Cloud Architecture

CSYE 6225: Network Structure & Cloud Computing
Northeastern University

Instructor: Raja Alomari,PhD

M9: Overview

Topics include:

- Overview of Cloud Architecture.
- Cloud Architect responsibilities and required skills.
- Overview of Microservices vs Monolithic Architectures.
- Pillars of AWS Well-Architected Framework
- Overview of AWS Core Services.

Cloud Architecture

Objectives:

- Understand Cloud Architecture Fundamentals and AWS Core Services
- Learn the AWS Well-Architected Framework and cloud architect responsibilities.
- Understand Microservices architecture.

What is Cloud Architecture?

Definition: The **design** and structure of various technologies and services that work together to deliver cloud computing solutions. It involves **organizing** and **managing** cloud resources to create scalable, secure, cost-effective, and fault-tolerant systems, allowing organizations to access computing power, storage, and services over the internet, on-demand.

Key Components of Cloud Architecture:

- Compute.
- Storage.
- Networking.
- Databases.
- Security.
- Monitoring & Management.
- Cost Optimization.

Why Cloud Architecture?

Scalability: Easily scale up or down based on demand.

Cost-Efficiency: Pay-as-you-go pricing models reduce upfront hardware investments.

Resilience: Built-in redundancy and disaster recovery options ensure high availability.

Security: Advanced security features protect against data breaches and vulnerabilities.

Agility: Faster deployment and easier management of applications and services.

Cloud Architect

A Cloud Architect is a critical role in designing, implementing, and managing cloud infrastructure.

Their responsibilities span across technical design, strategy alignment, and operational oversight.

Cloud Architect: Architecture Design and Planning

Cloud Strategy Development: Design cloud strategies that align with business objectives and IT goals.

Solution Architecture: Define and document architectural solutions, including high-level diagrams and component details.

Scalability Planning: Ensure solutions can scale with business needs, addressing future growth.

Cloud Service Selection: Evaluate and select appropriate cloud services (e.g., compute, storage, databases) based on requirements.

Hybrid Cloud Strategy: Design architectures integrating on-premise and cloud-based solutions.

Cloud Architect: Security and Compliance

Risk Assessment: Identify security risks and implement mitigation strategies.

Access Management: Enforce identity and access management (IAM) practices like least privilege and MFA.

Data Protection: Design secure data storage and transmission protocols, ensuring compliance with regulations (e.g., GDPR, HIPAA).

Monitoring and Response: Implement intrusion detection systems and incident response plans.

Cloud Architect: Performance and Optimization

Performance Monitoring: Design systems for optimal performance, using tools to monitor and manage workloads.

Cost Optimization: Implement cost-effective solutions, such as serverless computing and right-sizing resources.

Automation: Use infrastructure as code (IaC) tools (e.g., Terraform, CloudFormation) for deployment and scaling automation.

Cloud Architect: Collaboration and Stakeholder Engagement

Business Alignment: Work with stakeholders to align cloud solutions with organizational objectives.

Team Leadership: Guide development and operations teams in implementing cloud architectures.

Vendor Management: Collaborate with cloud providers (e.g., AWS, Azure, Google Cloud) to optimize service use.

Cloud Architect: Governance and Best Practices

Standards Development: Define cloud governance policies, including tagging strategies and usage limits.

Framework Compliance: Ensure systems adhere to frameworks such AWS Well-Architected Framework or Azure CAF.

Documentation: Maintain comprehensive documentation for all cloud architectures and implementations.

Cloud Architect: Innovation and Continuous Improvement

Research and Development: Stay updated on emerging cloud technologies and trends.

Prototype Development: Build proofs of concept for new technologies and assess feasibility.

Optimization: Continuously review and improve existing architectures.

Skill Set Required for a Cloud Architect

Cloud Architect: Core Technical Skills

Cloud Platforms: In-depth knowledge of AWS, Azure, Google Cloud, or hybrid environments.

Networking: Understanding of VPCs, VPNs, load balancers, and DNS configurations.

Security:

- Identity and Access Management (IAM).
- Encryption protocols (TLS, AES).
- Security tools like AWS GuardDuty or Azure Security Center.

