Evgeni Dyulgerov

Senior .NET Developer at Digitall

Google Cloud Design Patterns

Infrastructure, Machine Learning, Containerization and more...

Software University

SoftUni

Software University

https://about.softuni.bg

About Me

- ✓ .NET and Cloud enthusiast
- ✓ Developer and IT Consultant with 8+ years of experience
- ✓ Worked on 25+ projects
- ✓ Working as a Senior .NET Developer at Digitall
- ✓ Assistant at Technical University of Sofia
- ✓ Ph.D. candidate in Artificial Intelligence
- ✓ https://www.linkedin.com/in/evgeni-dyulgerov/

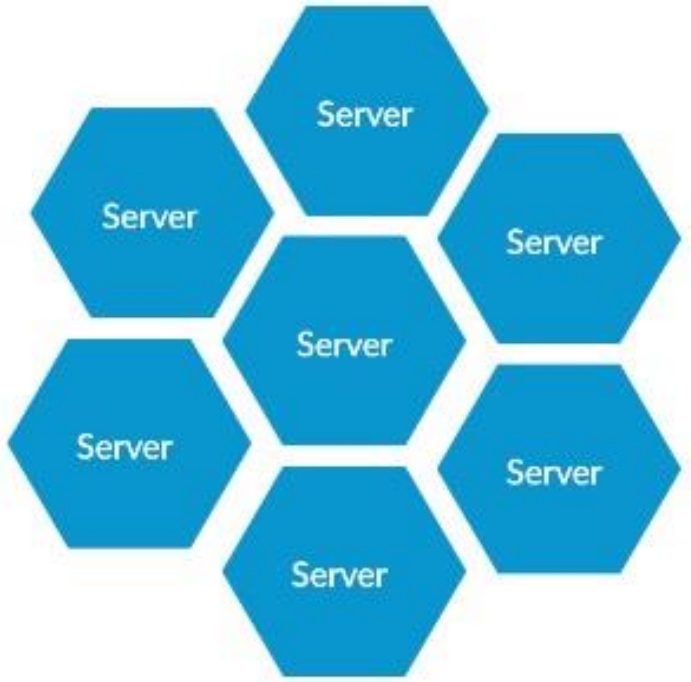
Content

- Infrastructure design patterns
- Platform services design patterns
- Containerization design patterns
- Big data design patterns
- Machine Learning design patterns
- Load balancing design patterns
- Additional resources
- Additional design patterns

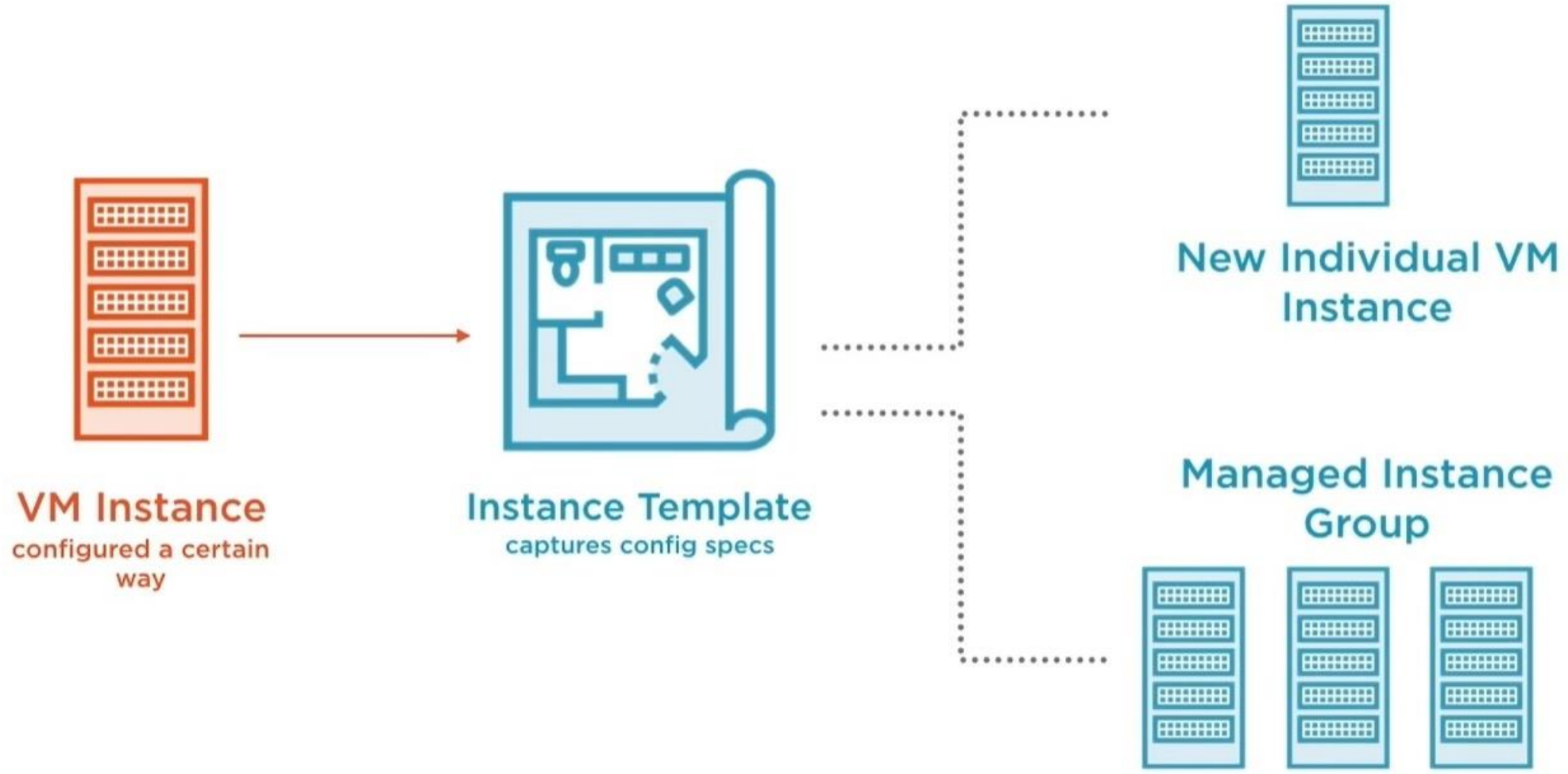
Software University

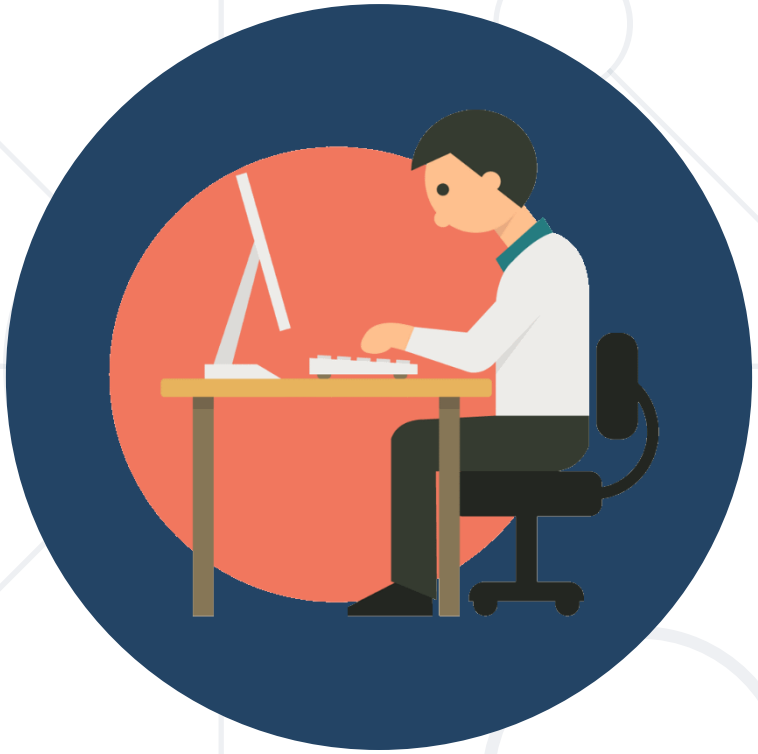
Managed Instance Group

Group of identical GCE VM instances, created from the same instance template that are managed by the platform



