image ausgewählt, das dann vom Glance-Image-Service bezogen wird. Über sogenannte „Flavors“ können vordefinierte Kombinationen von virtualisierten Hardware-Ressourcen gewählt werden, etwa die Anzahl von vCPUs, Hauptspeicher und Plattenkapazität.

Glance (Image Service)

Glance verwaltet die Images für die virtuellen Maschinen, die je nach Virtualisierungstechnologie das Betriebssystem oder die Betriebsumgebung enthalten, sowie eventuell einen vorkonfigurierten Anwendungs-Stack. Der Service speichert und liefert Images sowie zugehörige Metadaten wie Format, Größe, Prüfsumme, Erzeugungs- und Änderungsdaten oder Status. Diese werden über einen URI identifiziert und als Rohdaten gespeichert und geliefert.

In Solaris 11.2 wurde mit „Unified Archives“ ein überaus flexibles Werkzeug für System-Images zum Wiederherstellen oder zum Klonen von Solaris-Systemen eingeführt. Es erlaubt zudem einen Wechsel der Plattform zwischen Erzeugung und Nutzung, etwa zwischen einer physischen Installation direkt auf der Hardware und einer virtualisierten Umgebung etwa einer Zone oder einer Kernel-Zone (P2V oder V2V). Damit sind Unified

Oracle Database Appliance

Profitieren Sie vom ODA-Workshop!

Jetzt am DOAG Schulungstag
in Nürnberg am 21. November!

dbi InSite
Workshops

Als Oracle Platinum Partner hat dbi services die Single-Box Datenbanklösung ODA intensiv getestet. Unsere Experten teilen ihr Praxiswissen mit Ihnen: In nur einem Tag lernen Sie, wie man ODA effizient implementiert, betreibt und optimiert!

Phone +41 32 422 96 00 · BaselArea · Lausanne

dbi-services.com/ODA



Infrastructure at your Service.

dbi services

Archives ein ideales Image-Format für OpenStack auf Solaris.

Neutron (Netzwerk)

Neutron realisiert „Networking as a Service“ zwischen OpenStack-Diensten, wobei komplexe virtualisierte Netzwerk-Topologien isoliert zwischen unterschiedlichen Nutzern umgesetzt werden können. Neutron ist das Gegenstück zu den in den Diensten definierten Interfaces und stellt (vNICs) Ports auf virtuellen Switches bereit, denen MAC- und IP-Adressen zugeordnet sind. Dazu gibt es als Netzwerke isolierte OSI-Layer-2-Netzwerk-Segmente – analog zu VLANs in der physischen Welt. Ein solches Netzwerk ist nur für das Projekt (Tenant) sichtbar, unter dem oder für das es angelegt wurde, es sei denn, es wurde mit dem Shared-Attribut versehen.

Ein Administrator kann zudem externe Provider-Netzwerke definieren. Damit werden dann Subnetze assoziiert, also IPv4- oder IPv6-Adressblöcke (CIDR), sowie Routing-Informationen. Darüber können Projekte im Rahmen zugeteilter Quoten etwa für IP-Adressen virtuelle Netzwerk-Topologien aufbauen.

Über eine Plug-in-Architektur lassen sich flexibel Software-Defined-Networking-Technologien aufsetzen wie der Elastic Virtual Switch (EVS) in Solaris 11.2. Dieser ermöglicht die Konfiguration einer verteil-

ten Netzwerk-Infrastruktur mit virtuellen Switches auf mehreren Systemen sowie zentralem Management und Monitoring, ohne dass dazu Eingriffe in die zugrunde liegende physische Netzwerk-Infrastruktur etwa für das Routing nötig sind. Aufgesetzt wird auf VLANs oder VxLANs. Mit Virtual Extensible LANs (VxLANs) kann ein Layer-2-Netzwerk über ein Layer-3-Netzwerk gefahren werden. Während maximal 4096 VLANs möglich sind, sind es bei VxLANs bis zu 16 Millionen.

Cinder (Block Storage)

Das Cinder-API bietet eine Verwaltung für den Lebenszyklus von Volumes, angefangen vom Anlegen unter Berücksichtigung von Nutzerquoten und Quality-of-Service-Kriterien über das Verknüpfen mit Instanzen („Attach“ an beziehungsweise „Detach“ von einer virtuellen Maschine; ein Volume kann zu jedem Zeitpunkt maximal an einer Instanz hängen) bis hin zum Löschen. Dazu ein Beispiel aus dem REST-API [3]: Über ein HTTP-GET kann eine Liste aller Volumes, auf die ein bestimmter Nutzer (Tenant – hier mit der ID 441446) zugreifen kann, abgefragt werden (siehe Listing 1). Cinder liefert die Antwort formatiert in XML (wahlweise auch in JSON, nachfolgend ohne den HTTP-ohne Header, siehe Listing 2).

