Serverless

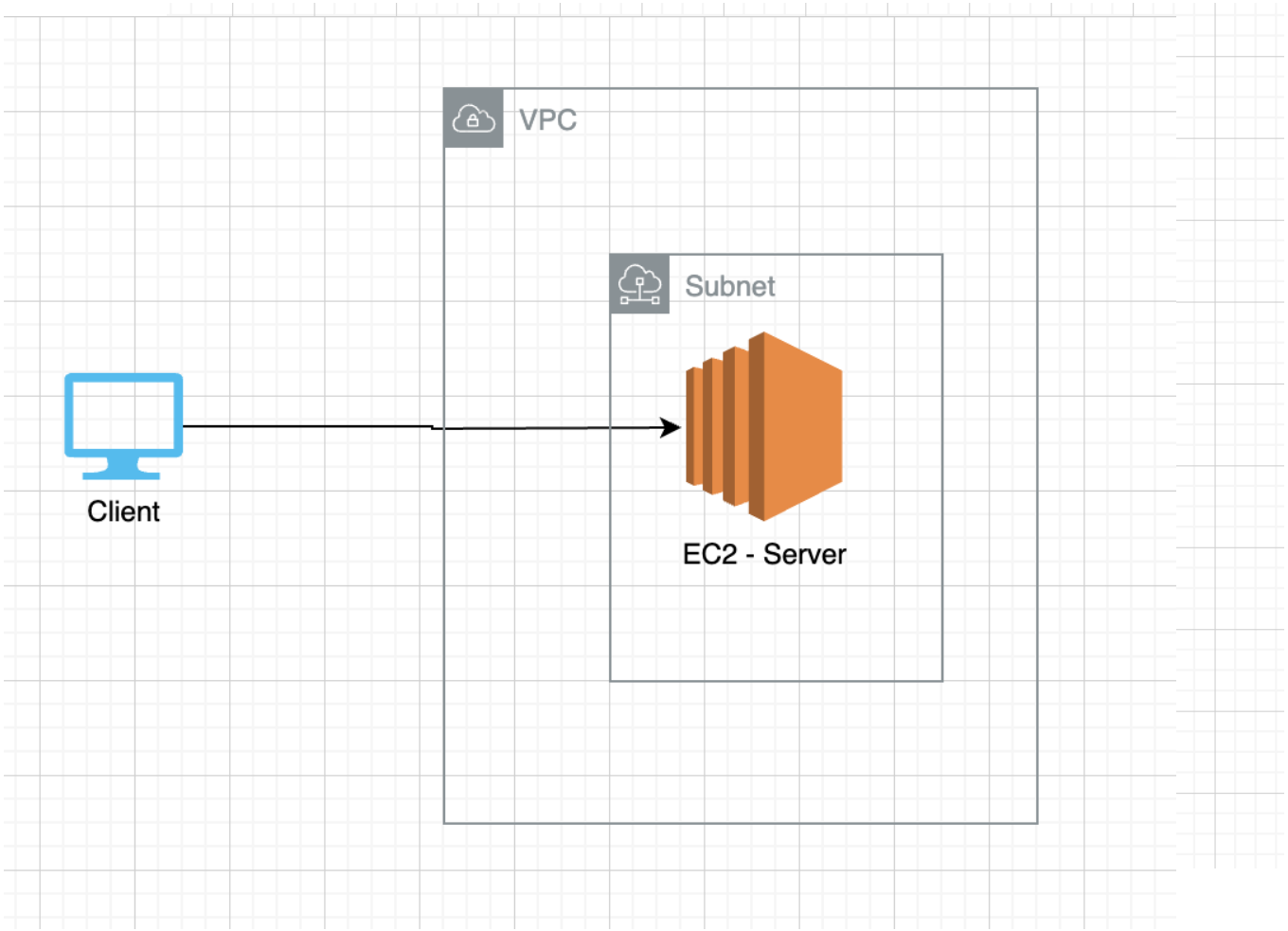
Thao Huy Vu

Maharishi International University - Fairfield, Iowa

Agenda

- Lambda Function
- API Gateway
- DynamoDB

Server-based Backend

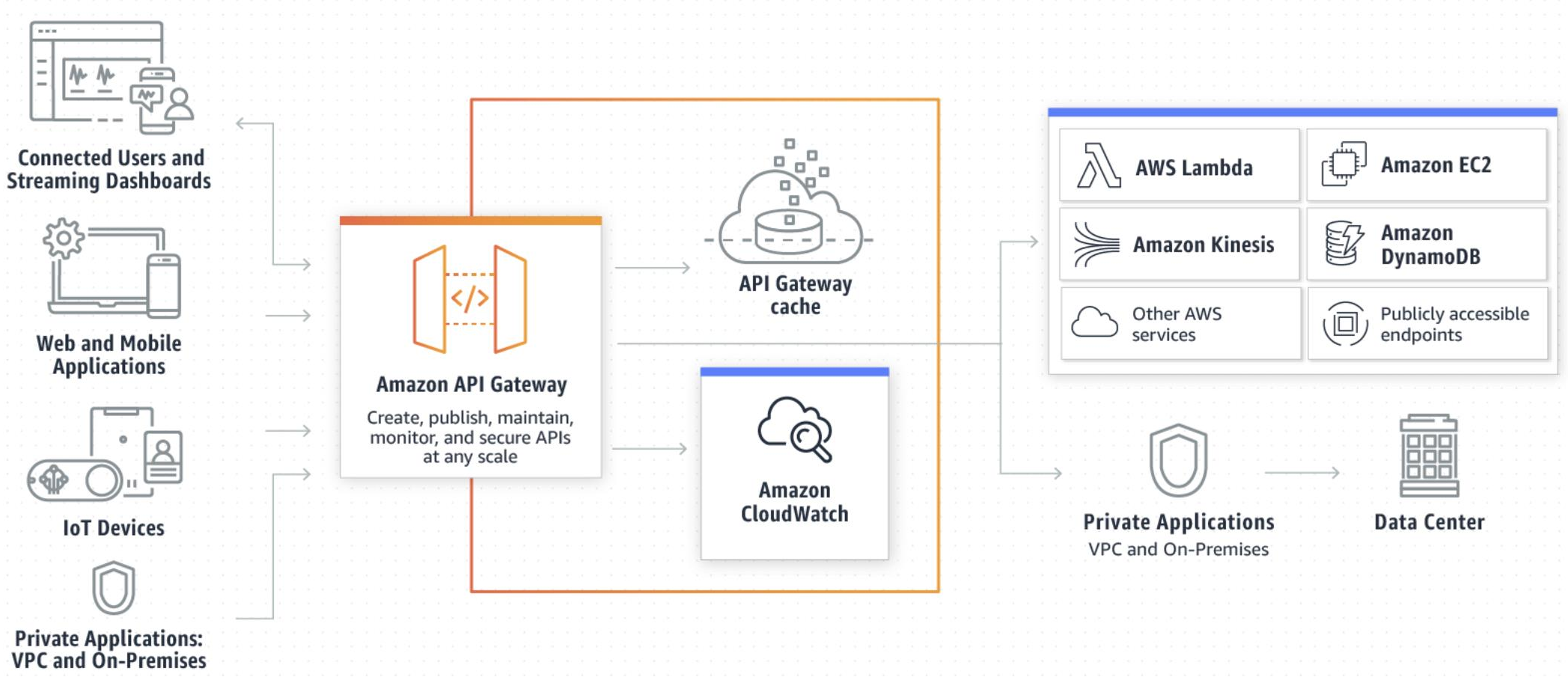


Serverless Backend

AWS Lambda

- **Serverless computing**: AWS Lamda is a serverless computing service that allows developers to run their code without caring about servers.
- **No administrative**: Zero administrative effort required for scalability, back-ups, or multi-data center redundancy.
- **Scalability**: Automatically scales with demand, handling from a few requests to thousands per second.
- **Supports Multiple Languages**: Compatible with Python, JavaScript, Java, Go, Ruby, and more.

API Gateway


Connected Users and Streaming Dashboards
Web and Mobile Applications
IoT Devices
Private Applications: VPC and On-Premises
Amazon API Gateway
Create, publish, maintain, monitor, and secure APIs at any scale
API Gateway cache
Amazon CloudWatch
AWS Lambda
Amazon EC2
Amazon Kinesis
Amazon DynamoDB
Other AWS services
Publicly accessible endpoints
Private Applications
VPC and On-Premises
Data Center

API Gateway

- **Definition**: A fully managed service to create, publish, maintain, monitor, and secure APIs at any scale.
- **Front Door**: Acts as the entry point for applications to access data, business logic, or functionality from backend services.
- **Restful API**: Supports serverless workloads and HTTP backends for scalable websites.
- **WebSocket API**: Enables real-time two-way communications, such as chat apps or live streaming dashboards.