Infrastructure design patterns – instance template

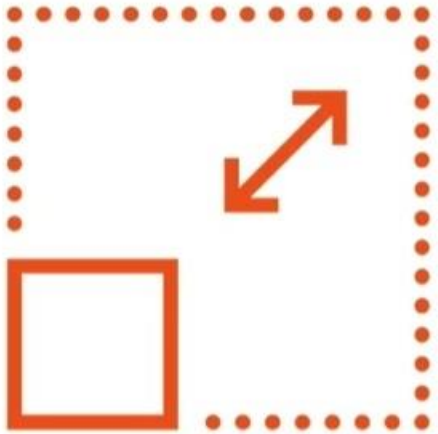




Demo

Instance template

Infrastructure design patterns – managed instance group



Autoscaling

Associate autoscaling policy with MIG

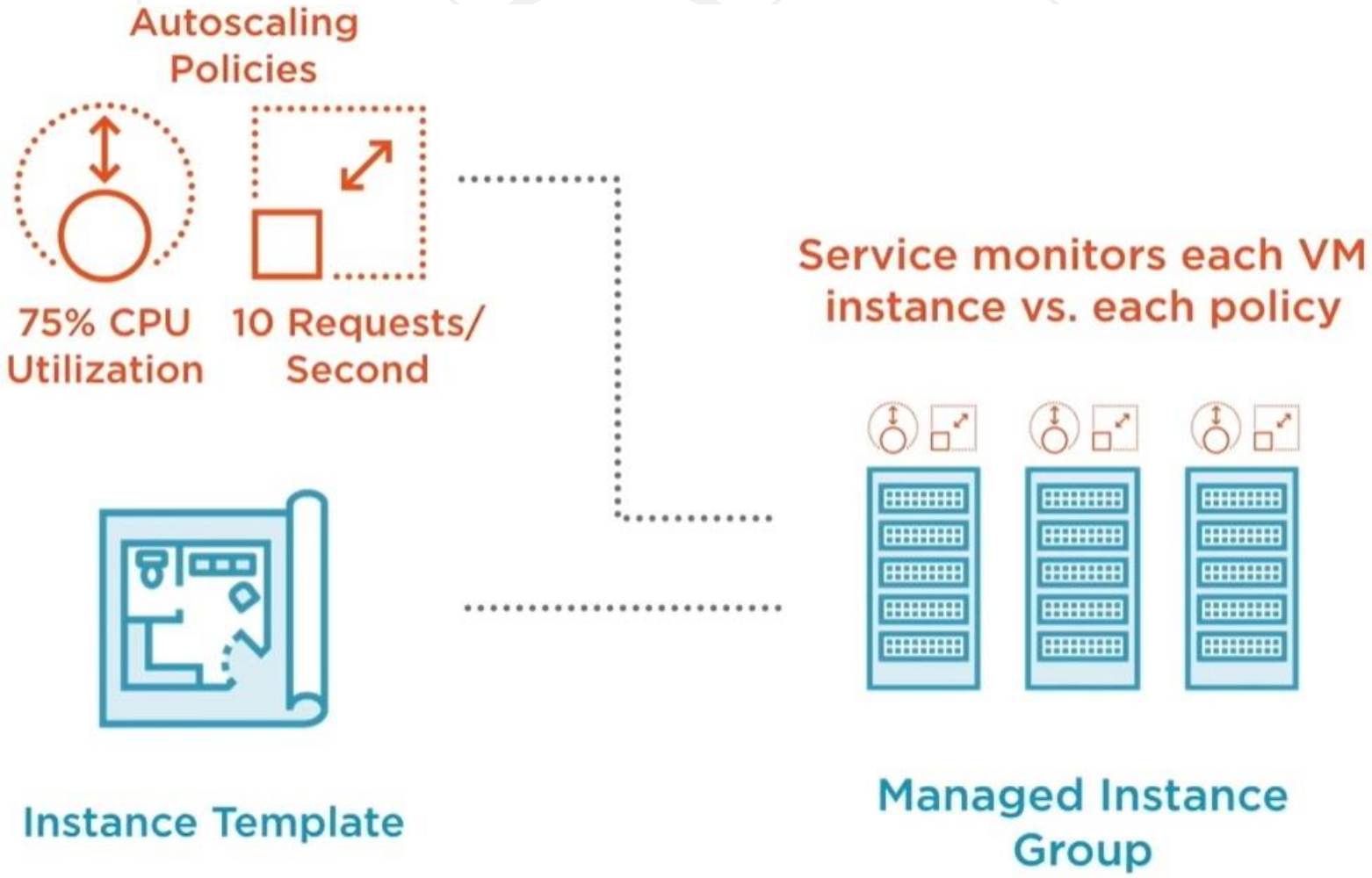
Autohealing

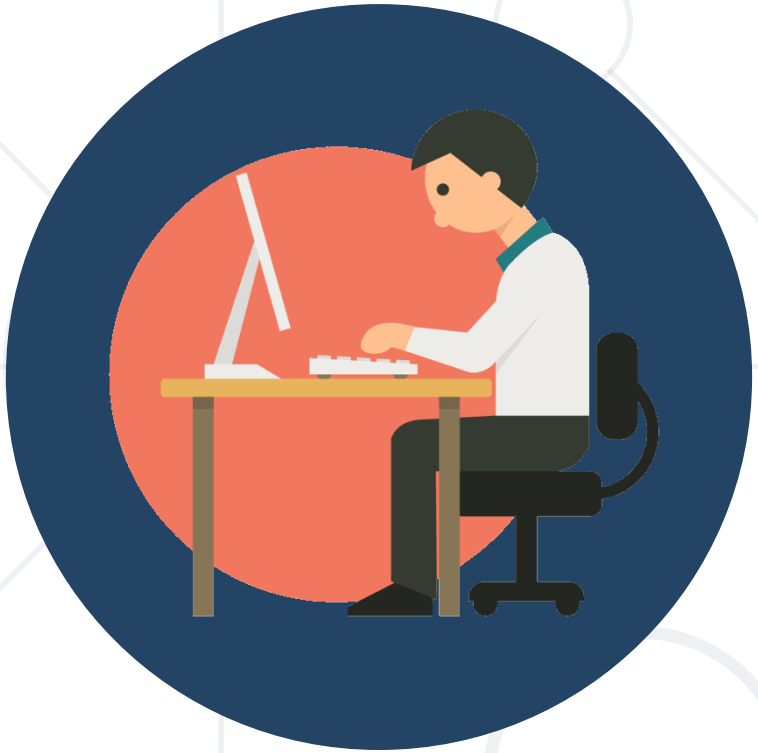
Associate health check and autohealing policy with MIG

Demo

Managed instance group

Infrastructure design patterns – auto-scaling policies





Demo

Auto-scaling policies

Software University

Cloud Functions

Event-driven serverless compute platform

Platform services design patterns – event driven compute

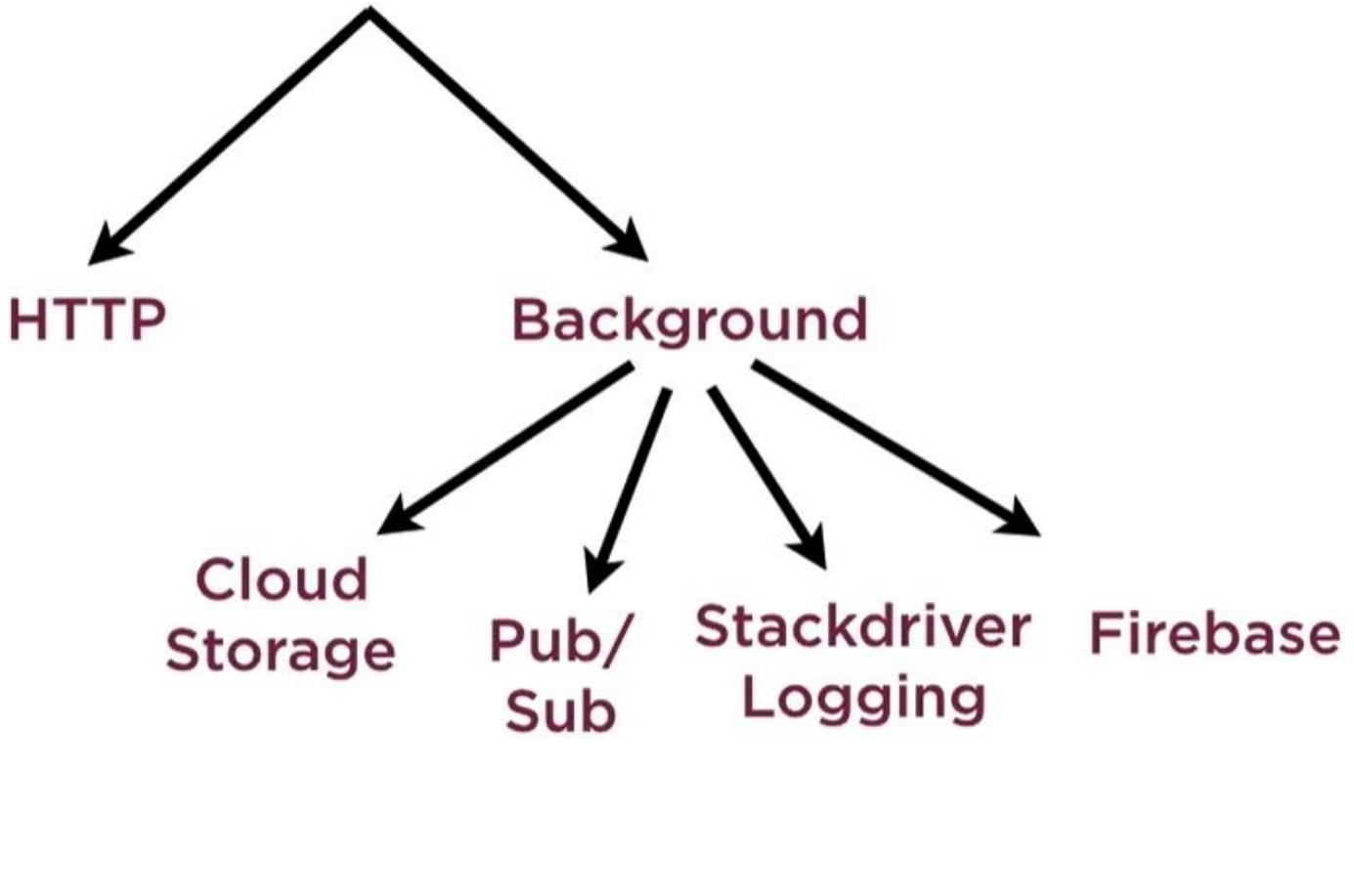
Platform services design patterns – events