Über ein HTTP-POST „/v2/{tenant_id}/volumes“ mit einer Liste von Request-Para-

metern wird ein Volume angelegt. Ein HTTP-DELETE „/v2/{tenant_id}/volumes/{volume_id}“ löscht das Volume [4]. Dieses API wird vom CLI sowie von der Browser-basierten Administrationsoberfläche Horizon genutzt.

Zu den Volumes gibt es das Konzept von Snapshots als „read only“-Point-in-Time-Kopien und von Backups, die von nicht angehängten Volumes gezogen werden können. Ein Volume kann in der generischen OpenStack-Implementierung beispielsweise ein LVM-Volume sein, auf das über iSCSI zugegriffen wird.

Über spezifische Volume-Driver ist eine Vielzahl von Speichersystemen als Backend möglich. Auch für NFS gibt es einen Volume-Driver, der Volumes auf Dateien abbildet. Die Implementierung auf Solaris verwendet ZFS als Backend mit Treibern für lokalen Zugriff, den Zugriff über iSCSI oder FC.

OpenStack-Volumes sind auf „zvols“ in einem ZFS-Datenset abgebildet, das in der Cinder-Konfiguration („/etc/cinder/cinder.conf“) spezifiziert ist. Dieses Dataset wird von einem Solaris-SMF-Service angelegt und mit den für den Cinder-User nötigen Berechtigungen versehen [5]. Mit Solaris-Mitteln ist auch eine Verschlüsselung möglich [6]. Oracle ZFS Storage Appliances [7] können mit dem OS 8.2 direkt als Storage-Backend für Cinder genutzt werden [8].

Swift (Object Storage)

Swift bietet redundanten, skalierbaren Object Storage auf Basis einer verteilten Speicher-Infrastruktur. Daten können über mehrere Server und deren Platten verteilt und repliziert werden. Swift-Service ist kein klassisches Dateisystem mit einer Baumstruktur aus Verzeichnissen und Dateien, sondern bietet sogenannte „Container“, die Objekte aufnehmen. Diese dienen als Namensraum, sind aber auch die Granularität für Access Control Lists oder für Mechanismen wie eine Objekt-Versionierung. Objekte sind Inhalte mit anwendungsspezifischen Metadaten. Sehr große Objekte (voreingestellt mehr als 5 GB) werden als segmentierte Objekte gespeichert.

In Solaris 11.2 ist Swift wie Cinder auf der Basis von ZFS implementiert. Auch die Oracle Public Storage Cloud, die bislang

```
GET /v2/441446/volumes HTTP/1.1
Host: dfw.blockstorage.api.openstackcloud.com
X-Auth-Token: eaaafd18-0fed-4b3a-81b4-663c99ec1cbb
Accept: application/xml
```

Listing 1

```
<?xml version='1.0' encoding='UTF-8'?>
<volumes xmlns:atom="http://www.w3.org/2005/Atom"
  xmlns="http://docs.openstack.org/api/openstack-block-storage/2.0/content">
  <volume name="vol-004" id="45baf976-c20a-4894-a7c3-c94b7376bf55">
    <attachments/>
    <metadata/>
  </volume>
  <volume name="vol-003" id="5aa119a8-d25b-45a7-8d1b-88e127885635">
    <attachments/>
    <metadata/>
  </volume>
</volumes>
```

