Introduction

Introduction to Web Systems Architecture

- Web Systems
- The Cloud
- Web Application Architectures
- Tradeoffs and Challenges
- Security of Web Systems

Web Systems

“

*a **system** that uses internet **web** technologies to deliver services, to users or other systems or applications*

”

Reference: Wikipedia

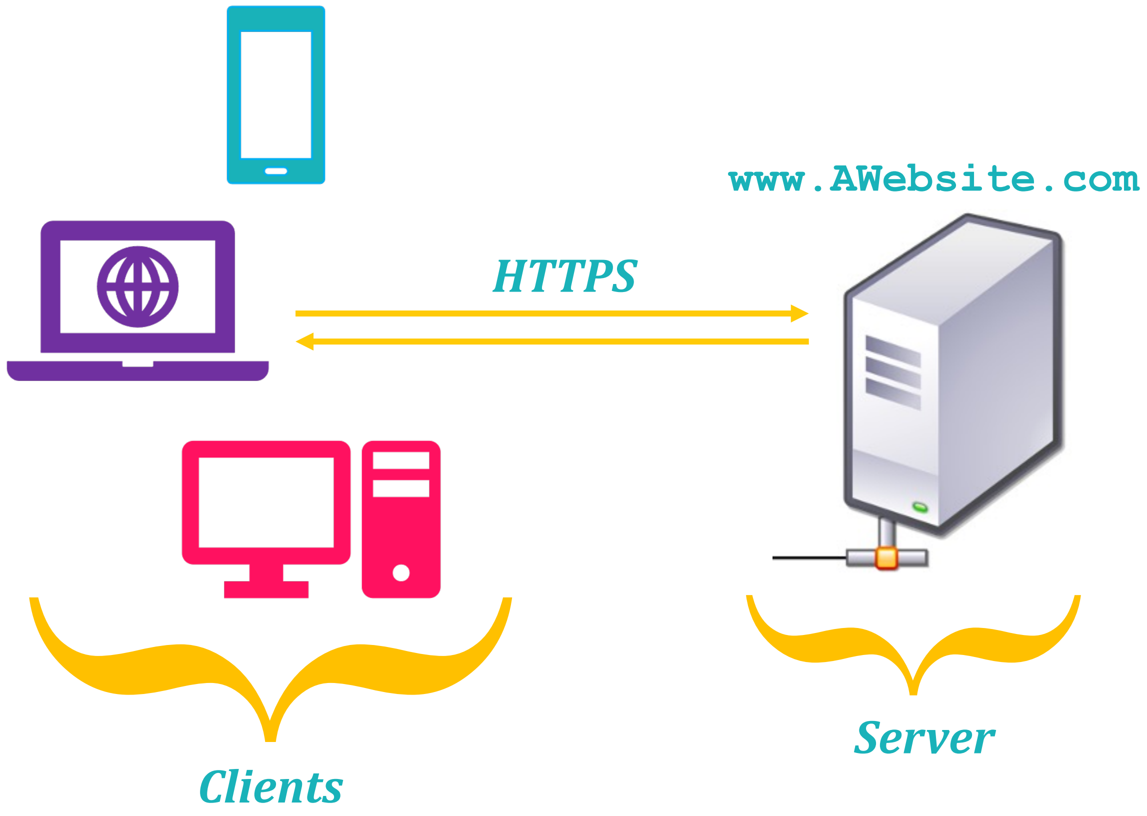
Web Systems

- Client-Server
  - Client: Service Requester
  - Server: Service Provider

Traditional

- On-premises server owned by service provider
- Server maintained by service provider
- Communicate over the internet

- What are ***Web 1.0***, ***Web 2.0***, and ***Web 3.0***?



Web Systems: Cloud-based

*"Instead of buying, owning, and maintaining physical data centers and servers, you can access technology services, such as computing power, storage, and databases, on an **as-needed basis from a cloud**."*

- From on-premises servers to the cloud-provider's infrastructure

- Infrastructure Services
  - Patching
  - Operating system maintenance
  - **Capacity provisioning!**

- Benefits
  - Agility
  - Cost Savings
  - Availability

Reference: https://aws.amazon.com/what-is-cloud-computing/

Cloud Computing: What is it?

