

- Der Emulator ist rein als Software zu betrachten, welches das „echte“ Gegenstück simuliert. Somit können wir beispielsweise Linux oder Dos auf einem anderen Computer mit anderer Architektur laufen lassen. Auch ältere Videospielekonsolen können auf PCs mit Emulatoren simuliert werden. Schlüsselsatz: Keine Hardware, rein Software.
- Die Virtuelle Maschine wird mit der Hardware des PCs ausgeführt. Das bedeutet die VM bildet die Hardware eines Rechners nach. Ich habe mir das wie eine Matroschka Puppe vorgestellt.

Scenario	Beschreibung
Hardware und Software getrennt behandeln	Beim testen von Software, um Schäden an der Hardware zu vermeiden.
Ressourcennutzung	Das verteilen vieler Systeme auf einem, mit leistungsstarker Hardware, damit können auch ältere Systeme mit neuer HW laufen.
Software Testen	Software einer anderen Maschine kann auf der Hauptmaschine getestet werden.
Cloning	Kopien von dem System unbegrenzt kopieren und auf anderen Geräten abspielen

Der Snapshot ist eine Kopie der gesamten VM laufenden zum Zeitpunkt der Kopie. Diese Kopie kann als Wiederherstellungsknoten benutzt werden oder verbreitet werden.

(Siehe Video: Das erstellen eines Snapshots, während DSL auf meiner VM läuft.)

- Mit virtuellen Netzwerken werden beliebig viele Rechner, Server und VMs über das Internet verbunden. Zur Bewerksstellung werden keine Kabeln benutzt sondern Tools wie Switches und Adapter.
- VPN (Virtual Private Network):
 - Gesicherte Verbindung zu einem Intranet oder Netzwerk über das Internet.
- VLAN (Virtual Local Area Network):
 - Fasst ausgewählte Geräte zusammen und isoliert sie in ein lokales Netzwerk. Damit kann man die ausgewählten Teilnehmer beobachten.
- VXLAN (Virtual Extensible Local Area Network):
 - Eine Erweiterung von VLAN womit alle Geräte und ein ganzes Netzwerk virtualisiert wird.

Dashboard:

- Die UI zum verwalten des Systems

API:

- REST API, empfängt HTTP requests und kommuniziert über die oslo.messaging queue oder HTTP

Keystone:

- API Client Authentifizierung, Service Discovery über OpenStack's Identity API]

Network:

- IP Verteilung und VLAN Verwaltung

Conductor:

- Kann als Datenbank genutzt werden und Anfragen verwalten

Compute:

- Verwaltet Kommunikation zwischen VMs und Hypervisor

Hypervisor:

- Trennschicht zwischen VM und Hardware und VM Verwaltung

Glance & Cinder:

