Storing and Managing Data with Redis and Apache Kafka on Heroku

PROVISION APACHE KAFKA ON HEROKU

Saravanan Dhandapani
SOFTWARE ARCHITECT

@dsharu

Course Structure

Part 1

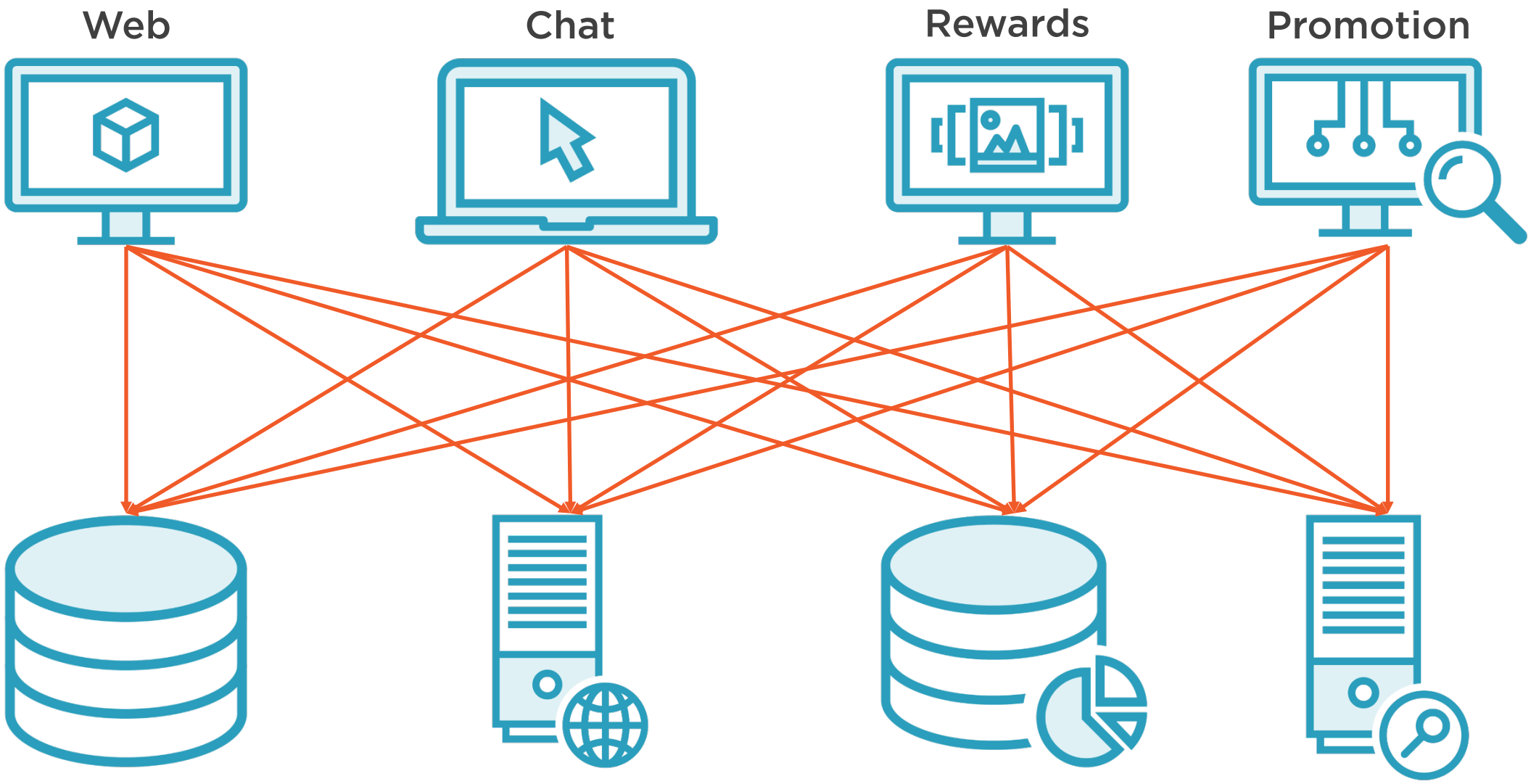
Apache Kafka on Heroku

Part2

Redis on Heroku

Provision Apache Kafka on Heroku

Globomantics E-Commerce Portal



LinkedIn Issue

