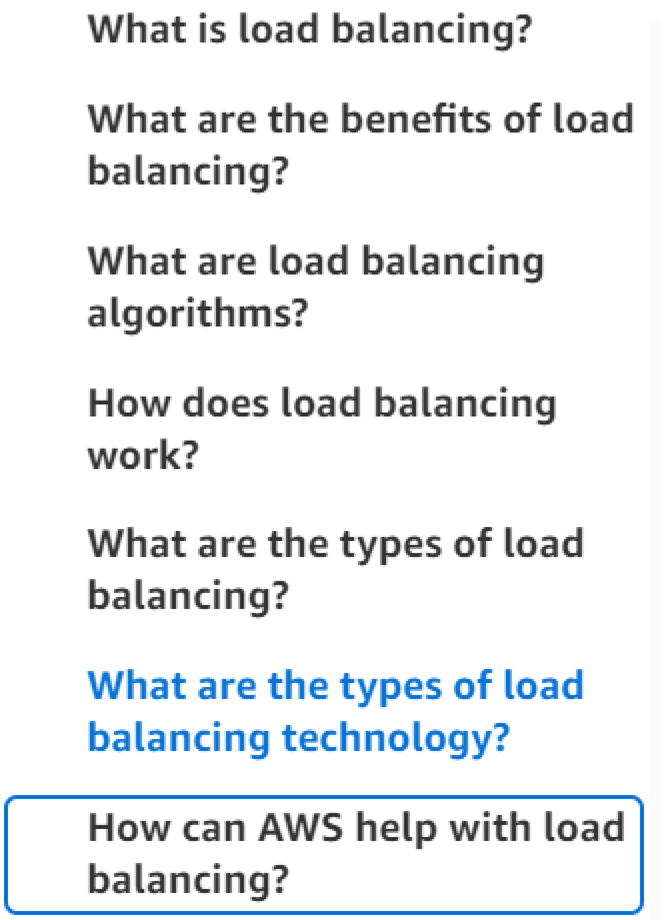
Load Balancing



What is load balancing?

What are the benefits of load balancing?

What are load balancing algorithms?

How does load balancing work?

What are the types of load balancing?

What are the types of load balancing technology?

How can AWS help with load balancing?

What is load balancing?

- Load balancing is the method of distributing network traffic equally across a pool of resources that support an application. Modern applications must process millions of users simultaneously and return the correct text, videos, images, and other data to each user in a fast and reliable manner.
- To handle such high volumes of traffic, most applications have many resource servers with duplicate data between them.
- A load balancer is a device that sits between the user and the server group and acts as an invisible facilitator, ensuring that all resource servers are used equally.

What are the benefits of load balancing?

- Load balancing directs and controls internet traffic between the application servers and their visitors or clients. As a result, it improves an application’s availability, scalability, security, and performance.

Application availability

- Server failure or maintenance can increase application downtime, making your application unavailable to visitors. Load balancers increase the fault tolerance of your systems by automatically detecting server problems and redirecting client traffic to available servers. You can use load balancing to make these tasks easier:
- Run application server maintenance or upgrades without application downtime
- Provide automatic disaster recovery to backup sites
- Perform health checks and prevent issues that can cause downtime

Application scalability

- You can use load balancers to direct network traffic intelligently among multiple servers. Your applications can handle thousands of client requests because load balancing does the following:
- Prevents traffic bottlenecks at any one server
- Predicts application traffic so that you can add or remove different servers, if needed
- Adds redundancy to your system so that you can scale with confidence

Contd.

Application security

- Load balancers come with built-in security features to add another layer of security to your internet applications. They are a useful tool to deal with distributed denial of service attacks, in which attackers flood an application server with millions of concurrent requests that cause server failure. Load balancers can also do the following:
- Monitor traffic and block malicious content
- Automatically redirect attack traffic to multiple backend servers to minimize impact
- Route traffic through a group of network firewalls for additional security

Application performance

- Load balancers improve application performance by increasing response time and reducing network latency. They perform several critical tasks such as the following:
- Distribute the load evenly between servers to improve application performance
- Redirect client requests to a geographically closer server to reduce latency
- Ensure the reliability and performance of physical and virtual computing resources

WHAT ARE LOAD BALANCING ALGORITHMS?

- A load balancing algorithm is the set of rules that a load balancer follows to determine the best server for each of the different client requests. Load balancing algorithms fall into two main categories.

Static load balancing

- Static load balancing algorithms follow fixed rules and are independent of the current server state. The following are examples of static load balancing.
- ***Round-robin method***
- Servers have IP addresses that tell the client where to send requests. The IP address is a long number that is difficult to remember. To make it easy, a Domain Name System maps website names to servers. When you enter aws.amazon.com into your browser, the request first goes to our name server, which returns our IP address to your browser.
- In the round-robin method, an authoritative name server does the load balancing instead of specialized hardware or software. The name server returns the IP addresses of different servers in the server farm turn by turn or in a round-robin fashion.
- ***Weighted round-robin method***
- In weighted round-robin load balancing, you can assign different weights to each server based on their priority or capacity. Servers with higher weights will receive more incoming application traffic from the name server.
- ***IP hash method***
- In the IP hash method, the load balancer performs a mathematical computation, called hashing, on the client IP address. It converts the client IP address to a number, which is then mapped to individual servers.