Occurs in the external environment

Functions can choose to respond to an event

Events are wired up to trigger functions

Platform services design patterns – event types



Platform services design patterns – function specifics

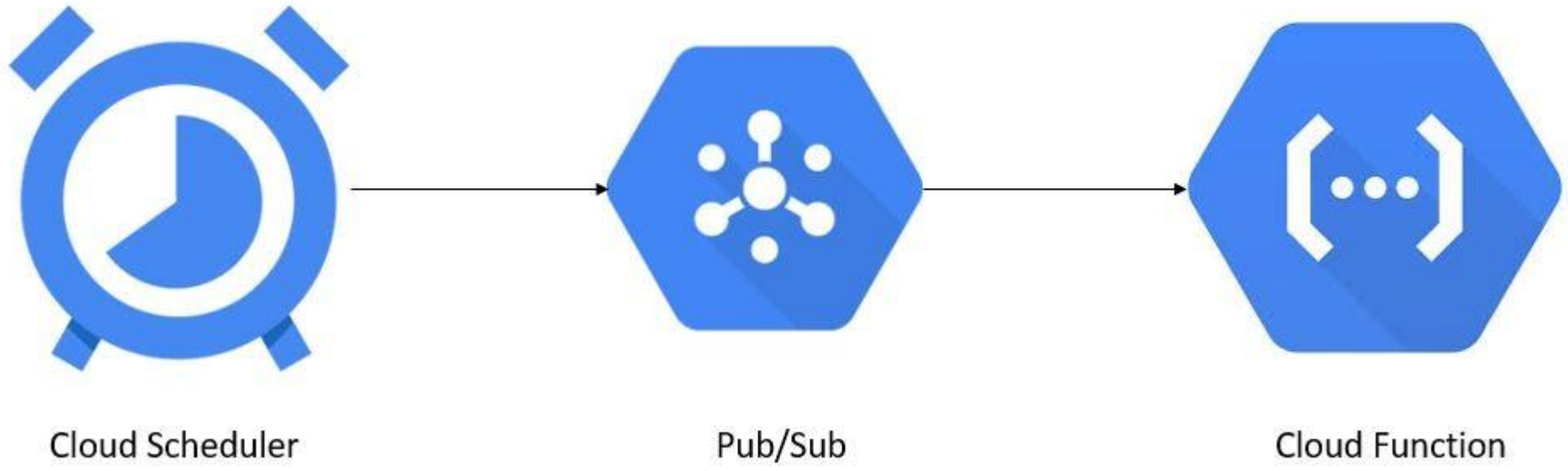
Multiple function instances based on current load

Functions do not share memory or variables

An instance processes a single request

Functions should be stateless

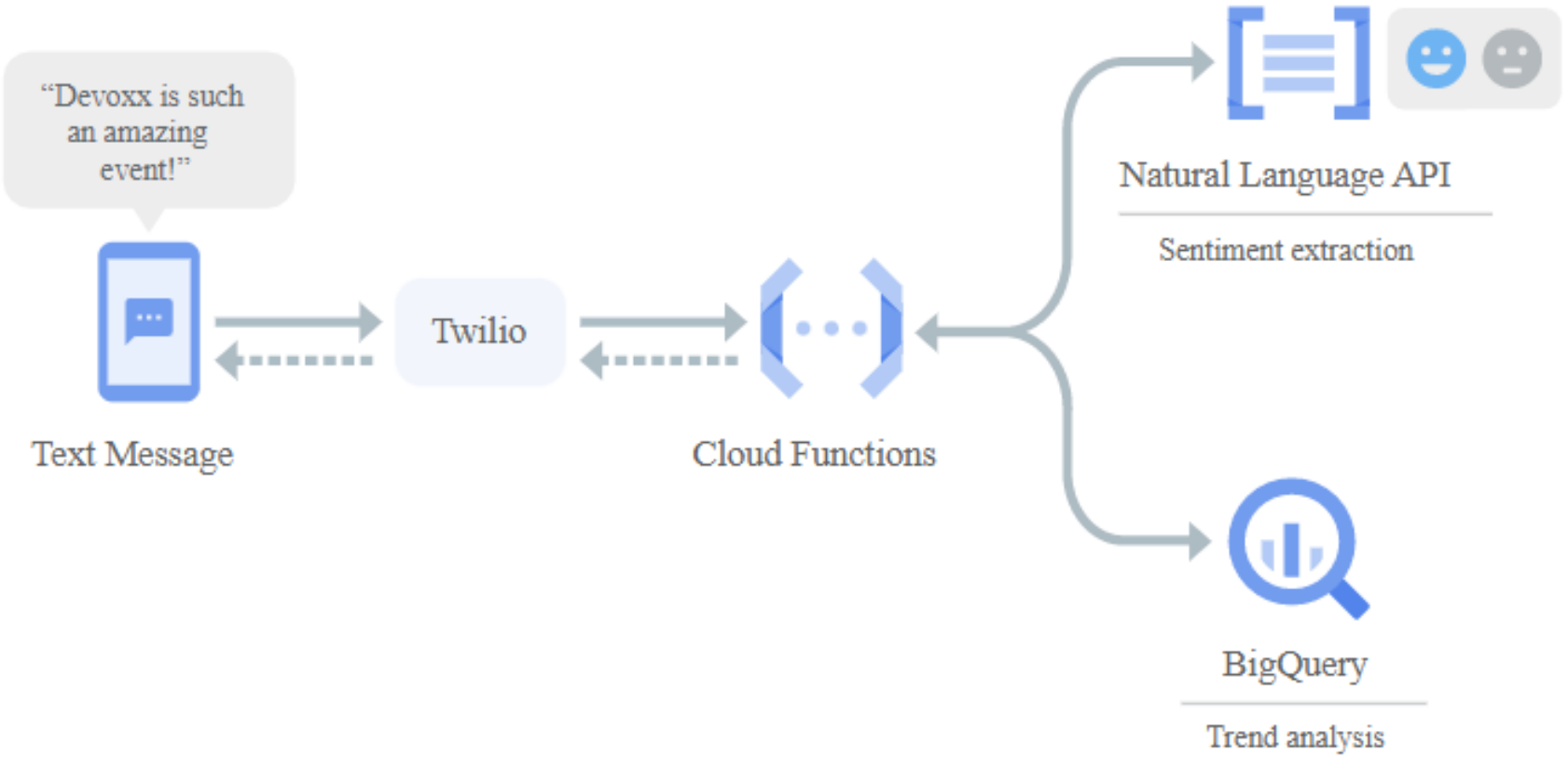
Platform services design patterns – example