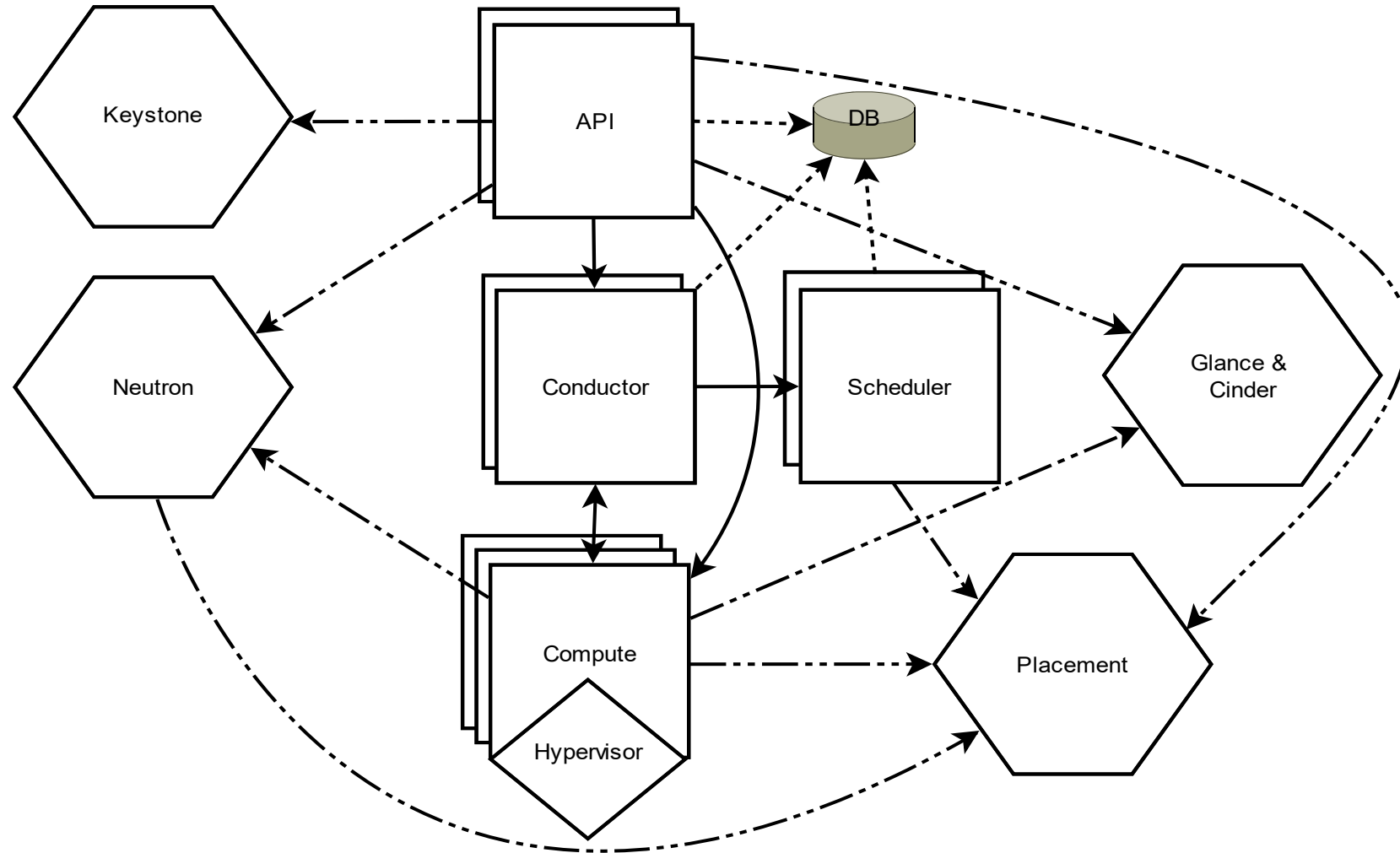
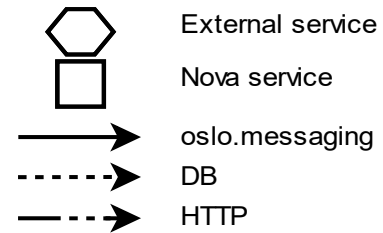
- Cinder – Virtueller Blockspeicher
- Glance – erstellt Images von VMs

Scheduler:

- Verwaltet die Instanzen für die Hosts


DB:

- SQL Datenbank und Speicher



1. Mit meiner VM und Ubuntu die Instanz erzeugen:

1 Eintrag wird angezeigt

<input type="checkbox"/>	Instanzname	Abbildname	IP-Adresse	Variante	bwCloud Hostname	Schlüsselpaar	Status	Verfügbarkeitszone
<input type="checkbox"/>	Le08	Ubuntu 20.04	192.52.35.97	m1.tiny	9c66d0bb-4156-41e1-b84b-b8340bfe5f13.fr.bw-cloud-instance.org	muh	Aktiv 	nova

2. Datenträger erzeugt und verbunden

Datenträger erstellen

Datenträgername

MCC-Data-Storage

Beschreibung

For Lecture Unit 08

Datenträger Quelle

Keine Quelle, leerer Datenträger

Typ

default

Größe (GiB) *

8

Beschreibung:

Datenträger sind blockorientierte Geräte, die mit Instanzen verbunden werden können.

Datenträger-Typ Beschreibung:

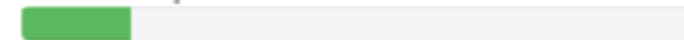
default

Keine Beschreibung verfügbar.