Cloud Architect: Core Technical Skills …. Cont'd

Compute Services: Proficiency in compute services such as AWS EC2, Lambda, Azure VMs.

Storage Solutions: Knowledge of S3, Azure Blob, EBS, and data lifecycle policies.

Databases: Familiarity with RDS, DynamoDB, MongoDB, or equivalent.

Containers and Orchestration: Expertise in Docker, Kubernetes, ECS, or EKS.

Cloud Architect: Core Technical Skills …. Cont'd

Automation Tools:

Infrastructure as Code (Terraform, CloudFormation).

CI/CD pipelines (CodePipeline, Jenkins).

Programming and Scripting:

Languages: Python, Java, Bash, PowerShell.

APIs: RESTful services for automating tasks.

Cloud Architect: Analytical and Problem-Solving Skills

Design Thinking: Ability to conceptualize and design scalable, resilient, and secure architectures.

Troubleshooting: Identify and resolve complex system issues.

Performance Analysis: Use metrics to assess and optimize resource utilization.

Cloud Architect: Business and Leadership Skills

Communication: Explain technical concepts to non-technical stakeholders.

Project Management: Manage timelines, budgets, and deliverables for cloud migration or deployment projects.

Team Collaboration: Work effectively with cross-functional teams, including DevOps, security, and business units.

Cloud Architect: Certifications

Certifications validate a Cloud Architect's expertise and are often required by employers:

- AWS Certifications:
  - AWS Certified Solutions Architect – Associate/Professional.
  - AWS Certified Security – Specialty.
- Microsoft Azure Certifications:
  - Azure Solutions Architect Expert.
- Google Cloud Certifications:
  - Professional Cloud Architect.
- Vendor-Neutral Certifications:
  - Certified Cloud Security Professional (CCSP).
  - TOGAF Certification (for enterprise architecture).

The *TOGAF*® Standard — *Version 9.2*

Cloud Architect: Emerging Skills

Artificial Intelligence and Machine Learning (AI/ML): Knowledge of AWS SageMaker, AWS Glue, Azure ML, TensorFlow, and other emerging cloud technologies.

Edge Computing: Understanding edge-based services such as AWS IoT Greengrass.

Sustainability: Designing energy-efficient systems and using low-carbon resources.

Key Questions a Cloud Architect Will Ask

Cloud Architect: Questions

These questions guide the Cloud Architect in gathering the necessary information to design a **secure**, **scalable**, and **cost-effective cloud architecture** that meets the organization's needs.

They also facilitate collaboration with stakeholders, ensuring alignment with both technical and business objectives.

Cloud Architect: Questions

Business Requirements and Goals

- What are the primary business objectives for adopting cloud solutions?
- What specific business problems are we trying to solve?
- Are there any upcoming projects or expansions that the architecture should support?
- What are the key performance indicators (KPIs) for success?
- How does the cloud align with the organization's digital transformation strategy?

Cloud Architect: Questions

Current State Analysis

- What does the current IT infrastructure look like?
- Are there any legacy systems or workloads that need modernization?
- What are the pain points or limitations of the existing infrastructure?
- What percentage of the workloads are currently in the cloud?
- Are there existing dependencies between systems that must be maintained?

Cloud Architect: Questions

Functional and Non-Functional Requirements

- What are the scalability requirements for the solution?
- What uptime or availability targets must we meet (e.g., 99.9%, 99.99%)?
- What level of latency or performance is acceptable?
- Are there any specific compliance or regulatory requirements (e.g., GDPR, HIPAA)?
- What are the storage, compute, and networking requirements?

Cloud Architect: Questions

Security and Compliance

- What is the organization's security posture?
- Who should have access to what resources, and how will this be managed?
- Are there data residency or sovereignty concerns?
- What disaster recovery and backup policies need to be implemented?
- What is the policy for handling sensitive data (e.g., encryption in transit and at rest)?

Cloud Architect: Questions

Budget and Cost Management

- What is the budget for the project, including ongoing operational costs?
- Are there specific cost-saving measures or priorities (e.g., serverless, auto-scaling)?
- What tools or practices will be used for cost monitoring and optimization?
- Are there any financial risks associated with potential over-provisioning or underutilization?