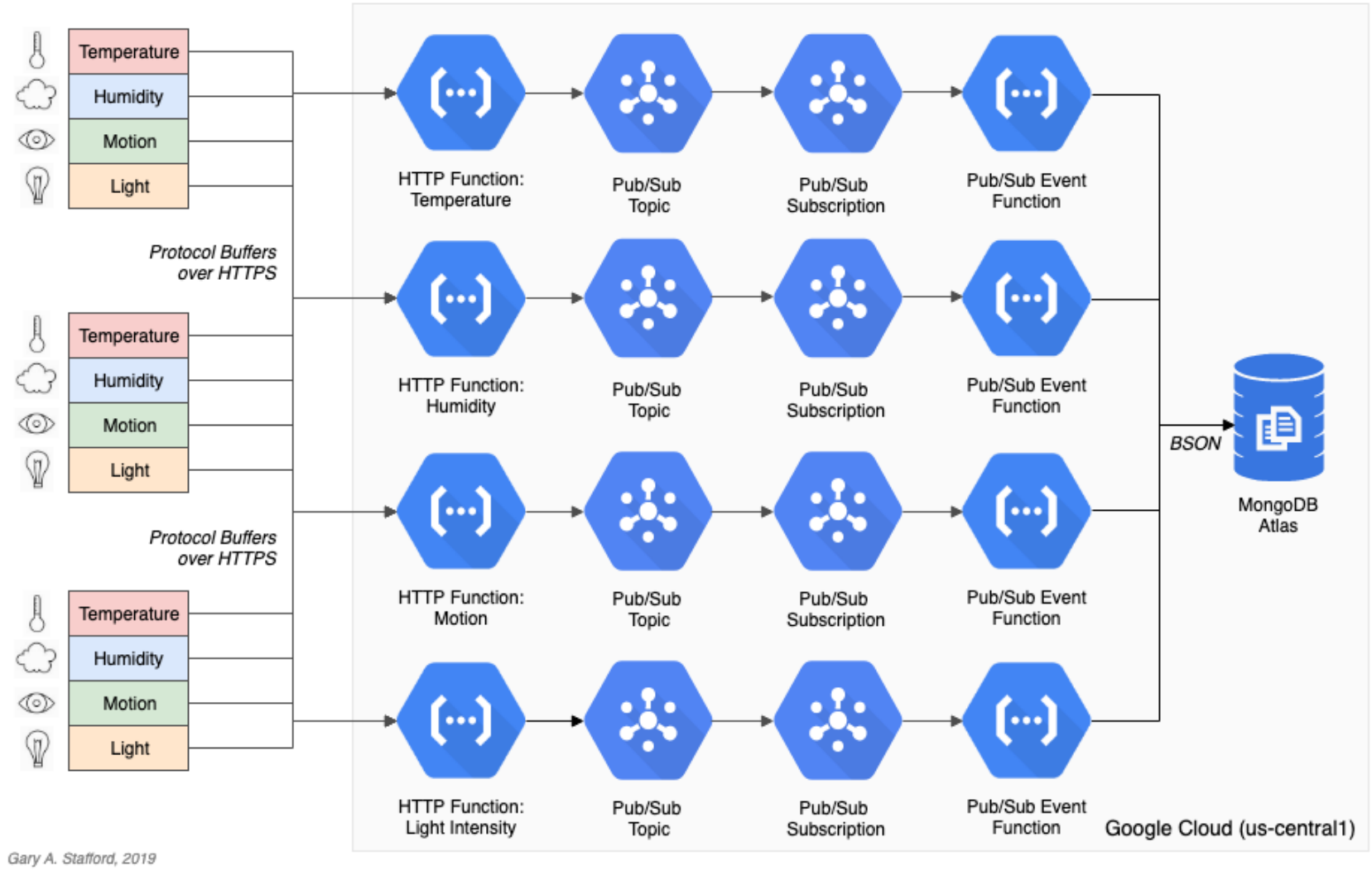
Demo

Scheduled cloud functions

Platform services design patterns – other examples (1)

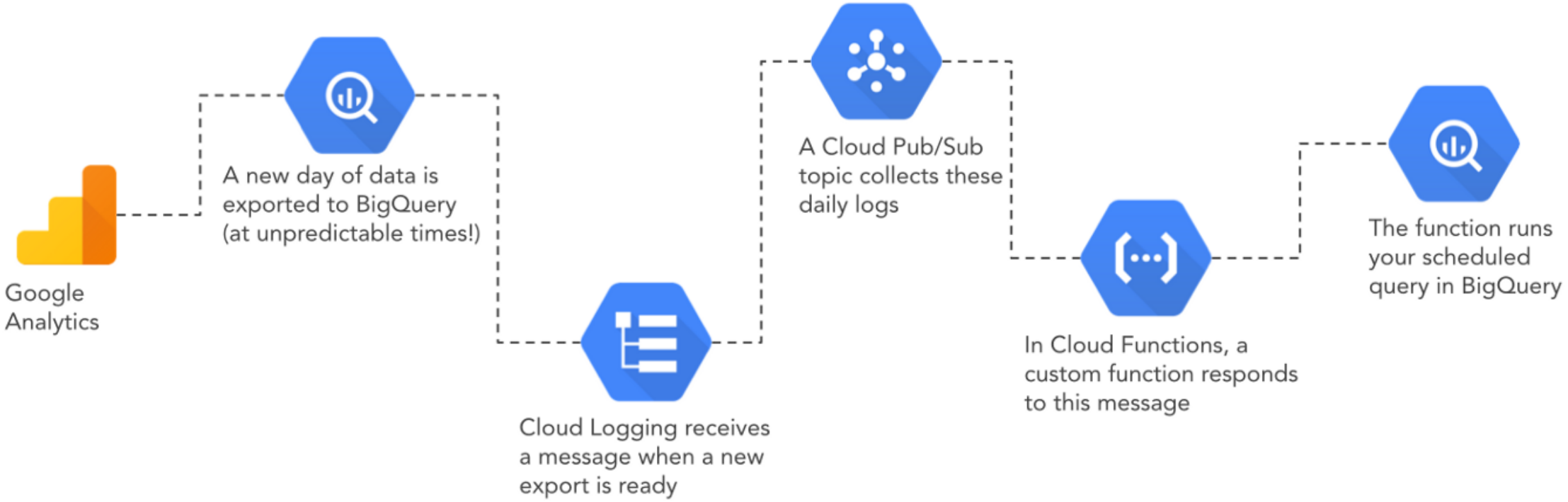


Platform services design patterns – other examples (2)



Gary A. Stafford, 2019

Platform services design patterns – other examples (3)

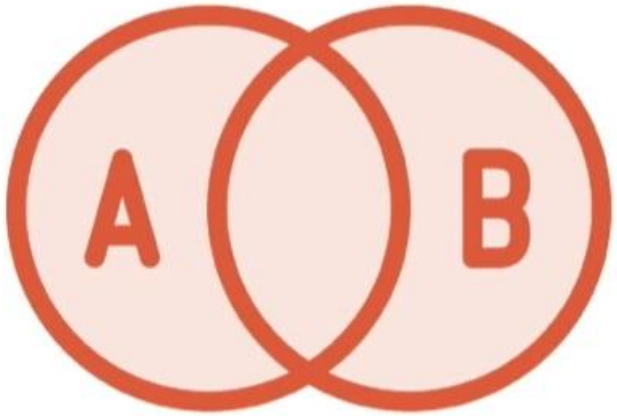


Modern data architecture

Allows you to process real-time streaming events. There are two primary approaches:

- ✓ Lambda Architecture - has two different components: batch processing and stream processing
- ✓ Kappa Architecture - where all data in your environment is treated as a stream

