Azure Service Fabric

Build always-on, hyper-scalable, microservice-based cloud applications

@ross_p_smith
Technical Evangelist

Microsoft

Services Powered by Service Fabric

SQL Database

2.1 million DBs

Cosmos DB

Billions transactions/day

IoT Hub

Millions of messages

Event Hubs

60bn events/day

30% of Azure cores run Service Fabric

Skype

Cortana

Intune

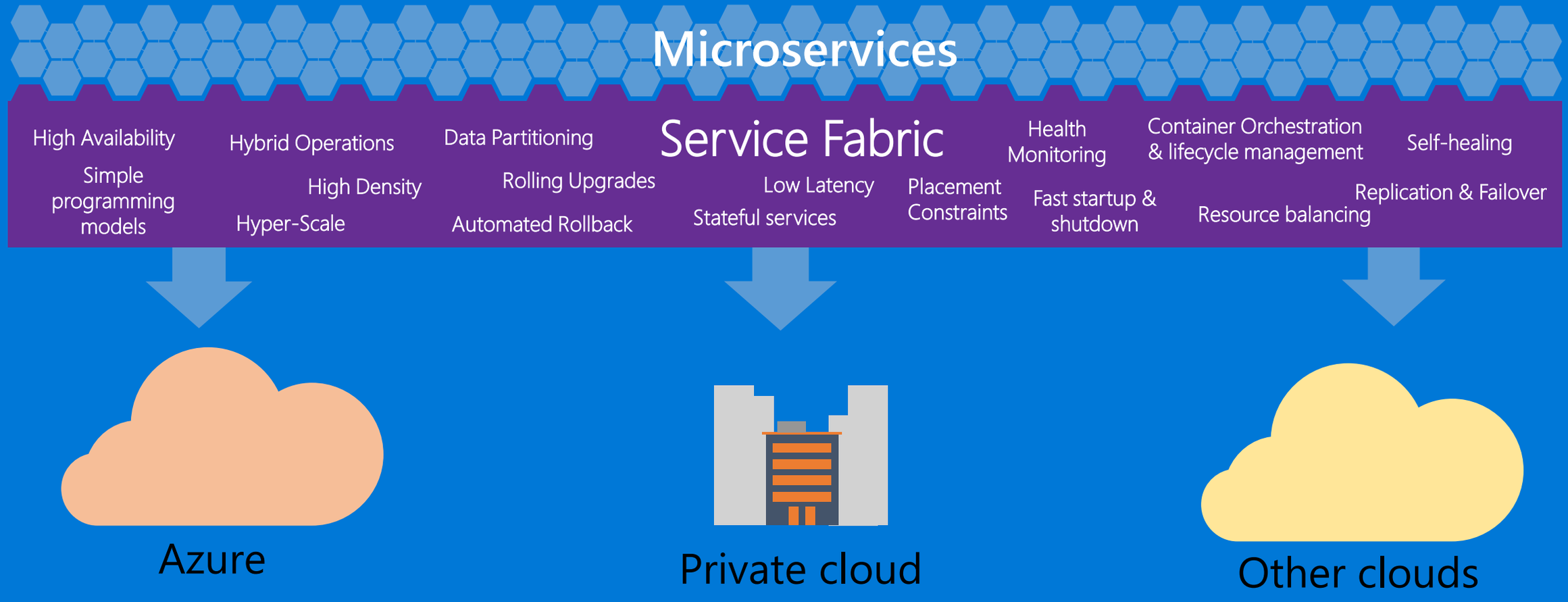
Dynamics

Power BI

Designed for mission critical tier 1 workloads

Microsoft Azure Service Fabric

A platform for reliable, hyperscale, microservice-based applications



To monolith or to Microservice?

