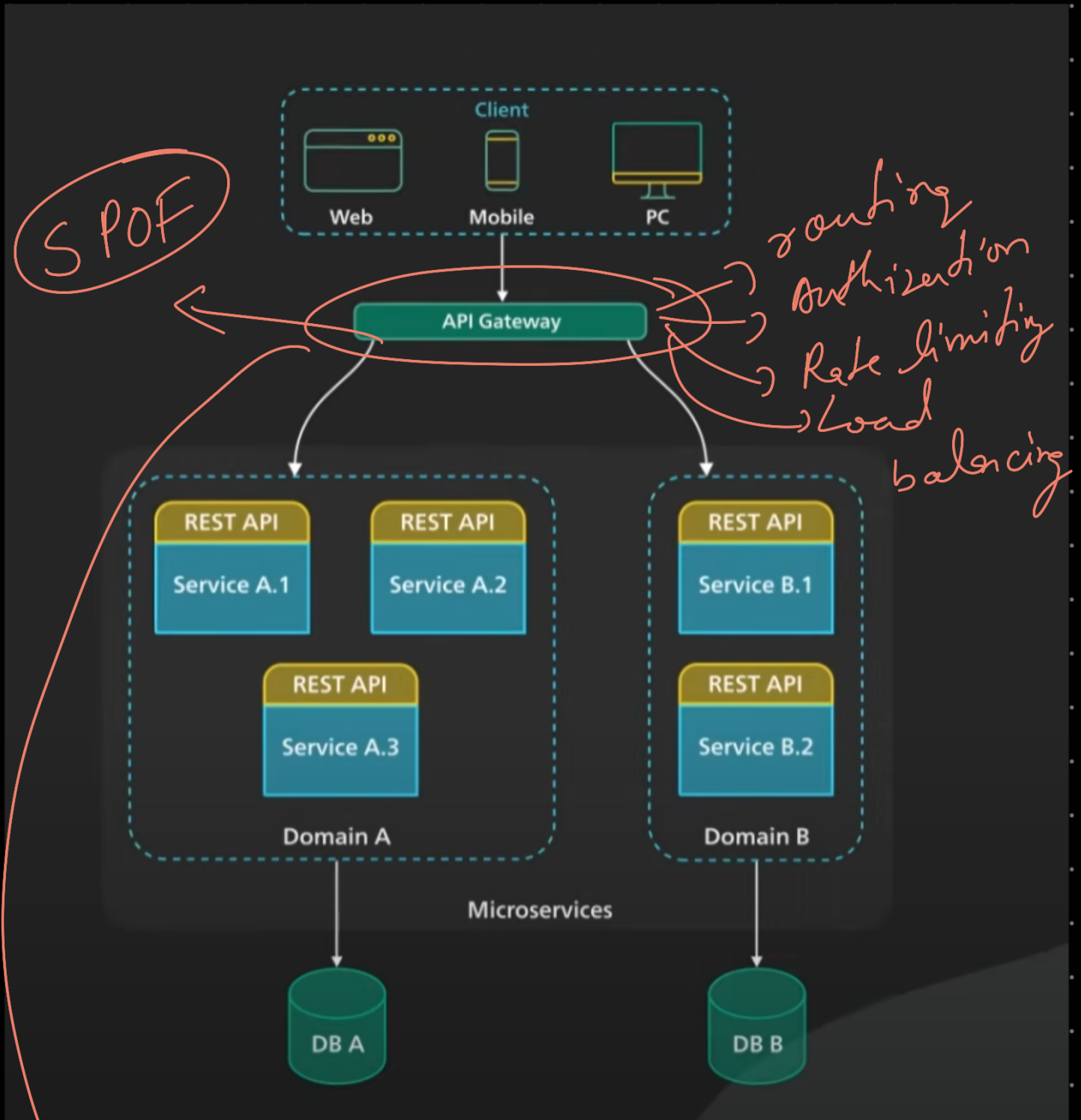


# [API Gateway]

↳ Reverse proxy



Need to scale in horizontally.

## Key Roles of an API Gateway

1. **Routing:** Directs client requests to the appropriate backend services based on endpoints.
2. **Security:** Enforces authentication and authorization policies to ensure secure access.
3. **Load Balancing:** Distributes incoming traffic across multiple service instances to ensure reliability and scalability.
4. **Rate Limiting and Throttling:** Controls the number of requests a client can make to prevent overuse or abuse.
5. **Protocol Translation:** Converts between protocols (e.g., HTTP to gRPC) to facilitate interoperability between clients and services.
6. **Request Transformation:** Modifies requests (e.g., adding headers) or responses to match the client's needs.
7. **Caching:** Stores frequently requested data to improve performance and reduce backend load.
8. **Logging and Monitoring:** Tracks request/response metrics and provides observability for debugging and analysis.

## How an API Gateway Works

1. **Client Request:** A client (e.g., a web or mobile app) sends a request to the API Gateway.
2. **Processing:** The gateway processes the request, applies security checks, and performs necessary transformations.
3. **Service Communication:** Routes the request to the appropriate backend service(s).
4. **Response Delivery:** Collects and consolidates responses (if needed) from backend services and sends them back to the client.

## Popular API Gateway Tools

1. **Kong:** Open-source, extensible gateway with advanced plugins.
2. **AWS API Gateway:** Fully managed gateway service for AWS environments.
3. **Apigee:** Google's API management platform for enterprise use.
4. **NGINX:** Can be configured as an API Gateway with additional modules.
5. **Spring Cloud Gateway:** Java-based API gateway for Spring Boot microservices.
6. **Traefik:** Cloud-native and lightweight gateway with support for modern protocols.

\* Mostly provide by Managed cloud services