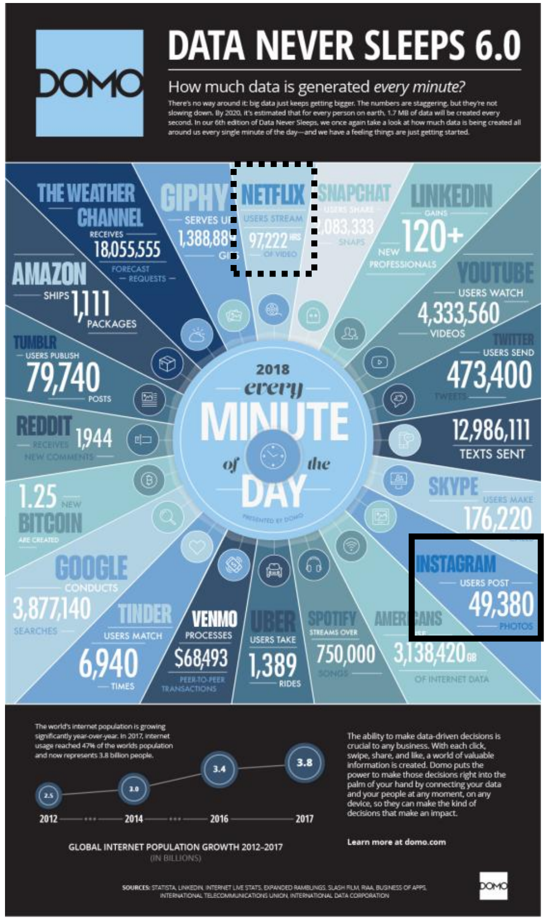
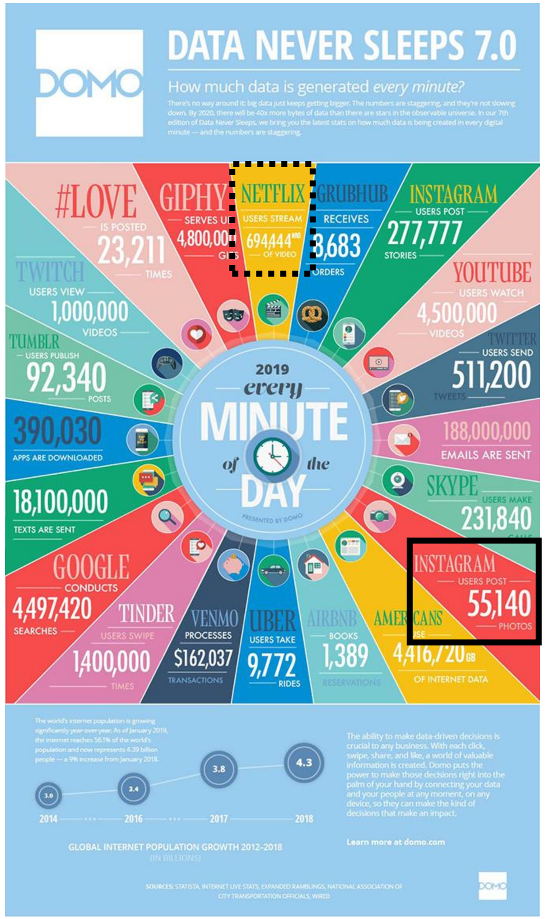
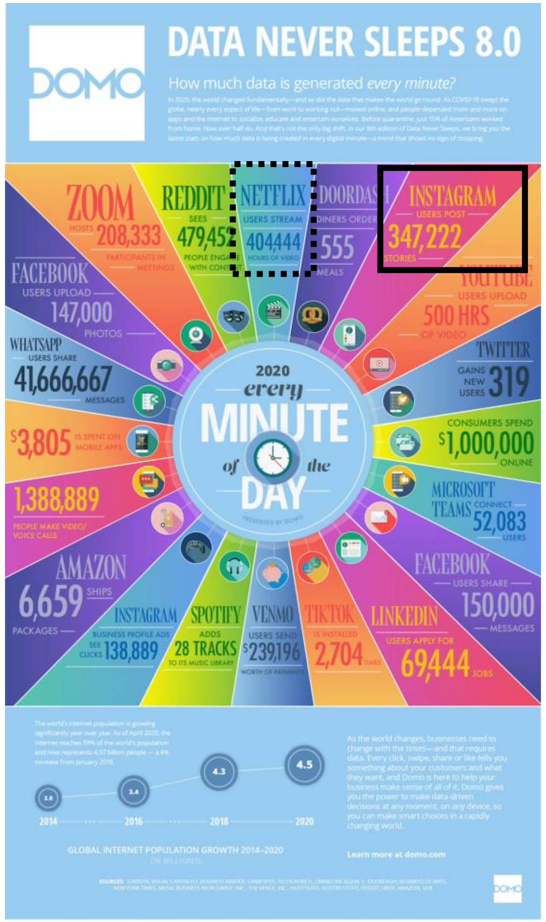
BIG DATA AND CLOUD PLATFORMS

