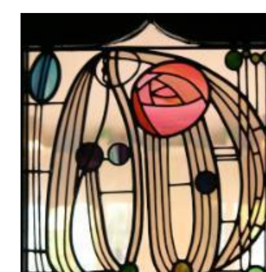
Cloud Systems

Chapter 1: Introduction

Dr Lauritz Thamsen
lauritz.thamsen@glasgow.ac.uk
School of Computing Science
University of Glasgow



Acknowledgements

- This part of the course draws on materials developed with colleagues at TU Berlin and Hasso Plattner Institute
  - Prof. Dr. Odej Kao
  - Dr. Daniel Warneke
  - Jossekin Beilharz
  - Dr. Anton Gulenko
  - Lukas Pirl
  - Morgan Geldenhuys
  - Philipp Wiesner

A Bit More on Me

Dr Lauritz Thamsen (he/him)

- Lecturer in Computer Systems
- Systems Research Section
- Low-Carbon and Sustainable Computing theme
- lauritz.thamsen@glasgow.ac.uk
- https://lauritzthamsen.org
- Office: S152
- Office hours: Tuesdays 11 am

Research interests:

- Distributed Systems
- Edge/Cloud Computing
- Resource Management & Scheduling
- Carbon-Aware Computing

First Half: Lecture Chapters

- Focus of this part: **Cloud Resource Management**
  - Management of virtual resources on host systems
  - Larger systems running on sets of virtual resources
  - Carbon impact assessment and optimization

1. Cloud Computing Intro
2. Virtual Machines
3. Containers
4. Cloud System Management
5. Cloud Sustainability

First Half: Labs and Exercise

- Weekly lab sessions:
  1. Lab 1: (Linux Systems) Benchmarking Basics
  2. Lab 2: Virtual Machines (Google Cloud and QEMU)
  3. Lab 3: Containers (Docker)
  4. AE1 formative feedback session
  5. General AE1 support

- Assessed Exercise 1 (AE1)
  - Virtual resources benchmarking project
  - Written report with plots showing own measurements
  - Release: 20.1. 10 am (Monday Week 2); Deadline: 14.2. 4:30 pm (Friday Week 5)
  - Weight 25%

The Audience

- Virtualization?
- Docker?
- Kubernetes?
- Infrastructure as Code?
- Edge computing?
- FaaS?
- Carbon-aware computing?

Outline of Chapter 1: Cloud Computing Intro

1.1 Clusters, Grids, Clouds, and Edge/Fog

1.2 The NIST Definition's Dimensions

– short break (10-15 minutes) –

1.3 Intro to Virtualization

1.4 Intro to VMs and Containers

Outline of Chapter 1: Cloud Computing Intro

1.1 Clusters, Grids, Clouds, and Edge/Fog

1.2 The NIST Definition’s Dimensions

– short break (10-15 minutes) –

1.3 Intro to Virtualization

1.4 Intro to VMs and Containers

Background

Clusters

- Many locally connected computers

Grids

- Loosely coupled and widely distributed computers or clusters

Clouds

- IT resources as a utility

Edge/Fog

- Cloud services in closer proximity to users and edge devices

Clusters

- Mostly homogeneous (commodity) compute resources and software stacks viewed as one system
- Interconnected by a low-latency and high-bandwidth (local area) network
- Goal: improving availability, resource utilization, price/performance

- Examples
  - Analytics clusters at Google, Microsoft, Facebook, Alibaba etc.!

Grids

- Widely distributed heterogeneous compute resources
- Connected via a slow network (i.e. the internet)
- Heterogeneous software stacks unified by a middleware
- Used for a common goal

- Examples
  - Worldwide LHC Computing Grid (WLCG)
  - Berkeley Open Infrastructure for Network Computing (BOINC)

Clouds

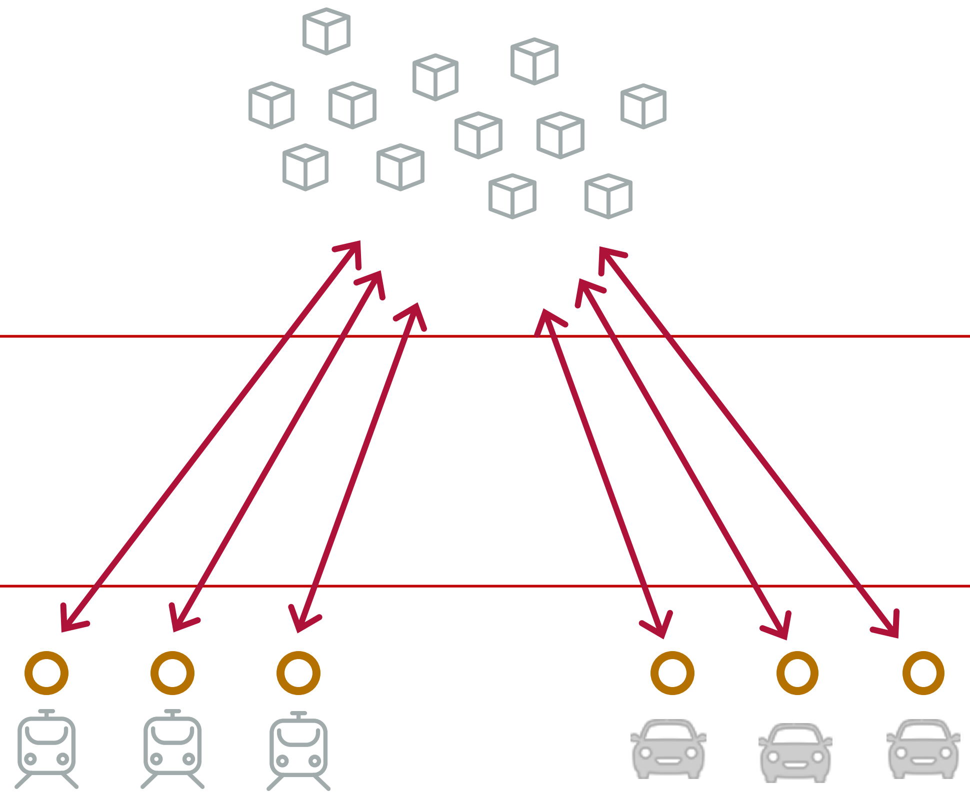
- Definition of Cloud Computing according to NIST [1]:

> "Cloud computing is a model for enabling **ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources** (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.
>
> This cloud model is composed of five essential characteristics, three service models, and four deployment models."
>
> Standard 800-145

- NIST: National Institute of Standards and Technology

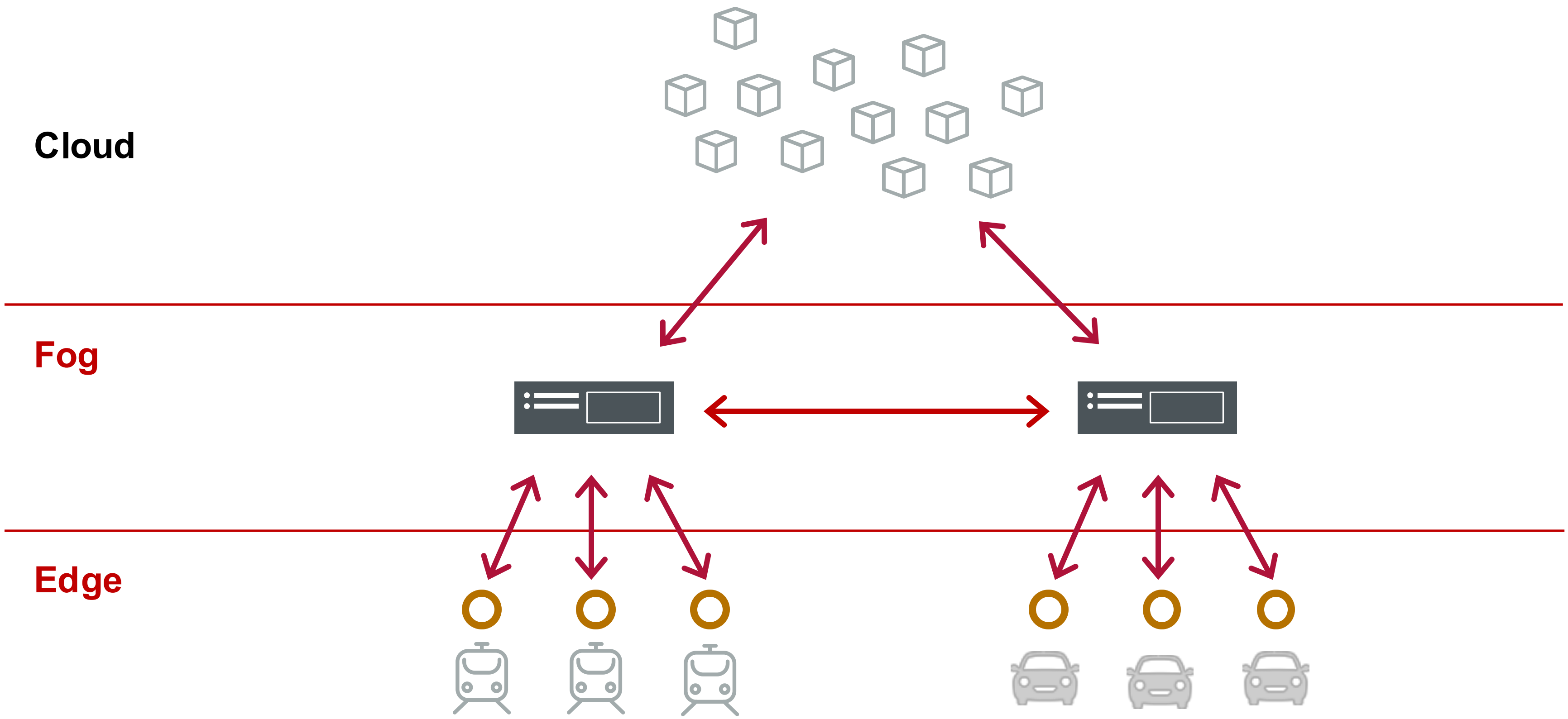
Cloud and the Internet of Things

Cloud



Devices and
Sensors

Fog Computing



Outline of Chapter 1: Cloud Computing Intro

1.1 Clusters, Grids, Clouds, and Edge/Fog

1.2 The NIST Definition's Dimensions

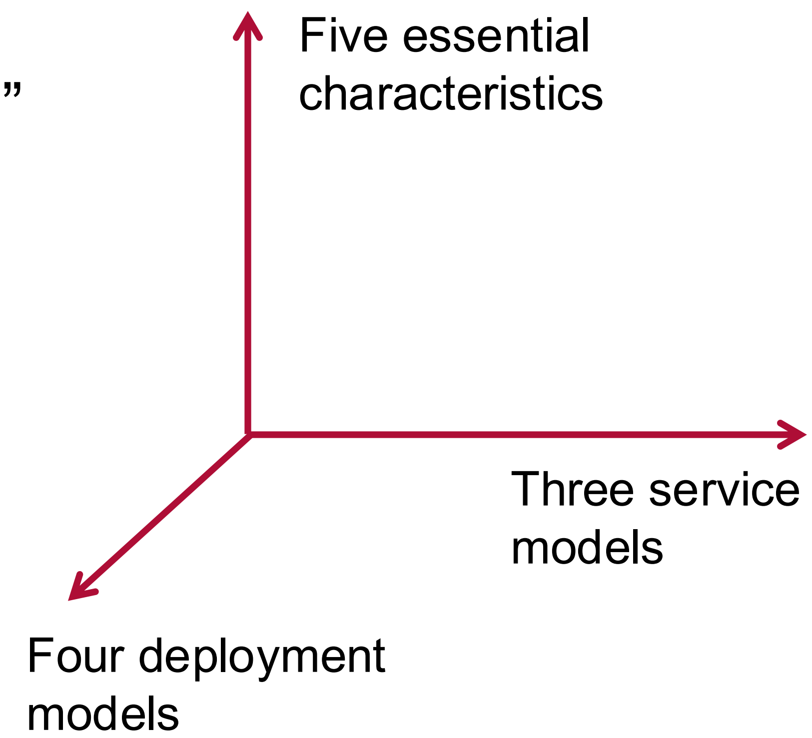
– short break (10-15 minutes) –

1.3 Intro to Virtualization

1.4 Intro to VMs and Containers

Dimensions of Cloud Computing (NIST)

- NIST: "Cloud model is composed of
  - Five essential characteristics
  - Three service models
  - Four deployment models"



Five Characteristics of Cloud Computing (NIST)

- On-demand self-service
  - No human interaction required for resource provisioning
- Broad network access
  - Accessible over network with standard mechanisms
- Resource pooling
  - Pooled resources dynamically shared among several consumers, with location independence
- Rapid elasticity
  - Capabilities can be provisioned/released on demand
- Measured service
  - Resource usage is monitored, controlled, and reported

Three Service Models of Cloud Computing (NIST)

- Software as a Service (SaaS)
  - Provider‘s application runs on cloud infrastructure and can be accessed over the network
  - Consumer does not control/manage underlying infrastructure
- Platform as a Service (PaaS)
  - Consumer can deploy custom application onto cloud infrastructure using programming languages, libraries, services, and tools supported by the provider
  - Consumer does not control/manage underlying infrastructure
- Infrastructure as a Service (IaaS)
  - Provider provides processing, storage, network resources to consumer
  - Consumer does not control/manage underlying infrastructure but has control over operating systems, storage, and deployed applications

