COMPARE LOADBALANCING AUTOSCALING SERVERLESS COMPUTING

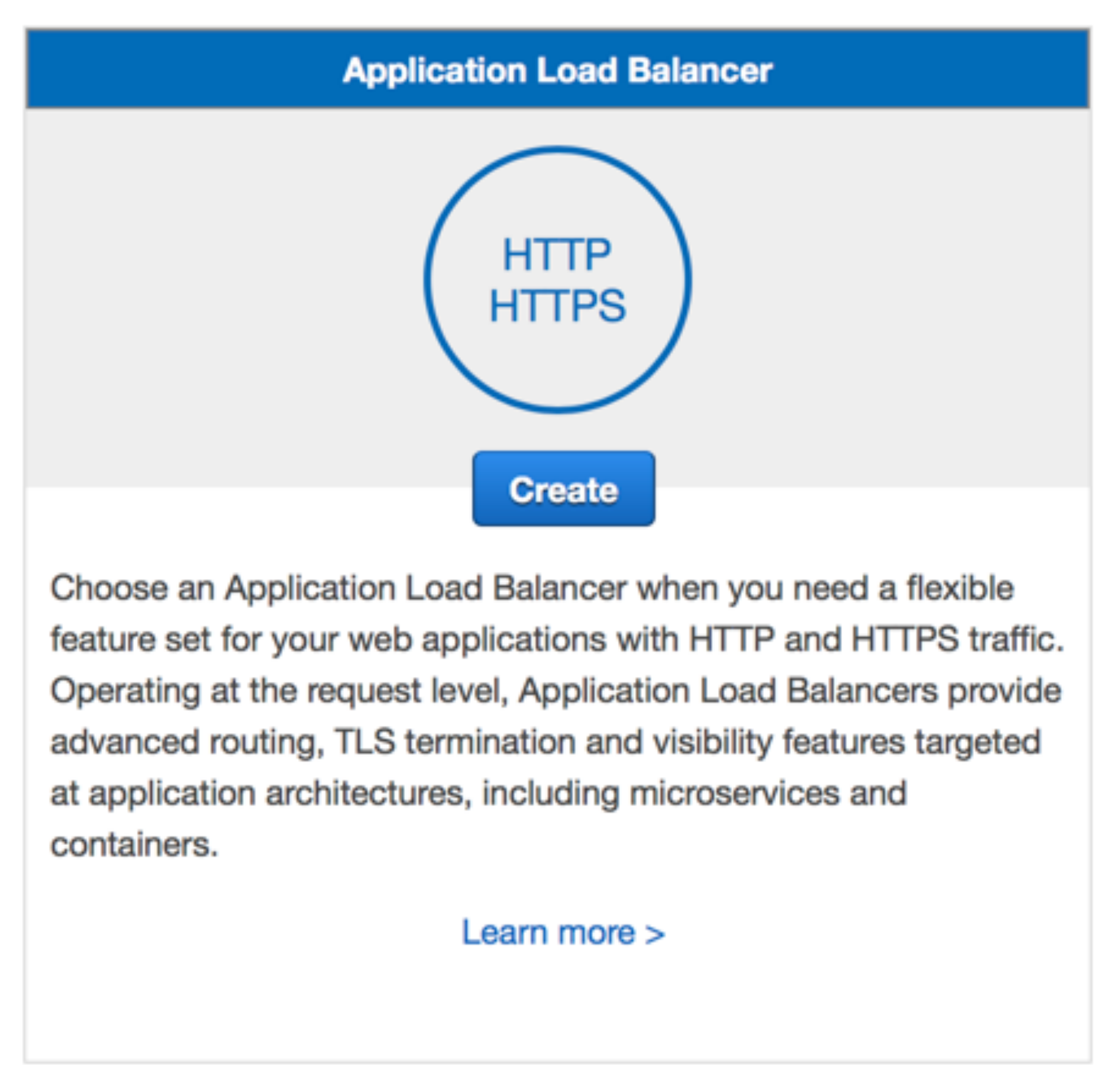
AWS

AWS Load Balancing

- Elastic Load Balancing supports three types of load balancers:
- Application Load Balancers
- Network Load Balancers
- Classic Load Balancers

AWS Application Load Balancers

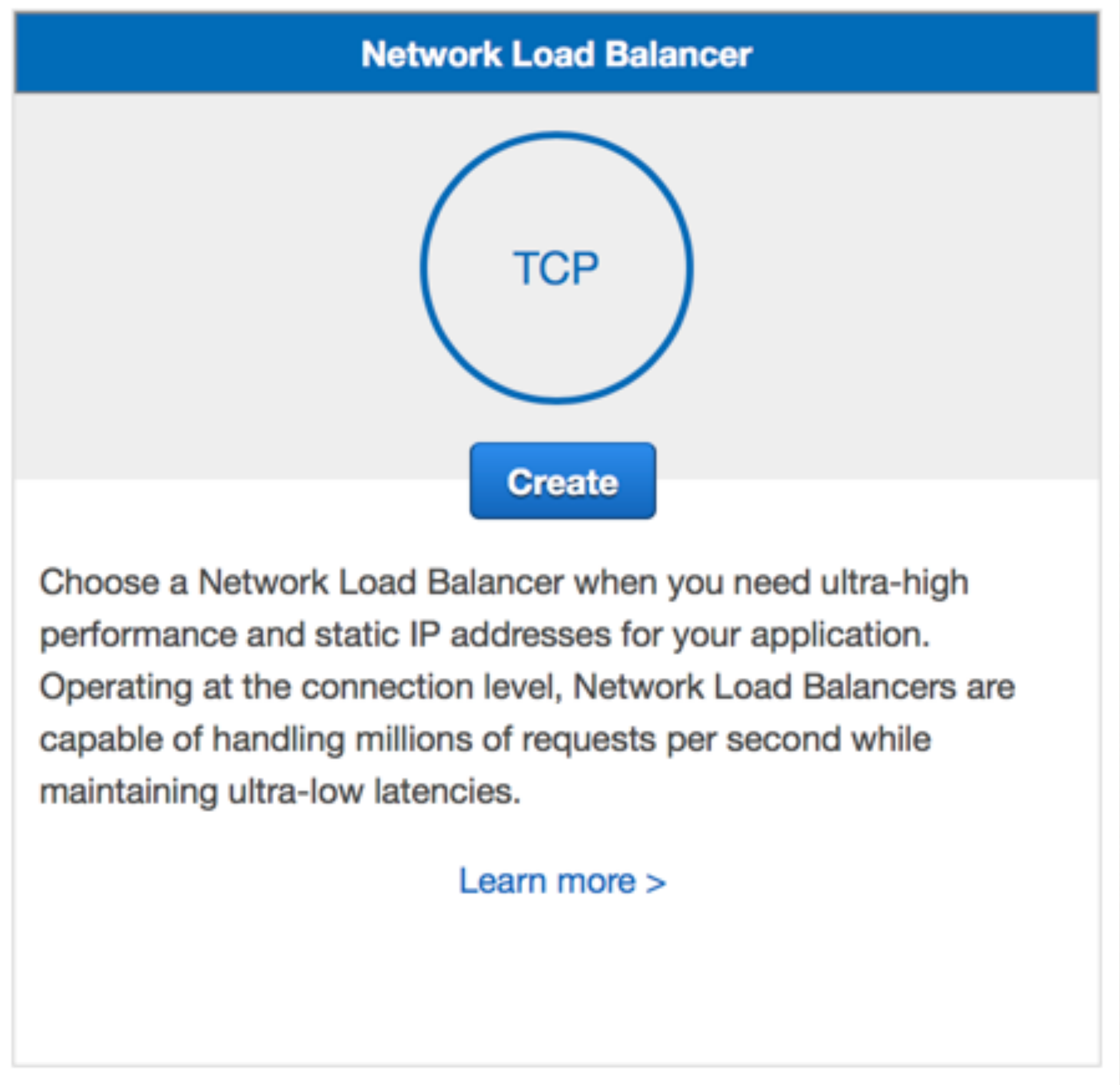
- Application Load Balancer is best suited for load balancing of HTTP and HTTPS traffic and provides advanced request routing targeted at the delivery of modern application architectures, including microservices and containers. Operating at the individual request level (Layer 7), Application Load Balancer routes traffic to targets within Amazon Virtual Private Cloud (Amazon VPC) based on the content of the request.