Cloud computing

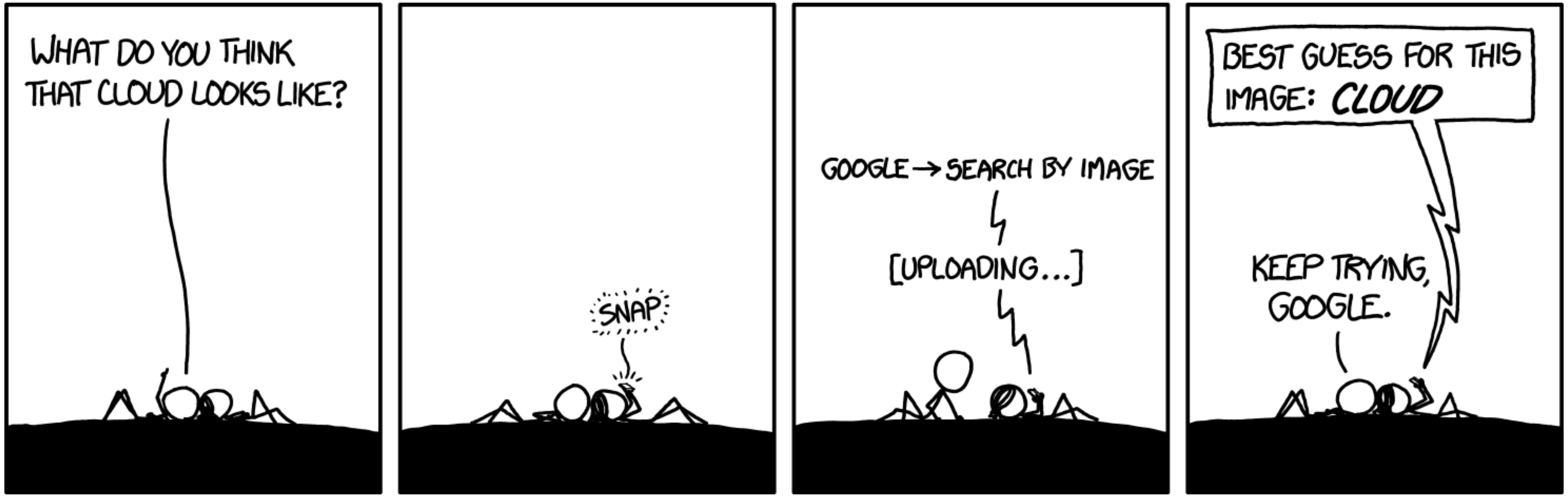
Why going cloud?







https://www.domo.com/learn/data-never-sleeps-8


WHAT DO YOU THINK THAT CLOUD LOOKS LIKE?
SNAP
GOOGLE → SEARCH BY IMAGE
[UPLOADING...]
BEST GUESS FOR THIS IMAGE: CLOUD
KEEP TRYING, GOOGLE.

https://xkcd.com/1444/

Why going cloud?

Cloud computing (National Institute of Standards and Technology)

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

- On-demand self-service (consume services when you want)
- Broad network access (consume services from anywhere)
- Resource pooling (infrastructure, virtual platforms, and applications)
- Rapid elasticity (enable horizontal scalability)
- Measured service (pay for the service you consume as you consume)

Digital transformation involves the **cloud** to create/change business flows

- Often involves changing the company culture to adapt to this new way of doing business
- One of the end goal is to meet ever-changing business and market demand

Why going cloud?

Goal: adjusts capacity to have predictable performance at the lowest cost

Scalability that is not possible on premises

- Scale from one to thousands of servers

Elasticity

- Automatically scale resources in response to run-time conditions
- Adapt to changes in workload by turning on/off resources to match the necessary capacity
- Core justification for the cloud adoption

Why going cloud?

Hardware scalability

- No longer think about rack space, switches, and power supplies, etc.

Grow storage from GBs to PBs

- 1PB: one hundred 10TB Enterprise Capacity 3.5 HDD hard drives

https://blog.seagate.com/business/linus-tech-tips-want-petabyte-system/

Why going cloud?

Resource pooling

- Enable cost-sharing, a resource to serve different consumers
- Resources are dynamically reassigned according to demands
- Based on virtualization, running multiple virtual instances on top of a physical computer system
- Economy of scale for physical resources

Reliability

- Built to handle failures
- Fault-tolerant or highly available

Why going cloud?