Four Deployment Models of Cloud Computing (NIST)

	Private Cloud	Community Cloud	Public Cloud	Hybrid Cloud
User of the cloud infrastructure?	Single organization	Organizations with shared concerns	Open for the general public	Composition of private / community / public cloud: • Clouds remain distinct, but bound together by standard mechanisms • General goal: Enable portability
Owner of the cloud infrastructure?	Organization, third party, combination thereof	Organizations, third party, combination thereof	Business academic, government organization, combination thereof	
Location of the cloud infrastructure?	On premise, off premise	On premise, off premise	On premise of cloud provider	

Public Cloud Examples

- AWS EC2
- DigitalOcean Droplets
- Google Kubernetes Engine
- AWS S3
- Amazon EMR
- AWS Lambda
- Azure Machine Learning Studio
- AWS SageMaker
- IBM Watson
- Google Mail, Google Docs, MS Office 365…

Outline of Chapter 1: Cloud Computing Intro

Outline of Chapter 1: Cloud Computing Intro

1.1 Clusters, Grids, Clouds, and Edge/Fog

1.2 The NIST Definition's Dimensions

– short break (10-15 minutes) –

1.3 Intro to Virtualization

1.4 Intro to VMs and Containers

Challenges for Cloud Providers

- Rapid provisioning
  - Resources must be available to the consumer quickly
  - No human interaction during provisioning
- Elasticity
  - Create illusion of infinite resources
  - Yet, manage data center in a cost-efficient manner
- Isolation of different consumers
  - Users must not interfere with each other
- Performance
  - Maintain good performance despite other challenges

Approach: Virtualization

> “Cloud computing is a model for enabling ubiquitous, convenient, on-demand **network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services)** that can be rapidly provisioned and released with minimal management effort or service provider interaction.
>
> Standard 800-145

- Main idea on IaaS-level: provide *virtual resources*
  - Provide compute resources in the form of virtual machines or containers
  - Also other computing resources such as virtual networks, virtual network functions, virtual storage

What is Virtualization?