Choose an Application Load Balancer when you need a flexible feature set for your web applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing, TLS termination and visibility features targeted at application architectures, including microservices and containers.

AWS Network Load Balancer

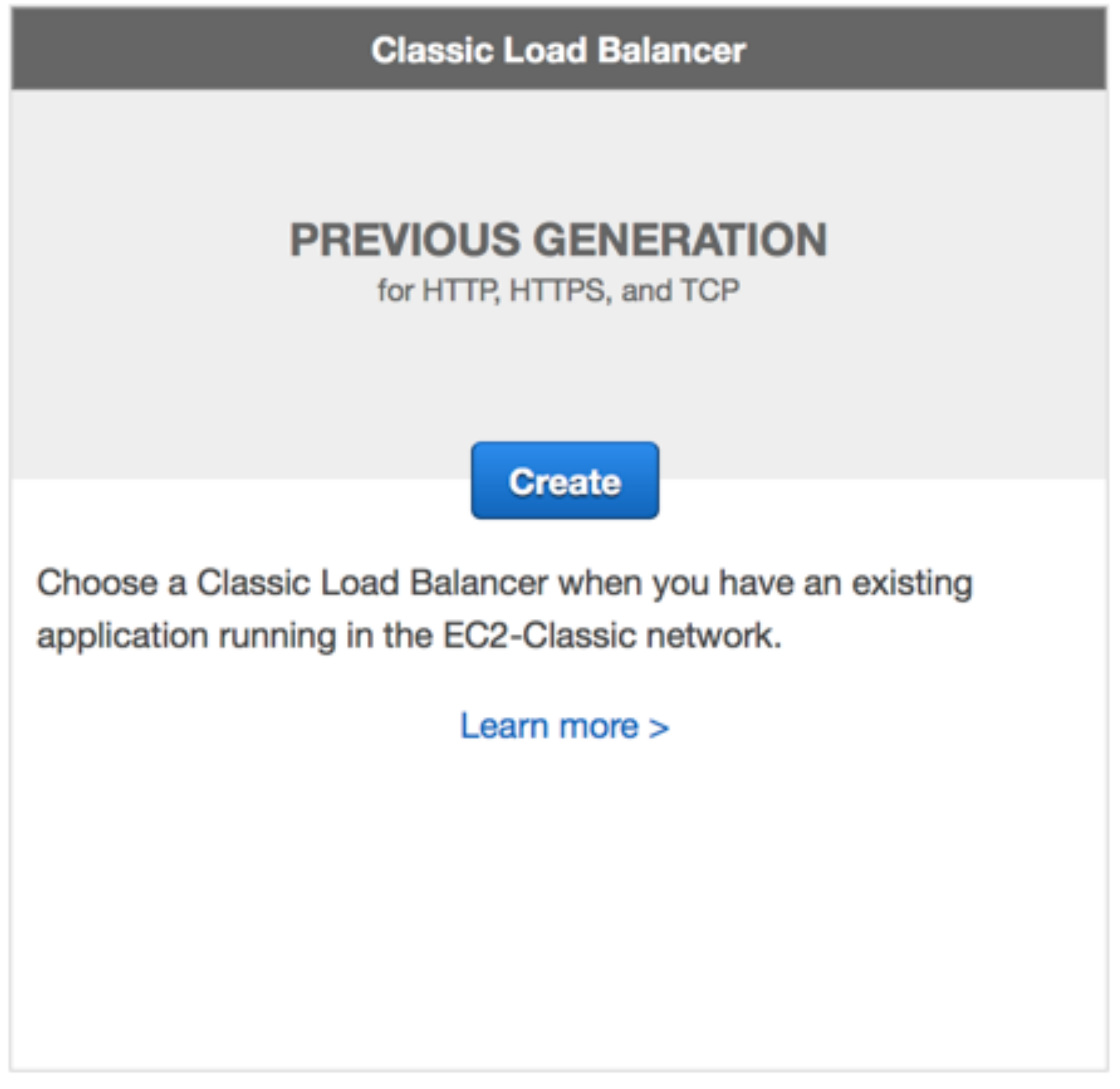
- Network Load Balancer is best suited for load balancing of TCP traffic where extreme performance is required. Operating at the connection level (Layer 4), Network Load Balancer routes traffic to targets within Amazon Virtual Private Cloud (Amazon VPC) and is capable of handling millions of requests per second while maintaining ultra-low latencies. Network Load Balancer is also optimized to handle sudden and volatile traffic patterns.



