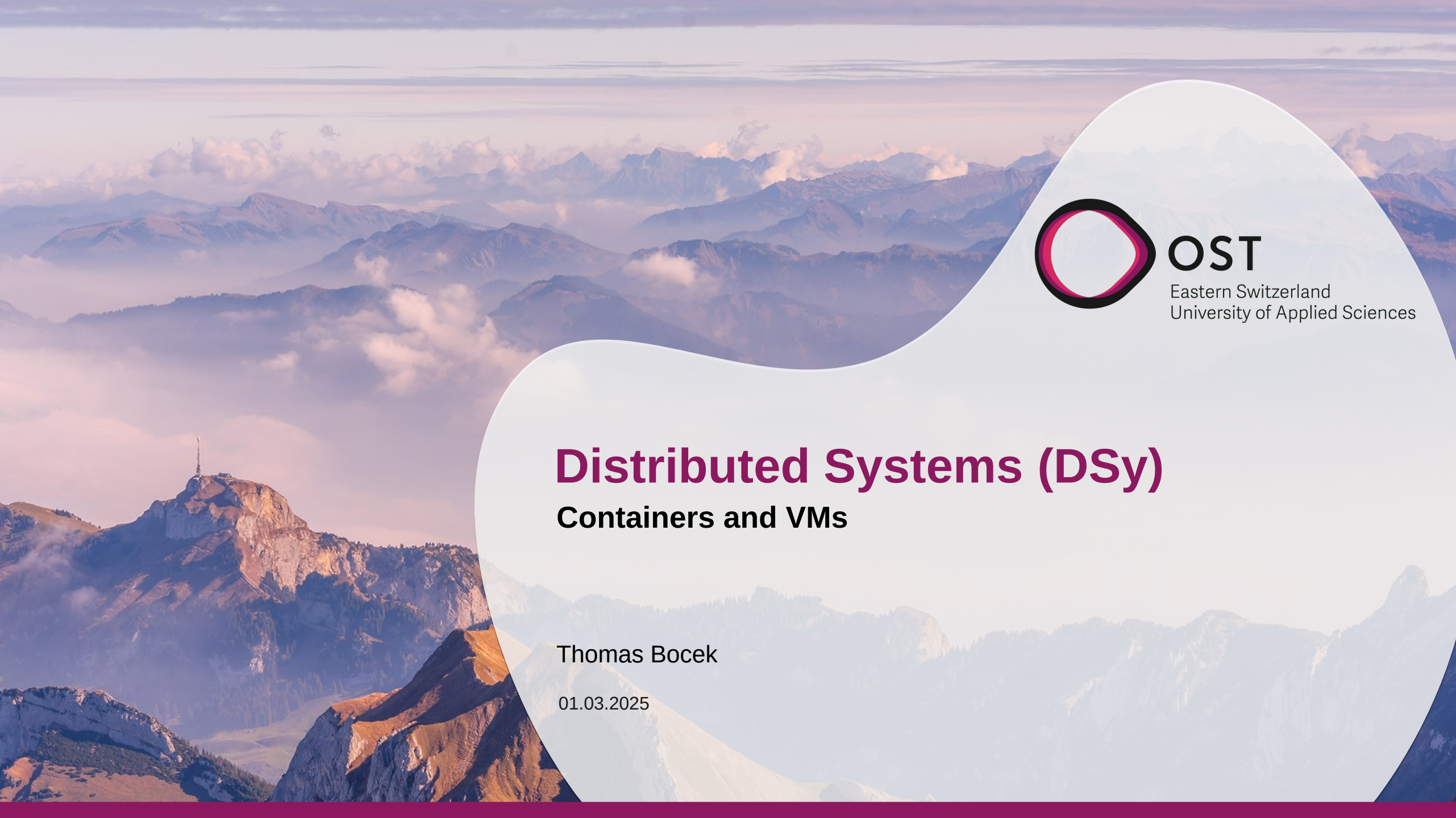


OST
Eastern Switzerland
University of Applied Sciences

Distributed Systems (DSy)