5 stages in a continuum…

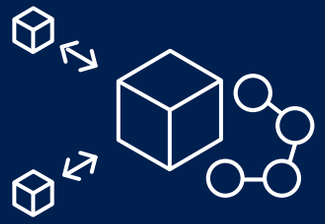
1

Traditional app

2

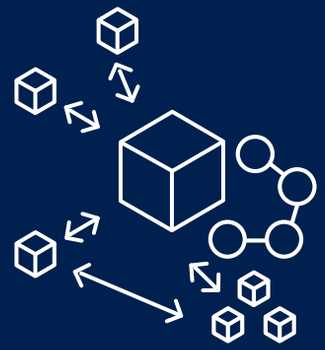
Monolith Hosted as guest executable or container

3



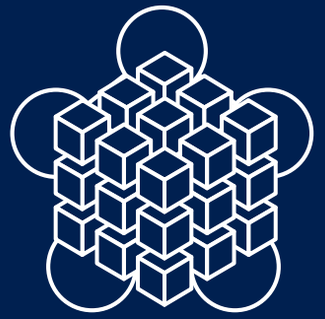
Existing Monolith + new microservices

4



Parts of existing monolith extracted

5

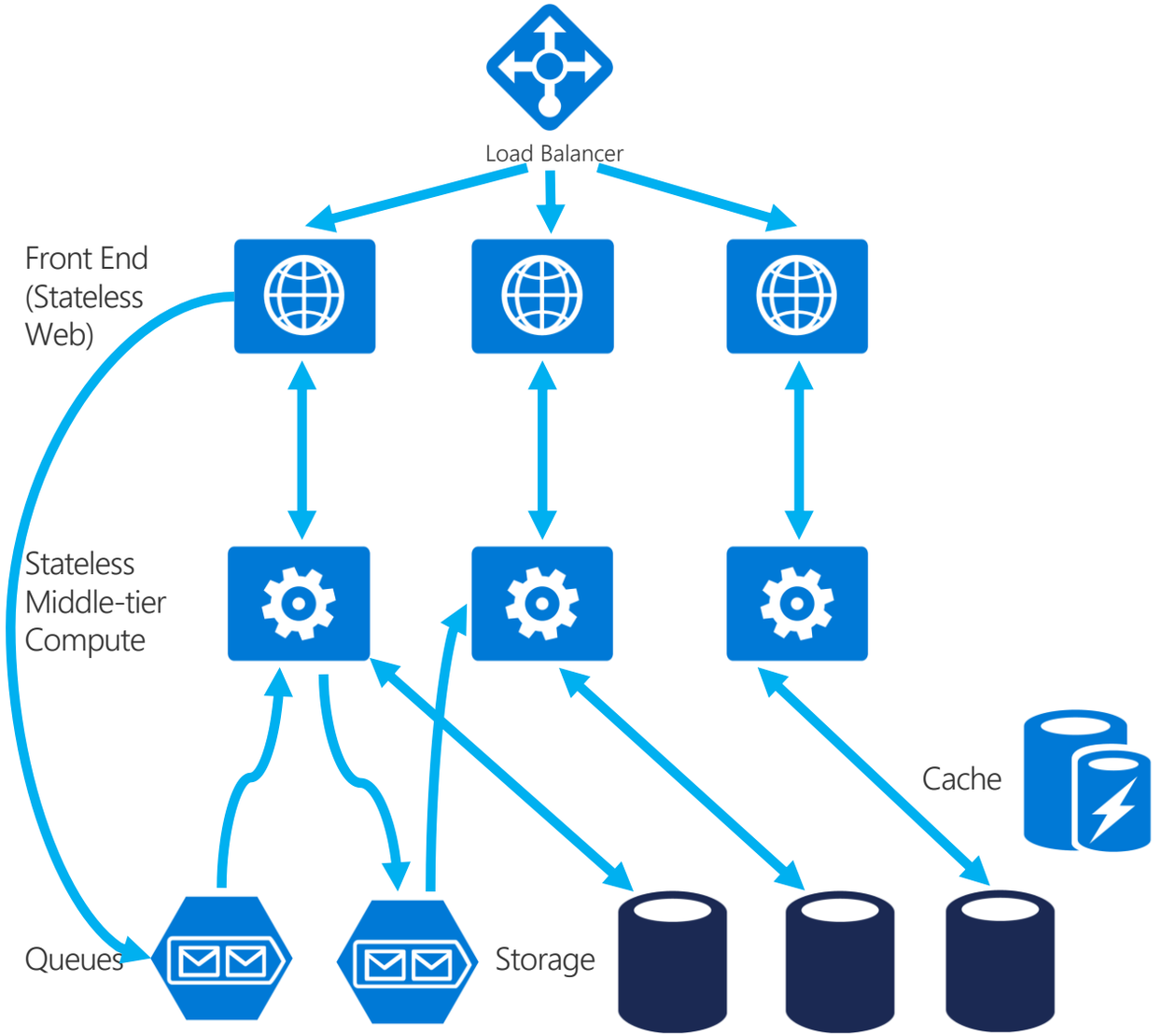