Choose a Network Load Balancer when you need ultra-high performance and static IP addresses for your application. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second while maintaining ultra-low latencies.

AWS Classic Load Balancer

- Classic Load Balancer provides basic load balancing across multiple Amazon EC2 instances and operates at both the request level and connection level. Classic Load Balancer is intended for applications that were built within the EC2-Classic network.



Classic Load Balancer

PREVIOUS GENERATION

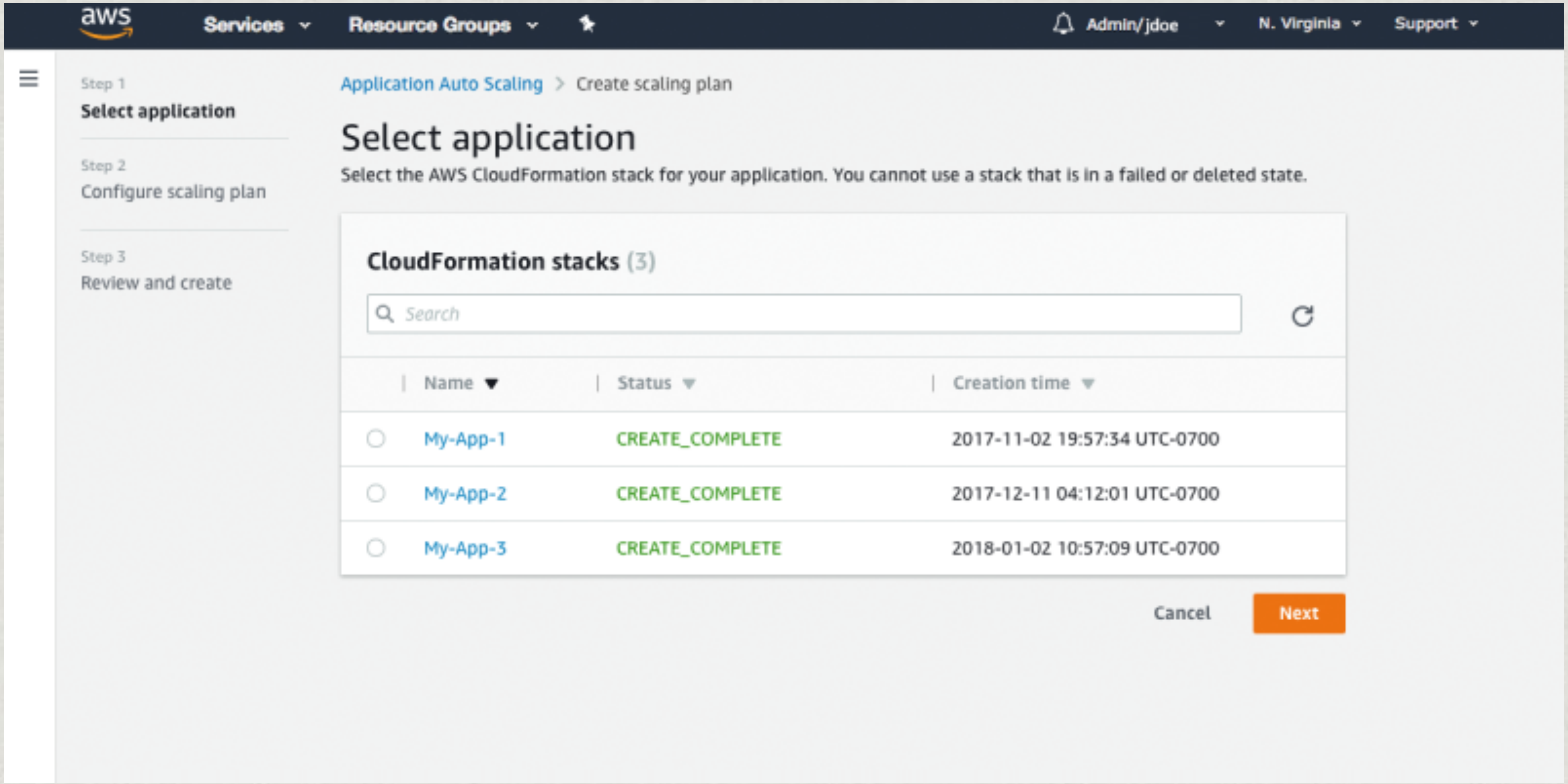
for HTTP, HTTPS, and TCP

Choose a Classic Load Balancer when you have an existing application running in the EC2-Classic network.

AWS Auto Scaling

- Amazon EC2 Auto Scaling is designed to automatically launch or terminate EC2 instances based on user-defined policies, schedules, and health checks. Use this service in conjunction with the AWS Auto Scaling, Amazon CloudWatch, and Elastic Load Balancing services.

AWS Auto Scaling



Select application

Select the AWS CloudFormation stack for your application. You cannot use a stack that is in a failed or deleted state.

CloudFormation stacks (3)

	Name ▼	Status ▼	Creation time ▼
○	My-App-1	CREATE_COMPLETE	2017-11-02 19:57:34 UTC-0700
○	My-App-2	CREATE_COMPLETE	2017-12-11 04:12:01 UTC-0700
○	My-App-3	CREATE_COMPLETE	2018-01-02 10:57:09 UTC-0700

AWS Serverless

- COMPUTE: AWS Lambda
- API PROXY: Amazon API Gateway
- STORAGE: Amazon Simple Storage Service (Amazon S3)
- DATA STORES: Amazon DynamoDB
- INTERPROCESS MESSAGING: Amazon SNS SQS
- ORCHESTRATION: AWS Step Functions
- ANALYTICS: Amazon Kinesis, Amazon Athena

Azure

Azure Load Balancer

- Load balance incoming Internet traffic to virtual machines. This configuration is known as a public Load Balancer.
- Load balance traffic between virtual machines inside a virtual network. You can also reach a Load Balancer frontend from an on-premises network in a hybrid scenario. Both of these scenarios use a configuration that is known as an internal Load Balancer.
- Port forward traffic to a specific port on specific virtual machines with inbound NAT rules.
- Provide outbound connectivity for virtual machines inside your virtual network by using a public Load Balancer.