Big data design patterns – Lambda and Kappa overview



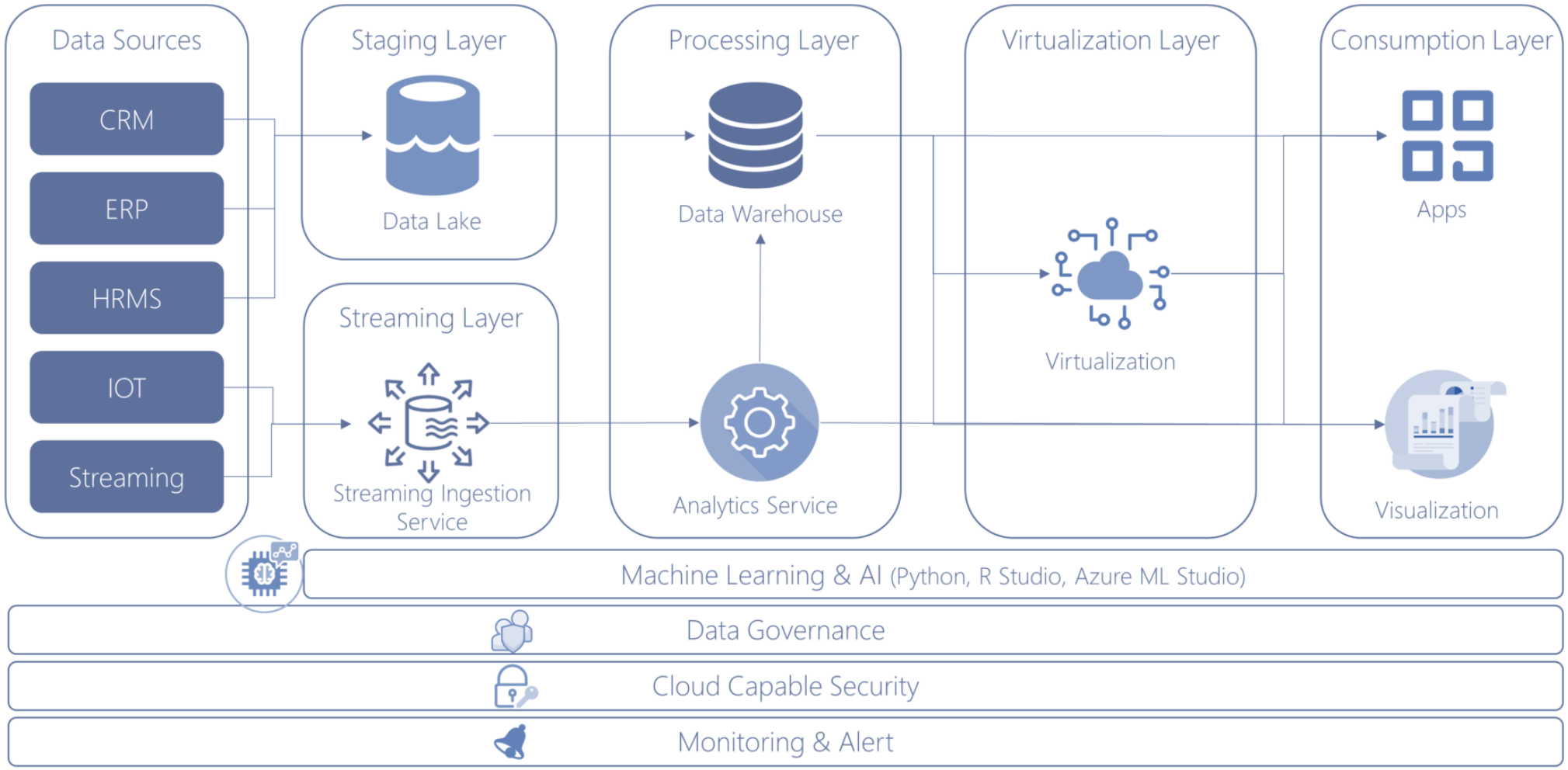
Lambda and Kappa architectures both combine batch and stream data

They do so in different ways

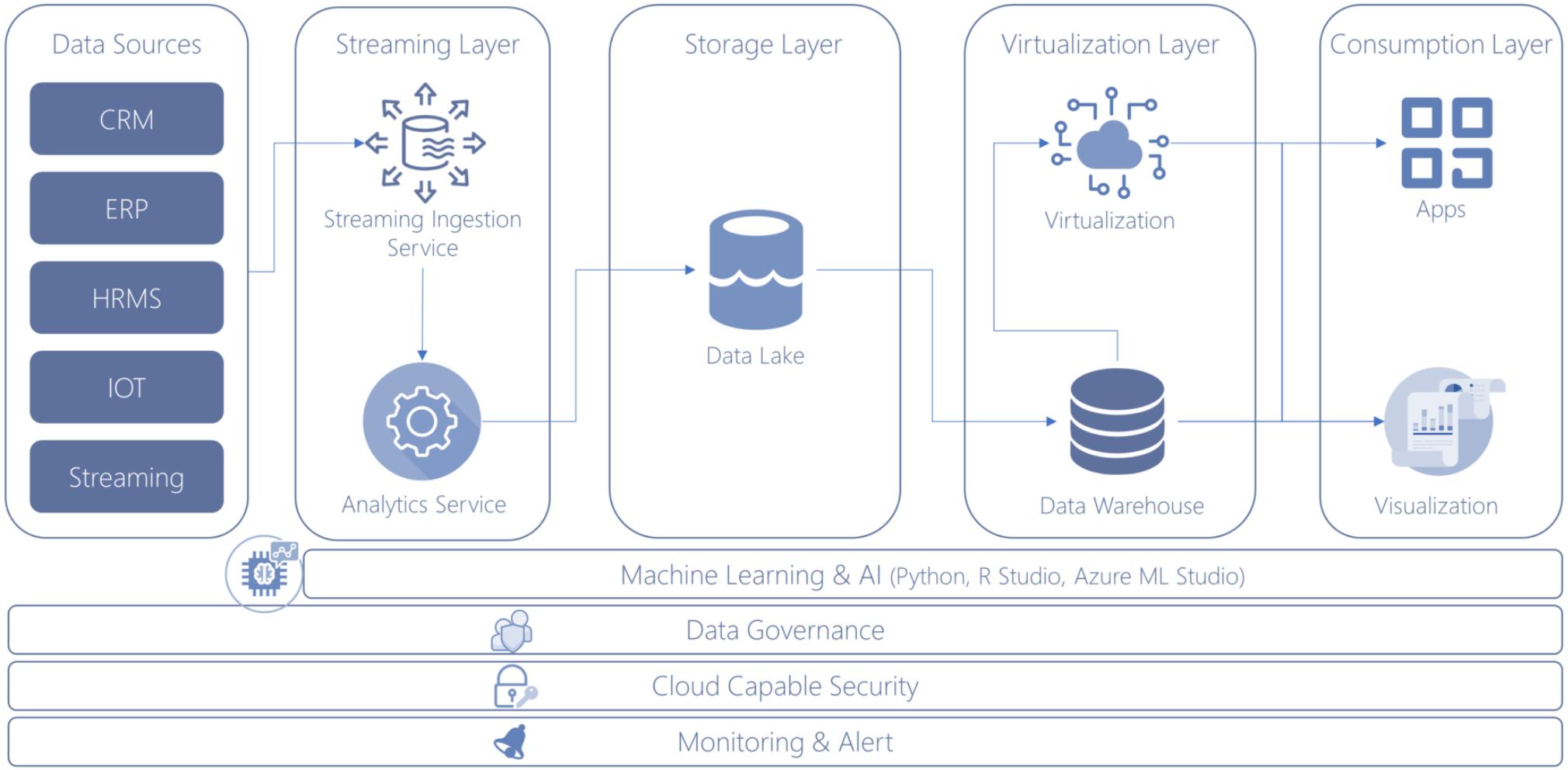
Lambda couples them less tightly

But is more robust

Big data design patterns – Lambda architecture



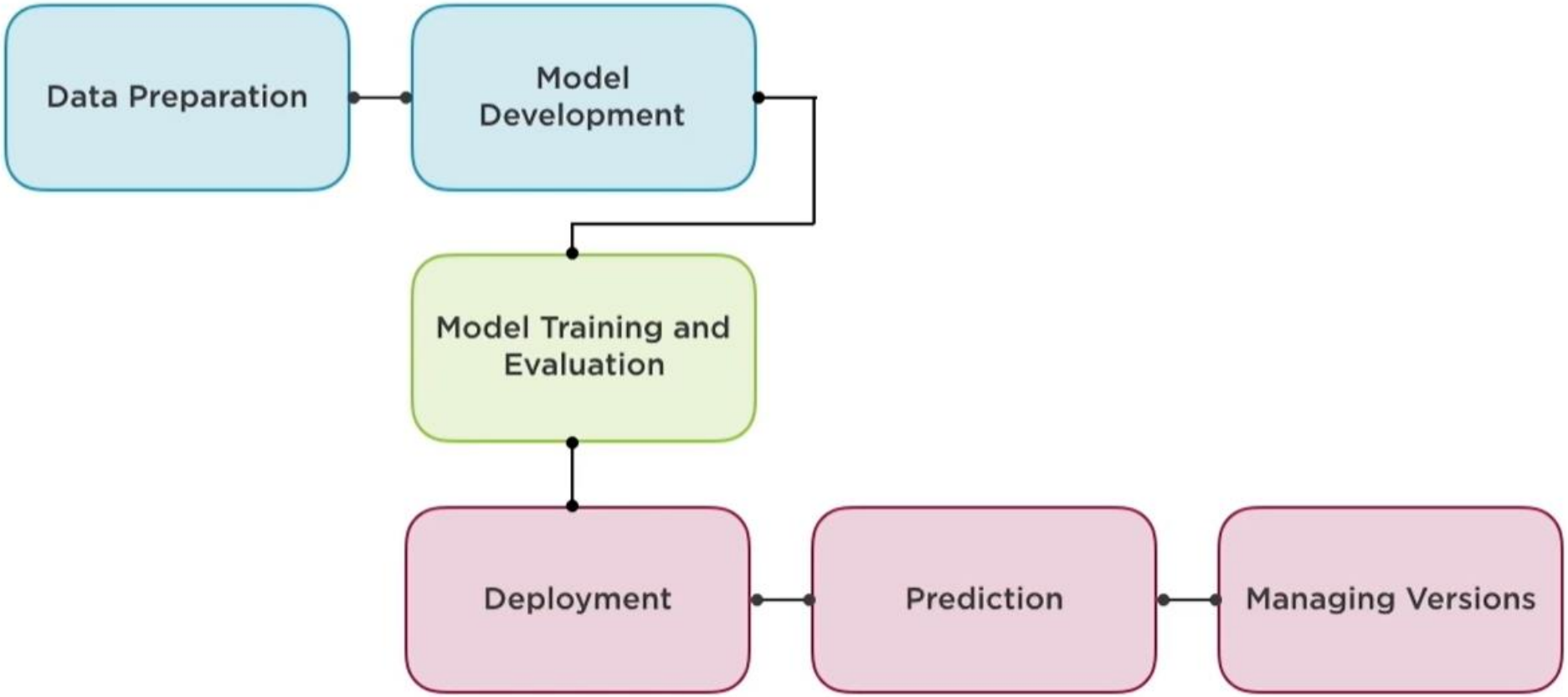
Big data design patterns – Kappa architecture