New or transformed microservices app

… we support any stage you choose

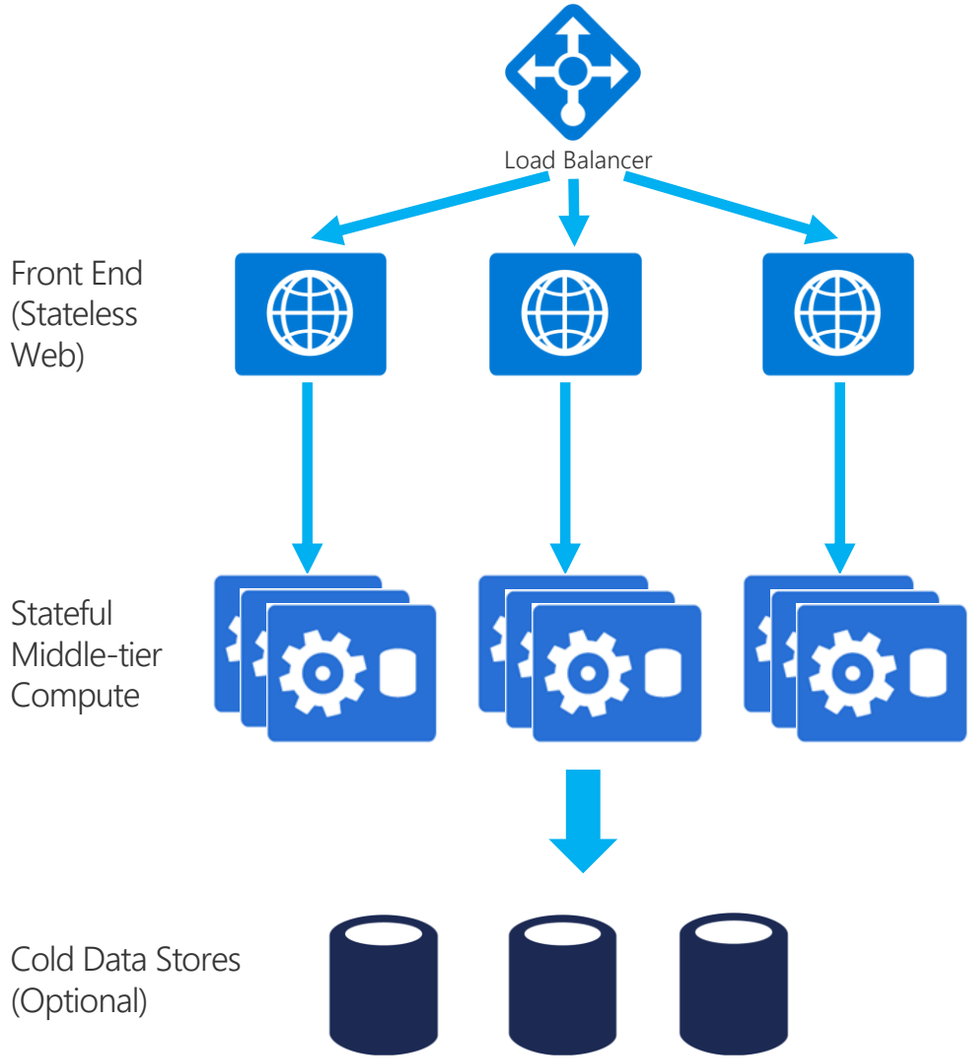
Stateless Services Pattern

- Scale stateless services backed by partitioned storage
- Increase reliability and ordering with queues
- Reduce read latency with caches
- Manage your own transactions for state consistency