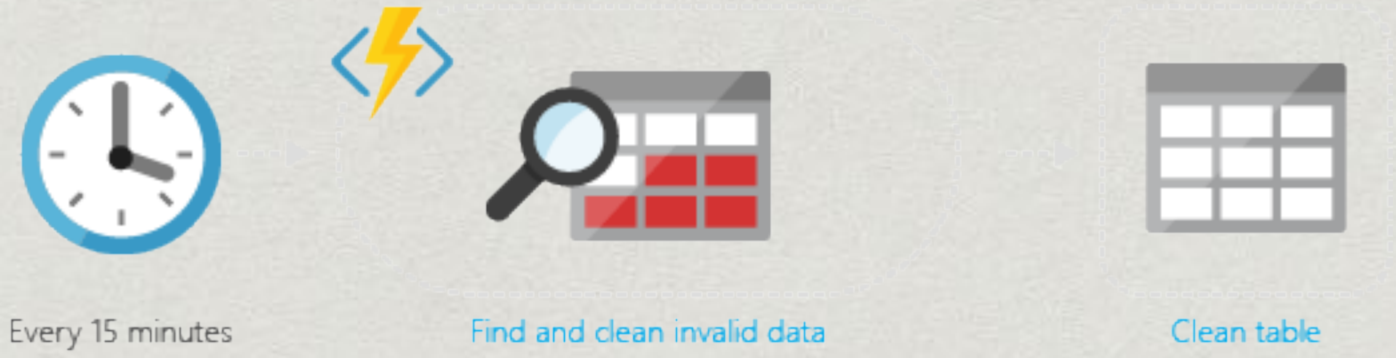
Autoscaling

- **Virtual Machines** support autoscaling through the use of VM Scale Sets, which are a way to manage a set of Azure virtual machines as a group.
- **Service Fabric** also supports auto-scaling through VM Scale Sets. Every node type in a Service Fabric cluster is set up as a separate VM scale set. That way, each node type can be scaled in or out independently.
- **Azure App Service** has built-in autoscaling. Autoscale settings apply to all of the apps within an App Service.
- **Azure Cloud Services** has built-in autoscaling at the role level.
- **Azure Functions** differs from the previous compute options, because you don't need to configure any autoscale rules. Instead, Azure Functions automatically allocates compute power when your code is running, scaling out as necessary to handle load.

Azure
Serverless Computing

Timer-based processing

- Functions supports an event based on a timer using Cron job syntax. For example, execute code that runs every 15 minutes and clean up a database table based on custom business logic.