- Computing as a **utility**
  - The illusion of infinite computing resources available on demand
  - The elimination of an up-front commitment by users
  - Ability to pay for use of computing resources in a short-term basis as needed and release them as needed

- **Applications** delivered as **services** over the internet

- **NIST (National Institute of Standards and Technology)** Definition

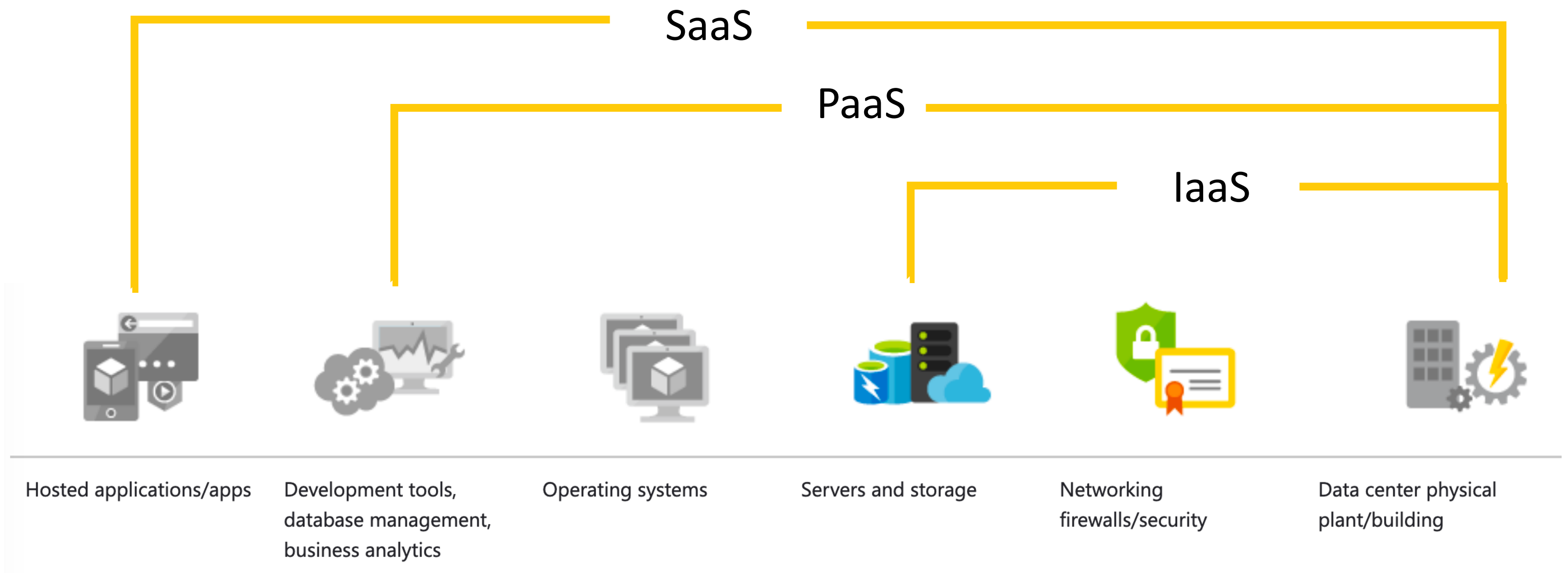
  *"a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This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

Cloud: Characteristics

- On-demand Self-service
- Broad Network Access
- Resource Pooling
- Rapid Elasticity
- Measured Services

Cloud: Service Models

Cloud computing {
- Application + Cloud = **SaaS** (Software as a service)
- Platform + Cloud = **PaaS** (Platform as a service)
- Infrastructure + Cloud = **IaaS** (Infrastructure as a service)


SaaS
PaaS
IaaS
Hosted applications/apps
Development tools, database management, business analytics
Operating systems
Servers and storage
Networking firewalls/security
Data center physical plant/building

Cloud: Deployment Models

- Private Cloud
- Community Cloud
- Public Cloud
- Hybrid Cloud

- When Utility Computing preferable to Private Cloud?
    - Parallel batch processing
    - When demand is unknown in advance
    - When demand for a service varies with time

Reference: https://csrc.nist.gov/publications/detail/sp/800-145/final (2011) and Above The Clouds: A Berkeley View of Cloud Computing (2009)

Conventional Cloud Computing

- Migration to the cloud of
    - More successful for batch style workloads (MapReduce, High Performance Computing)
    - Less successful for stateful services (Database Management Systems)

- Developers had to deal with
    - Redundancy for availability
    - Geographic distribution of redundant copies
    - Load balancing and request routing
    - Autoscaling in response to changes in load to scale up or down the system.
    - Monitoring
    - Logging for debugging or performance tuning
    - System upgrades, including security patching
    - Migration to new instances as they became available