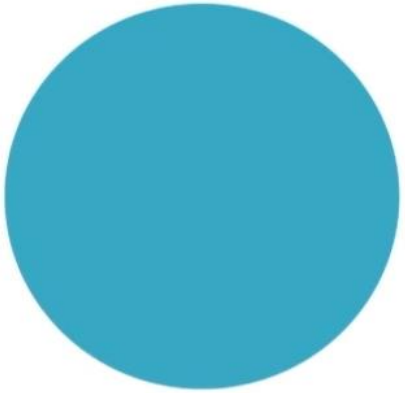
Machine learning architecture

A type of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so

Machine Learning design patterns – process

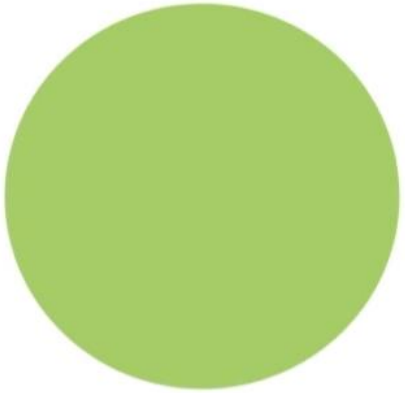


Machine Learning design patterns – tools



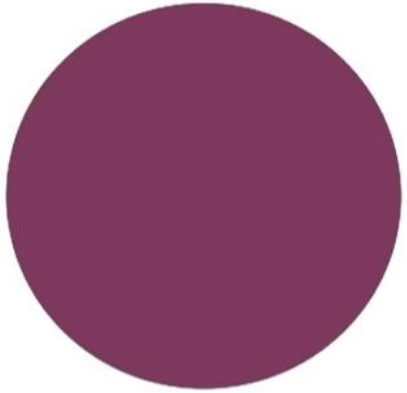
Google Cloud BigQuery

SQL Data Warehouse - powerful analytical tool



Google Cloud AI

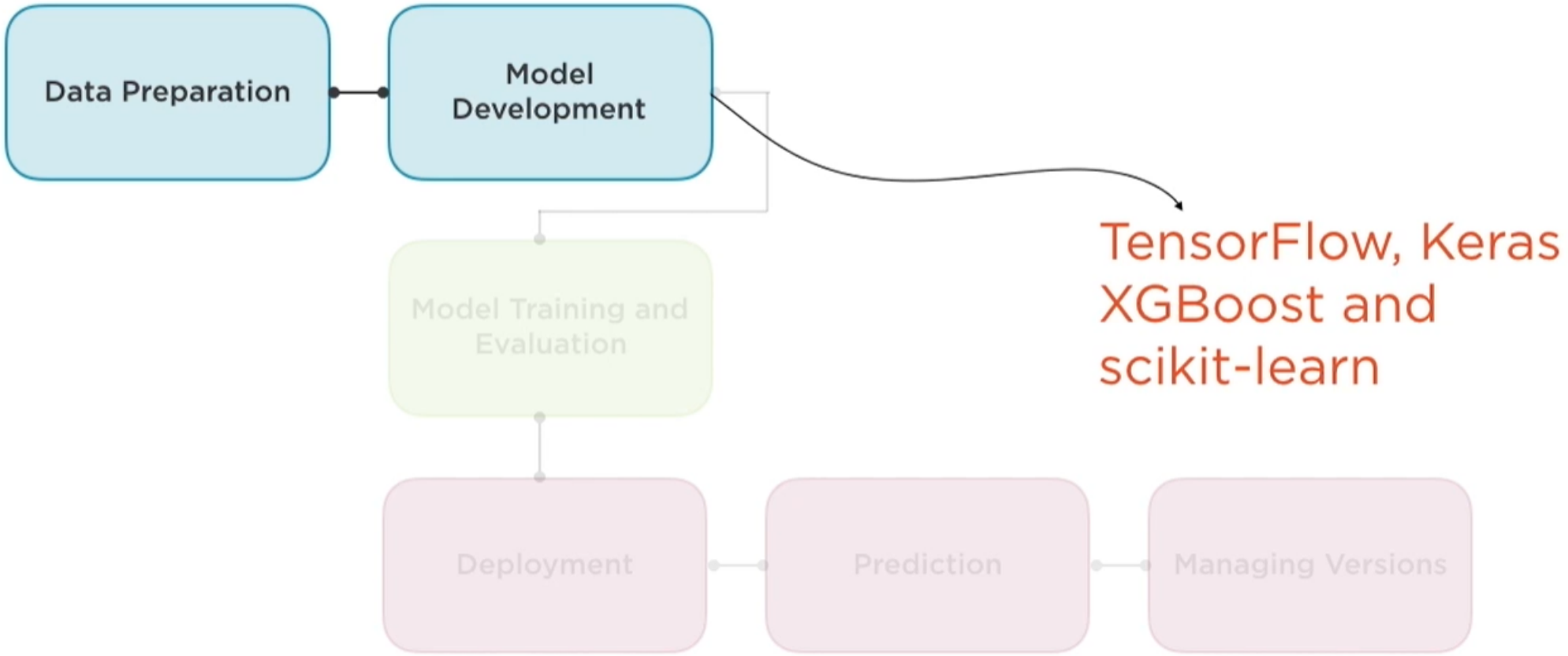
Suite of powerful ML and AI services developed at Google