Containers and VMs

Thomas Bocek

01.03.2025

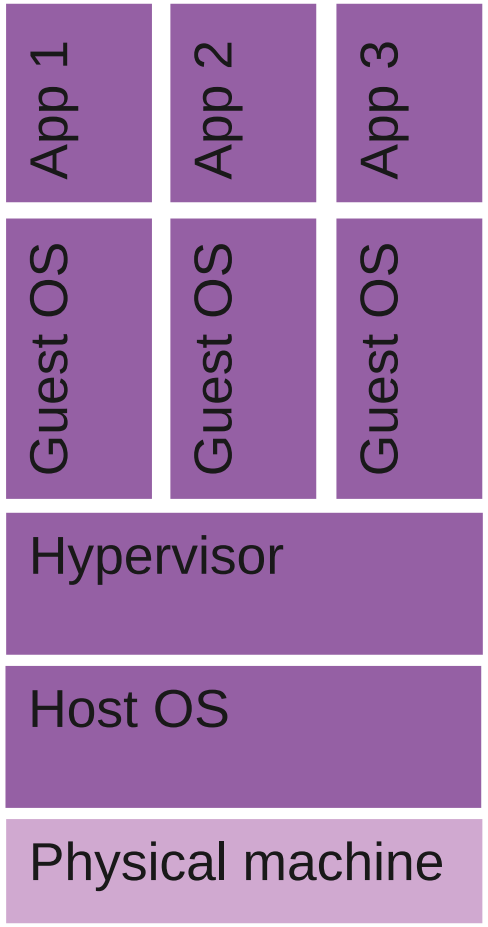
Learning Goals

- Lecture 3 (Containers and VMs)
  - What is the difference of VM / Container?
  - Next segment: practical examples

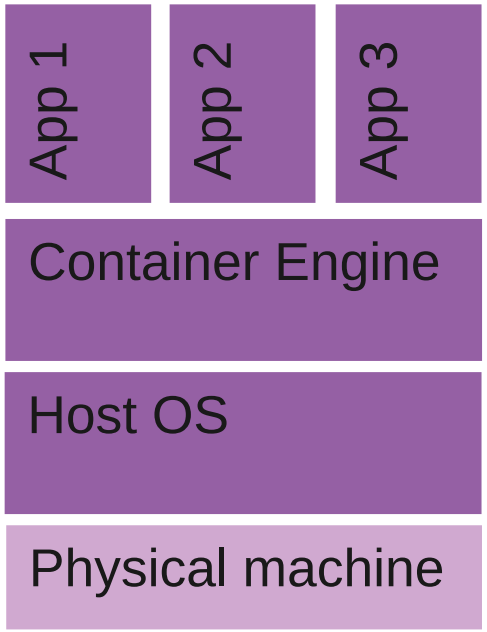
Virtualization

- Creation of a virtual machine that acts like a real computer with an operating system [source]
  - Host machine: machine where the virtualization software runs
  - Guest machine: virtual machine
- Hypervisor runs virtual machines
  - Type 1: bare-metal – e.g., Xen
    - "We built Amazon EC2 using a virtual machine monitor by the name of Xen" [source]
  - Type 2: hosted – e.g., VirtualBox
- Run unmodified OS with Intel VT-x and AMD-V, or paravirtualized if not present
  - E.g., VM should not access memory directly
- Needs to be the same architecture
  - Otherwise use emulation, e.g., QEMU
    - Ubuntu on a RISC-V processor
    - Qemu, opensbi, u-boot
  - Gaming console emulators: Snes9x, Mupen64 Plus, Switch [defunct]
- Virtual desktop infrastructure (VDI)
  - Interact with a virtual machine over a network
- Containers
  - Isolated user-space instances
  - OS support: isolations