Dynamic load balancing

- Dynamic load balancing algorithms examine the current state of the servers before distributing traffic. The following are some examples of dynamic load balancing algorithms.
- ***Least connection method***
- A connection is an open communication channel between a client and a server. When the client sends the first request to the server, they authenticate and establish an active connection between each other. In the least connection method, the load balancer checks which servers have the fewest active connections and sends traffic to those servers. This method assumes that all connections require equal processing power for all servers.
- ***Weighted least connection method***
- Weighted least connection algorithms assume that some servers can handle more active connections than others. Therefore, you can assign different weights or capacities to each server, and the load balancer sends the new client requests to the server with the least connections by capacity.
- ***Least response time method***
- The response time is the total time that the server takes to process the incoming requests and send a response. The least response time method combines the server response time and the active connections to determine the best server. Load balancers use this algorithm to ensure faster service for all users.
- ***Resource-based method***
- In the resource-based method, load balancers distribute traffic by analyzing the current server load. Specialized software called an agent runs on each server and calculates usage of server resources, such as its computing capacity and memory. Then, the load balancer checks the agent for sufficient free resources before distributing traffic to that server.

What are the types of load balancing?

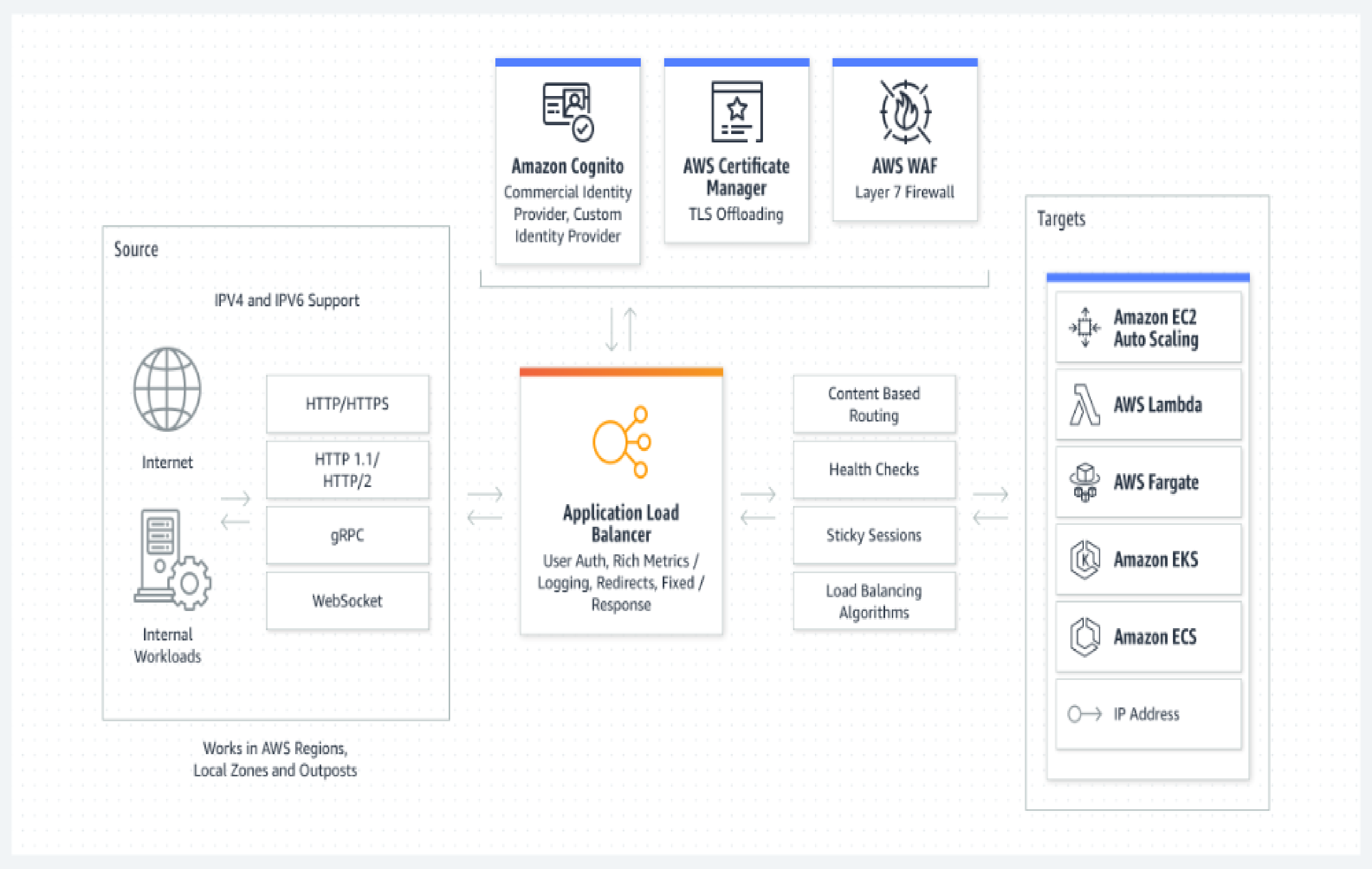
- **Application load balancing**
- Complex modern applications have several server farms with multiple servers dedicated to a single application function. Application load balancers look at the request content, such as HTTP headers or SSL session IDs, to redirect traffic.
- For example, an ecommerce application has a product directory, shopping cart, and checkout functions. The application load balancer sends requests for browsing products to servers that contain images and videos but do not need to maintain open connections. By comparison, it sends shopping cart requests to servers that can maintain many client connections and save cart data for a long time.
- **Network load balancing**
- Network load balancers examine IP addresses and other network information to redirect traffic optimally. They track the source of the application traffic and can assign a static IP address to several servers. Network load balancers use the static and dynamic load balancing algorithms described earlier to balance server load.
- **Global server load balancing**
- Global server load balancing occurs across several geographically distributed servers. For example, companies can have servers in multiple data centers, in different countries, and in third-party cloud providers around the globe. In this case, local load balancers manage the application load within a region or zone. They attempt to redirect traffic to a server destination that is geographically closer to the client. They might redirect traffic to servers outside the client’s geographic zone only in case of server failure.
- **DNS load balancing**
- In DNS load balancing, you configure your domain to route network requests across a pool of resources on your domain. A domain can correspond to a website, a mail system, a print server, or another service that is made accessible through the internet. DNS load balancing is helpful for maintaining application availability and balancing network traffic across a globally distributed pool of resources.