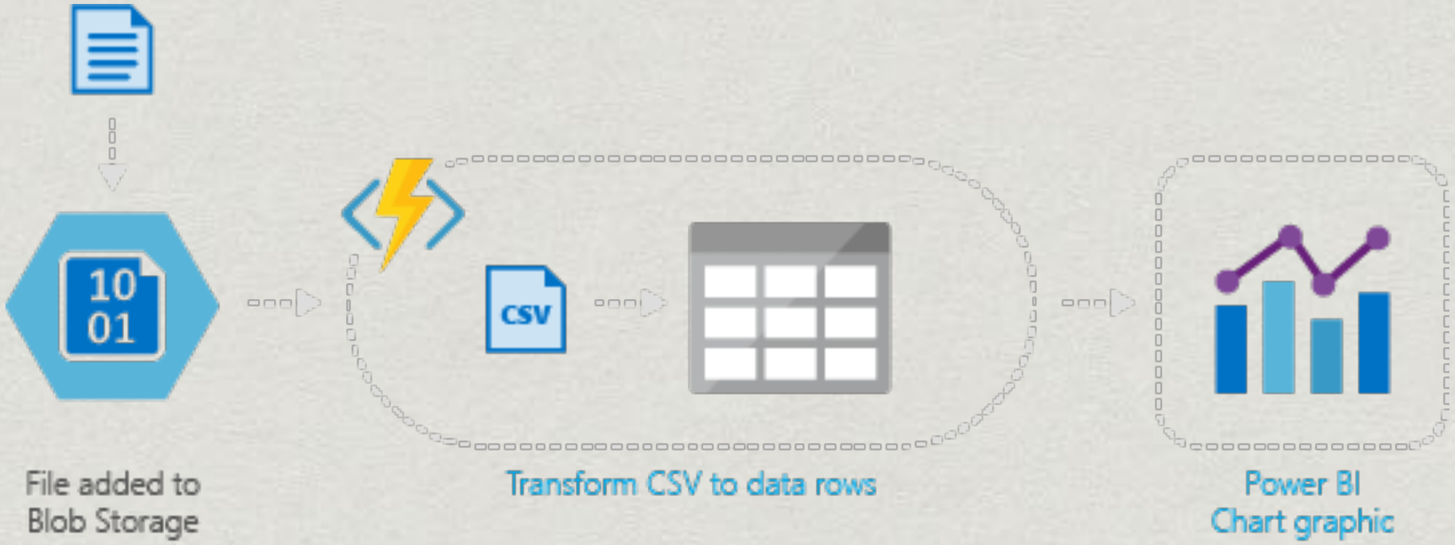
Every 15 minutes

Find and clean invalid data

Clean table

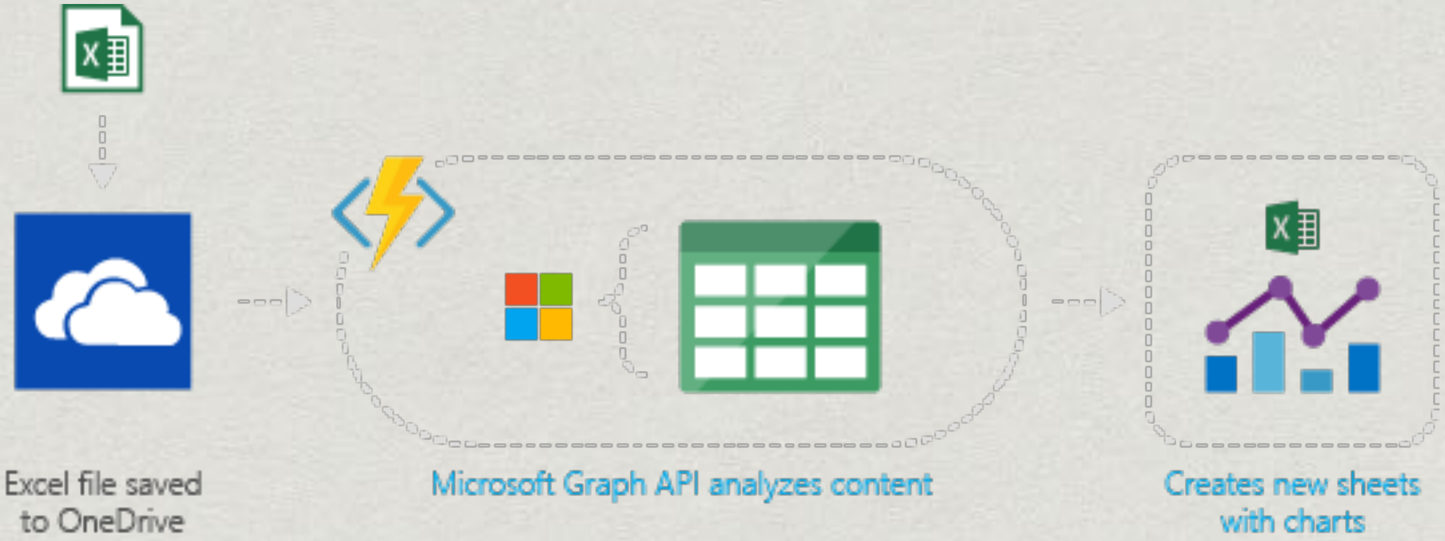
Azure service event processing

- Functions supports triggering an event based on an activity in an Azure service. For example, execute serverless code that reads newly discovered test log files in a Blob storage container, and transform this into a row in a SQL Database table.



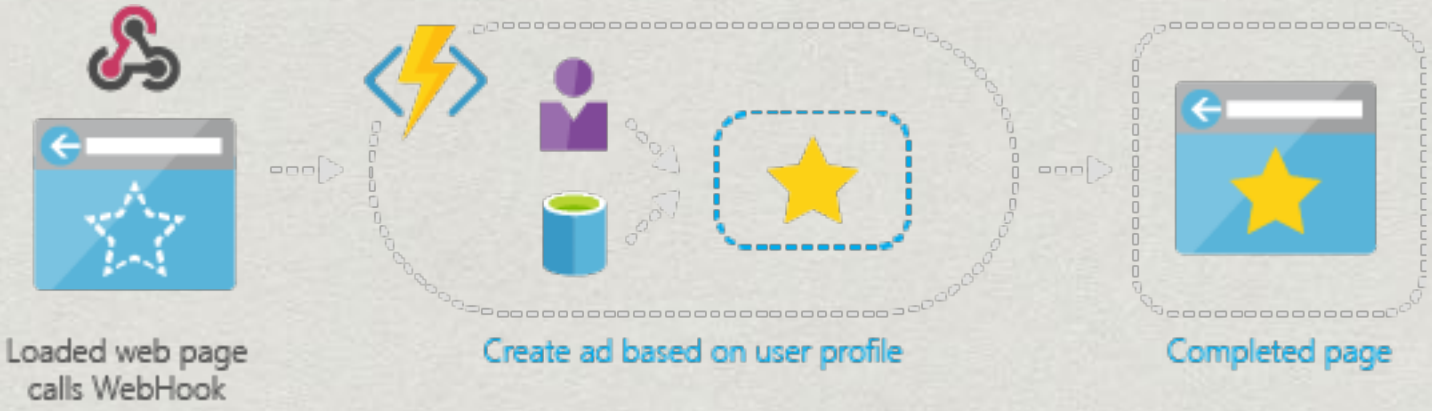
SaaS event processing

- Functions supports triggers based on activity in a Software as a service (SaaS)-based application. For example, save a file in OneDrive, which triggers a function that uses the Microsoft Graph API to modify the spreadsheet, and creates additional charts and calculated data.


Excel file saved to OneDrive
Microsoft Graph API analyzes content
Creates new sheets with charts

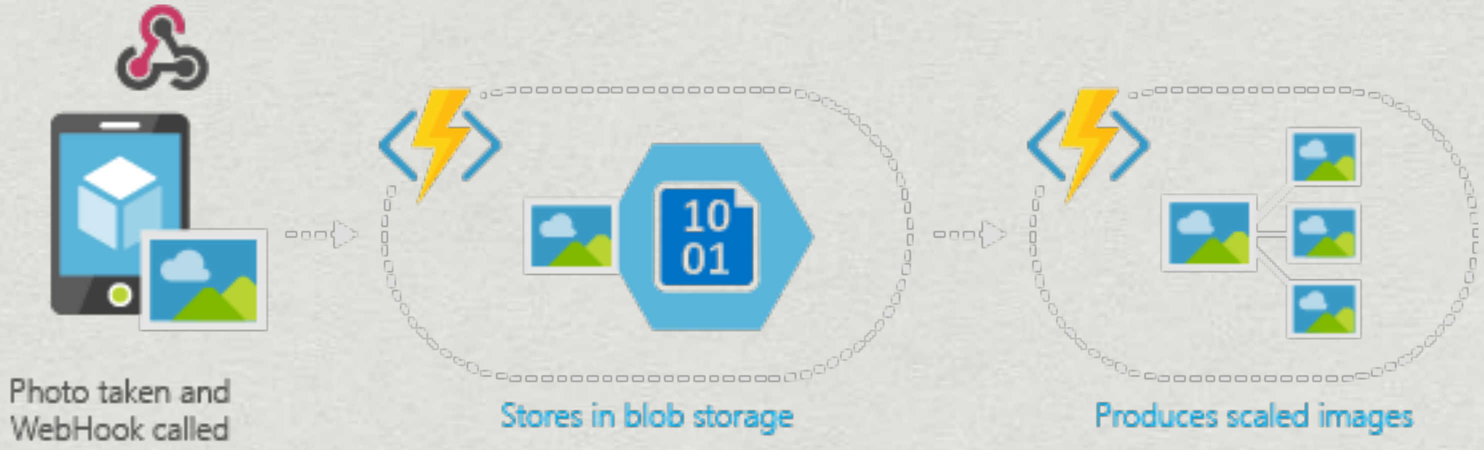
Serverless web application architectures

- Functions can power a single-page app. The app calls functions using the WebHook URL, saves user data, and decides what data to display. Or, do simple customizations, such as changing ad targeting by calling a function and passing it user profile information.