Cloud Architect: Questions

Workload and Application Design

- What are the types of workloads to be hosted (e.g., web applications, databases, analytics)?
- Are there workloads with variable demand requiring elastic scaling?
- Do applications need containerization or orchestration (e.g., Kubernetes)?
- Will the solution require serverless components such as AWS Lambda or Azure Functions?
- How will workloads be migrated to the cloud (lift-and-shift, re-architect, or hybrid)?

Cloud Architect: Questions

Integration and Interoperability

- What integrations are required with third-party systems or services?
- Are there any APIs that need to be developed or consumed?
- How will hybrid or multi-cloud systems communicate with each other?
- Are there existing DevOps pipelines that need integration?

Cloud Architect: Questions

Monitoring and Operations

- What tools or platforms will be used for monitoring and alerting (e.g., CloudWatch, Azure Monitor)?
- What are the log retention policies for auditing and troubleshooting?
- Who will manage and maintain the cloud environment post-deployment?
- Are there plans to automate operational tasks (e.g., backups, scaling)?

Cloud Architect: Questions

Training and Change Management

- What training or upskilling will the team require for the new architecture?
- Are there existing knowledge gaps in cloud technologies among the staff?
- How will documentation for the new architecture be managed?
- What is the plan for transitioning operations to a cloud-focused model?

Cloud Architect: Questions

Long-Term Planning and Innovation

- What are the organization's long-term goals for cloud adoption?
- Are there plans to explore new technologies such as AI/ML, IoT, or edge computing?
- How will we ensure continuous optimization and innovation?
- What metrics will we use to periodically evaluate the cloud strategy?

Microservices Architecture

What are Microservices?

- Definition:
  - A design pattern where applications are structured as a collection of loosely coupled, independently deployable services.
- Characteristics:
  - Small, focused on specific business capabilities.
  - Communicate through lightweight protocols (e.g., HTTP/REST, gRPC).
  - Can be independently developed, deployed, and scaled.

Monolithic vs. Microservices Architecture

Aspect	Monolithic	Microservices
Structure	Unified codebase, tightly integrated	Independent services, loosely coupled
Scalability	Whole system scales together	Individual services scale independently
Deployment	Single deployment	Services deployed independently
Fault Isolation	Failure affects entire application	Failure isolated to specific service
Technology	One tech stack	Different stacks per service

Benefits of Microservices

Scalability: Scale individual services based on demand (e.g., scaling the payment service during sales events).

Agility: Independent deployments reduce time-to-market for new features.

Fault Isolation: Failures in one service do not cascade to others.

Technology Diversity: Teams can choose the best technology stack for their service.

Key Components of a Microservices Architecture

Service Discovery: Services register themselves and are discoverable dynamically (e.g., AWS App Mesh).

API Gateway: Acts as the entry point for external traffic and routes requests to appropriate services (e.g., Amazon API Gateway).

Data Decentralization: Each service manages its own database, promoting data isolation.

Inter-Service Communication: Use asynchronous messaging (e.g., Amazon SQS) or synchronous APIs (e.g., REST, gRPC).

AWS Services for Microservices

Compute:

- AWS Lambda (Serverless microservices)
- Amazon ECS or EKS (Containerized services)

Service Discovery:

- AWS Cloud Map
- AWS X-Ray for tracing service dependencies.

Communication:

- Amazon SQS, Amazon SNS (Asynchronous)
- Amazon API Gateway (Synchronous)

Monitoring and Logging:

- Amazon CloudWatch

Challenges in Microservices

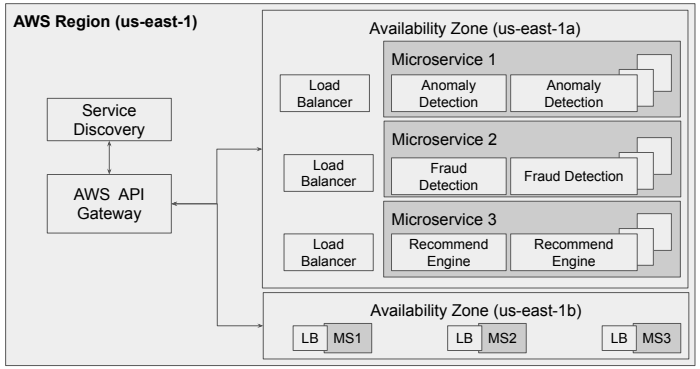
Distributed Systems Complexity: Debugging and monitoring can be complex due to the number of services.