Listing 2

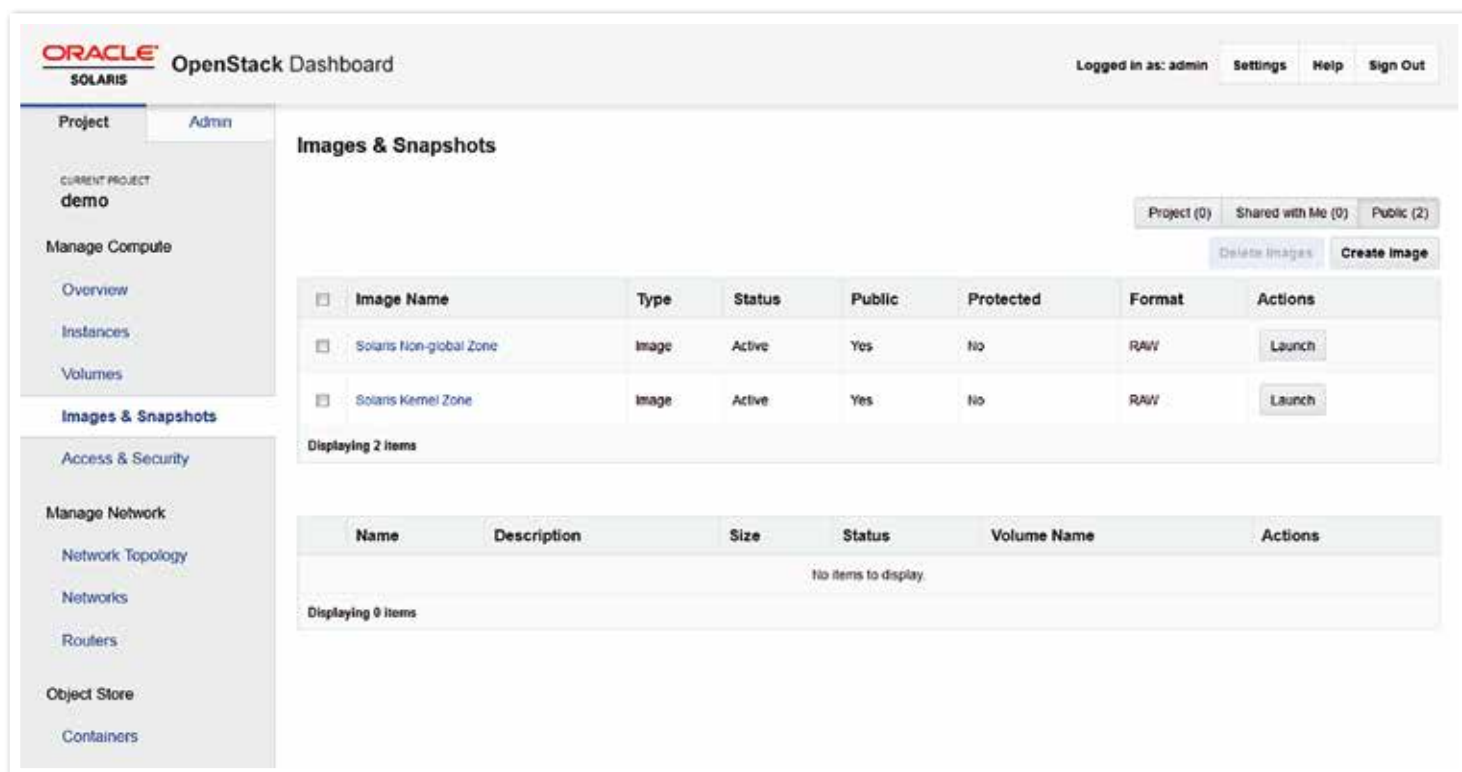


Abbildung 2: Horizon Dashboard

allerdings nur in Nordamerika verfügbar ist, unterstützt das Swift-API [9].

Keystone (Identity)

Keystone dient der Authentifizierung zwischen den einzelnen Diensten. OpenStack nutzt ein Token-basiertes Authentifizierungskonzept. Einzelne Nutzer werden initial über Name und Passwort authentifiziert. Zur Isolation von Ressourcen und als Namensraum für Objekte dienen sogenannte „Tenants“, denen ein Mandant, ein Kunde, eine Organisation oder ein Projekt entsprechen kann. Zudem gibt es Rollen, mit denen in den einzelnen OpenStack-Diensten Rechte und Privilegien verknüpft sein können. Mit einem Token, das ein Nutzer erhält, ist eine Liste von Rollen verknüpft, die der Nutzer annehmen kann. Die Semantik der Rollen wird von den Diensten definiert. Keystone ordnet Nutzern lediglich Rollennamen zu.