Worldwide **deployment**

- Deploy applications as close to customers as possible
    - E.g., to reduce network latency
- Improve data locality
- Compliant to privacy regulations (e.g., GDPR)

Measured **quality of service**

- Services leverage a quantitative qualitative metering capability making pay-as-you-go (or pay-per-use) billing and validation of the service quality available

Why going cloud?

Service **integration**

- Do not reinvent the wheel, eliminate repetitive tasks
  - Use services that solve common problems (e.g., load balancing, queuing)
- Abstract and automatically adapt the architecture to requirements
  - E.g., create (test) environments on demand

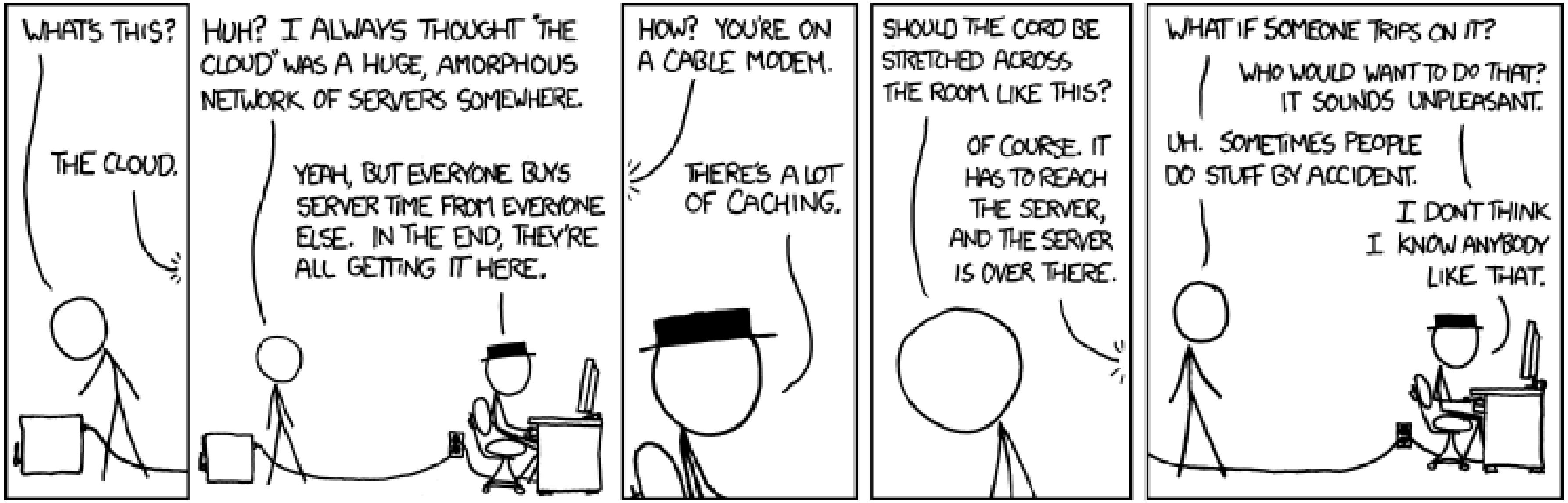
Integration and abstraction are drivers of change

- From databases to data platforms
- From on-premises to serverless architectures
- From custom to standardized data pipelines

Is cloud a silver bullet?

Cloud computing is the outsourcing of a company’s hardware and software architecture

- Which are the risks and issues?

