ACADEMIND

Compute Services: EC2 & More

Using AWS to run compute tasks in the cloud

- Which Compute Services Does AWS Offer?
- Getting Started with the EC2 Service
- Configuring & Using EC2

ACADEMIND

Which Services Does AWS Offer?

Compute

Data Storage

Database

Networking & Content Delivery

Application Integration

Security

Cloud Management

Migration & Edge Computing

Analytics & Data Ingestion

Machine Learning & Artificial Intelligence

Developer Tools

Business Applications

AWS Compute Services

ECS / EKS
(Elastic Container Service)

An alternative to EC2 - for **containerized workloads**

Run clusters of containers in the cloud

Managed (with vast amount of configuration options)

Run any kind of containerized workload

EC2
(Elastic Compute Cloud)

One of the **most important & popular** services!

Rent a (virtual) remote server / computer

Fully configurable

Run any kind of workload in the cloud

Lambda
(Serverless Code Execution)

The **most popular serverless** compute service

Run code without provisioning any infrastructure

No access to the underlying machine or OS

Run any kind of code upon pre-defined events

What Are "Containers"?

Containers are "**packages**" of code + the code's dependencies (e.g., OS, required software)

Containers allow developers to distribute and deploy reproducible code environments (including the code itself)

No server configuration required
(since the container already includes the operating system, software, configuration etc.)

Containers can be deployed into **all environments that support containers**

Supporting environments are still computers / servers — they **host the containers**, not the app itself though

ECS / EKS

What Are "Serverless" Services?

Serverless services allow you to run code **without configuring or controlling any servers**

You can perform tasks in response to events by just providing the code that should be executed

AWS Lambda

Serverless services allow you to **focus on your code**, instead of the environment that runs the code

Often, multiple serverless services / tasks **must be combined** to handle more complex workloads

AWS manages the (hidden) underlying server configuration

Understanding EC2

Rent a (virtual) remote machine

"A computer in the cloud"

It's actually (typically) not an entire machine

Rent an Instance

Fully configurable

Choose hardware profile, operating system & install any software

Run any kind of application or task on the instance

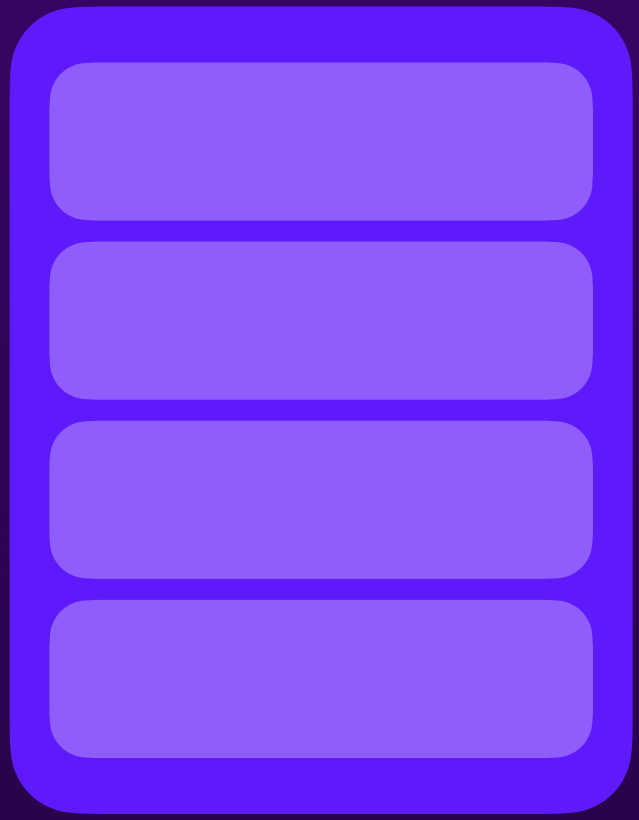
A virtual machine using a "slice" of a the physical machine