Data Consistency: Ensuring consistency across services is challenging without ACID (Atomicity, Consistency, Isolation, Durability) transactions.

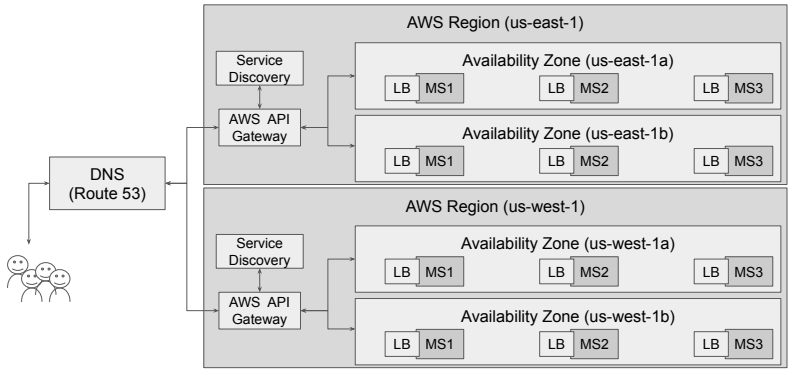
Network Latency: Increased number of network calls compared to monolithic architectures.

Security: Authentication and authorization for inter-service communication.

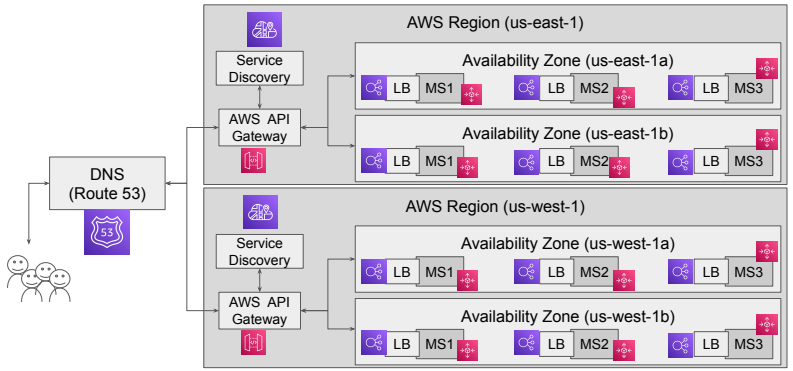
Case Study: Microservice Architecture



Case Study: Microservice Architecture



Case Study: Microservice Architecture



AWS Well-Architected Framework

What is the AWS Well-Architected Framework?

Guiding principles for building secure, high-performing, resilient, and efficient infrastructure.

Importance:

- Ensures alignment between business and technology.
- Identifies potential risks and improvements.

Six Pillars:

Operational Excellence, Security, Reliability, Performance Efficiency, Cost Optimization, and Sustainability.

Pillar 1 – Operational Excellence

Design Principles

- Define and document operational standards.
- Automate infrastructure and deployment processes.
- Design for failure; build mechanisms to anticipate and recover from errors.

AWS Services

- AWS CloudFormation: Automate infrastructure as code (IaC).
- Amazon CloudWatch: Centralized monitoring for metrics, logs, and alarms.
- AWS CodeDeploy: Automate application deployments and rollbacks.

Pillar 2 – Security

Design Principles

- Implement strong identity and access controls (least privilege).
- Detect and respond to threats in real time.
- Secure data at rest and in transit.

AWS Services

- AWS Identity and Access Management (IAM): Manage users, roles, and permissions.
- AWS Key Management Service (KMS): Encrypt sensitive data.
- Amazon GuardDuty: Continuously monitor for malicious activity.

Pillar 3 – Reliability

Design Principles

- Design systems with fault-tolerance.
- Implement multi-AZ (Availability Zone) and multi-region deployments.
- Enable automated recovery for failures.