What are the types of load balancing technology?

- **Hardware load balancers**
- A hardware-based load balancer is a hardware appliance that can securely process and redirect gigabytes of traffic to hundreds of different servers. You can store it in your data centers and use virtualization to create multiple digital or virtual load balancers that you can centrally manage.
- **Software load balancers**
- Software-based load balancers are applications that perform all load balancing functions. You can install them on any server or access them as a fully managed third-party service.
- **Comparison of hardware balancers to software load balancers**
- Hardware load balancers require an initial investment, configuration, and ongoing maintenance. You might also not use them to full capacity, especially if you purchase one only to handle peak-time traffic spikes. If traffic volume increases suddenly beyond its current capacity, this will affect users until you can purchase and set up another load balancer.
- In contrast, software-based load balancers are much more flexible. They can scale up or down easily and are more compatible with modern cloud computing environments. They also cost less to set up, manage, and use over time.

How does load balancing work?

- Companies usually have their application running on multiple servers. Such a server arrangement is called a server farm.
- User requests to the application first go to the load balancer. The load balancer then routes each request to a single server in the server farm best suited to handle the request.
- Load balancing is like the work done by a manager in a restaurant. Consider a restaurant with five waiters. If customers were allowed to choose their waiters, one or two waiters could be overloaded with work while the others are idle. To avoid this scenario, the restaurant manager assigns customers to the specific waiters who are best suited to serve them.

