Introduction to cloud computing

Unit objectives

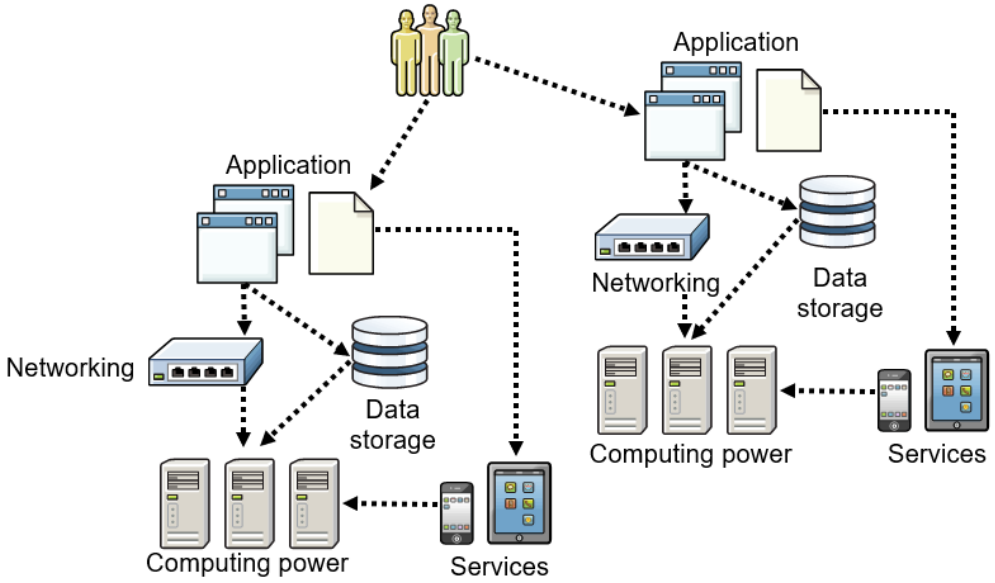
- Define cloud computing.
- Describe the characteristics of Cloud.
- Describe the benefits of Cloud and the factors contributing to its growth.
- Describe cloud services models (IaaS, PaaS and SaaS).
- Describe the cloud deployment options (Private, Public, Hybrid).
- Describe cloud-native applications and development methods.
- Explain the Twelve-Factor App methodology.
- Describe the choices that developers have when building cloud applications.

Introduction to cloud computing

Topics

- Introduction to cloud computing
- Cloud service models
- Cloud deployment models

Before cloud computing



Traditional on-premises

- Applications
- Data
- Runtime
- Middleware
- OS
- Virtualization
- Servers
- Storage
- Networking

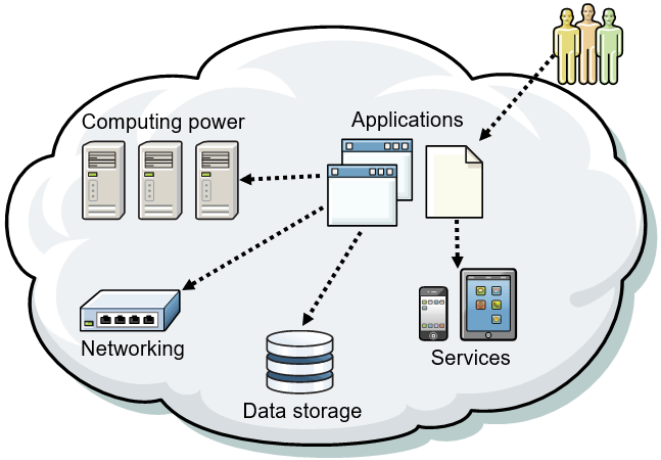
Challenges faced before cloud computing

- Cost
- Scalability
- Reliability
- Security
- Mobility

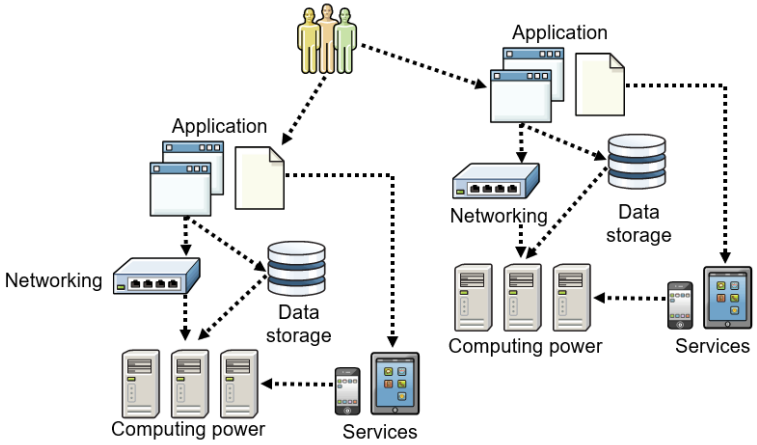
What is cloud computing

Cloud computing, often referred to as “the cloud,” is the delivery of on-demand computing resources (applications to data centers) on a pay-as-you-go basis.