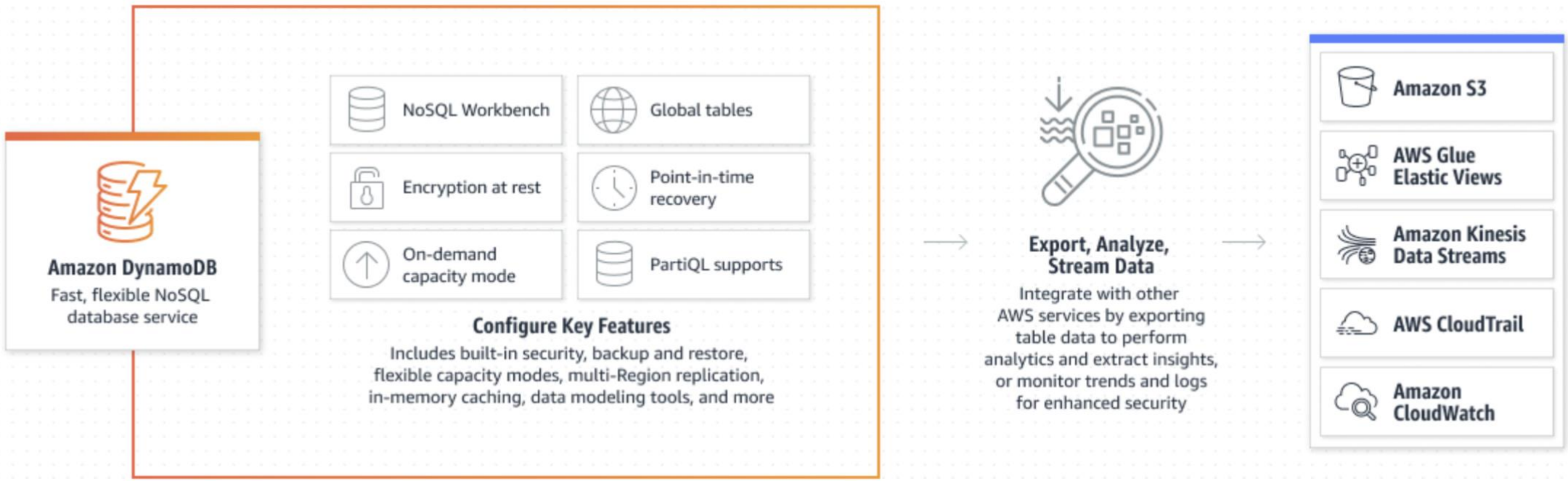
API Gateway

- **Efficient API Development**: Simplifies API creation by enabling developers to define APIs quickly and integrate with various backend services.
- **Performance at Any Scale**: Automatically scales to handle millions of API requests with low latency.
- **Cost Savings at Scale**: Pay-as-you-go pricing ensures cost-efficiency, charging only for the number of API calls and data transferred.
- **Easy Monitoring**: Built-in integration with Amazon CloudWatch provides detailed metrics and logging for API usage and performance.

Rest API

- **REST (Representational State Transfer)**: A software architectural style that uses a subset of HTTP for building web services.
- **RESTful Web Services**:
  - Provide resources in a textual format (e.g., JSON or XML).
  - Use a stateless protocol (typically HTTP).
  - Allow resource manipulation via predefined operations (e.g., GET, POST, PUT, DELETE).
- **Contract**: There is an implicit agreement between the service provider and consumer regarding how the API is structured and accessed.

Amazon DynamoDB



Amazon DynamoDB

- A fully managed NoSQL database service designed for document and key-value store models.
- **High Performance**:
  - Delivers consistent single-digit millisecond latency.
  - Optimized for high-read and high-write workloads.
- **Scalability**:
  - Automatically scales to handle virtually unlimited requests or data size.
  - Suitable for applications ranging from small workloads to enterprise-scale systems.
- **Serverless**:
  - No need to provision or manage servers.
  - Automatically adjusts capacity based on workload.

Amazon DynamoDB

- **Data Model**:
  - DynamoDB is a key-value and document database.
  - Stores data in tables with items (rows) and attributes (columns), offering flexibility for unstructured or semi-structured data.
- **Read and Write Consistency Models**:
  - **Eventually Consistent Reads**: Fast, less resource-intensive.
  - **Strongly Consistent Reads**: Guarantees the most recent data but may have higher latency.

Amazon DynamoDB

- **Backup and Restore**:
  - Provides on-demand and continuous backups with point-in-time recovery.
  - Ensures data durability and protection against accidental deletes or corruption.
- **Security**:
  - Supports encryption at rest and in transit.
  - Integrates with AWS IAM for fine-grained access control and AWS KMS for encryption keys.

Reference

- AWS: https://docs.aws.amazon.com
- ChatGPT: https://chatgpt.com
- Google AI: https://gemini.google.com
- Practical Tutorials: https://thaovu.org