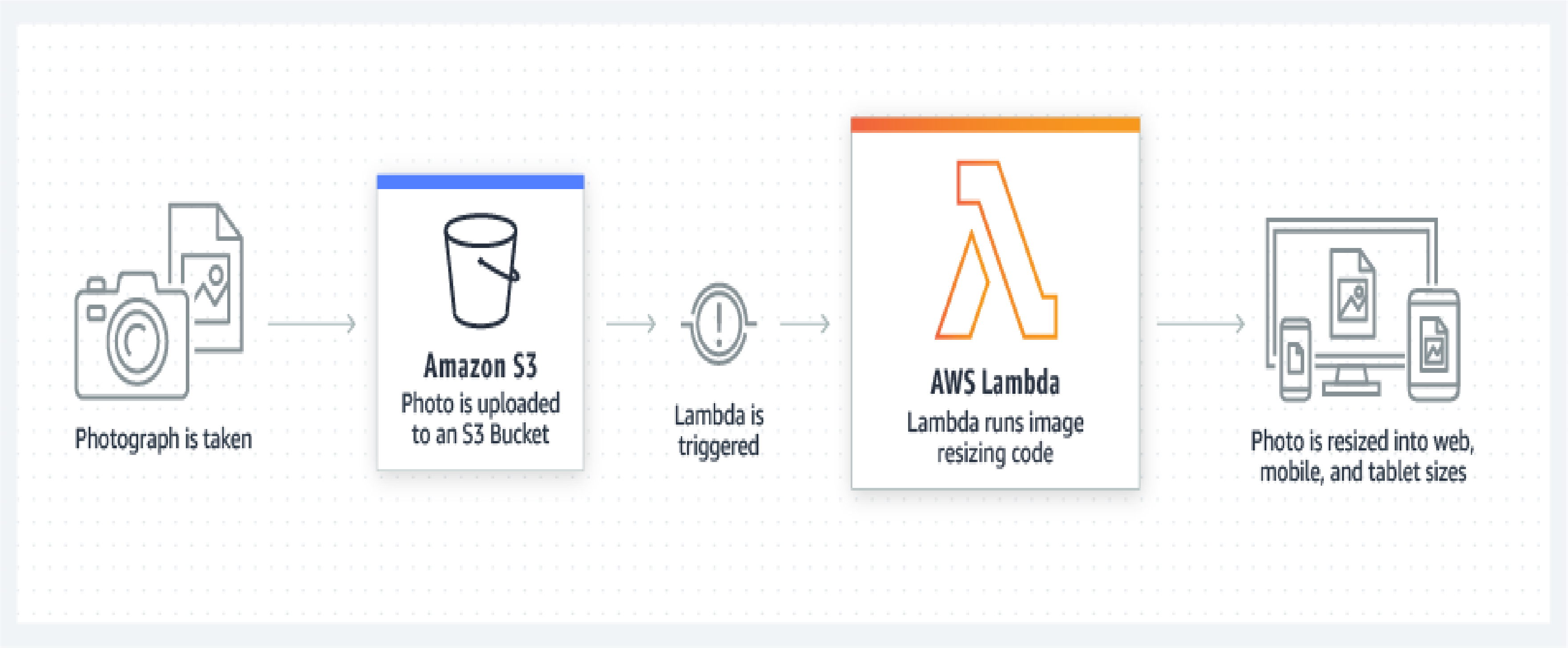

Amazon Cognito
Commercial Identity Provider, Custom Identity Provider
AWS Certificate Manager
TLS Offloading
AWS WAF
Layer 7 Firewall
Source
IPV4 and IPV6 Support
Internet
Internal Workloads
HTTP/HTTPS
HTTP 1.1/ HTTP/2
gRPC
WebSocket
Application Load Balancer
User Auth, Rich Metrics / Logging, Redirects, Fixed / Response
Content Based Routing
Health Checks
Sticky Sessions
Load Balancing Algorithms
Targets
Amazon EC2 Auto Scaling
AWS Lambda
AWS Fargate
Amazon EKS
Amazon ECS
IP Address
Works in AWS Regions, Local Zones and Outposts

AWS Lambda

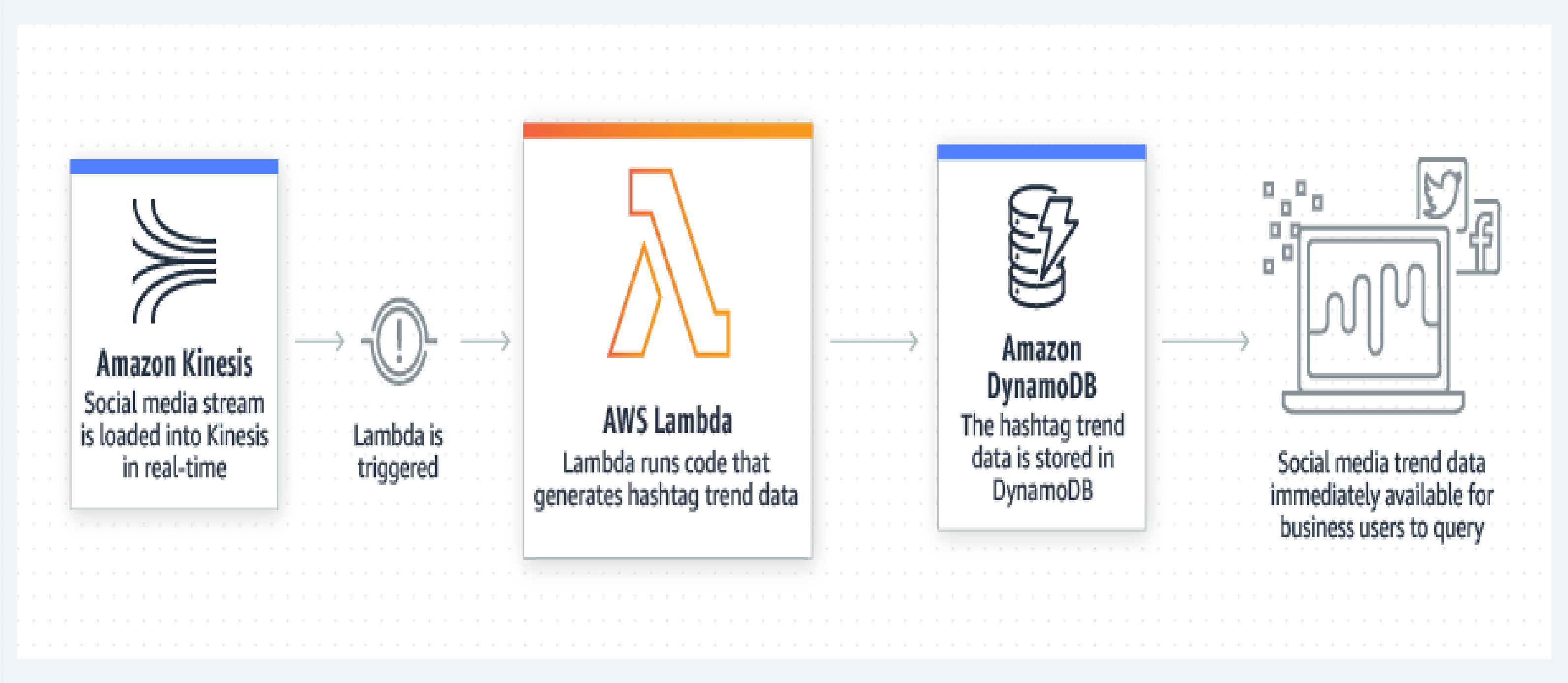
Run code without thinking about servers or clusters