- **Elastic resources**
- **Metered services**
- **Self-service**



Cloud computing model



Traditional on-premises computing model

Characteristics of the cloud

Cloud makes hardware and platform resources readily available and quick to configure. Cloud provides the following characteristics to developers:

- On-demand resources
- Self-service
- Ubiquitous access
- Resource pooling
- Rapid elasticity
- Measured service

Benefits of the cloud

- Achieves economies of scale.
- Goes from CAPEX to OPEX.
- Runs anytime and anywhere.
- Facilitates the agile methodology.
- Ensures global availability.
- Built-in security
- Provides advanced capabilities.

Factors contributing to the growth of the cloud

- Applications with a short lead time to delivery.
- Developers expect to have programming language options and interact with predefined services.
- Modern applications must be able to scale and be managed dynamically.
- Developers expect the “pay-as-you-go” utility computing billing method.

Cloud service models

Topics

- Introduction to cloud computing
- Cloud service models
- Cloud deployment models

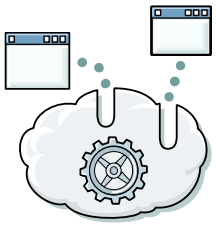
The pizza analogy

The cloud has different service models. With platform, infrastructure, and software offered as services, the pizza analogy is an easy way to understand this approach.

Traditional	Infrastructure as a Service (IaaS)	Platform as a Service (PaaS)	Software as a Service (SaaS)
○ Table & Chairs	○ Table & Chairs	○ Table & Chairs	● Table & Chairs
○ Drinks	○ Drinks	○ Drinks	● Drinks
○ Oven	○ Oven	● Oven	● Oven
○ Toppings	● Toppings	● Toppings	● Toppings
○ Dough Base	● Dough Base	● Dough Base	● Dough Base
Make from scratch at home	**Buy pizza and bake home**	**Get pizza delivered**	**Dine at Pizza Restaurant**
	○ = you furnish; ● = vendor furnishes		

Cloud service models

IaaS: Infrastructure as a Service



PaaS: Platform as a Service

SaaS: Software as a Service

Infrastructure as a Service

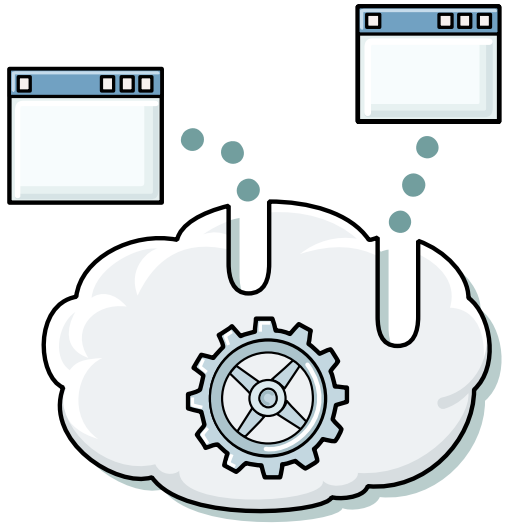
Key features:

- Instead of purchasing hardware, users pay for IaaS on demand.
- Infrastructure is scalable depending on your processing and storage needs.
- You avoid the cost of buying and maintaining your own hardware.
- Enables the virtualization of administrative tasks, which frees time for other work.

Platform as a Service

Key features:

- PaaS provides a platform with tools to test, develop, and host applications in the same environment.
- Enables organizations to focus on software development without having to worry about the underlying infrastructure.
- Providers manage security, operating systems, server software, and backups.
- Facilitates collaborative work even if teams work remotely.



Software as a Service

Key features:

- SaaS vendors provide users with software and applications through a subscription model.
- Users do not have to manage, install, or upgrade software; SaaS providers manage all of those items.
- Data is secure in the cloud; equipment failure does not result in loss of data.
- Applications are accessible from almost any internet-connected device from anywhere in the world.