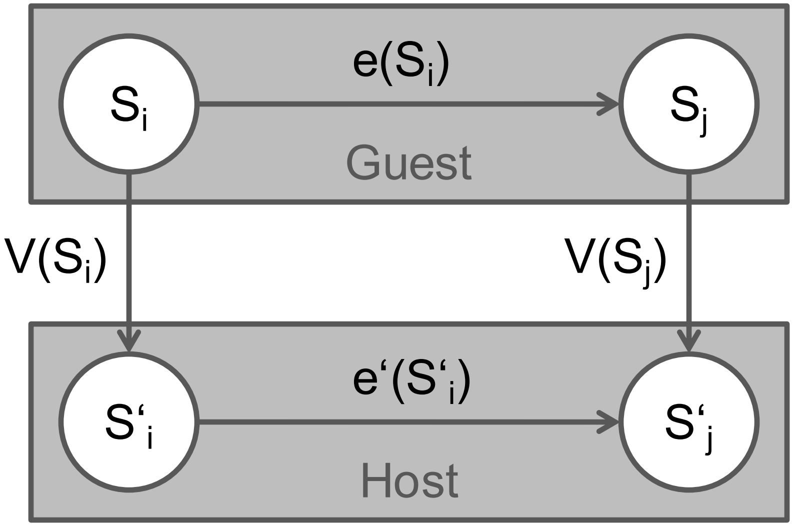
- Definition of virtualization according to NIST [2]:

> "Virtualization is the **simulation of the software and/or hardware upon which other software runs**. This simulated environment is called a virtual machine (VM)."
>
> Standard 800-125

- Virtualization can transform a real system so
  - it looks like a different virtual system
  - multiple virtual systems

- **Real system** is often referred to as **host (system)**
- **Virtual system** is often referred to as **guest (system)**