- AWS Lambda is a serverless, event-driven compute service that lets you run code for virtually any type of application or backend service without provisioning or managing servers.
- File processing
- Stream Processing
- Web application’
- IOT backends
- Mobile backends

File Processing


Photograph is taken
Amazon S3
Photo is uploaded to an S3 Bucket
Lambda is triggered
AWS Lambda
Lambda runs image resizing code
Photo is resized into web, mobile, and tablet sizes

Stream Processing



Amazon Kinesis
Social media stream is loaded into Kinesis in real-time

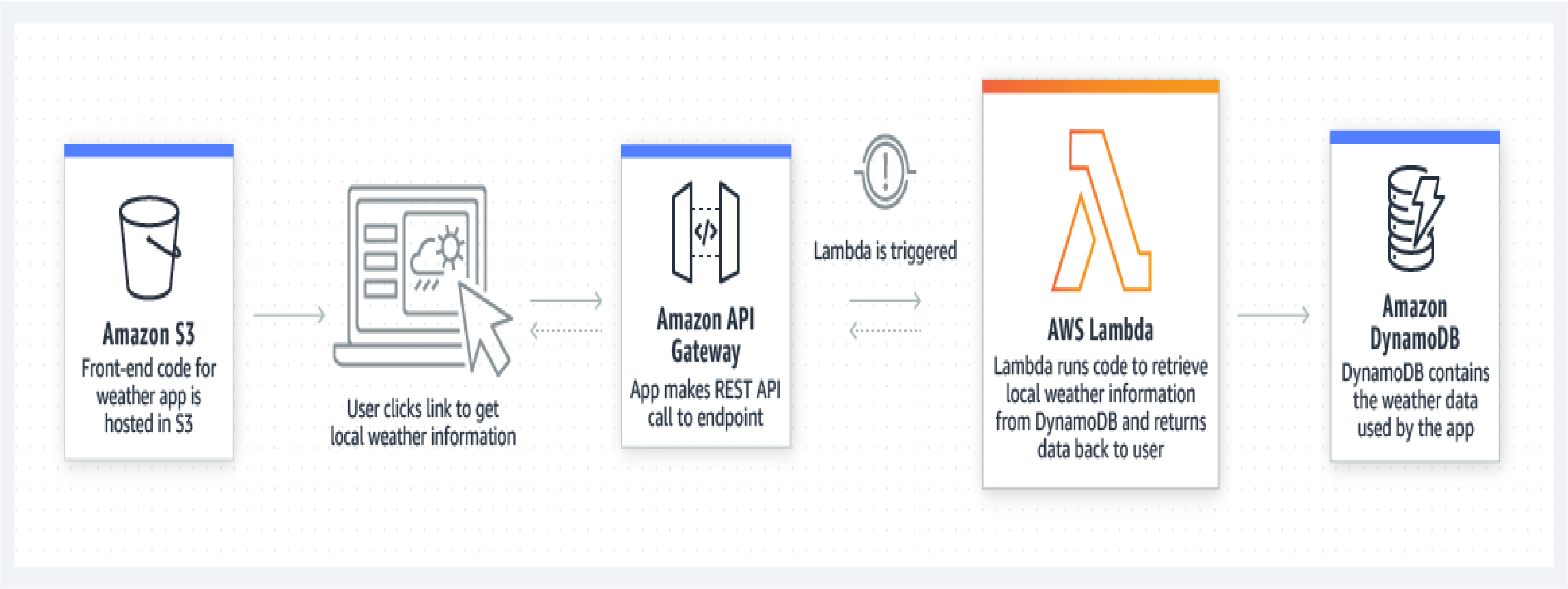
Lambda is triggered

AWS Lambda
Lambda runs code that generates hashtag trend data

Amazon DynamoDB