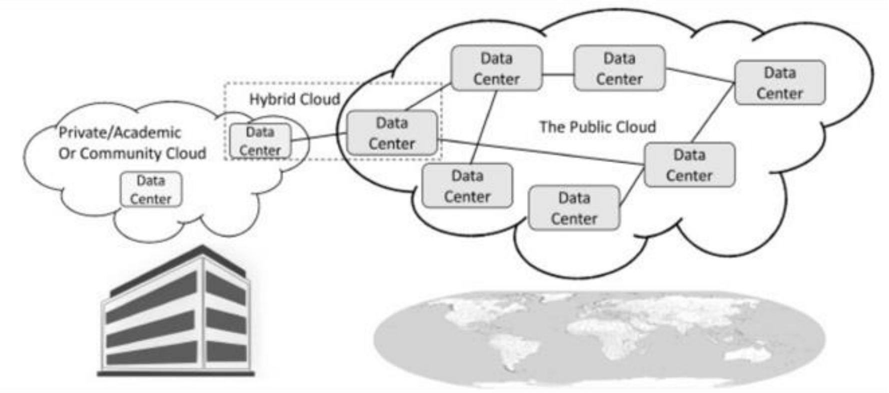

WHAT'S THIS?
THE CLOUD.
HUH? I ALWAYS THOUGHT "THE CLOUD" WAS A HUGE, AMORPHOUS NETWORK OF SERVERS SOMEWHERE.
YEAH, BUT EVERYONE BUYS SERVER TIME FROM EVERYONE ELSE. IN THE END, THEY'RE ALL GETTING IT HERE.
HOW? YOU'RE ON A CABLE MODEM.
THERE'S A LOT OF CACHING.
SHOULD THE CORD BE STRETCHED ACROSS THE ROOM LIKE THIS?
OF COURSE. IT HAS TO REACH THE SERVER, AND THE SERVER IS OVER THERE.
WHAT IF SOMEONE TRIPS ON IT?
WHO WOULD WANT TO DO THAT? IT SOUNDS UNPLEASANT.
UH. SOMETIMES PEOPLE DO STUFF BY ACCIDENT.
I DON'T THINK I KNOW ANYBODY LIKE THAT.

https://xkcd.com/908/

Cloud computing: types of cloud

There are different types of cloud

- **Public**: accessible to anyone willing to pay (e.g., Microsoft, AWS, Google)
- **Private**: accessible by individuals within an institution
    - In public cloud, any resources that you are not using can be used by other
    - Users share the costs
    - Cost-sharing disappears in private clouds
- **Hybrid**: a mix of the previous



Cloud computing: types of cloud

Cloud services are hosted in separate geographic areas

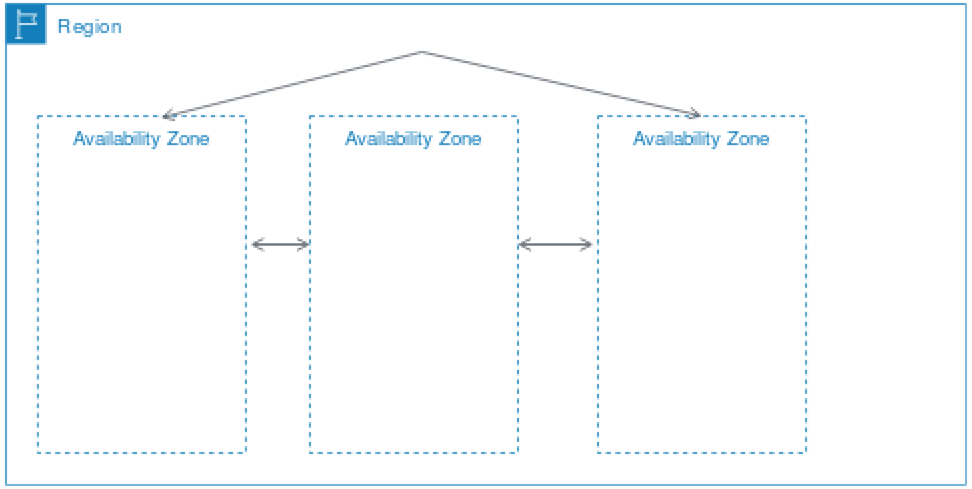
- Locations are composed of **regions** and **availability zones**

Region (e.g., us-east-1)

- Is an independent geographical area that groups data centers
- Has availability zones

Availability zones in a region

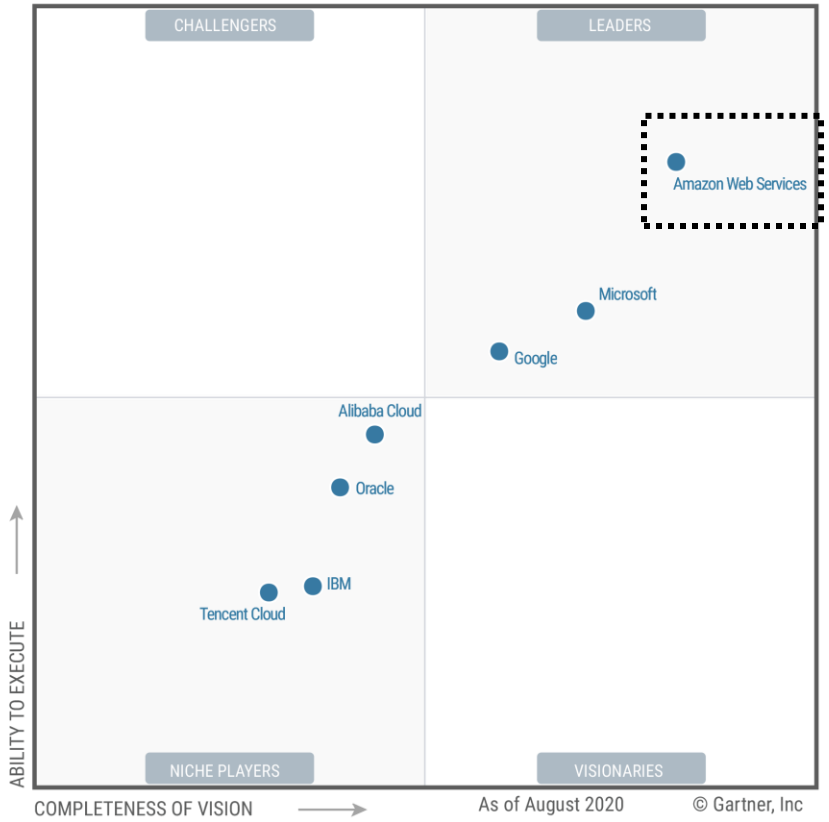
- A data center
- Connected through low-latency links
- Resources are usually replicated across zones but not regions