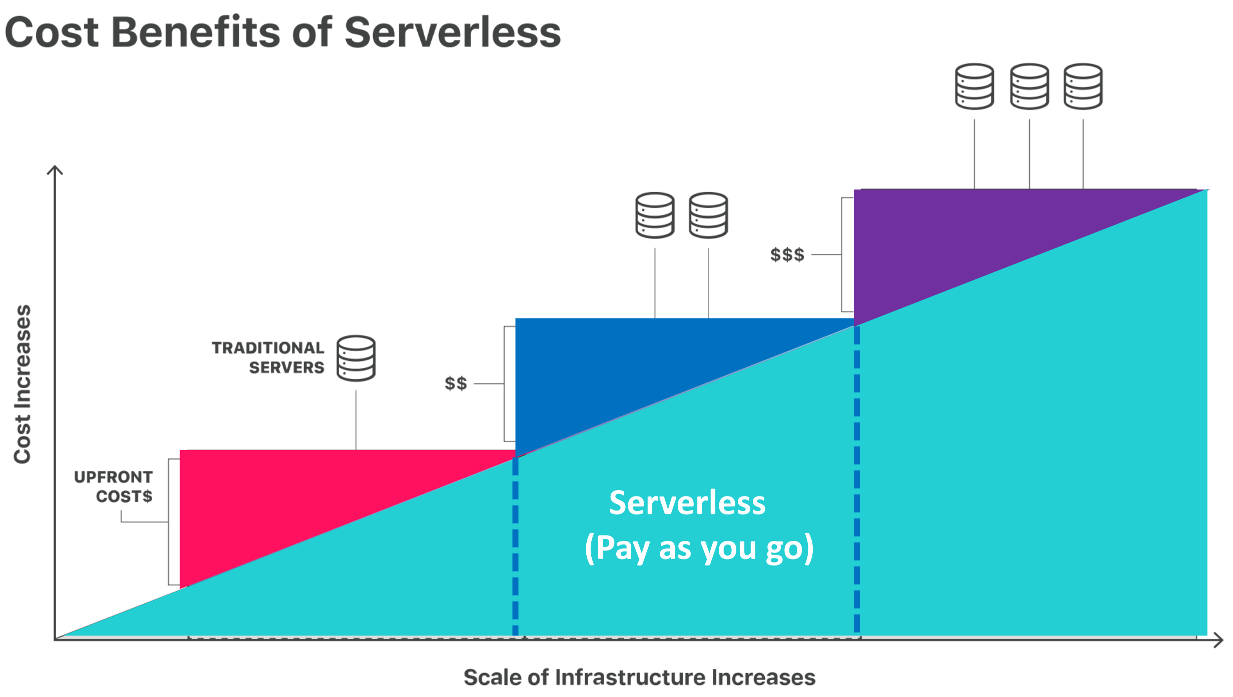
Reference: Cloud Programming Simplified: A Berkeley View on Serverless Computing (2019)

Cloud Resources: Serverless

- Developers to purchase infrastructure services on a flexible pay-as-you-go basis
  - Lower costs: Pay for value
  - Resource Elasticity

- Servers still used!
  - Usage managed by cloud provider

- Serverless means that developers can do their work without having to worry about server and resource provisioning



Reference (including [edited] figure): https://www.cloudflare.com/learning/serverless/what-is-serverless/

Cloud Resources: Serverless

- Serverless: Function as a Service + Backend as a Service (FaaS + BaaS)
- Performed by Cloud System
  - Instance selection
  - Scaling
  - Deployment
  - Fault tolerance
  - Monitoring, logging
  - Security patches
- Three fundamental differences between Serverless and conventional Cloud Computing:
  - Decoupling of computation and storage (Scale separately, priced independently)
  - Executing code without managing resource allocation
  - Paying for resources used instead of resources allocated

Reference: Cloud Programming Simplified: A Berkeley View on Serverless Computing (2019)

Web Applications

- Defines the relation of different components of the system (application, middleware, data stores) that work together for an application
    - N-Tier
    - Microservices
    - Big Compute
    - Big Data
    - Event-Driven
    - Web-Queue-Worker

    . . .

User Requirements

Availability: To run with no significant downtime
Every client request to the service should succeed and receive a response (Data, Service)

Consistency: A model that describes the consistent behavior given a set of rules (Memory, DB)

Latency: Time between a request and a provider's response

Resiliency: The ability to recover from failures and continue to function
With minimal downtime and data loss before full recovery

Scalability: The ability to handle increased load

Security: Identity of the involved parties in the interactions and secrecy of the information exchanged are ensured

Acknowledgements

For complete list of references, please check:

`https://canvas.sfu.ca/courses/63676/pages/references`