The hashtag trend data is stored in DynamoDB

Social media trend data immediately available for business users to query

Web application



Amazon S3

Front-end code for weather app is hosted in S3

User clicks link to get local weather information

Amazon API Gateway

App makes REST API call to endpoint

Lambda is triggered

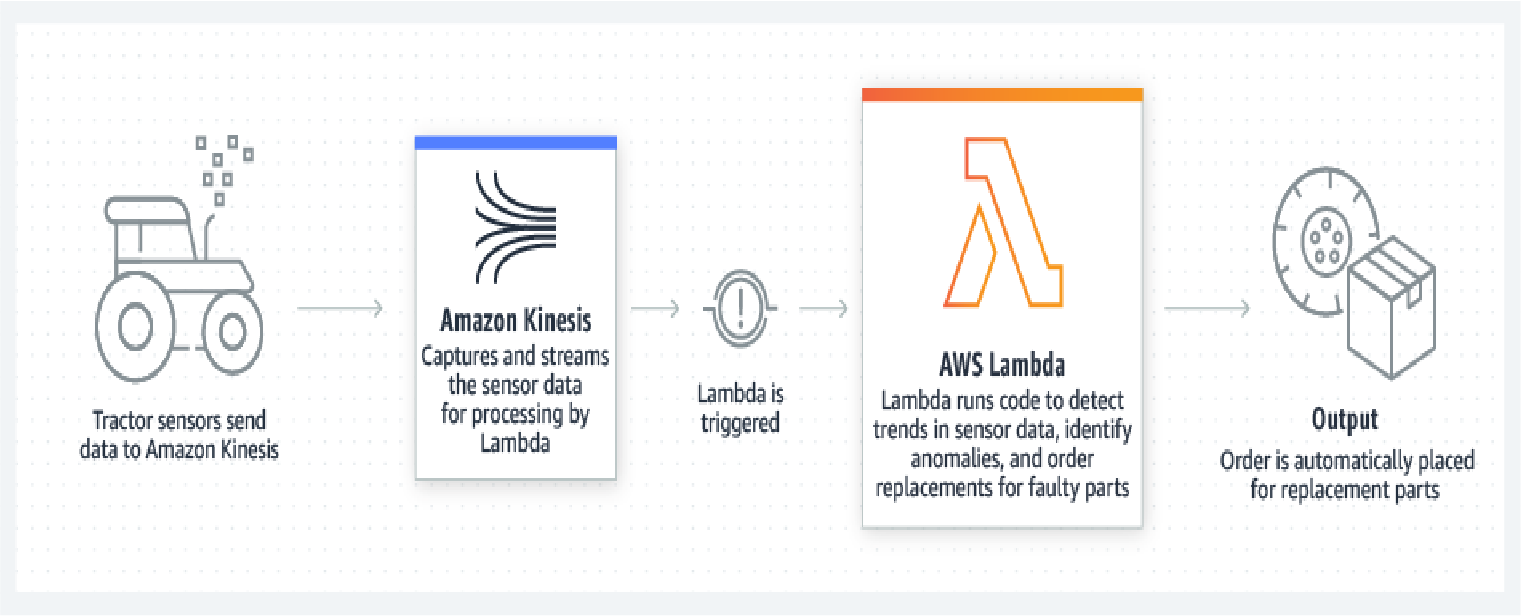
AWS Lambda

Lambda runs code to retrieve local weather information from DynamoDB and returns data back to user

Amazon DynamoDB

DynamoDB contains the weather data used by the app

IOT backends



Tractor sensors send data to Amazon Kinesis

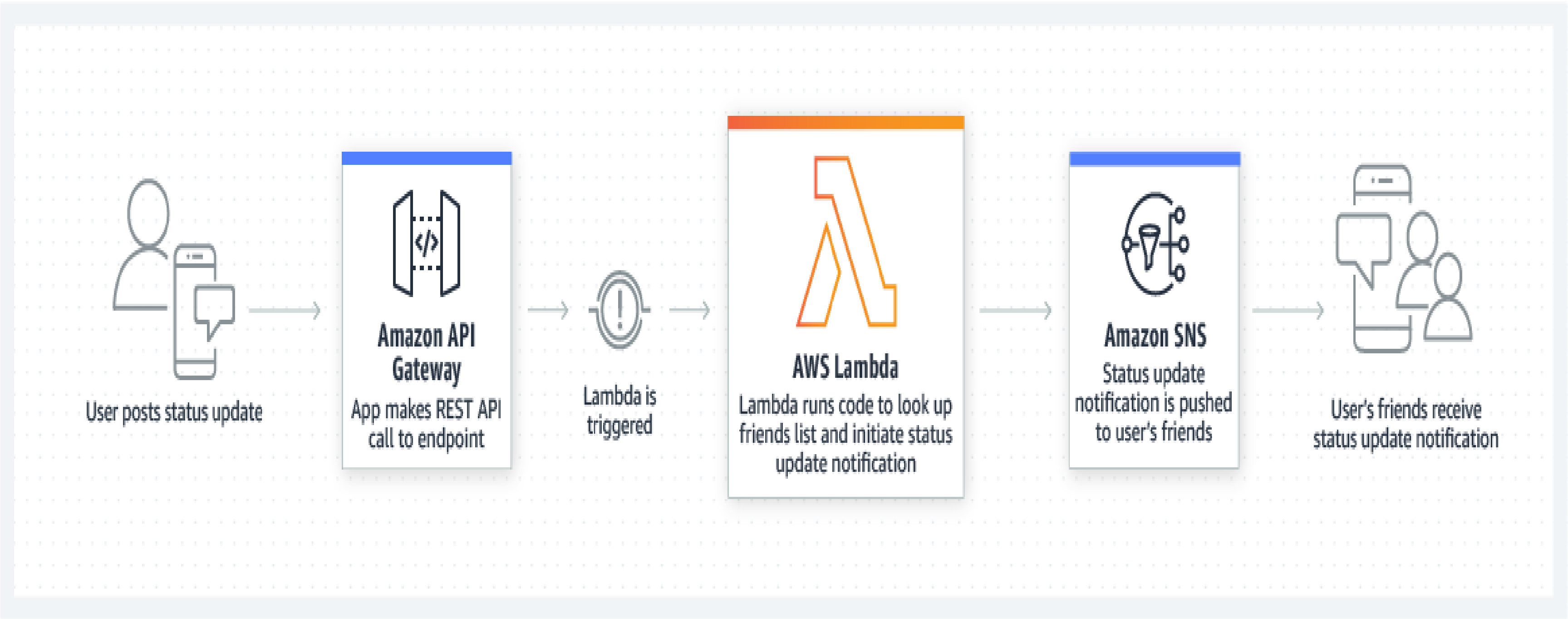
Amazon Kinesis
Captures and streams the sensor data for processing by Lambda