https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-regions-availability-zones.html

Cloud computing: principal vendors

Figure 1. Magic Quadrant for Cloud Infrastructure and Platform Services



https://www.gartner.com/en/research/methodologies/magic-quadrants-research

Gartner Magic Quadrant

- Understanding the technology providers to consider for an investment
- **Leaders** execute well and are well positioned for tomorrow
- **Visionaries** understand where the market is going but do not yet execute well
- **Niche Players** focus successfully on a small segment, or are unfocused and do not out-innovate or outperform others
- **Challengers** execute well but do not demonstrate an understanding of market direction
- Focusing on leaders isn’t always the best
  - A niche player may support needs better than a market leader. It depends on how the provider aligns with business goals

Cloud computing: deployment models

On a cloud architecture, you can rely on serverless or managed services

Serverless

- Standalone independent services built for a specific purpose and integrated by cloud service provider
- No visibility into the machines
  - There are still servers in serverless, but they are abstracted away
  - No server management, do not have to manage any servers or scale them
  - E.g., when you run a query on BigQuery you do not know how many machines were used
- Pay for what your application uses, usually per request or per usage

(Fully) Managed

- Visibility and control of machines
  - You can choose the number of machines that are being used to run your application
- Do not have to set up any machines, the management and backup are taken care for you
- Pay for machine runtime, however long you run the machines and resources that your application uses

https://cloud.google.com/blog/topics/developers-practitioners/serverless-vs-fully-managed-whats-difference (accessed 2020-08-01)

