Serverless vs Serverful

A traditional (serverful) approach is to have one large code base on a server, like with monolithic multi-tier systems. This is a conservative and safe engineering approach. With serverless computing, the monolithic code base is broken into microservices or functions. These can be developed in different languages and platforms as needed, making updates and additions easier. The advantages of having a server are retained, but the management of the servers is left to the cloud service provider. This approach enhances scalability without leaving any idle capacity.

API Gateway

API Gateway is a feature that lets you define a RESTful API to lead to other services or AWS lambda functions. For example, you can define an HTTP verb and a URL path like GET /movies, and specify a target. This lets you break a monolithic system into microservices or functions that are easier to handle. However, AWS lambda has longer latency and to reduce it a faster technology will be needed with more expense. API Gateway allows for serverless computing.

Partition Tolerance

Partition tolerance is one of the three main components of the CAP theorem in distributed systems theory. It means the system should continue operating even in the presence of failures like software, hardware, network issues, or power issues. A system that maintains Partition tolerance is called fault-tolerant. It achieves this by redundancy i.e., running on multiple systems, so if one part crashes it doesn't affect the whole system. Architects designing distributed systems have to choose between Consistency, Availability and Partition tolerance, they can only pick two.