Completely isolated from other instances

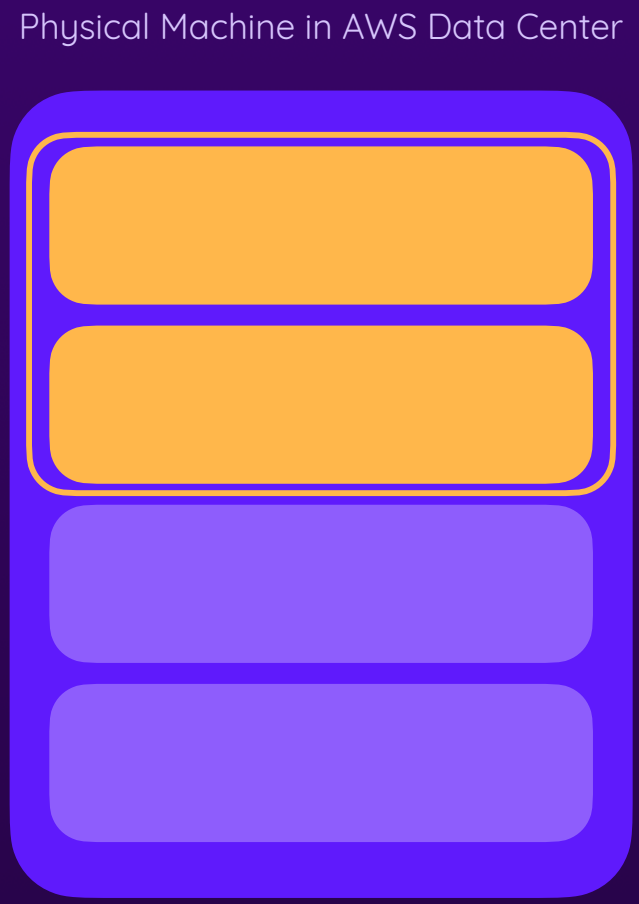
Dedicated hardware assigned to the virtual machine