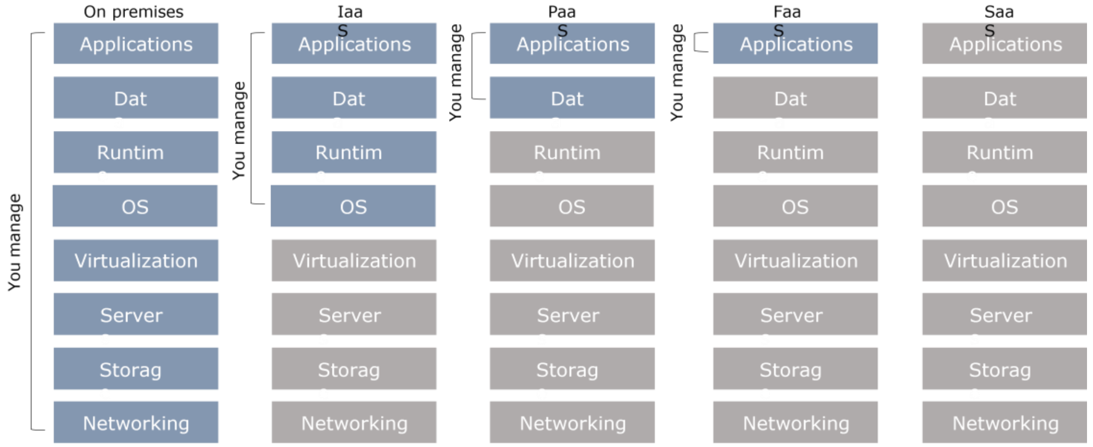
Cloud computing: deployment models

Understanding architectures is paramount to successful systems

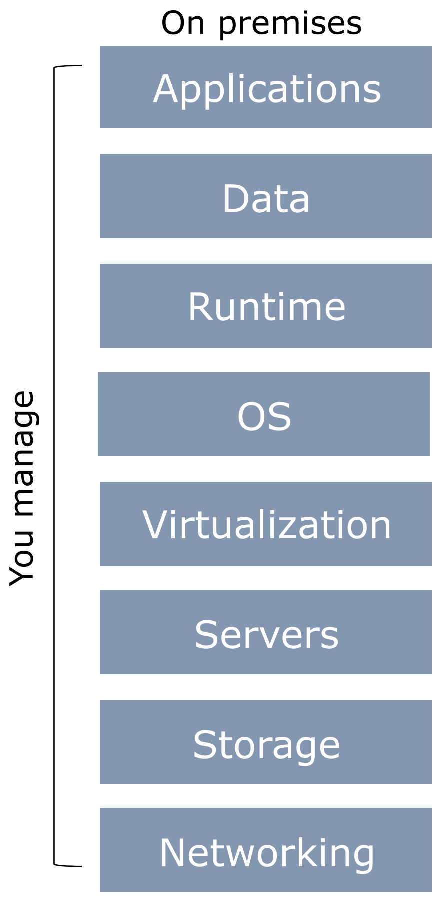
- Good architectures help to scale
- Poor architectures cause issues that necessitate a costly rewrite

XaaS (anything as a service)

- A collective term that refers to the delivery of anything as a service
- It encompasses the products, tools and technologies that vendors deliver to users



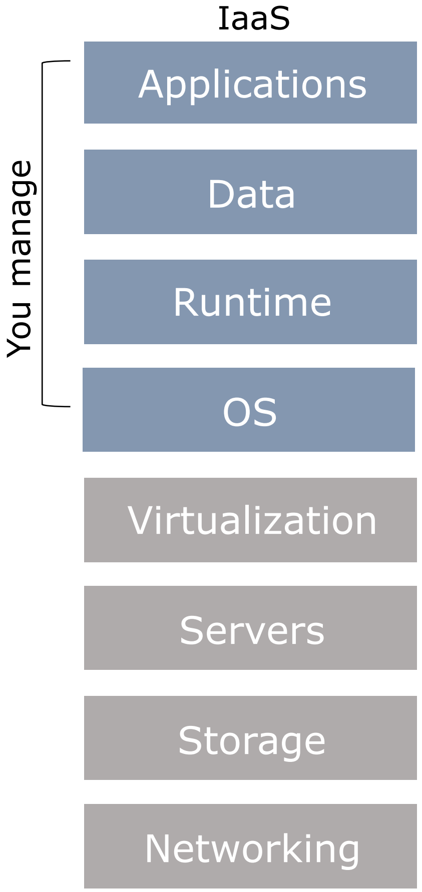
Cloud computing: deployment models



On-premises

- Provisioning servers is time-consuming
  - A non-trivial environment is hard to set up
- Require dedicated operations people
- Often a distraction from strategic tasks

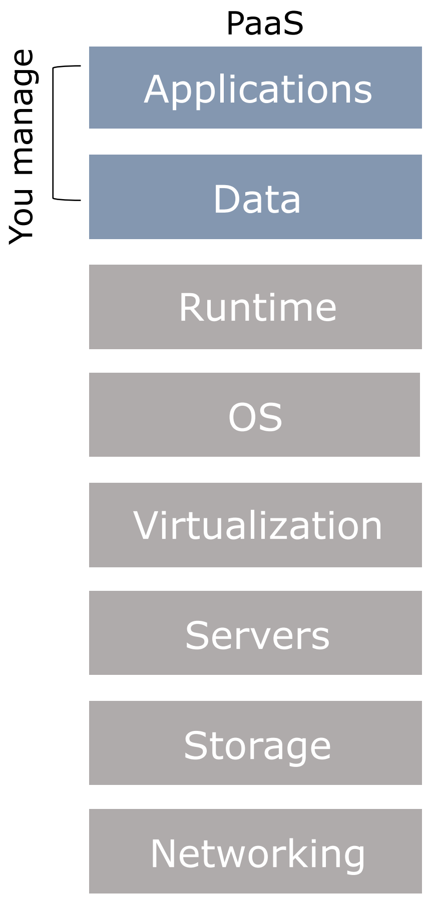
Cloud computing: deployment models



Infrastructure as a service (IaaS)

- A computing infrastructure provisioned and managed over the internet (e.g., AWS EC2)
- Avoid expense/complexity of buying/managing physical servers/data-centers
- IaaS overcomes issues on-premises
- Possibly requires to manage many environments

Cloud computing: deployment models



Platform as a Service (PaaS)

- A development and deployment environment in the cloud (e.g., AWS Elastic Beanstalk)
- Support complete application life-cycle: building, testing, deploying, etc.
- Avoid expense/complexity of managing licenses and application infrastructure

Cloud computing: deployment models

PaaS and **containers** are potential solutions to inconsistent infrastructures

PaaS provides a platform for users to run their software

- Developers write software targeting features/capabilities of the platform

Containerization isolates an application with its own environment

- Lightweight alternative to full virtualization
- Containers are isolated but need to be deployed to (public/private) server
- Excellent solution when dependencies are in play
- Housekeeping challenges and complexities