Lambda is triggered

AWS Lambda
Lambda runs code to detect trends in sensor data, identify anomalies, and order replacements for faulty parts

Output
Order is automatically placed for replacement parts

Mobile backends



User posts status update

Amazon API Gateway

App makes REST API call to endpoint

Lambda is triggered

AWS Lambda

Lambda runs code to look up friends list and initiate status update notification

Amazon SNS

Status update notification is pushed to user's friends

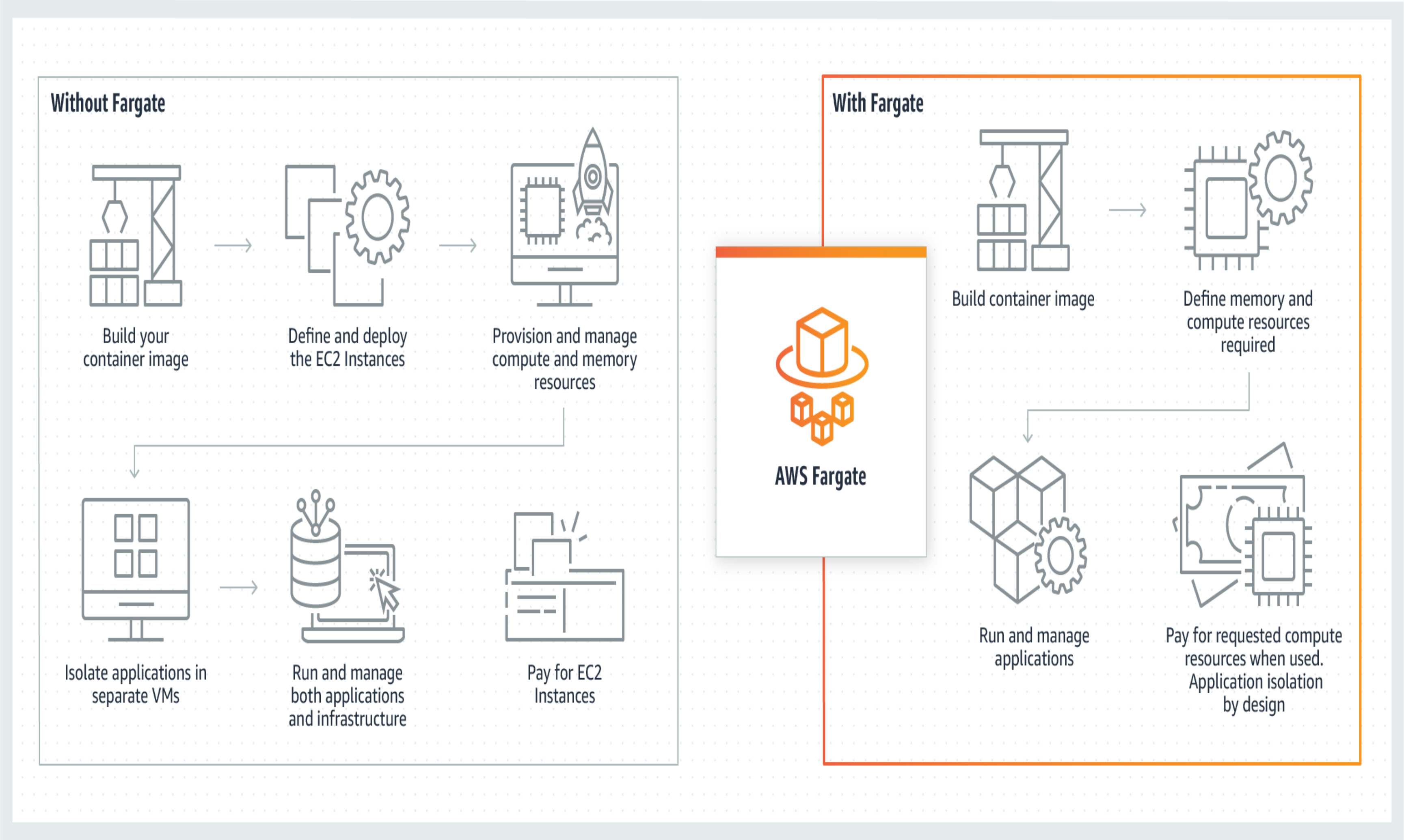
User's friends receive status update notification

AWS Fargate

Serverless compute for containers

How it works ?