LinkedIn Issue

1 trillion messages per day

Peak volume of 10 million messages per second

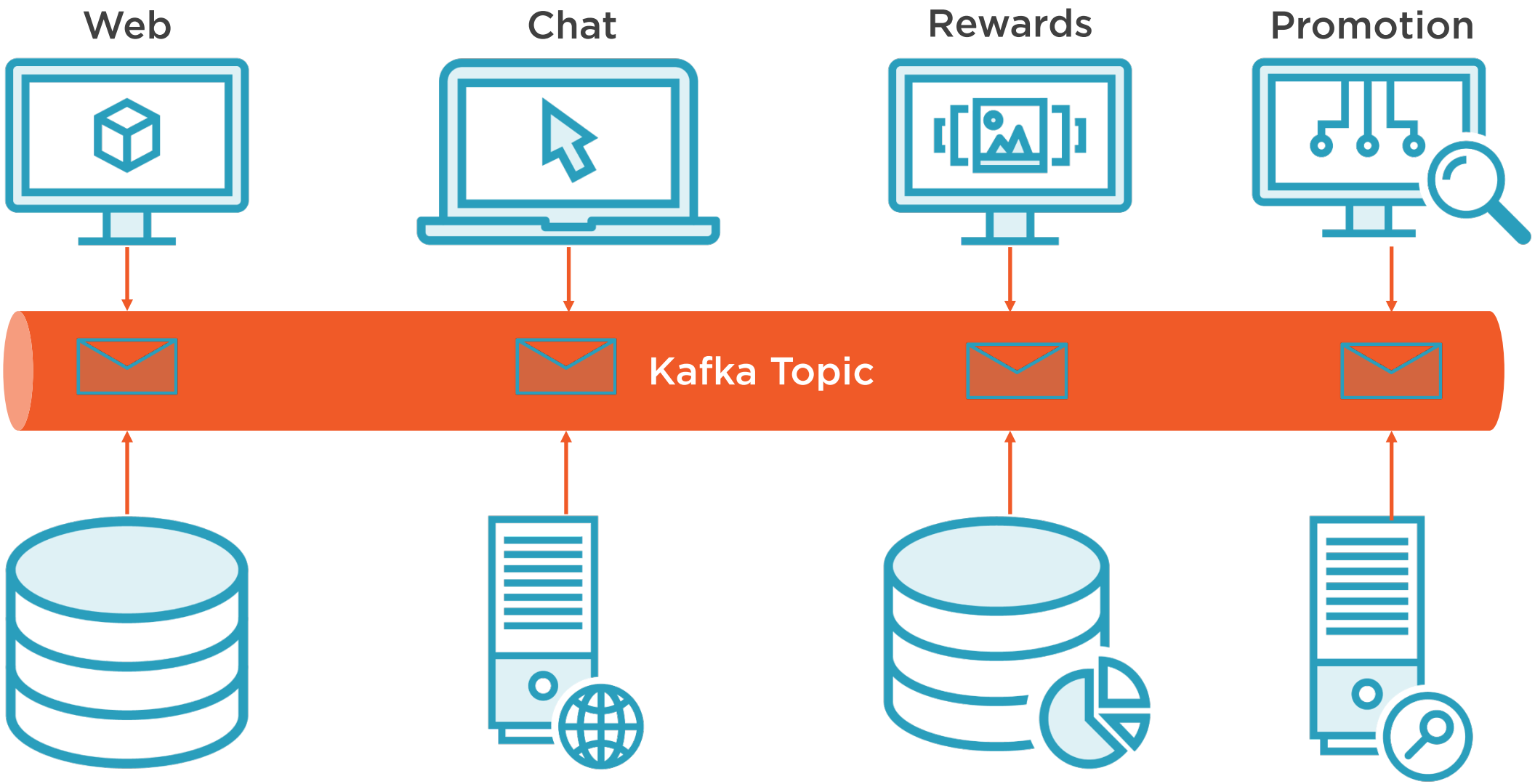
More than 400 millions users

Independently created pipelines routing this data

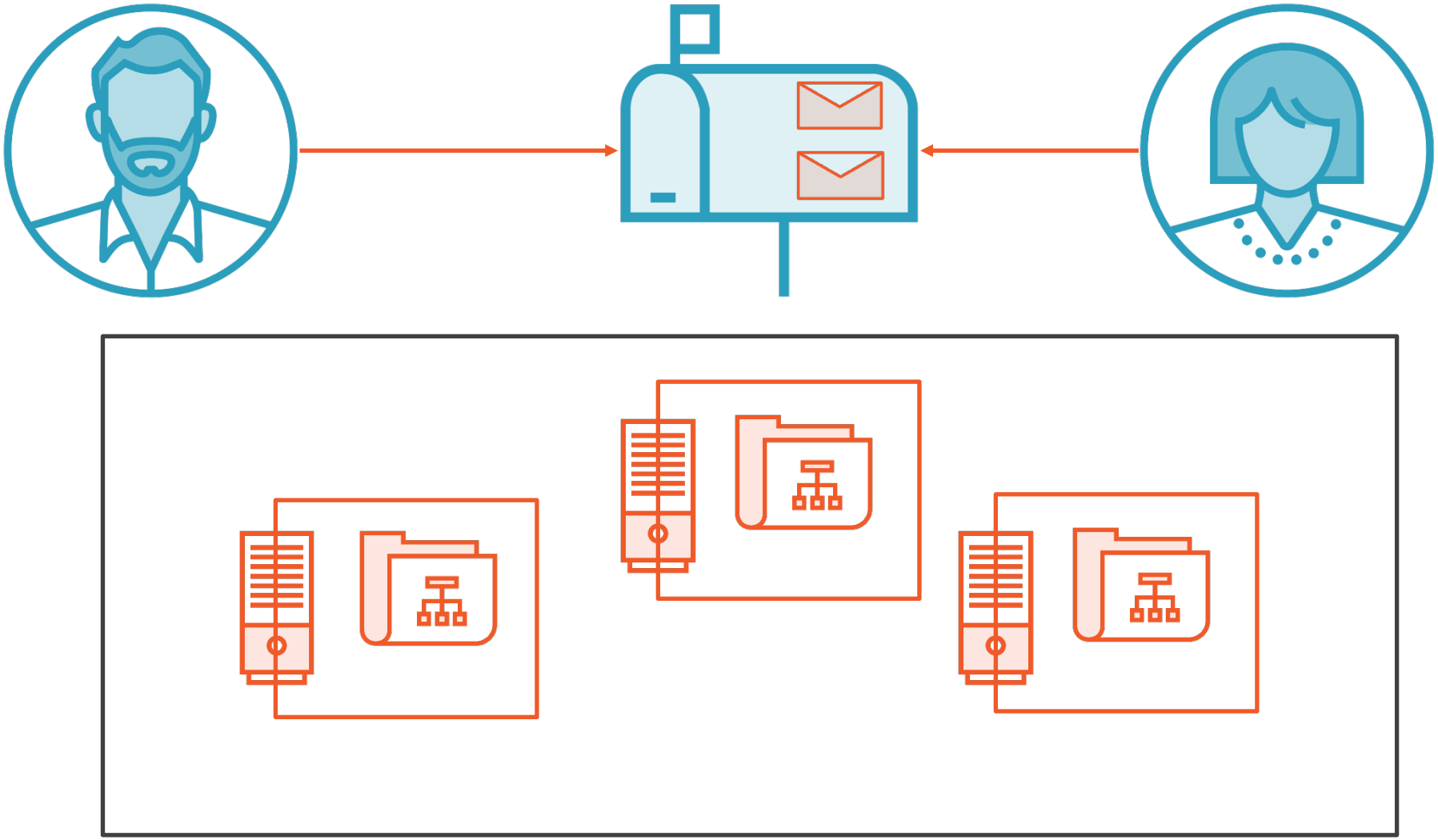
Kafka

- Open sourced in 2012

Globomantics E-Commerce Portal



Postal Analogy