Cloud computing: deployment models

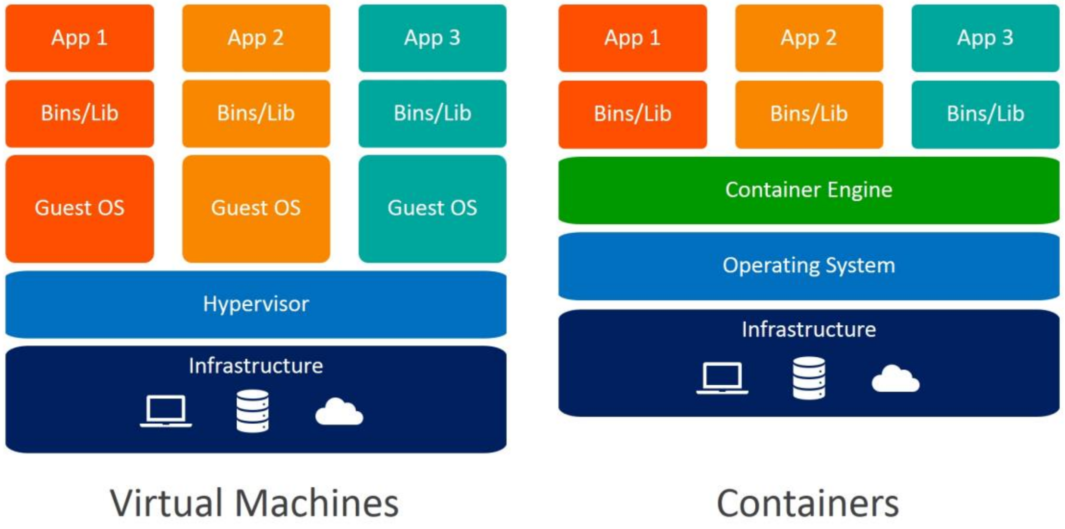
Containers and virtual machines are packaged computing environments

Containers

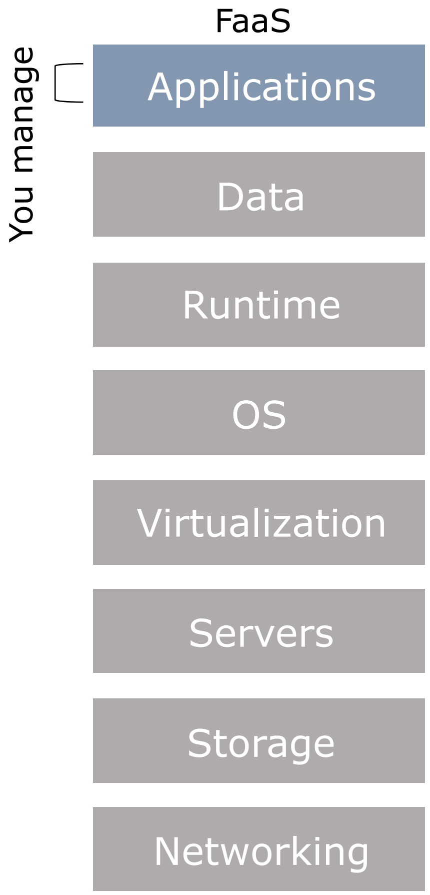
- On top of physical server and its host OS
- Share the host OS kernel
- Shared components are read-only
- “Light”, take seconds to start

Virtual machines

- Emulate a hardware/software system
- On top of a hypervisor (VM monitor)



Cloud computing: deployment models



Function as a Service (FaaS)

- A coding environment, cloud provider provisions platform to run the code (e.g., AWS Lambda)
- Infrastructure provisioning and management are invisible to the developer

Cloud computing: deployment models

Principles of FaaS architectures

- FaaS is based on a serverless approach, use a compute service to execute code on demand
- Every function could be considered as a standalone service
- Write single-purpose stateless functions

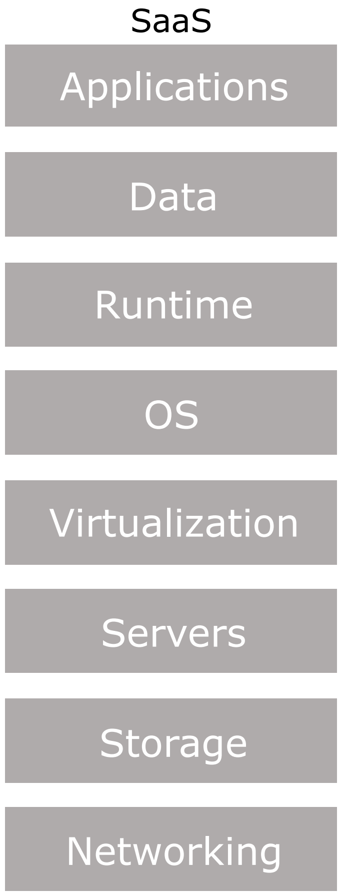
Functions react to events

- Design push-based, event-driven pipelines
- Create thicker, more powerful front ends
- Embrace third-party services (e.g., security)