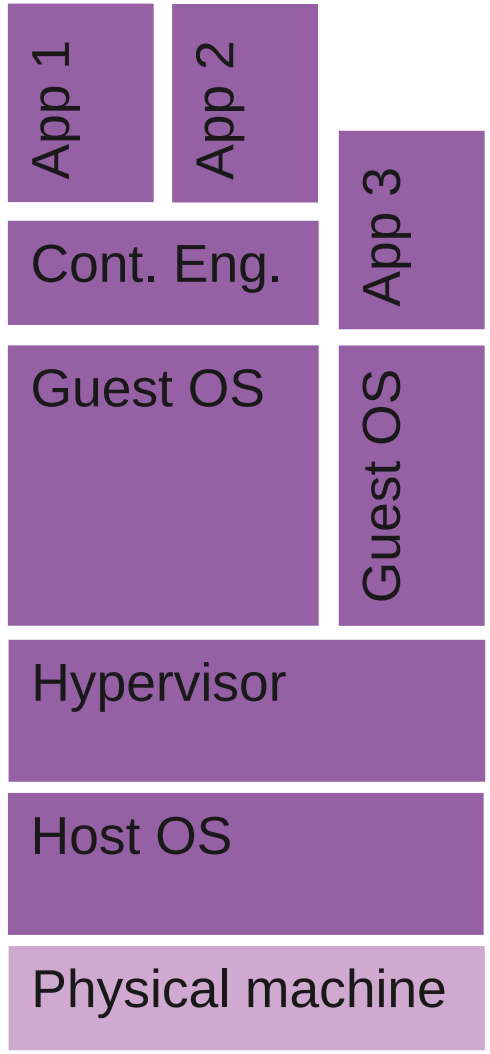
Virtualization - Visualization



- Virtual machines



- Container



- Both

OST

Comparison

Container

+ Reduced size of snapshots 2MB vs 45MB

+ Quicker spinning up apps

+ / - Available memory is shared

+ / - Process-based isolation (share same kernel)

Use case: complex application setup, with container less complex configuration

Providers: ECS, Google Cloud Run, Digital Ocean App Platform, ...

Virtual Machine

+ App can access all OS resources

+ Live migrations

+ / - Pre allocates memory

+ / - Full isolation

Use case: better hardware utilization / resource sharing

EC2, Virtual Machines, Compute Engine, Droplets

Market shares, others

Prices / VM on e.g., AWS

Virtual Machines

- On-Demand
  - Machine
  - Data transfer
  - IP address
- Spot instances (discount when not needed)
- Reserved Instances
- Comparison, comparison, comparison
  - Not easy to compare
  - Optimize for cost → provider changes cost structure, you need to adapt again for optimizing

On-Demand Pricing

Instance Type	AWS	Azure	Google	AWS pricing (per hour)	Azure Pricing (per hour)	Google pricing (per hour)
General purpose	m6g.xlarge	B4MS	e2-standard-4	$0.154	$0.166	$0.134
Compute optimized	c6g.xlarge	F4s v2	c2-standard-4	$0.136	$0.169	$0.208
Memory optimized	r6g.xlarge	E4a v4	m1-ultramem-40	$0.202	$0.252	$6.293
Accelerated computing	p2.xlarge	NC4as T4 v3	a2-highcpu-1g	$0.90	$0.526	$3.678

SIMFORM

https://www.simform.com/blog/compute-pricing-comparison-aws-azure-googlecloud/

	aws	Microsoft Azure	Google Compute Engine
CPUs	1	1	1
RAM	2GB	2GB	3.75GB
Storage	30GB free	16GB	$.02/GB per month
Bandwidth	10GB free	5GB free	$.12/GB per month
Price	$7.00/month	$18.97/month	$15.60/month
	Visit site »	Visit site »	Visit site »

https://www.hostingadvice.com/how-to/aws-azure-google-cloud-alternatives/

OST

Introduction

- Containers – docker?
  - LXC (Linux Containers) - Lower abstraction level and direct use of Linux kernel features
  - systemd-nspawn - Part of the systemd project, minimalist container manager
  - Solaris Zones - Oracle/Sun-specific container technology
  - Linux-VServer - Kernel patch for Linux, older virtualization technology
  - OpenVz - Operating system-level virtualization, popular hosting tool
  - rkt [ended]- CoreOS-developed alternative to Docker, focus on security
  - Singularity - Scientifically oriented container solution, HPC-friendly
  - Application kernel to sandbox applications
    - syd, Bubblewrap, Firejail, GVisor, and minijail
- Docker / docker-compose for software development

OST

Introduction - Docker

- VMs “Startup time in minutes” [link]
- MicroVM
  - Purpose-built for serverless workloads, everything else removed → lightweight virtual machine
  - E.g., Firecracker
    - Startup in 125ms
    - Used and developed by AWS
  - E.g,. microvm
    - Inspired by Firecracker
- Choice: Security (e.g., Firecracker) vs. convenience (e.g., Docker) – both are secure if everything implemented correctly

- Docker is a containerization platform
  - Packages software into containers
    - Existing images on Docker Hub
  - Containers are isolated from each other
    - Communicate over well-defined channels
  - Docker, Inc is the company behind its tooling
  - Alternatives: Podman
    - Different architecture, docker runs a daemon and you connect via CLI, podman does not [source]
    - Podman supports docker-compose