Horizon (Dashboard)

Horizon bietet zum einen eine Sicht für den Cloud-Administrator („Admin“), zum anderen ein Self-Service-Portal („Project“), in dem Nutzer jeweils auch nur die Ressourcen sehen, die ihren Projekten („Tenant“) zugeordnet sind. Das Portal liefert

auch eine Übersicht über verfügbare Quoten für Ressourcen, u.a. Instanzen, vCPUs, RAM und IP-Adressen, sowie deren Nutzung in definierbaren Zeitfenstern (siehe Abbildung 2). In der Admin-Sicht können projekt-übergreifende Ressourcen angelegt werden, etwa Netzwerk, allgemein verfügbare Images sowie Flavors (Resource-Klassen). Außerdem gibt es Übersichten über Objekte, die in Projekten angelegt sind.

In zwei kurzen Videos im OTN Garage Channel auf YouTube zeigt David Comay, einer der Architekten von OpenStack in Solaris, Abläufe im Admin- und das Project-Panel (allerdings noch des Grizzly-Release von OpenStack, das im Solaris 11.2 Beta enthalten war). Im final freigegebenen Solaris 11.2 wird auf OpenStack Havana aufgesetzt [10].

OpenStack ausprobieren

Das Aufsetzen von OpenStack in Oracle Linux und Oracle VM, die OpenStack als Technology-Preview unterstützen, ist in Einzelschritten in einem Whitepaper beschrieben [11]. Besonders einfach kann man erste Erfahrungen mit OpenStack auf Solaris 11.2 sammeln. Hier gibt es eine weitgehend

vorkonfigurierte Installation als Unified Archive, das man als USB-Boot-Image herunterladen, auf einen USB-Stick kopieren (mit „dd“ oder Solaris 11.2 „usbcopy(1M)“) und damit dann ein physisches System booten und installieren kann [12].

Leider ist eine Installation in einer VirtualBox nicht ganz so einfach, da VirtualBox nicht von USB booten kann und das Image mit gut 8 GB für ein ISO-Image zu groß ist. Als Workaround kann man das Image zusätzlich auf eine virtuelle Platte kopieren, die temporär als erste Platte der virtuellen Maschine hinzugefügt wird. Damit lässt sich der Boot-Vorgang von dieser Platte starten und anschließend kann die Installation des Unified Archive vom in die virtuelle Maschine eingebundenen USB-Stick erfolgen.

Nach der Installation sind über ein Skript (siehe Abschnitt „Post-installation“ im README) „ssh“-Keys für die System-Nutzer zu generieren, der Elastic Virtual Switch zu konfigurieren und die Neutron-SMF-Services zu starten [13]. Die enge Integration von OpenStack in Solaris 11.2 ist natürlich nicht nur für einen schnellen ersten Blick, sondern auch beim produktiven Einsatz von Vorteil.

Aktueller Stand und weitere Entwicklung

OpenStack hat einen raschen Entwicklungszyklus – alle sechs Monate findet ein Design-Summit statt, der mit einem neuen Release und einer großen Nutzer-Konferenz verbunden ist. Auf dem letzten OpenStack Summit in Atlanta (USA) im Mai 2014 waren mehr 4.500 Teilnehmer und namhafte Unternehmen präsentierten, wie sie OpenStack produktiv einsetzen. Der nächste OpenStack Summit wird Anfang November in Paris stattfinden.

Als Codename für das neue Release wird ein geografischer Ort mit Bezug zum Summit gewählt, mit einem auf das vorige Release folgenden Anfangsbuchstaben. Aktuell verbreitet ist das Havana-Release, das auch als Grundlage der Implementierung von Solaris 11.2 dient. Neuestes verfügbares Release ist Icehouse, das auch als Technology-Preview für Oracle Linux und Oracle VM verfügbar ist; Juno ist das aktuelle Entwicklungs-Release. Ein Release-Wechsel bedeutet heute im Betrieb einen Einschnitt: Eine Cloud muss für einen Release-Wechsel heruntergefahren werden.

Zu den neueren Projekten gehört Heat für die Orchestrierung komplexer Cloud-Anwendungen. Es wurde von AWS Cloud-Formation inspiriert. In Templates werden benötigte Infrastruktur-Ressourcen wie Server, Volumes, User etc. und ihre Zusammenhänge, beispielsweise welche Volumes mit welchen Servern verbunden sind, beschrieben, sodass eine komplexe Anwendung etwa als Ganzes ausgerollt und gestartet werden kann. Auch dynamische Zusammenhänge für Last-Balancierung, Auto-Skalierung und Verfügbarkeit können definiert sein.