FaaS is not a silver bullet

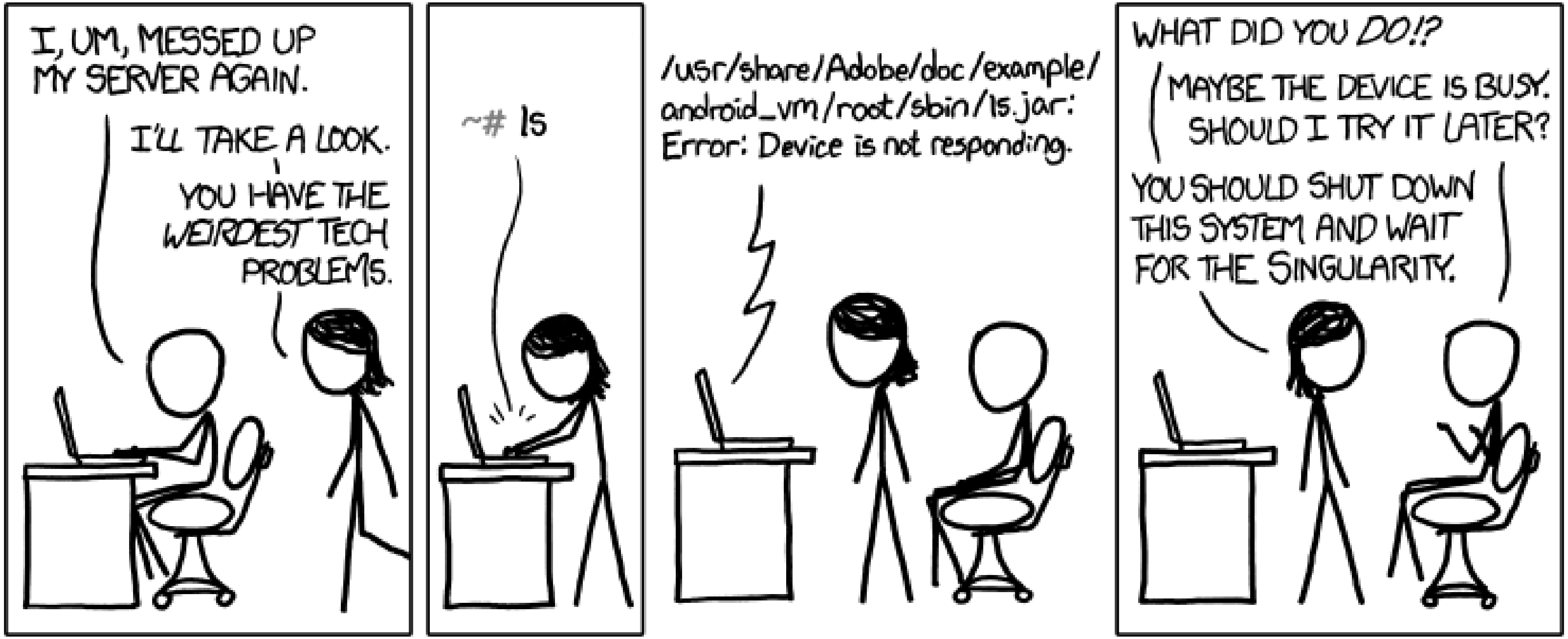
- Not appropriate for latency-sensitive applications
- Strict specific service-level agreements
- Migration costs
- Vendor lock-in can be an issue

Cloud computing: deployment models



Software as a service (SaaS)

- An application environment
- Access cloud-based apps over the Internet (e.g., email, Microsoft Office 365)



I, UM, MESSED UP MY SERVER AGAIN.
I'LL TAKE A LOOK.
YOU HAVE THE WEIRDEST TECH PROBLEMS.

```
~# ls
```

```
/usr/share/Adobe/doc/example/android_vm/root/sbin/ls.jar:
Error: Device is not responding.
```

WHAT DID YOU *DO!?*
MAYBE THE DEVICE IS BUSY. SHOULD I TRY IT LATER?
YOU SHOULD SHUT DOWN THIS SYSTEM AND WAIT FOR THE SINGULARITY.

https://xkcd.com/1084/