AWS Services

- Amazon Route 53: DNS with health checks for failover.
- Auto Scaling: Automatic recovery of instances.
- Amazon RDS Multi-AZ: High availability for databases.

Pillar 4 – Performance Efficiency

Design Principles

- Use the right services to match workload requirements.
- Optimize resource configurations for compute, memory, and storage.
- Employ content delivery and caching strategies.

AWS Services

- Amazon EC2: Scalable compute options with instances tailored to workload.
- Amazon S3: Object storage with high availability and durability.
- Amazon CloudFront: Content Delivery Network (CDN) for accelerated content delivery.
- AWS Lambda: Serverless compute for event-driven workloads.

Pillar 5 – Cost Optimization

Design Principles

- Match capacity with demand using Auto Scaling and right-sizing.
- Leverage Spot Instances for cost efficiency.
- Use storage lifecycle policies to optimize data costs.

AWS Services

- AWS Cost Explorer: Track and manage cloud expenses.
- Savings Plans: Optimize costs for committed usage.
- S3 Lifecycle Policies: Move infrequent data to lower-cost tiers.

Pillar 6 – Sustainability

Design Principles

- Design with energy efficiency in mind.
- Minimize idle resources.
- Use serverless and containerized solutions for efficient compute utilization.

AWS Services

- Graviton2 Instances: Energy-efficient processors for EC2.
- AWS Lambda: Serverless architecture with pay-per-use pricing.
- Elastic Kubernetes Service (EKS): Optimized container orchestration for efficient workloads.

AWS Core services Overview

AWS Core Services

Categories:

- Compute
- Storage
- Database
- Networking
- Serverless Architecture
- Data Analytics
- AI & Machine Learning

Compute Services

Amazon EC2 (Elastic Compute Cloud)

- Overview:
  - Scalable virtual servers for various workloads.
- Instance Types and Use Cases:
  - General Purpose (T-Series, M-Series): Web servers, development environments.
  - Compute Optimized (C-Series): High-performance compute tasks like gaming or machine learning.
  - Memory Optimized (R-Series, X-Series): Large databases, in-memory caching.
  - Storage Optimized (I-Series, D-Series): High I/O databases, big data.
  - GPU Instances (P-Series, G-Series): Machine learning, rendering.

Compute Services

AWS Lambda

- **Overview:** Event-driven, serverless computing that executes code in response to triggers.
- **Key Features:**
  - Supports multiple languages (Node.js, Python, Java, etc.).
  - Scales automatically based on events.
  - Pay-per-use model.

Compute Services

AWS Elastic Beanstalk

- **Overview:** Platform-as-a-Service (PaaS) for simplified deployment of web applications.
- **Features:**
  - Automatically provisions EC2 instances, load balancers, and scaling.
  - Compatible with multiple programming languages (Java, Python, Ruby, etc.).

Storage Services

Amazon S3 (Simple Storage Service)

- Overview: Object storage designed for scalability, durability, and availability.
- Storage Classes:
  - Standard: Frequently accessed data.
  - Intelligent-Tiering: Optimizes costs by moving data between tiers.
  - Glacier/Deep Archive: Low-cost, long-term archival storage.
  - One Zone-IA: Infrequently accessed data stored in a single availability zone.

Storage Services

Amazon EBS (Elastic Block Store)

- Overview: Persistent block storage for use with EC2 instances.
- Use Cases:
  - Databases, file systems, and any application requiring low-latency storage.
- Performance Options:
  - General Purpose (SSD), Provisioned IOPS (SSD), and HDD options for throughput-intensive workloads.

Storage Services

Amazon EFS (Elastic File System)

- Overview: Fully managed, scalable file storage.
- Features:
  - Automatically grows and shrinks as files are added or removed.
  - Supports multiple EC2 instances accessing the same file system.

Database Services

Amazon RDS (Relational Database Service)

- Overview: Fully managed relational database service supporting MySQL, PostgreSQL, MariaDB, SQL Server, and Oracle.
- Features:
  - Automated backups, scaling, and replication.
  - Supports Multi-AZ deployments for high availability.

Database Services

Amazon DynamoDB

- Overview: Managed NoSQL database for key-value and document data.
- Features:
  - Scales seamlessly to handle millions of requests per second.
  - Built-in backup and restore capabilities.