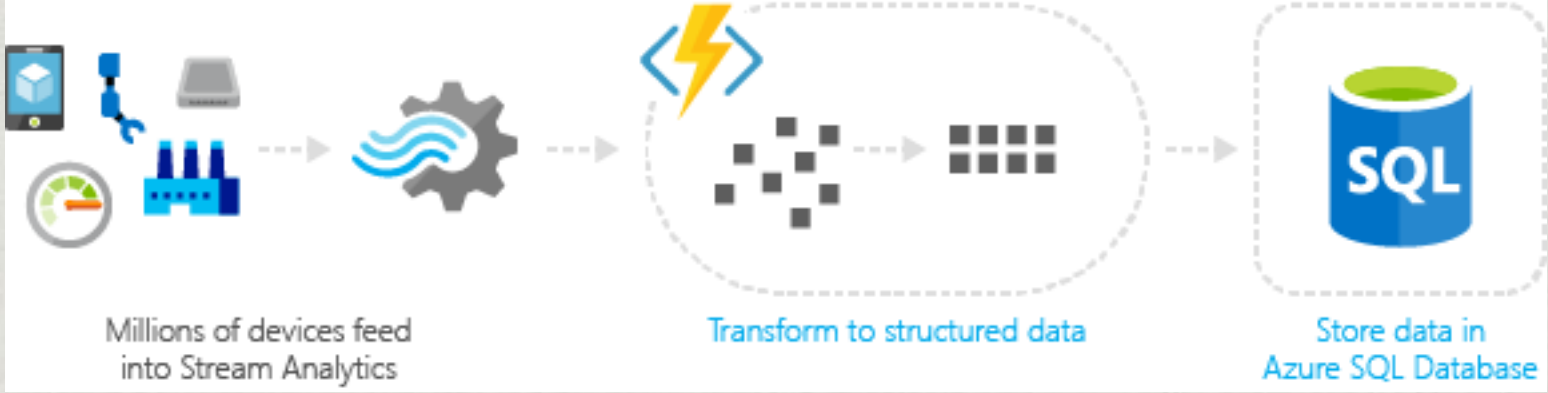
Serverless mobile back ends

- A mobile back end can be a set of HTTP APIs that are called from a mobile client using the WebHook URL. For example, a mobile application can capture an image, and then call a function to get an access token for uploading to blob storage. A second function is triggered by the blob upload and resizes the image to be mobile-friendly.



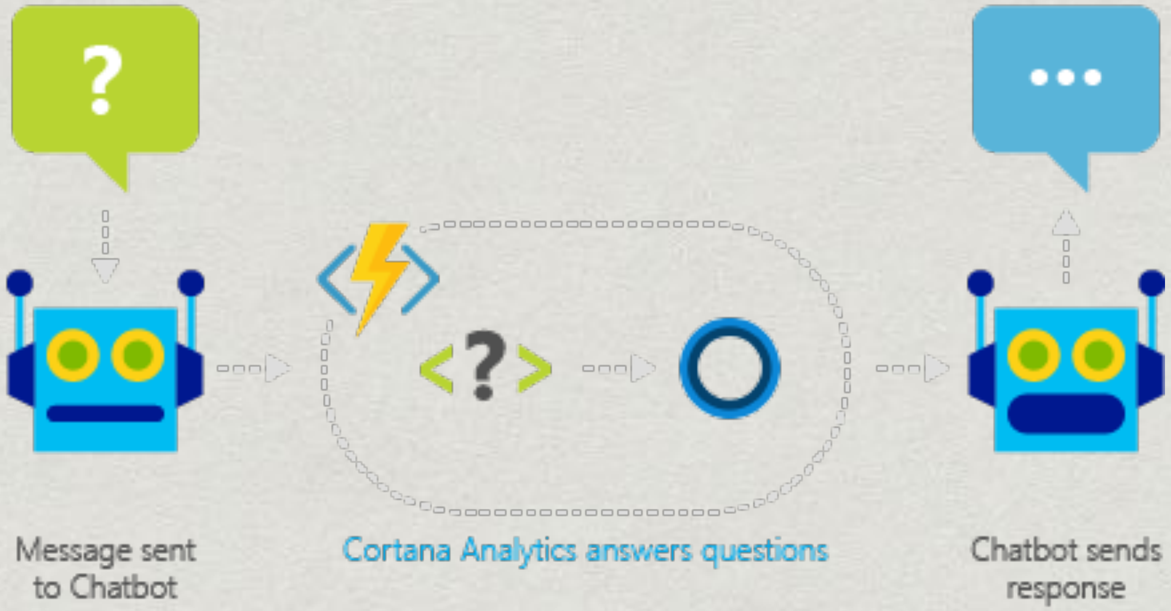
Real-time stream processing

- For example, Internet of Things devices send messages to Stream Analytics, which then calls a function to transform the message. This function processes the data and creates a new record in a SQL database.


SQL
Millions of devices feed into Stream Analytics
Transform to structured data
Store data in Azure SQL Database

Real-time bot messaging

- Use Functions to customize the behavior of a bot using a WebHook. For example, create a function that processes a message using Cortana Analytics and call this function using Microsoft Bot Framework.



Google Cloud Compute

Global external load balancing

- HTTP(S) load balancing distributes HTTP(S) traffic among groups of instances based on proximity to the user, the requested URL, or both.
- SSL Proxy load balancing distributes SSL traffic among groups of instances based on proximity to the user.
- TCP Proxy load balancing distributes TCP traffic among groups of instances based on proximity to the user.

Regional external load balancing

- Network load balancing distributes traffic among a pool of instances within a region. Network load balancing can balance any kind of TCP/UDP traffic.