Datenträger-Begrenzungen

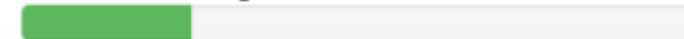
Gesamt Gibibytes

0 von 50 GiB verwendet



Anzahl an Datenträgern

0 von 4 verwendet



3. HTTPS Regel in der Security-Group erstellen

<input type="checkbox"/>	Richtung	Netzwerktyp	IP-Protokoll	Port-Bereich
<input type="checkbox"/>	Austritt	IPv4	Jede(s)	Jede(s)
<input type="checkbox"/>	Austritt	IPv6	Jede(s)	Jede(s)
<input type="checkbox"/>	Eintritt	IPv4	TCP	443 (HTTPS)

3 Einträge werden angezeigt

4. Login SSH

1 Eintrag wird angezeigt

<input type="checkbox"/>	Instanzname	Abbildname	IP-Adresse	Variante	bwCloud Hostname	Schlüsselpaar	Status	Verfügbar
<input type="checkbox"/>	Le08	Ubuntu 20.04	192.52.35.97	m1.tiny	9c66d0			

1 Eintrag wird angezeigt

```
ubuntu@le08: ~  
System load:  0.77          Processes:           104  
Usage of /:   10.4% of 11.46GB Users logged in:       0  
Memory usage: 18%          IPv4 address for ens3: 192.52.35.97  
Swap usage:   0%  
  
0 updates can be installed immediately.  
0 of these updates are security updates.  
  
The list of available updates is more than a week old.  
To check for new updates run: sudo apt update  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
ubuntu@le08:~$
```

5. Package downloaden und Hello World vom Docker laufen lassen

```
ubuntu@le08: ~  
Selecting previously unselected package docker.io.  
Preparing to unpack .../8-docker.io_19.03.8-0ubuntu1.20.04.2_amd64.deb ...  
Unpacking docker.io (19.03.8-0ubuntu1.20.04.2) ...  
Selecting previously unselected package ubuntu-fan.  
Preparing to unpack .../9-ubuntu-fan_0.12.13_all.deb ...  
Unpacking ubuntu-fan (0.12.13) ...  
Setting up runc (1.0.0~rc10-0ubuntu1) ...  
Setting up dns-root-data (2019052802) ...  
Setting up libidn11:amd64 (1.33-2.2ubuntu2) ...  
Setting up bridge-utils (1.6-2ubuntu1) ...  
Setting up pigz (2.4-1) ...  
Setting up cgroupfs-mount (1.4) ...  
Setting up containerd (1.3.3-0ubuntu2.3) ...  
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service.  
Setting up docker.io (19.03.8-0ubuntu1.20.04.2) ...  
Adding group `docker' (GID 121) ...  
Done.  
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket.  
docker.service is a disabled or a static unit, not starting it.  
Setting up dnsmasq-base (2.80-1.1ubuntu1.3) ...  
Progress: [ 90%] [#####ubuntu@le08:~$  
Status: Downloaded newer image for hello-world:latest  
Hello from Docker!  
This message shows that your installation appears to be working correctly.  
To generate this message, Docker took the following steps:  
1. The Docker client contacted the Docker daemon.  
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.  
   (amd64)  
3. The Docker daemon created a new container from that image which runs the  
   executable that produces the output you are currently reading.  
4. The Docker daemon streamed that output to the Docker client, which sent it  
   to your terminal.  
To try something more ambitious, you can run an Ubuntu container with:  
$ docker run -it ubuntu bash  
Share images, automate workflows, and more with a free Docker ID:  
https://hub.docker.com/  
For more examples and ideas, visit:  
https://docs.docker.com/get-started/
```

6. MariaDB als Host user angelegt und erster Start ohne Domain Setup

```
MariaDB [(none)]> select Host, User from mysql.user;
+-----+-----+
| Host      | User      |
+-----+-----+
| localhost | mariadb.sys |
| localhost | mysql      |
| localhost | nextcloud  |
| localhost | root       |
+-----+-----+
4 rows in set (0.009 sec)
```

Zugriff über eine nicht vertrauenswürdige Domain

Bitte kontaktiere Deinen Administrator. Wenn Du Administrator bist, bearbeite die „trusted_domains“-Einstellung in config/config.php. Siehe Beispiel in config/config.sample.php.

Weitere Informationen zur Konfiguration finden sich in der **Dokumentation**.

LU08 – Mobile Cloud Computing: Cloud IaaS