Rent A Slice Of A Computer

Physical Machine in AWS Data Center



Rent A Slice Of A Computer



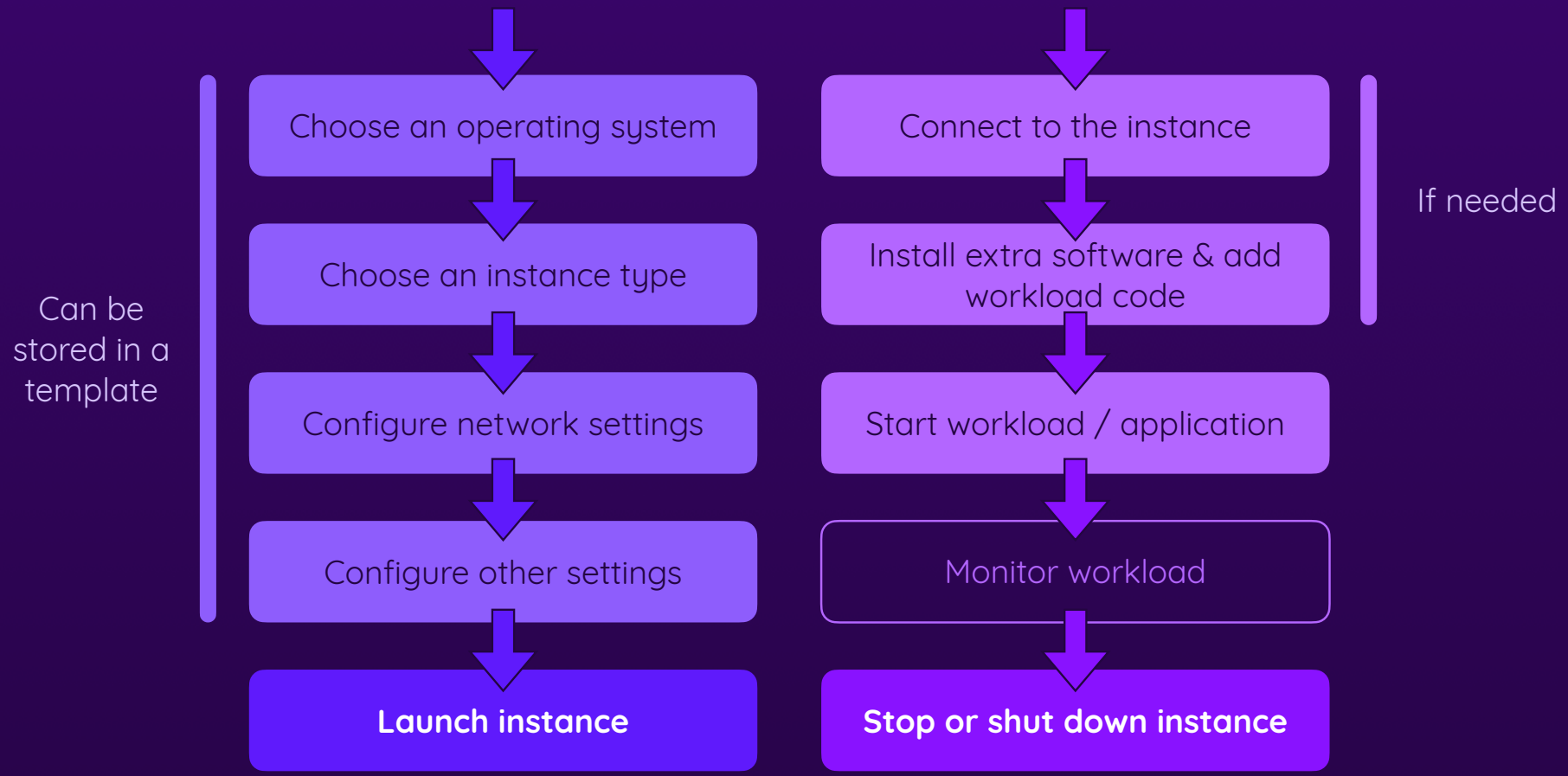
You rent a “virtual server”