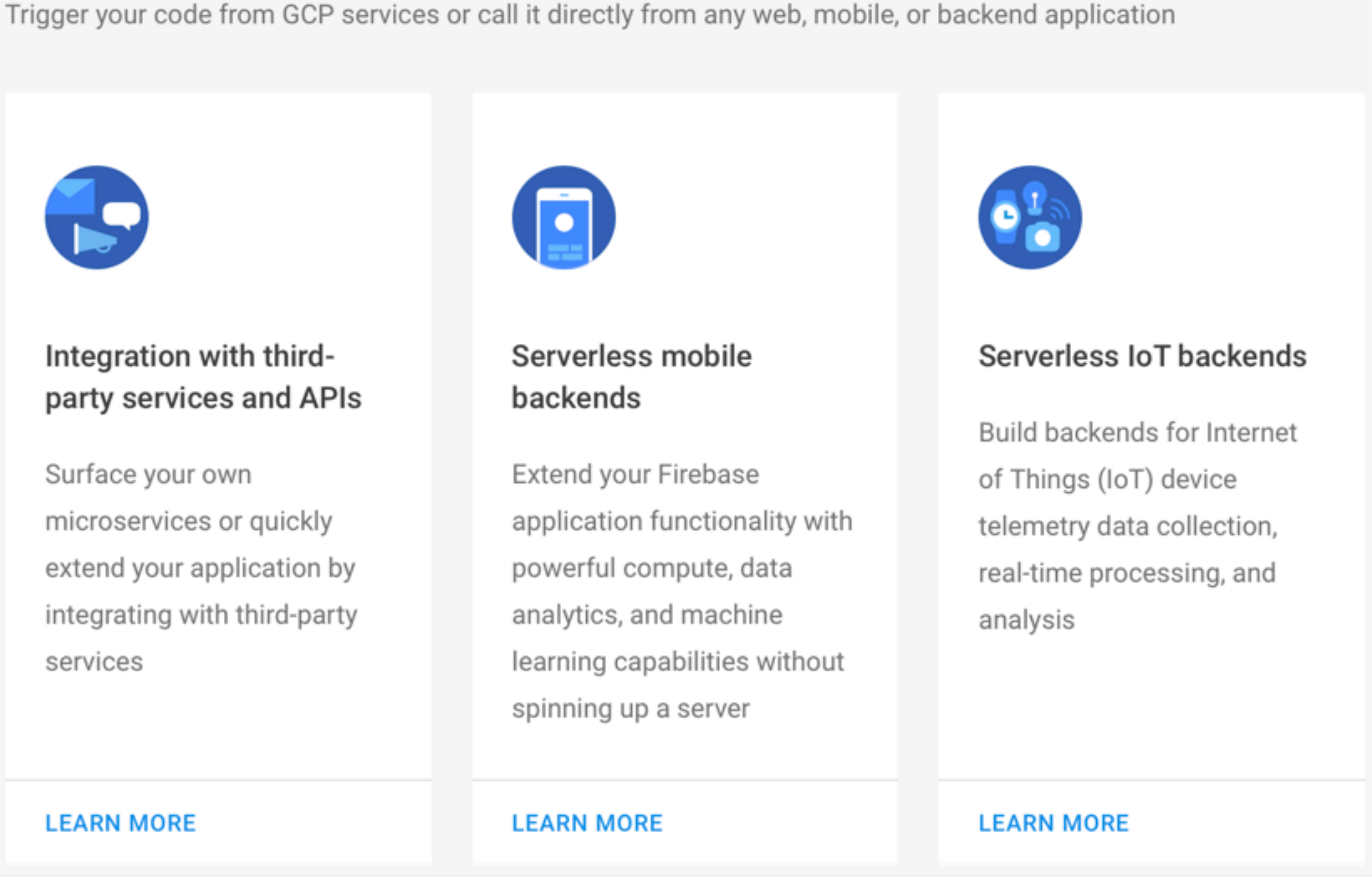
Regional internal load balancing

- Internal load balancing distributes traffic from Google Cloud Platform virtual machine instances to a group of instances in the same region.

Autoscaling

- CPU utilization: CPU utilization is the most basic autoscaling that you can perform. This policy tells the autoscaler to watch the average CPU utilization of a group of virtual machines and add or remove virtual machines from the group to maintain your desired utilization. This is useful for configurations that are CPU-intensive but might fluctuate in CPU usage.
- Load balancing serving capacity: Set up an autoscaler to scale based on load balancing serving capacity and the autoscaler will watch the serving capacity of an instance group, and scale if the virtual machines are over or under capacity. The serving capacity of an instance can be defined in the load balancer's backend service and can be based on either utilization or requests per second.
- Stackdriver Monitoring metrics: If you export or use Stackdriver Monitoring metrics, you can set up autoscaling to collect data of a specific metric and perform scaling based on your desired utilization level. It is possible to scale based on standard metrics provided by Stackdriver Monitoring, or using any custom metrics you create as well.

Serverless application backends



Trigger your code from GCP services or call it directly from any web, mobile, or backend application

Integration with third-party services and APIs

Surface your own microservices or quickly extend your application by integrating with third-party services