Core Kafka Concepts

Kafka brokers

- Servers running Kafka topics

Kafka cluster

Zookeeper

- Manages the brokers in the cluster

Producers and Consumers

Topics

- Message containers

Partitions

Configuration options

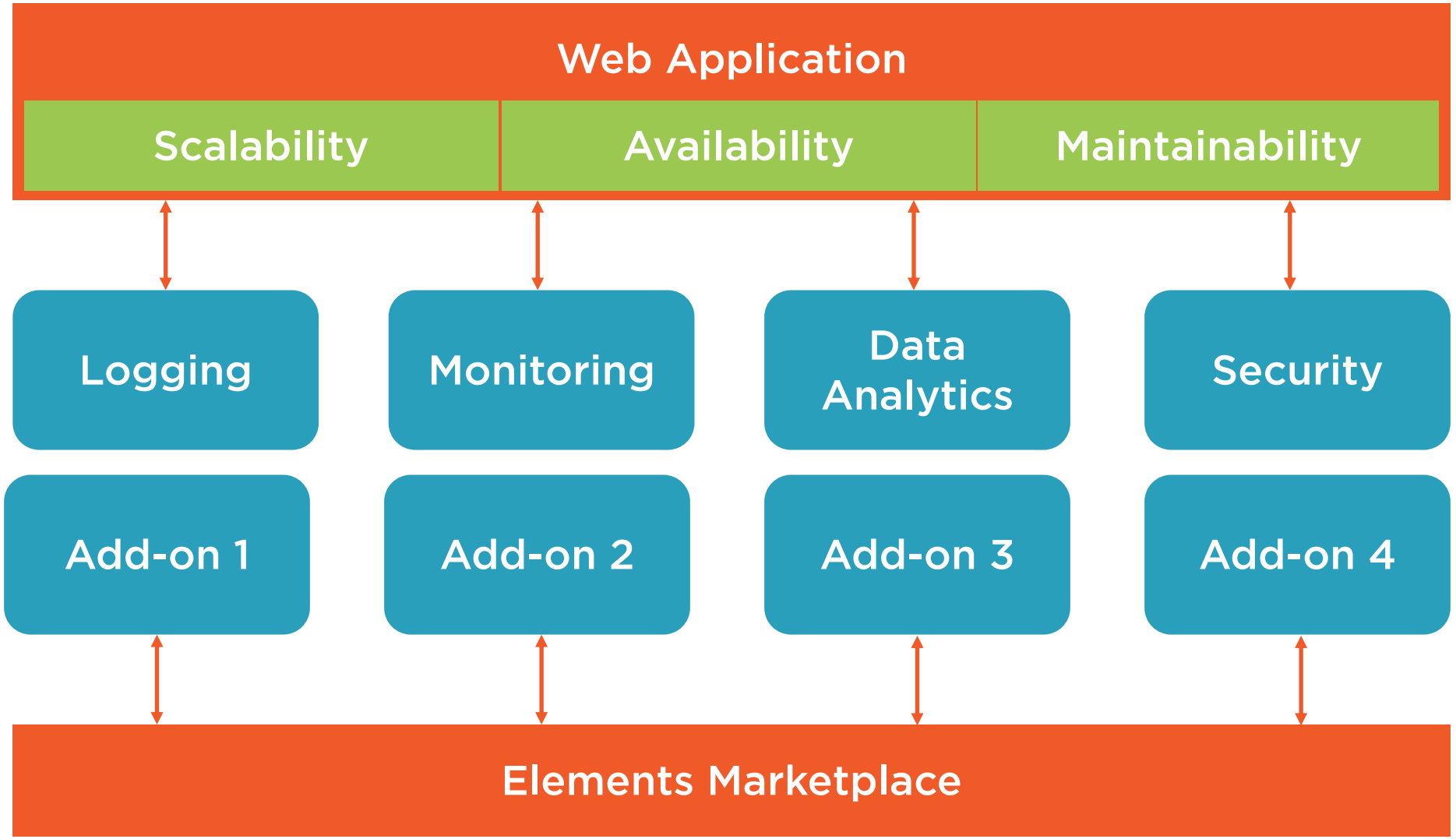
- Message retention
- Replication factor

Why Kafka? Why Heroku?

Kafka Business Cases

Smart devices Process streaming data	Infrastructure and application errors Publish metrics and create dashboards	Replay events and determine root cause	Microservice architecture Service to service communication
Streaming applications	**Monitoring and Alerting**	**Complex error handling**	**Promote loose coupling**

Heroku Architecture



Why Heroku?