An **EC2 Instance**

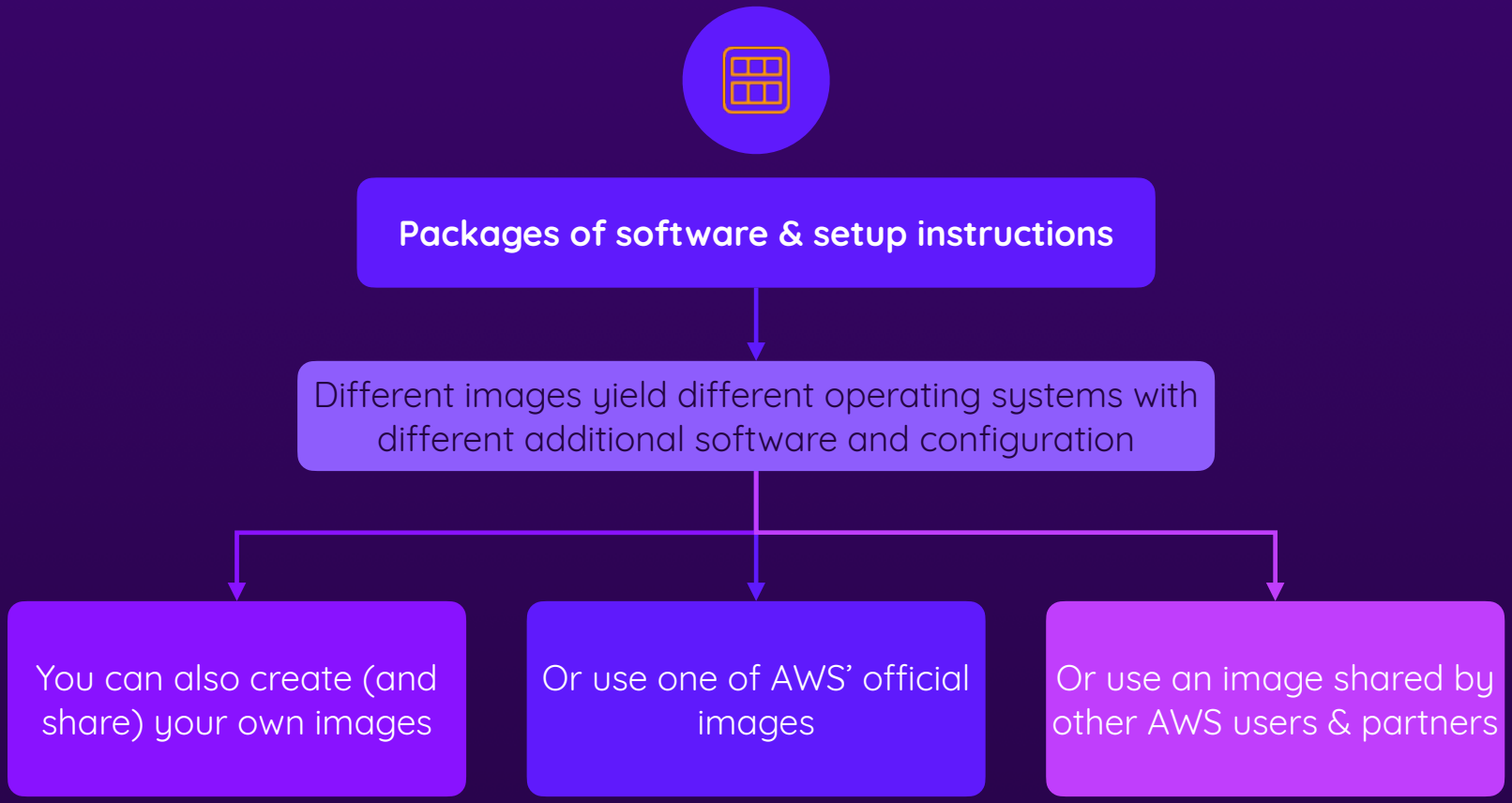
A slice of the physical machine

Fully isolated from other slices (other instances), with its own **dedicated hardware**

Using EC2 Instances



Amazon Machine Images (AMIs)



EC2 Pricing

On Demand Instances	Spot Instances	Savings Plans	Reserved Instances
Default & most flexible option	Must be selected	Must be bought separately	Must be bought separately
Pay for usage	Spare instances, can be reclaimed any time	Pay in advance (for chosen amt. of usage)	Pay in advance (for chosen instance types)
No discounts	Discounts over on-demand pricing	Discounts over on-demand pricing	Discounts over on-demand pricing
Price depends on chosen config	Price depends on chosen config	You can use other compute services	Not very flexible & only EC2
	Ideal for workloads that can be interrupted	Ideal if you can commit long-term	

Summary

Multiple Compute Services	EC2 Instances	Running Workloads via EC2
ECS / EKS for containerized workloads	EC2 allows you to “rent” “slices” of real machines: Instances	Connect to EC2 instances via ssh or EC2 Instance Connect
Lambda for serverless compute tasks	Each instance is fully isolated from other instances	Run commands, install software, download code, etc.
“Serverless” = You only provide the code, no server config	Instance configuration (AMI, instance type, etc.) is up to you	Run one or multiple scripts / commands / programs
EC2 for fully customizable server configuration	AMIs define the operating system + base software / config	Stop or terminate whenever you want
Run any workload / task on EC2	Control network access via security groups	Advanced options & different pricing models

What About Lambda & ECS / EKS?

Will be covered later in the course!

But also less important for the exam