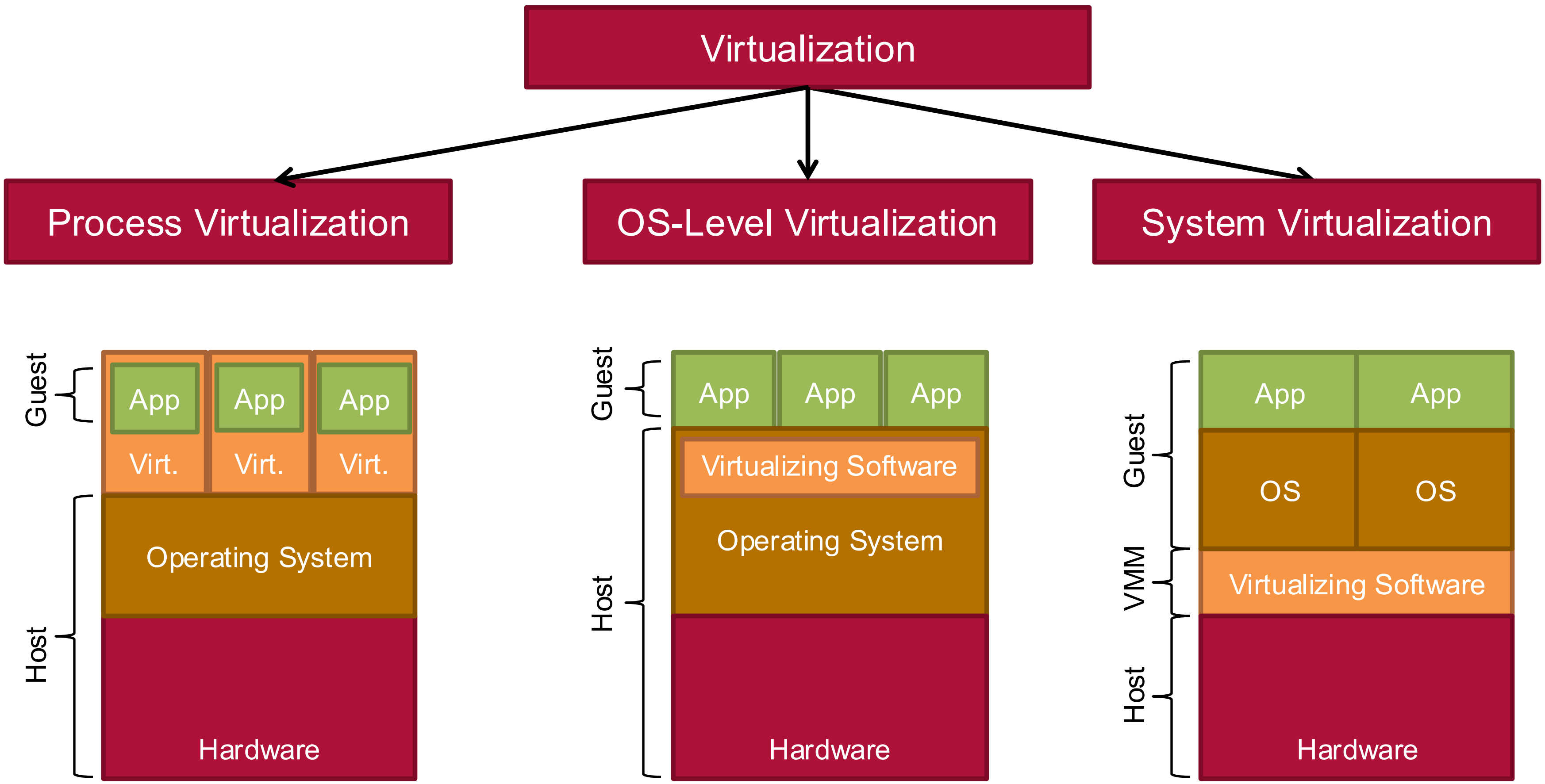
Formal Definition of Virtualization



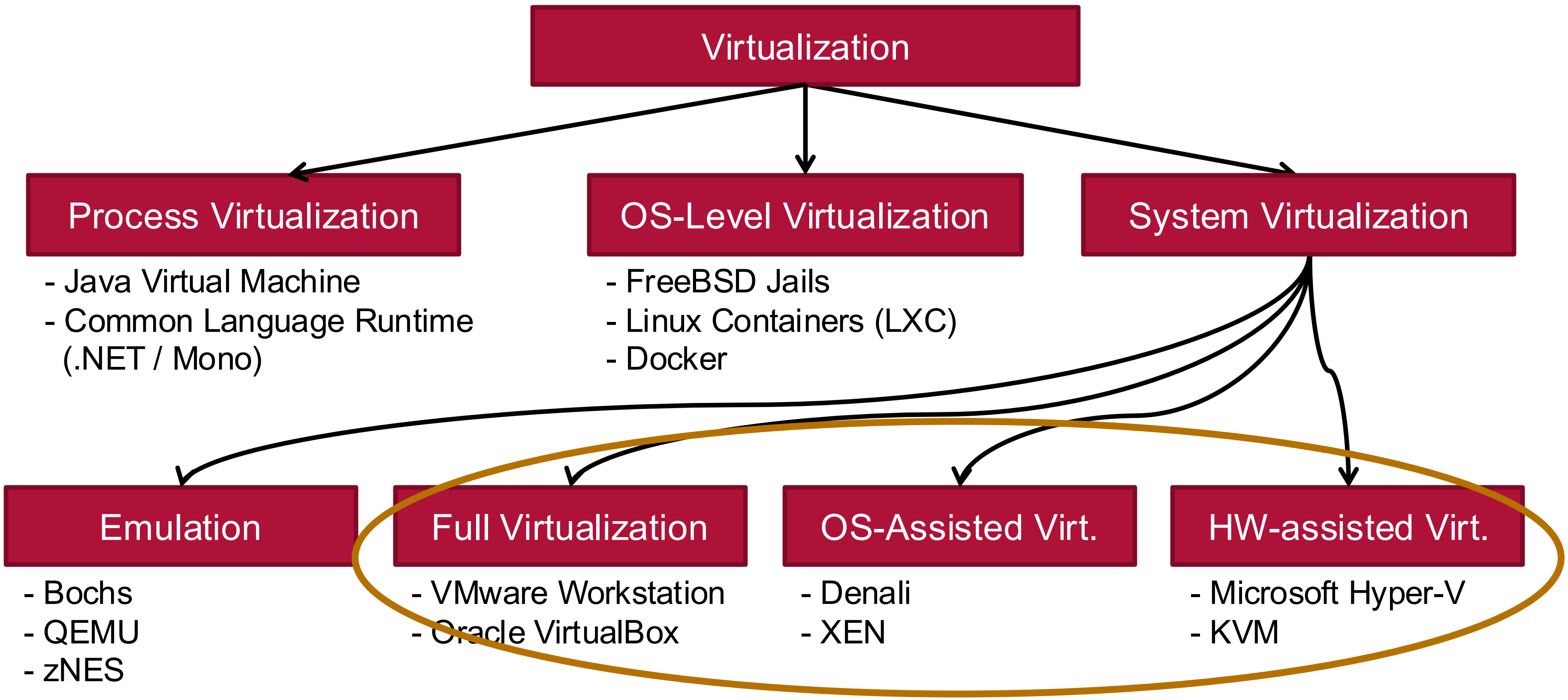
- Isomorphism V:
  - Si, Sj: States of machine
  - e: Sequence of operations

- Isomorphism V maps guest state to host state such that
  - for e that modifies the guest's state from S_i to S_j
  - there exists a corresponding sequence of operations e' that performs an equivalent modification between host's states (S'_i to S'_j)

Categories of Virtualization (1/2)



Categories of Virtualization (2/2)