Google Cloud BigQuery ML

Build ML models using SQL without leaving BigQuery

Machine Learning design patterns – using frameworks



Machine Learning design patterns – frameworks

TensorFlow

- Deep learning
- Tight coupling with ML Engine
- Relatively complex to use

Keras

- Deep learning
- Easy to use

scikit-learn

- Traditional ML
- Easy to use

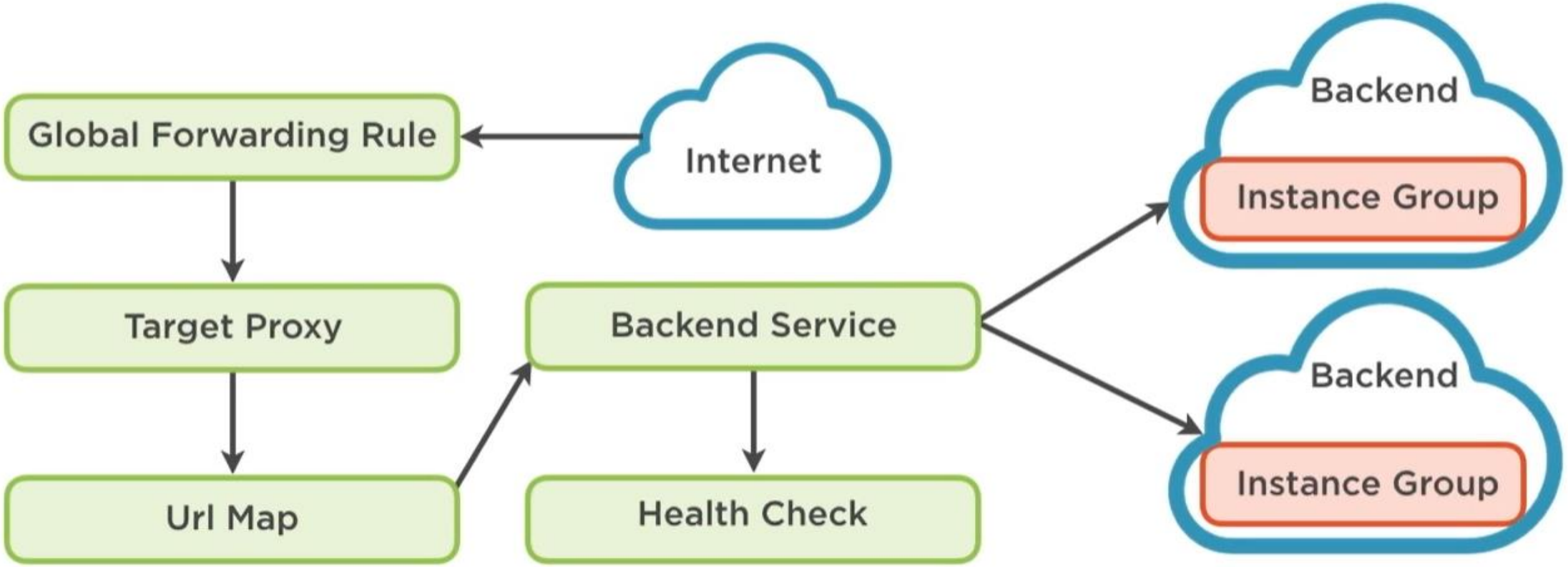
XGBoost

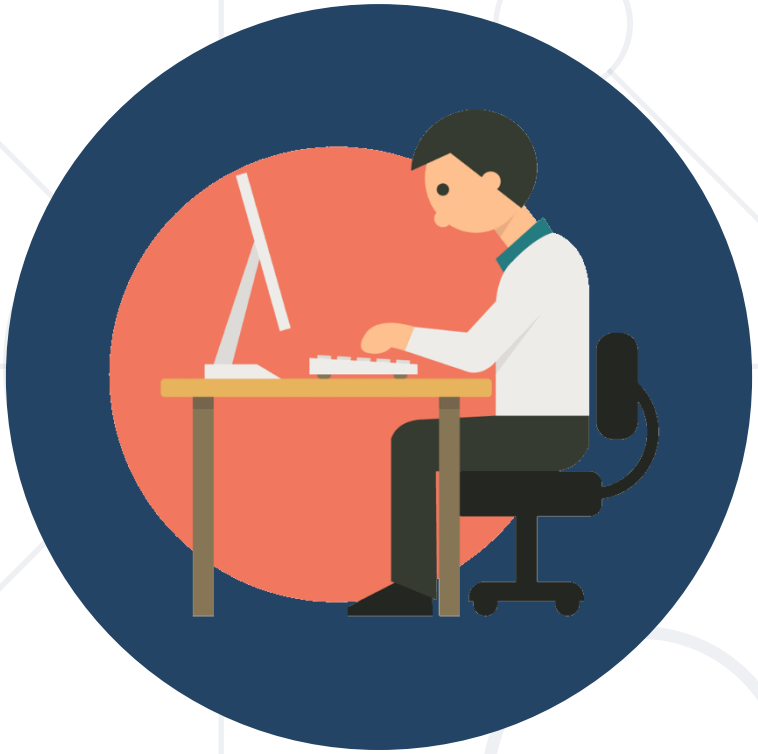
- Fast prototyping and training
- Small datasets with missing data

Cloud load balancing

Efficiently distributing incoming network traffic across a group of backend servers.