Example of cloud services

IaaS

- Virtual servers
- Bare metal machines
- Block storage
- File share storage
- Object storage
- Backup
- IP management
- Virtual private networks
- Firewalls
- Load balancers
- Automation

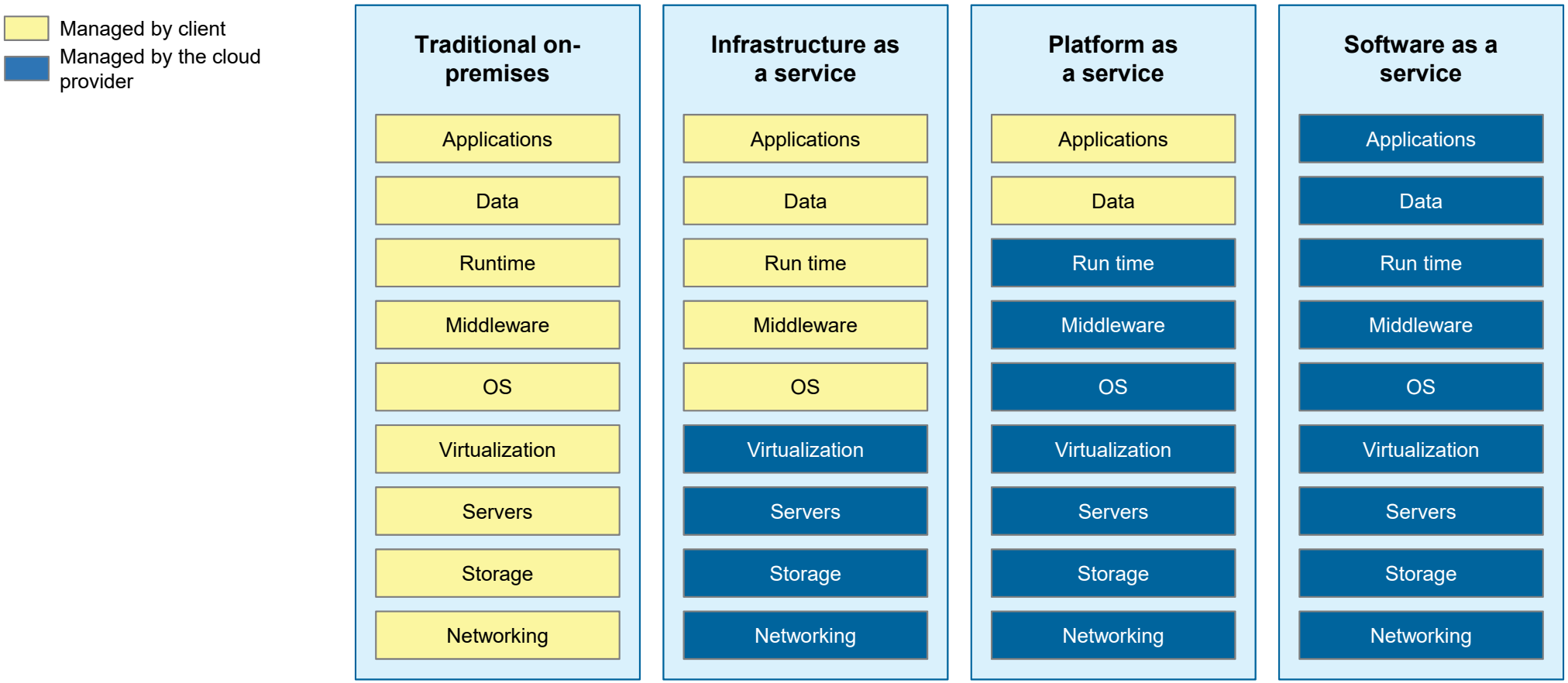
PaaS

- Run times and development platforms
- Databases
- Analytics
- Integration
- Starter kits
- Mobile platforms
- Push notifications
- Messaging
- Developer tools
- Continuous integration / continuous delivery

SaaS

- Email and Collaboration
- Customer relationship manager (CRM)
- Enterprise resource planning (ERP)
- Video streaming
- Marketing
- Talent management
- Advertising

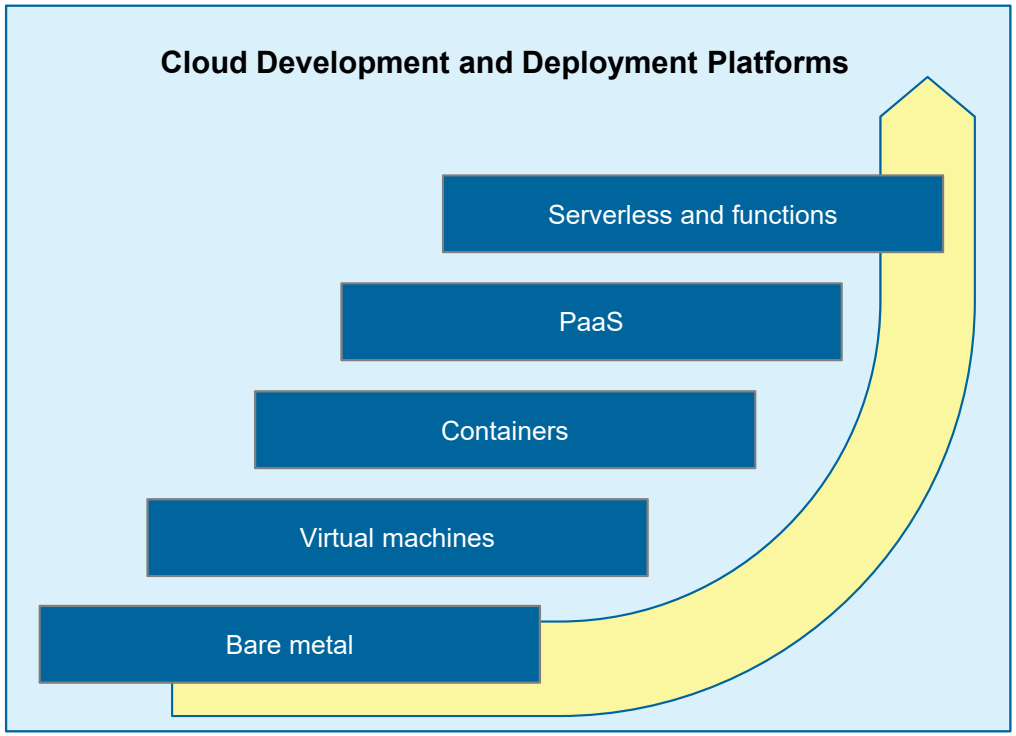
Cloud provider and client responsibilities