- AWS Fargate is a serverless, pay-as-you-go compute engine that lets you focus on building applications without managing servers. AWS Fargate is compatible with both Amazon Elastic Container Service (ECS) and Amazon Elastic Kubernetes Service (EKS).



Without Fargate

Build your container image

Define and deploy the EC2 Instances

Provision and manage compute and memory resources

Isolate applications in separate VMs

Run and manage both applications and infrastructure

Pay for EC2 Instances

AWS Fargate

With Fargate

Build container image

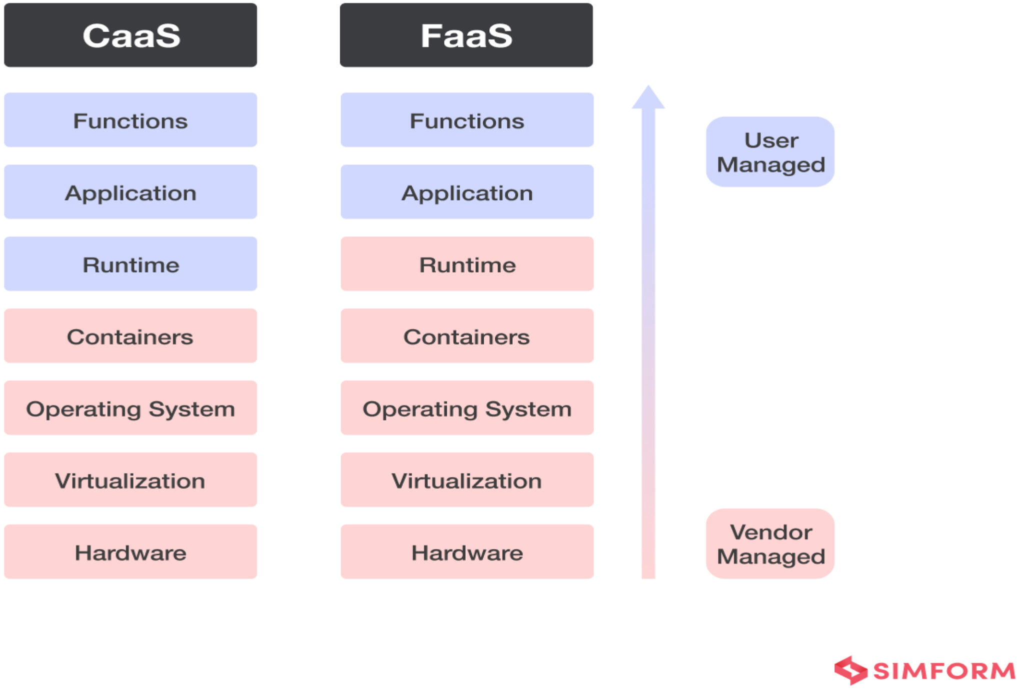
Define memory and compute resources required

Run and manage applications

Pay for requested compute resources when used. Application isolation by design

What is the difference between EC2 Lambda and Fargate?

- **While Fargate is a Container as a Service (CaaS) offering, AWS Lambda is a Function as a Service (FaaS offering)**. Therefore, Lambda functions do not necessarily need to be packaged into containers, making it easier to get started with Lambda. But if you have containerized applications, Fargate is the way to go.



How can AWS help with load balancing?

- Elastic Load Balancing (ELB) is a fully managed load balancing service that automatically distributes incoming application traffic to multiple targets and virtual appliances across AWS and on-premises resources.
- You can use it to scale modern applications without complex configurations or API gateways.
- You can use ELB to set up four different types of software load balancers.
- An Application Load Balancer routes traffic for HTTP-based requests.
- A Network Load Balancer routes traffic based on IP addresses. It is ideal for balancing TCP and User Datagram Protocol (UDP)-based requests.
- A Gateway Load Balancer routes traffic to third-party virtual appliances. It is ideal for incorporating a third-party appliance, such as a network firewall, into your network traffic in a scalable and easy-to-manage way.
- A Classic Load Balancer routes traffic to applications in the Amazon EC2-Classic network—a single, flat network that you share with other customers.

Example

- For example, Terminix, a global pest control brand, uses Gateway Load Balancer to handle 300% more throughput.
- Second Spectrum, a company that provides artificial intelligence-driven tracking technology for sports broadcasts, uses AWS Load Balancer Controller to reduce hosting costs by 90%.
- Code.org, a nonprofit dedicated to expanding access to computer science in schools, uses Application Load Balancer to handle a 400% spike in traffic efficiently during online coding events.

References

- https://aws.amazon.com/blogs/containers/amazon-ecs-vs-amazon-eks-making-sense-of-aws-container-services/
- https://aws.amazon.com/ecs/
- https://aws.amazon.com/eks/
- https://bluexp.netapp.com/blog/aws-cvo-blg-aws-ecs-vs-eks-6-key-differences#:~:text=4.-,Portability,support%20for%20portability%20of%20workloads.
- https://docs.aws.amazon.com/prescriptive-guidance/latest/patterns/deploy-a-grpc-based-application-on-an-amazon-eks-cluster-and-access-it-with-an-application-load-balancer.html