Load balancing design patterns – process





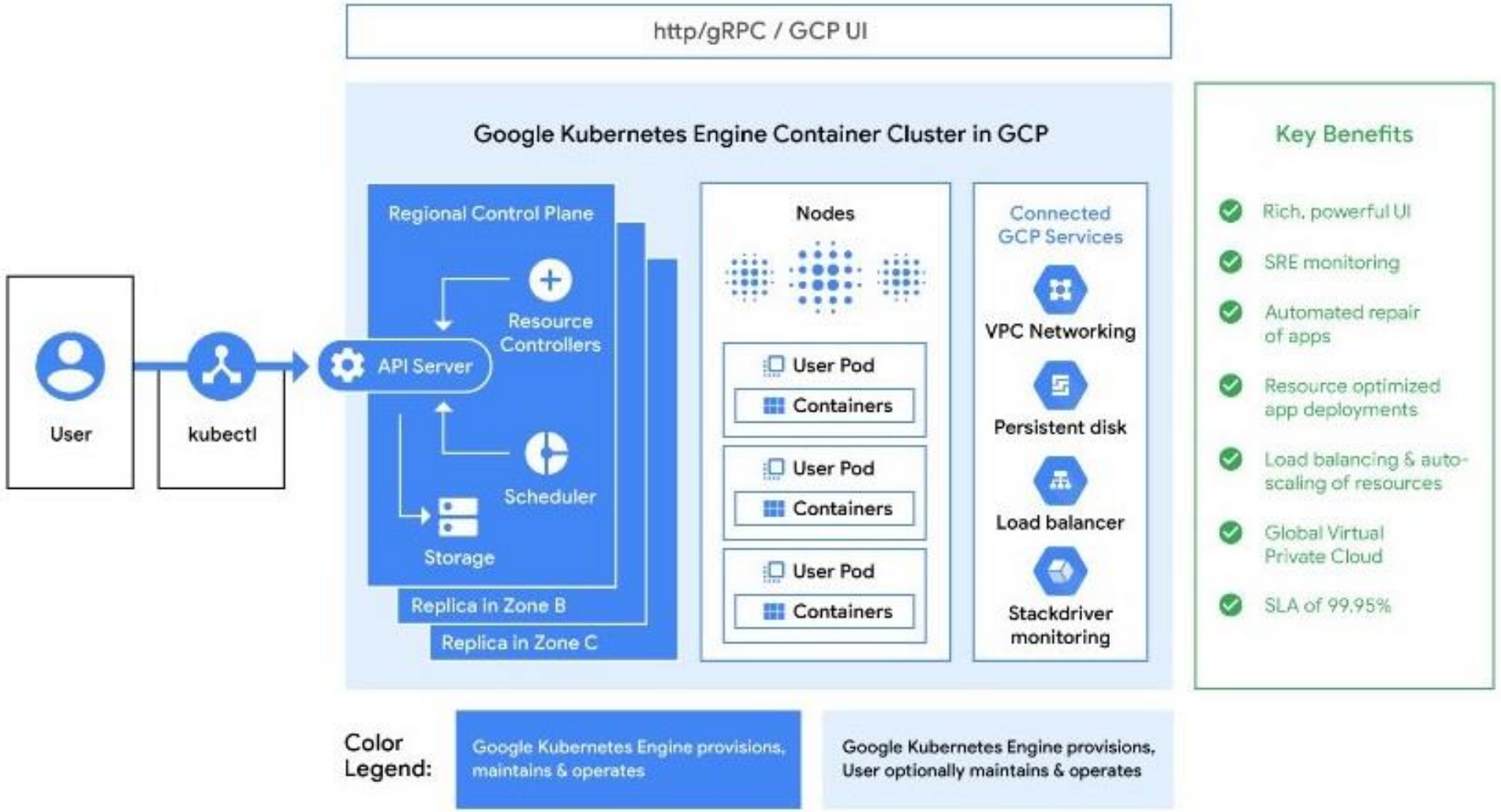
Demo

Load balancing

Containerization

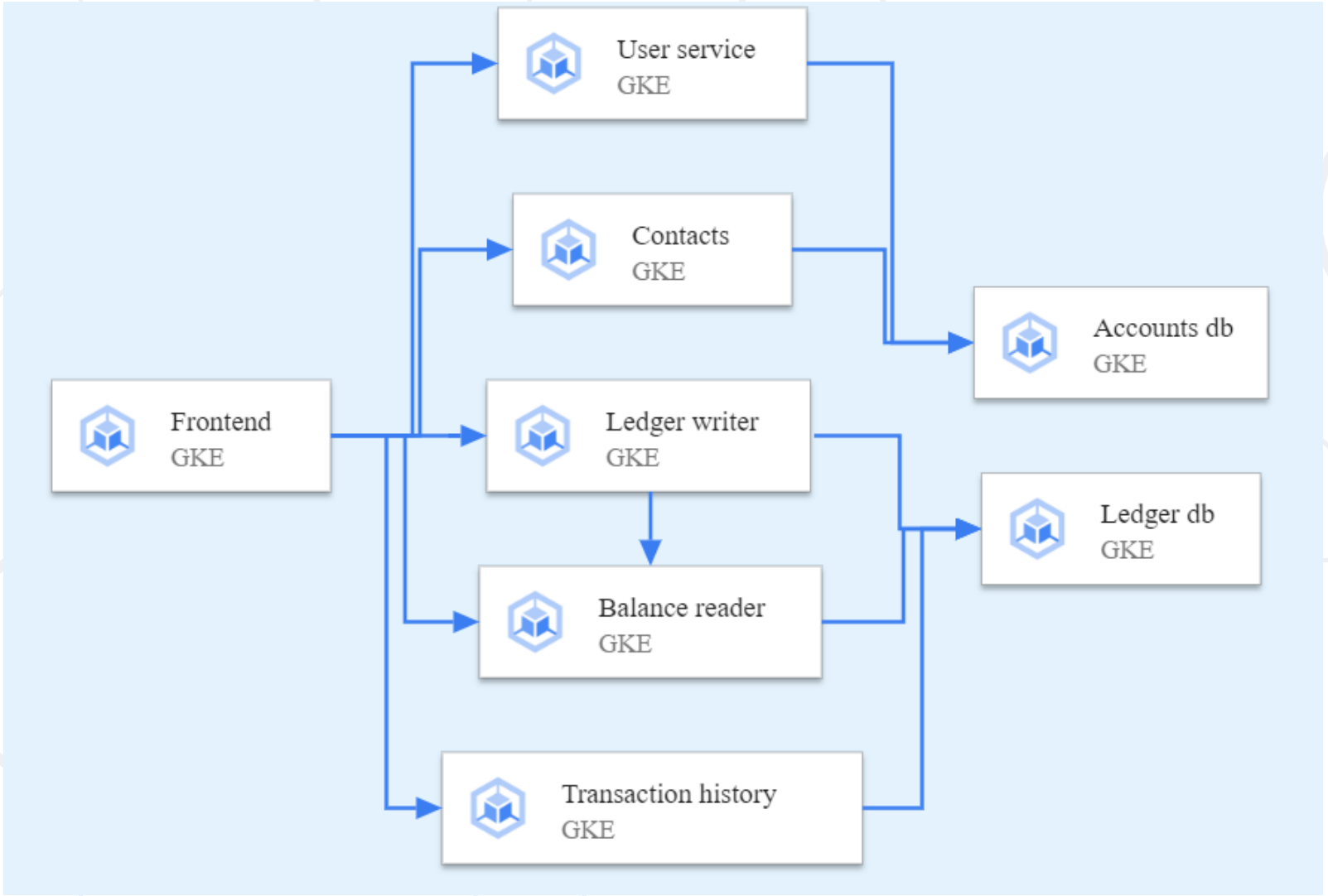
The packaging together of software code with all it's necessary components like libraries, frameworks, and other dependencies so that they are isolated in their own "container"

Containerization design patterns – GKE specifics



Containerization design patterns – containerized apps

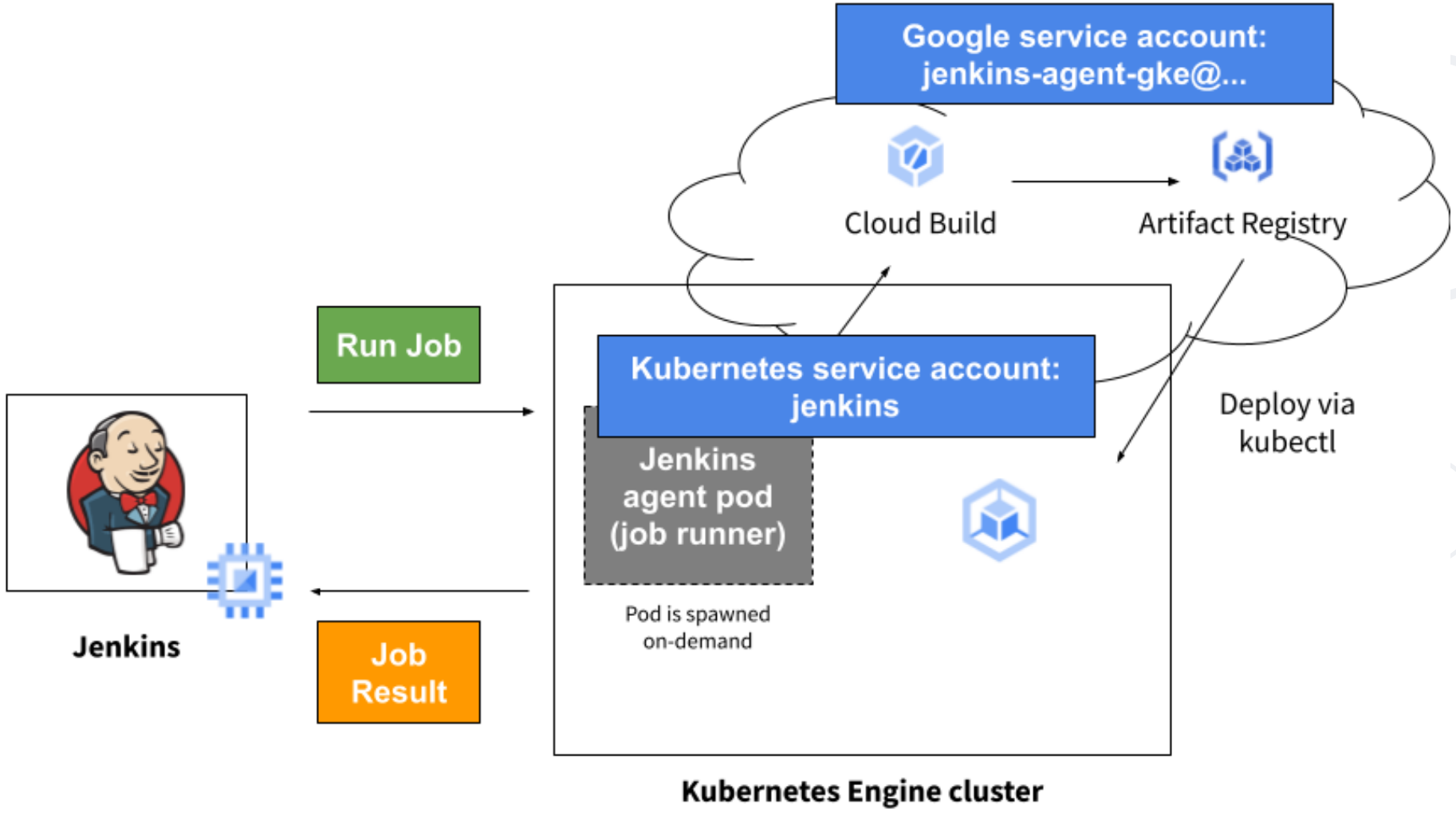
Software University

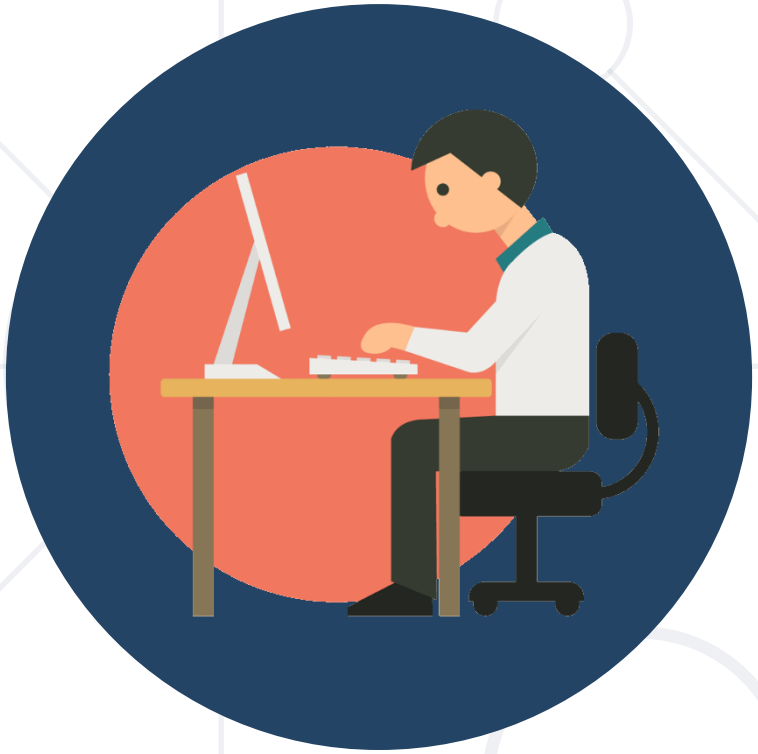


Demo

Modular banking app

Containerization design patterns – CI/CD pipeline





Demo

CI/CD pipeline with Jenkins

Additional resources

- Google Cloud Architecture - https://cloud.google.com/architecture
- Google Cloud Tutorials - https://cloud.google.com/docs/open-tutorials
- Google Cloud (Documentation and Tutorials) - https://cloud.google.com/

Additional design patterns

- AWS Design Patterns - https://en.clouddesignpattern.org/index.php/Main_Page.html
- Azure Design Patterns - https://learn.microsoft.com/en-us/azure/architecture/patterns/

Questions?