Bei Ceilometer geht es um Metering für Chargeback/Showback. Trove adressiert „Database as a Service“ – zunächst mit MySQL- und NoSQL-Datenbanken, und zeigt, dass OpenStack nicht mehr auf „Infrastructure as a Service“ beschränkt ist, sondern auch in Richtung „Platform as a Service“ weiterentwickelt wird. Ein weiteres Projekt in dieser Richtung ist Sahara zur Provisionierung von Hadoop-Clustern auf OpenStack.

Hinweis: Auf der DOAG 2014 Konferenz + Ausstellung stehen mehrere Vorträge rund um OpenStack auf dem Programm, die Gelegenheit bieten werden, OpenStack als Demo insbesondere auf Solaris 11.2 zu sehen, inhaltlich zu vertiefen und zu diskutieren.

Hinweis: Auf der DOAG 2014 Konferenz + Ausstellung stehen mehrere Vorträge rund um OpenStack auf dem Programm, die Gelegenheit bieten werden, OpenStack als Demo insbesondere auf Solaris 11.2 zu sehen, inhaltlich zu vertiefen und zu diskutieren.

Weitere Informationen

- [1] OpenStack Project Homepage: <http://openstack.org>
- [2] Oracle Solaris OpenStack Cloud Management: <http://www.oracle.com/technetwork/server-storage/solaris11/technologies/openstack-2135773.html>
- [3] OpenStack Block Storage API v2 General API Information: http://docs.openstack.org/api/openstack-block-storage/2.0/content/General_API_Information.html, Reference: <http://developer.openstack.org/api-ref-blockstorage-v2.html>
- [4] OpenStack Cinder Commands: http://docs.openstack.org/user-guide/content/cinder_commands.html
- [5] Cindy Swearingen, How to Build OpenStack Block Storage on ZFS, July 2014: <http://www.oracle.com/technetwork/articles/servers-storage-admin/howto-build-openstack-zfs-2248817.html>
- [6] Darren Moffet, OpenStack Cinder Volume Encryption with ZFS, 31.7.2014: https://blogs.oracle.com/openstack/entry/cinder_volume_encryption_with_zfs

- [7] Oracle ZFS Storage Appliance: <http://www.oracle.com/de/products/servers-storage/storage/nas/overview/index.html>
- [8] Ronen Kofman, Running OpenStack Icehouse with ZFS Storage Appliance, 18.8.2014, https://blogs.oracle.com/ronen/entry/running_openstack_icehouse_with_zfs
- [9] Oracle Storage Cloud: <https://cloud.oracle.com/storage>
- [10] OpenStack on Oracle Solaris 11.2 Dashboard Walkthrough, Admin Panel: <https://www.youtube.com/watch?v=SZqlfJLkSQ>, Project Panel: https://www.youtube.com/watch?v=_ob5qj1UM-wY
- [11] Getting Started with Oracle VM, Oracle Linux, OpenStack Technology Preview, An Oracle White Paper, August 2014: <http://www.oracle.com/technetwork/server-storage/vm/ovm-linux-openstack-2202503.pdf>
- [12] Oracle Solaris 11.2 OpenStack Unified Archives: <http://www.oracle.com/technetwork/server-storage/solaris11/downloads/unified-archives-2245488.html>
- [13] Getting Started with OpenStack on Oracle Solaris 11.2, July 2014: <http://www.oracle.com/technetwork/articles/servers-storage-admin/getting-started-openstack-os11-2-2195380.html>



Franz Haberhauer
franz.haberhauer@oracle.com



Auf dem Prüfstand Oracle Datenbank Infrastrukturen

Besuchen Sie uns: **DOAG** Konferenz und Ausstellung 2014, 1. Ebene, Stand 124

Vortrag, 18. November
10:00 - 10:45 Uhr, Raum 7 „Kiew“
BYODBIT (Bring your own Database IT),
Converged oder Engineered? –
die optimale Datenbank-Plattform

Das erwartet Sie in 45 Minuten:
• **Unterschiedliche Infrastruktur-Szenarien**
Dual-Vendor-Konzepte im Vergleich
• **Experten-Know-how** Vor- und Nachteile unterschiedlicher Infrastruktur-Designs

Anhand von Beispielkonfigurationen mit Hardware bekannter Hersteller, lernen Sie die für Ihre Anforderungen optimale Datenbankinfrastruktur kennen und einschätzen.

Nicht auf der DOAG? Download der Vortragsfolien und der Referenzarchitektur „Oracle on EMC“: www.inforSacom.com/DOAG2014