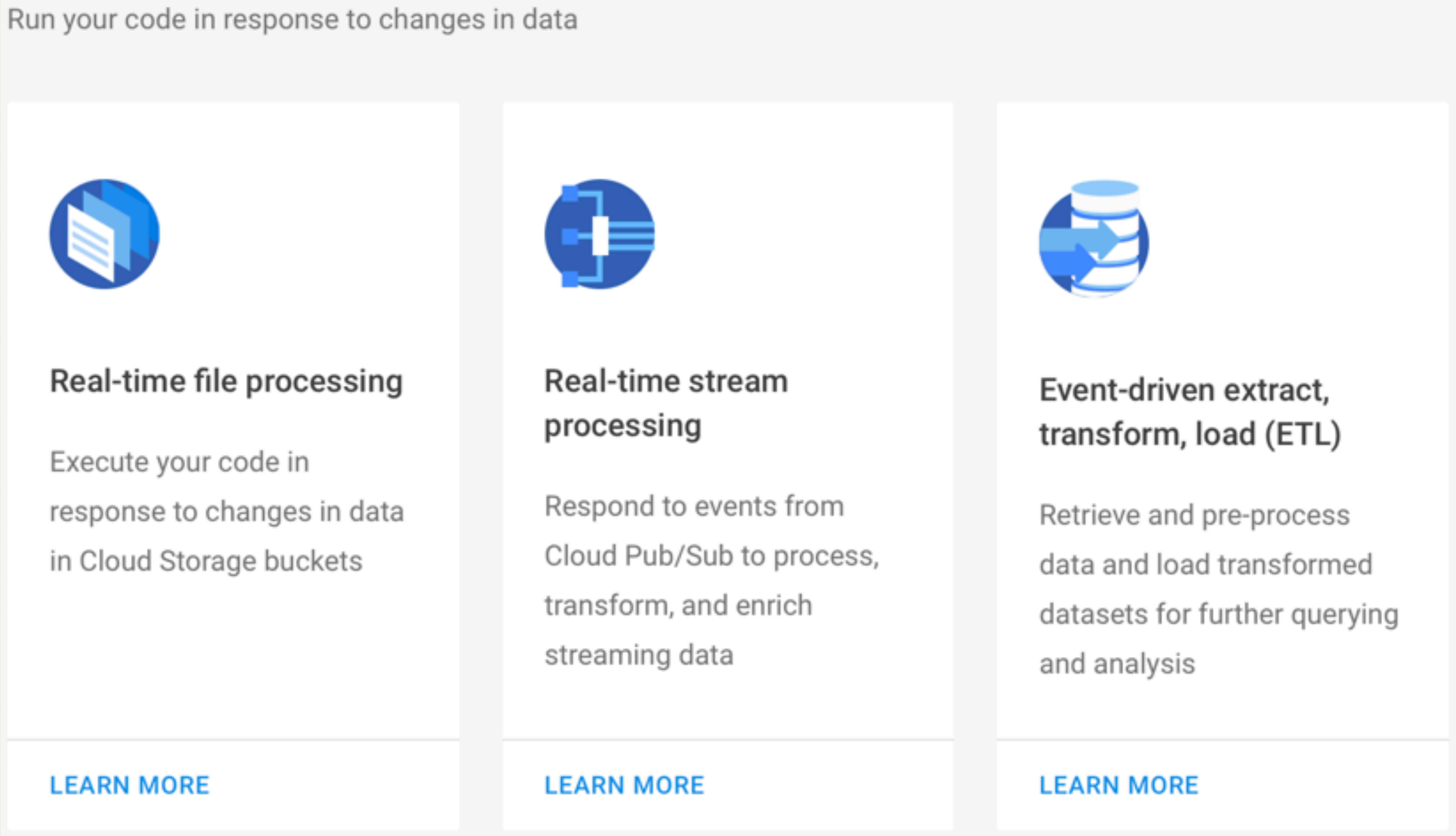
Serverless mobile backends

Extend your Firebase application functionality with powerful compute, data analytics, and machine learning capabilities without spinning up a server

Serverless IoT backends

Build backends for Internet of Things (IoT) device telemetry data collection, real-time processing, and analysis

Real-time data processing systems



Run your code in response to changes in data

Real-time file processing

Execute your code in response to changes in data in Cloud Storage buckets

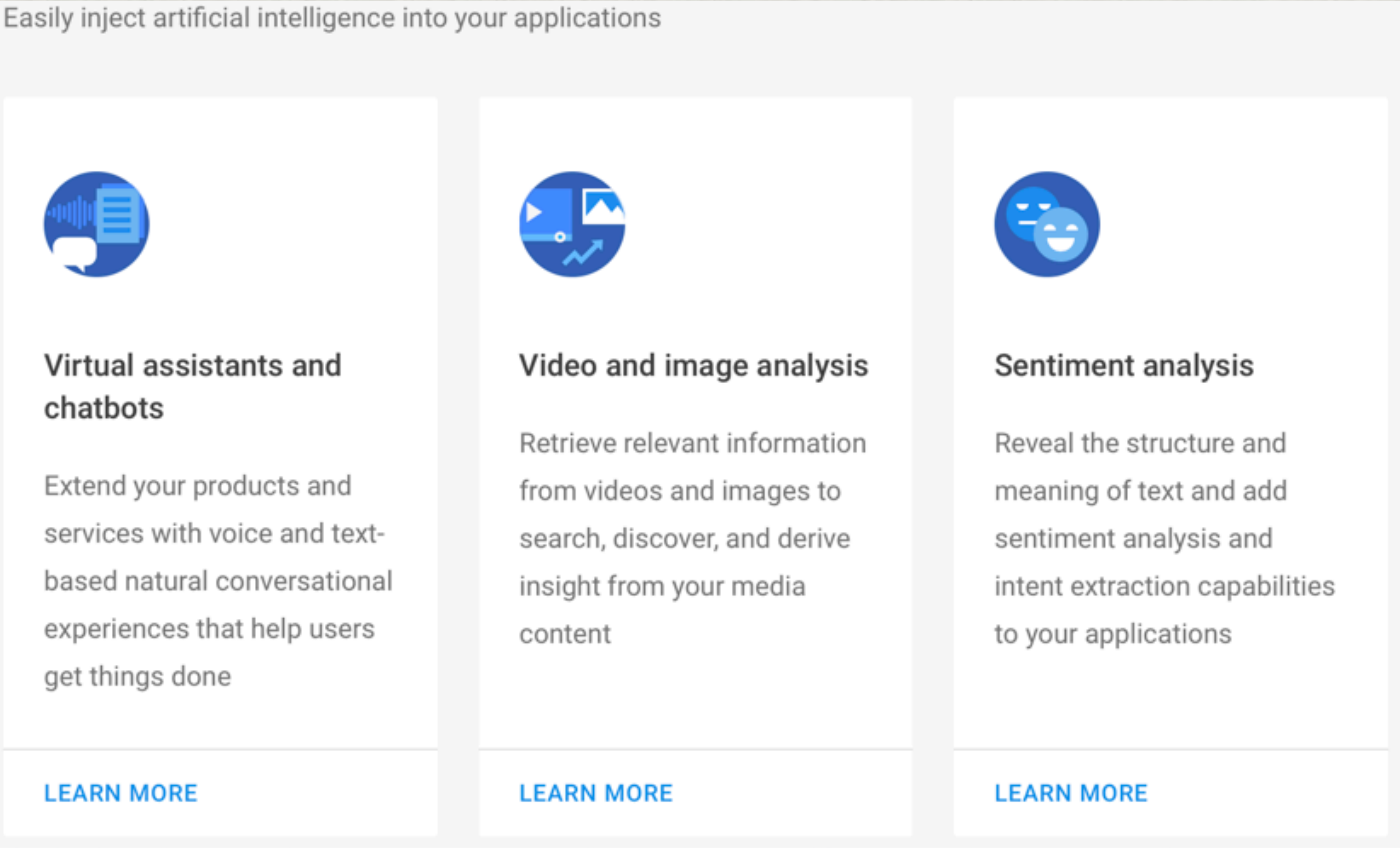
Real-time stream processing

Respond to events from Cloud Pub/Sub to process, transform, and enrich streaming data

Event-driven extract, transform, load (ETL)

Retrieve and pre-process data and load transformed datasets for further querying and analysis

Intelligent applications



Easily inject artificial intelligence into your applications

Virtual assistants and chatbots

Extend your products and services with voice and text-based natural conversational experiences that help users get things done

Video and image analysis

Retrieve relevant information from videos and images to search, discover, and derive insight from your media content

Sentiment analysis

Reveal the structure and meaning of text and add sentiment analysis and intent extraction capabilities to your applications