Most relevant for Infrastructure as a Services

Outline of Chapter 1: Cloud Computing Intro

1.1 Clusters, Grids, Clouds, and Edge/Fog

1.2 The NIST Definition's Dimensions

– short break (10-15 minutes) –

1.3 Intro to Virtualization

1.4 Intro to VMs and Containers

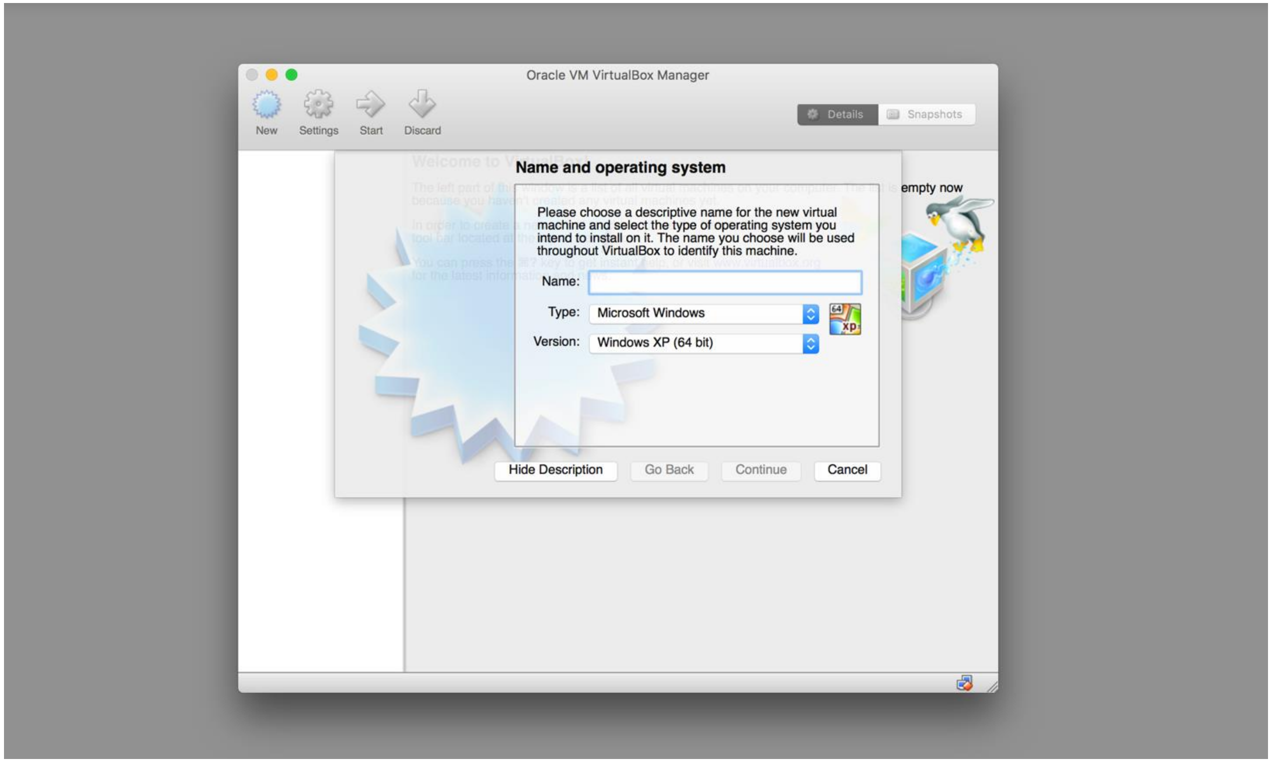
Virtual Machines

- Virtual machine (VM): hardware that is used by operating system and running processes is not real, but virtual → OS and processes run on software emulating hardware

- VMs can have different hardware characteristics (including different from the physical host)
  - e.g. number of cores, assigned memory, disk space
- VMs are often started from available images (with an operating system and some applications installed)
  - e.g. Windows with SQL Enterprise

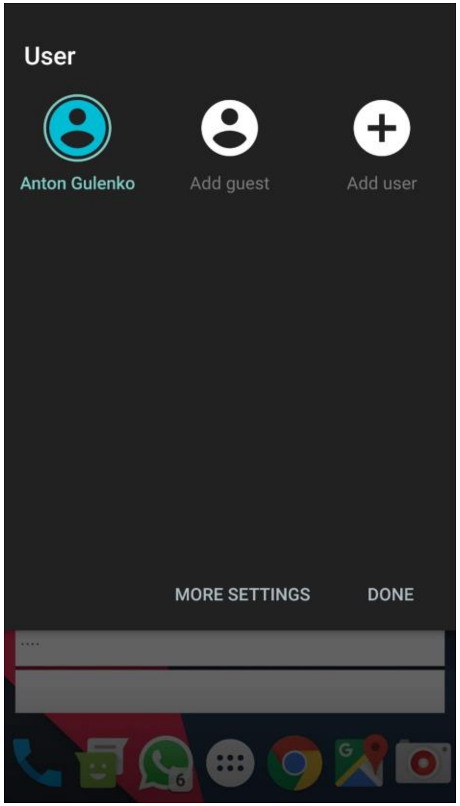
Use Cases for Virtual Machines (1/4)

- Run a different operating system than installed on the host system



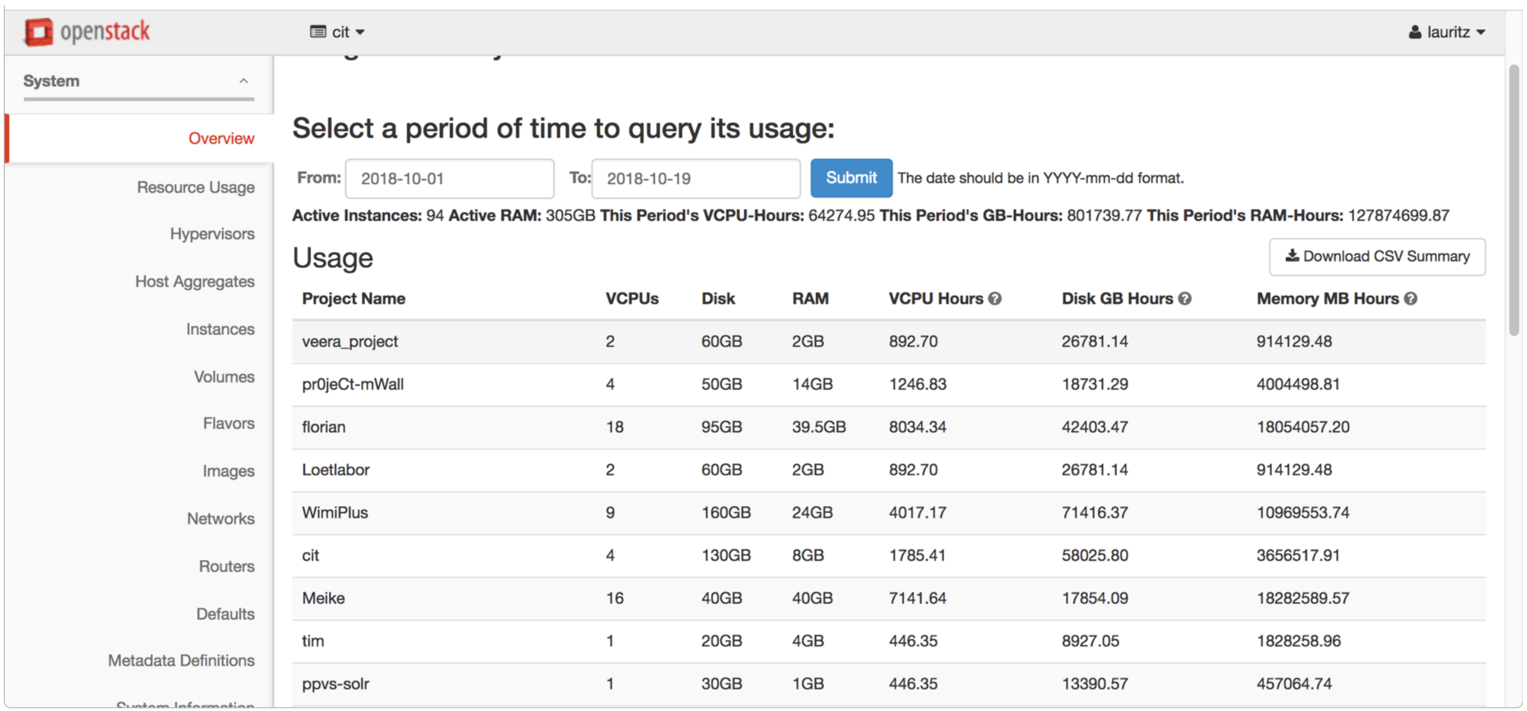
Use Cases for Virtual Machines (2/4)