Database Services

Amazon Aurora

- Overview: High-performance, managed relational database engine.
- Benefits:
  - Up to five times faster than MySQL and three times faster than PostgreSQL.
  - Automatic scaling, fault tolerance, and global database support.

Networking

Amazon VPC (Virtual Private Cloud)

- Overview: Enables users to create isolated networks within AWS.
- Features:
  - Subnets, route tables, internet gateways, and NAT gateways.
  - Security features include security groups and network ACLs.

Networking

Amazon Route 53

- Overview: Scalable and reliable DNS web service.
- Features:
  - Domain registration, DNS routing, and health checks.
  - Supports failover, latency, and geolocation-based routing.

Networking

Amazon CloudFront

- Overview: Content Delivery Network (CDN) for accelerated delivery of static and dynamic web content.
- Features:
  - Integrates seamlessly with S3 for website hosting.
  - Supports encryption and edge computing (Lambda@Edge).

Serverless Architecture

Key Benefits

- Scalability: Automatically adjusts to handle fluctuating workloads.
- Cost-Efficiency: Pay only for the compute time used.
- Simplified Management: No need to provision or manage servers.

Use Cases

- APIs, chatbots, real-time file processing, background tasks, and IoT applications.

Data Analytics

Amazon Redshift

- Overview: Fully managed data warehousing service for structured and semi-structured data.
- Features:
  - Handles petabyte-scale data.
  - Supports complex queries and integrates with visualization tools like QuickSight.

Data Analytics

Amazon Athena

- Overview: Query service for data stored in S3.
- Key Features:
  - Serverless and supports SQL queries.
  - Pay-per-query pricing.

Data Analytics

Amazon EMR (Elastic MapReduce)

- Overview: Big data processing framework for running Spark, Hadoop, Presto, etc.
- Use Cases:
  - Data transformations, streaming data analytics, and log analysis.

AI & Machine Learning

Amazon Rekognition

- Overview: Image and video analysis.
- Use Cases: Facial recognition, object detection, content moderation.

Amazon Comprehend

- Overview: Natural language processing (NLP) service.
- Use Cases: Sentiment analysis, key phrase extraction, and language detection.

Amazon Polly

- Overview: Text-to-speech conversion.
- Use Cases: Interactive voice systems, educational tools.

Amazon Translate

- Overview: Language translation.
- Use Cases: Multilingual support for applications.

AI & Machine Learning

Amazon SageMaker

- Overview: End-to-end ML development and deployment platform.
- Core Features:
  - Data Preparation: Built-in data wrangling tools.
  - Model Building: Supports pre-built algorithms and custom models.
  - Training: Distributed training capabilities with managed infrastructure.
  - Deployment: Real-time and batch deployment options.

AI & Machine Learning

ML Frameworks

- AWS Deep Learning AMIs: Pre-installed frameworks like TensorFlow, PyTorch, MXNet.
- Elastic Kubernetes Service (EKS): Train ML models on containerized environments.

Amazon Augmented AI (A2I)

- Overview: Human-in-the-loop for ML predictions.
- Use Cases: Reviewing sensitive predictions such as loan approvals.

AI & Machine Learning

ML for Data Analytics

Amazon Kinesis: Real-time data streaming for ML pipelines. Process live sensor data for anomaly detection.

Amazon QuickSight with ML Insights: BI service with integrated ML predictions. Predict trends and forecast sales.

AWS Glue: ETL service with ML-powered data preparation. Automate schema detection for large datasets.

AI & Machine Learning

Specialized ML Services

Amazon Forecast: Time-series forecasting using ML. Demand planning, inventory management.

Amazon Personalize: Recommendation engine. Personalized shopping experiences.

Amazon Textract: Extract text and data from scanned documents. Invoice processing, compliance checks.

Module 9 Conclusion

- Cloud Architecture is Key to the Success of Any Product Leveraging Cloud Computing.
- The Choice Between Monolithic and Microservices Architecture Depends on the Technology Stack and Other Factors.
- Serverless Computing is Becoming a Trend in Cloud Computing, Despite Its Vendor Lock-in.