Choices when building cloud applications

When developing applications for the cloud, developers have many options to choose from in terms of platforms, frameworks, tools, and services:

1. Traditional development
   - Example: Bare metal or VMs
2. Containerization
   - Example: Docker or Kubernetes
3. PaaS
   - Example: Cloud Foundry
4. Serverless and functions
   - Example: OpenWhisk



Cloud deployment models

Topics

- Introduction to cloud computing
- Cloud service models
- Cloud deployment models

Cloud deployment models

The various types of cloud-computing deployment models include *public cloud*, *private cloud*, and *hybrid cloud*.

Public

Public clouds are owned and operated by cloud providers that offer rapid access over a public network to affordable computing resources.

Private

A private cloud is infrastructure that is operated solely for a single organization.

Hybrid

A hybrid cloud uses a private cloud foundation that is combined with the strategic integration and use of public cloud services.

Cloud-native applications

Cloud-native applications capitalize on the scalability and flexibility of the cloud:

- Applications are broken into separate services called *microservices*.
- Microservices can be developed in different programming languages (polyglot development).
- Microservices communicate with each other by using an agreed upon protocol (such as REST or gRPC).
- Microservices work together as a whole to make up an application, yet each can be independently scaled, continuously improved, and quickly iterated through automation and orchestration processes.

Cloud-native development methods

When developing cloud-native applications, developers must understand and adopt new methods and patterns to maximize the capability that is provided by the cloud provider:

- Readily available sandbox and production environments
- Programming languages and frameworks
- APIs
- Developer toolchains

Cloud-native development methods (cont.)

Cloud-native development introduces the 12-factor app methods and patterns to development:

I. Codebase: One codebase that is tracked in revision control, but there are many deployments.
II. Dependencies: Explicitly declare and isolate dependencies.
III. Configuration: Store the configuration in the environment.
IV. Backing services: Treat backing services as attached resources.
V. Build, release, and run: Strictly separate build and run stages.
VI. Processes: Run the app as one or more stateless processes.

Cloud-native development methods (cont.)

Cloud-native development introduces the 12-actor app methods and patterns to development:

VII. Port binding: Export services by using port binding.
VIII. Concurrency: Scale out by using the process model.
IX. Disposability: Maximize robustness with fast startup and graceful shutdown.
X. Dev/prod parity: Keep development, staging, and production similar.
XI. Logs: Treat logs as event streams.
XII. Admin processes: Run admin and management tasks as one-off processes.

Unit summary

- Define cloud computing.
- Describe the characteristics of Cloud.
- Describe the benefits of Cloud and the factors contributing to its growth.
- Describe cloud services models (IaaS, PaaS and SaaS).
- Describe the cloud deployment options (Private, Public, Hybrid).
- Describe cloud-native applications and development methods.
- Explain the Twelve-Factor App methodology.
- Describe the choices that developers have when building cloud applications.

References

- What is Cloud Computing?
  https://www.ibm.com/cloud-computing/learn-more/what-is-cloud-computing/
- Cloud Service Models
  https://www.ibm.com/cloud/learn/iaas-paas-saas
- IBM Cloud Docs
  https://cloud.ibm.com/docs/
- Cloud Native
  https://www.ibm.com/cloud/cloud-native
- IBM Cloud Services
  https://www.ibm.com/cloud/products
  https://cloud.ibm.com/docs/home/alldocs
- Cloud Explained with a Pizza Analogy
  https://www.ibm.com/blogs/insights-on-business/government/cloud-explained-with-a-pizza-analogy/
- 12 Factor Application
  https://12factor.net