- Operating multiple isolated environments on a host (e.g. different user accounts on a phone)



Use Cases for Virtual Machines (3/4)

- Pool of resources shared by multiple users and applications in a private cloud



Project Name	VCPUs	Disk	RAM	VCPU Hours	Disk GB Hours	Memory MB Hours
veera_project	2	60GB	2GB	892.70	26781.14	914129.48
pr0jeCt-mWall	4	50GB	14GB	1246.83	18731.29	4004498.81
florian	18	95GB	39.5GB	8034.34	42403.47	18054057.20
Loetlabor	2	60GB	2GB	892.70	26781.14	914129.48
WimiPlus	9	160GB	24GB	4017.17	71416.37	10969553.74
cit	4	130GB	8GB	1785.41	58025.80	3656517.91
Meike	16	40GB	40GB	7141.64	17854.09	18282589.57
tim	1	20GB	4GB	446.35	8927.05	1828258.96
ppvs-solr	1	30GB	1GB	446.35	13390.57	457064.74

Use Cases for Virtual Machines (4/4)

- Virtual machines available in a public cloud (AWS)

Linux | RHEL | SLES | Windows | Windows with SQL Standard | Windows with SQL Web
Windows with SQL Enterprise | Linux with SQL Standard | Linux with SQL Web | Linux with SQL Enterprise

Region: EU (Frankfurt)

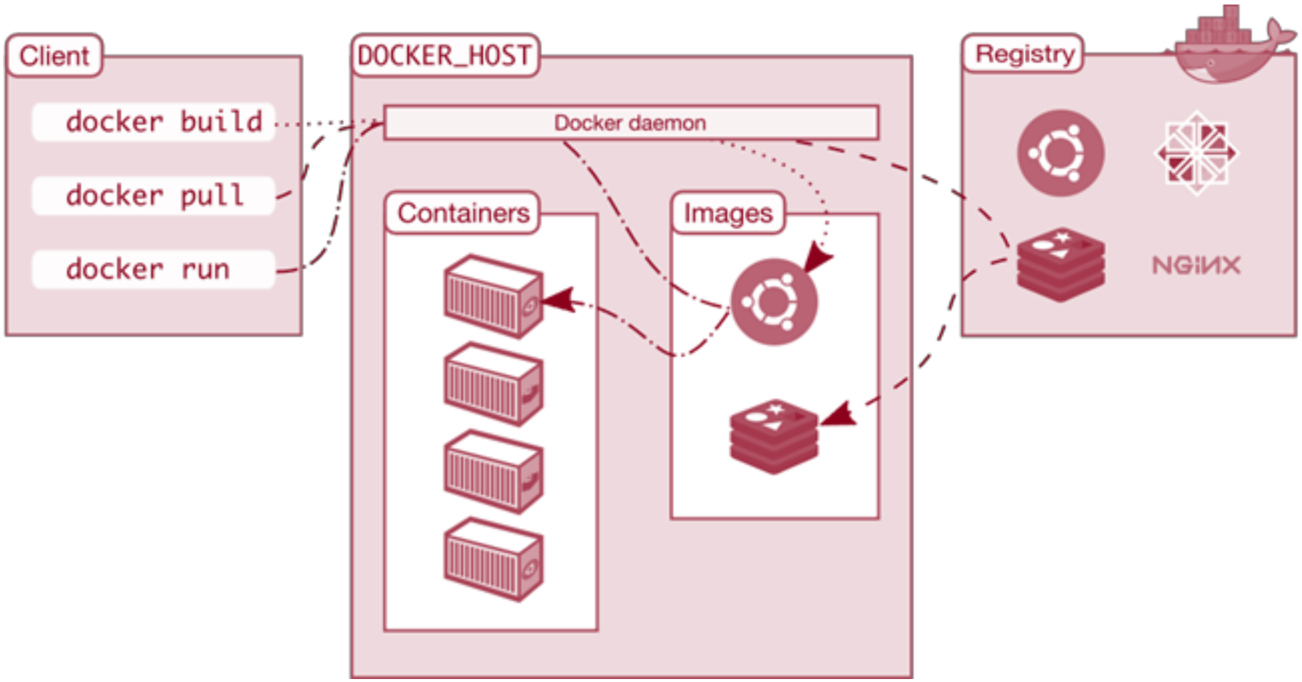
	vCPU	ECU	Memory (GiB)	Instance Storage (GB)	Linux/UNIX Usage
General Purpose - Current Generation					
t3.nano	2	Variable	0.5 GiB	EBS Only	$0.006 per Hour
t3.micro	2	Variable	1 GiB	EBS Only	$0.012 per Hour
t3.small	2	Variable	2 GiB	EBS Only	$0.024 per Hour
t3.medium	2	Variable	4 GiB	EBS Only	$0.048 per Hour
t3.large	2	Variable	8 GiB	EBS Only	$0.096 per Hour
t3.xlarge	4	Variable	16 GiB	EBS Only	$0.192 per Hour
t3.2xlarge	8	Variable	32 GiB	EBS Only	$0.384 per Hour
t2.nano	1	Variable	0.5 GiB	EBS Only	$0.0067 per Hour