Scale based on your needs

Resources are highly available and addressable

Maintain external configuration variables

Rich CLI experience to provision and monitor

Stream PHI and PII data

Provision Apache Kafka Addon

Kafka Basic Plan Comparison

Plan name	Producer quota (KB/s)	Consumer quota (KB/s)	Capacity (GB)	Max partitions	Price/ month
Basic-0	64	256	4	240	$100
Basic-1	512	2048	32	480	$125
Basic-2	4096	16384	64	960	$175

Kafka Basic Plan Use Cases

Limited production usage

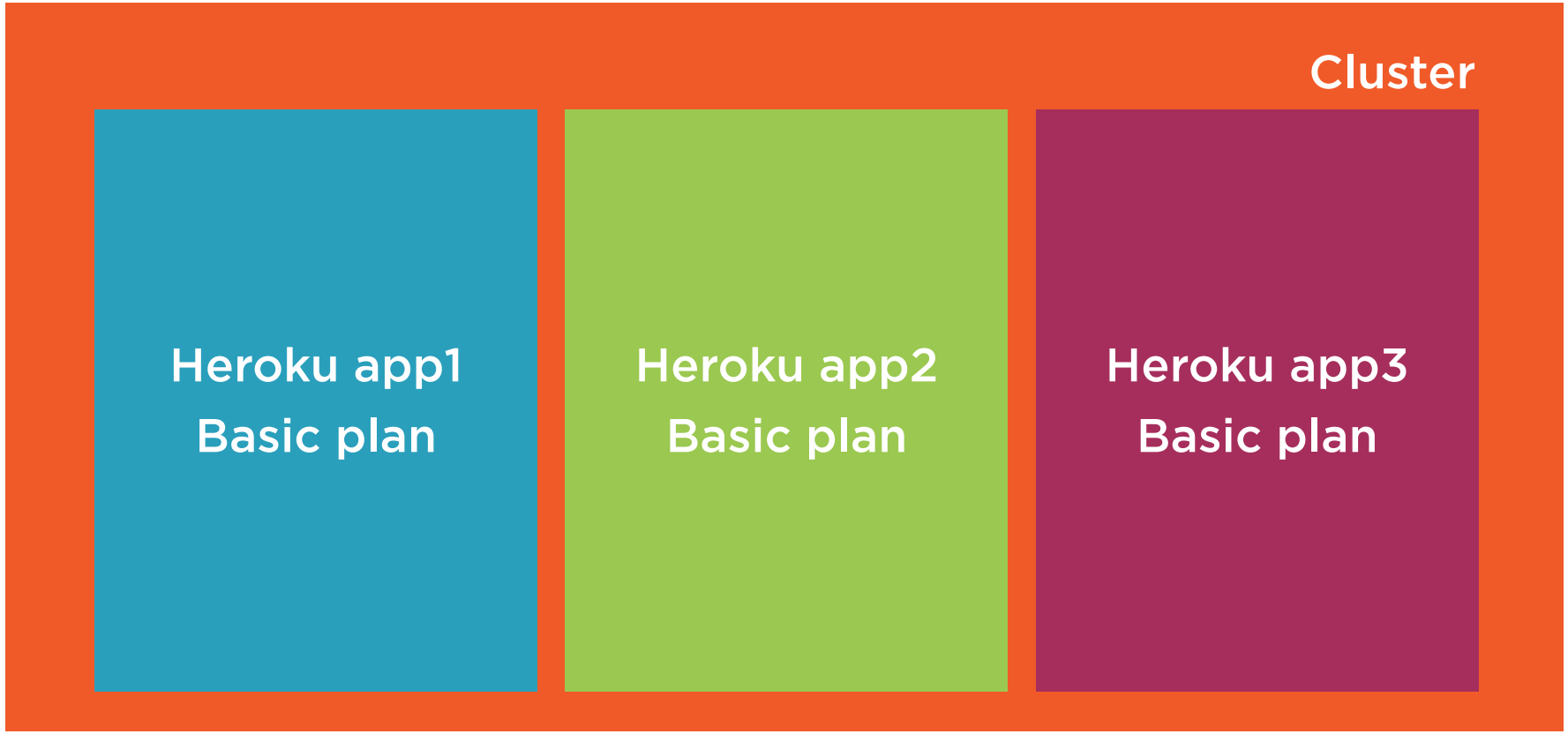
Development and testing environments

Experimentation and prototyping purposes

Multi-Tenant Kafka on Heroku

Multi-Tenancy

Heroku Basic Plan



Dedicated Cluster

- A minimum of 3 Kafka brokers
- 2 weeks of data retention
- Price starting from $1500/month
- Unique prefix not needed

Features unavailable in Basic Plan

Staging environment testing

- heroku kafka:fail

Upgrading environment

- heroku kafka:upgrade

Connect and Manage Kafka on Heroku