Containers

- Lightweight "OS-level virtualization": virtual environments for single applications → processes run in isolation <u>within an OS</u>
  - Containers rely on containment features of the OS kernel (e.g. the *chroot* and *cgroups* features of Linux)
- Use cases:
  - Isolating applications on the same host from each other (→ resource sharing/pooling in a cluster)
  - Bundling dependencies and the environment of applications (libs and config) to ship software
- Containers can have different sizes, are usually started from images, and can be run on cloud services

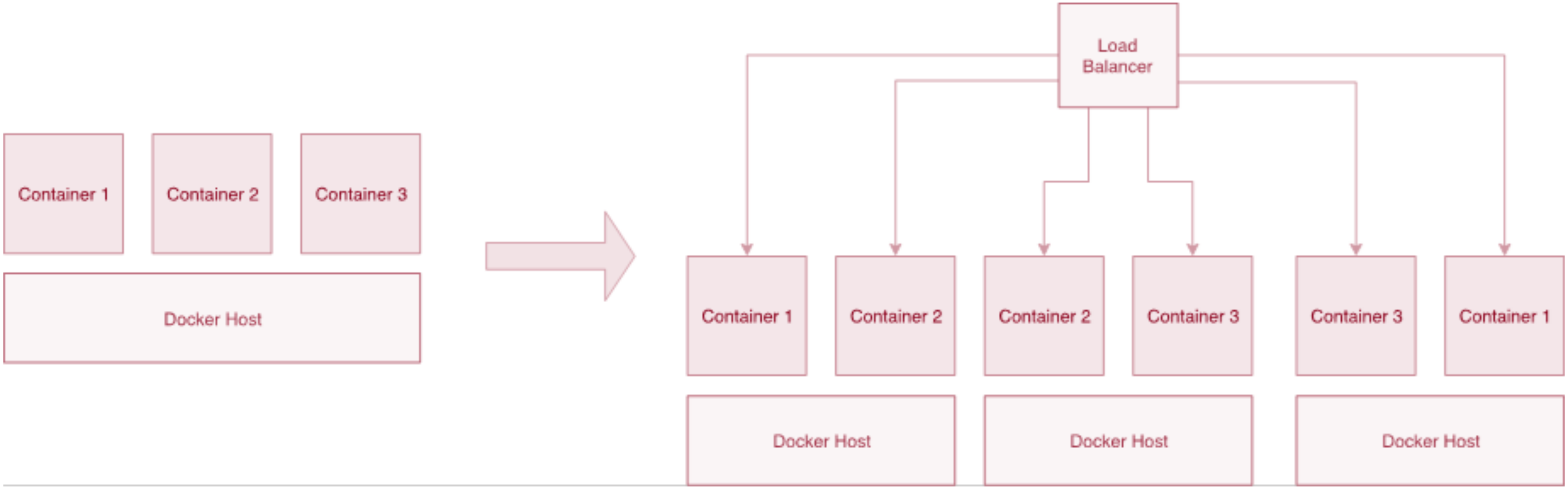
Use Cases for Containers (1/2)

- Build and ship your application in a container to be able to run it everywhere, e.g. development on your host, but then deployment to staging / production



Use Cases for Containers (2/2)

- Having many applications share the resources of a cluster



Containers vs. Virtual Machines

- Containers
  - No virtual hardware: cannot run an application build for a different architecture or operating system kernel
  - Reduced scope: single application in pre-build environments
  - Reduced isolation: containerized applications share the same kernel, but are isolated on process-level
  - Live migration of containers is less mature
- VMs
  - Larger images, slower VM startup, and more overhead per VM on a physical node

(Literature and) References

- References:

  - [1] P. Mell, T. Grance: “The NIST Definition of Cloud Computing”, Technical Report, National Institute of Standards and Technology, 2011, https://csrc.nist.gov/pubs/sp/800/145/final
  - [2] K. Scarfone, M. Souppaya, P. Hoffman: “Guide to Security for Full Virtualization Technologies”, 2010, https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-125.pdf