Stateful Services Pattern Simplify design, reduce latency

- Application state resides in the compute tier
- Low latency reads and writes
- Partitions are first class at the service layer for scale-out
- Built in transactions
- External stores for exhaust and offline analytics

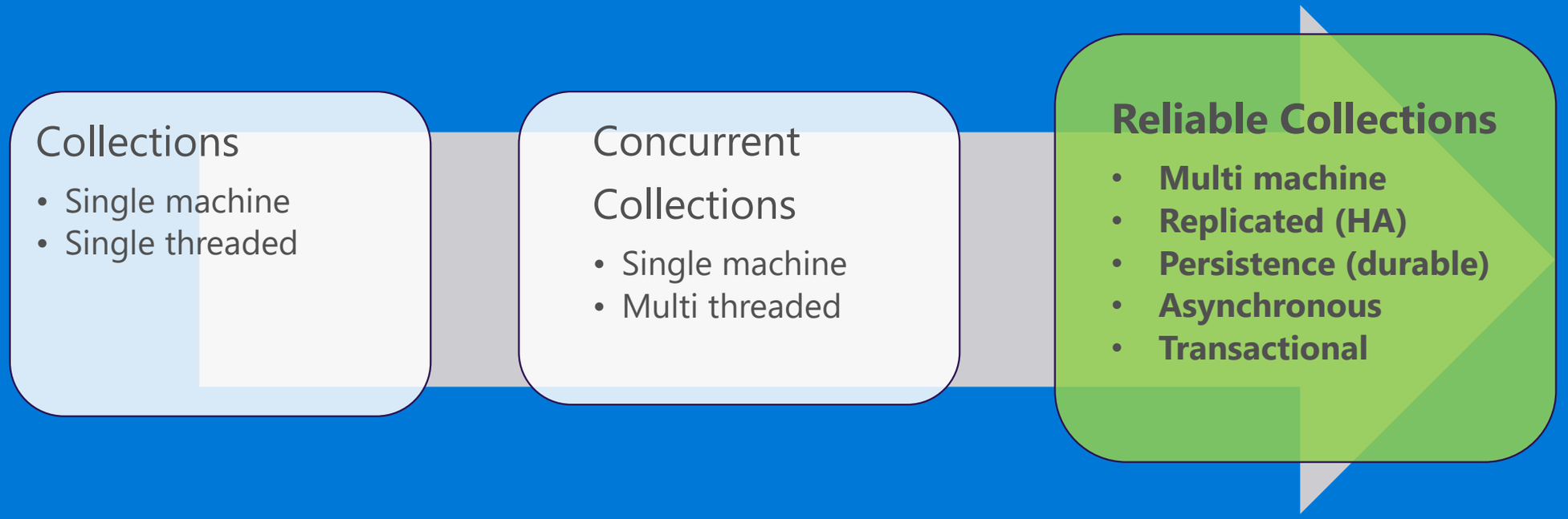


Reliable Services

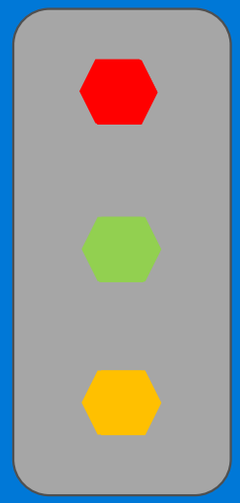
- Stateless microservice
  - Has either no state or it can be retrieved from an external store
  - There can be N instances
  - e.g. web frontends, protocol gateways, Azure Cloud Services etc.
- Stateful microservice
  - Maintain hard, authoritative state
  - N consistent copies achieved through replication and local persistence
  - e.g. database, documents, workflow, user profile, shopping cart etc.

Reliable Collections

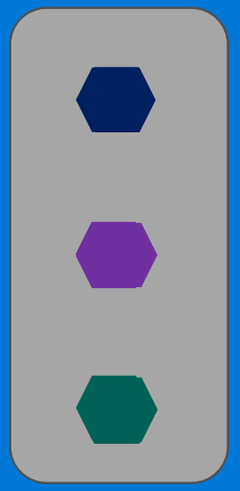
- Reliable collections make it easy to build stateful services.
- An evolution of .NET collections for the cloud.



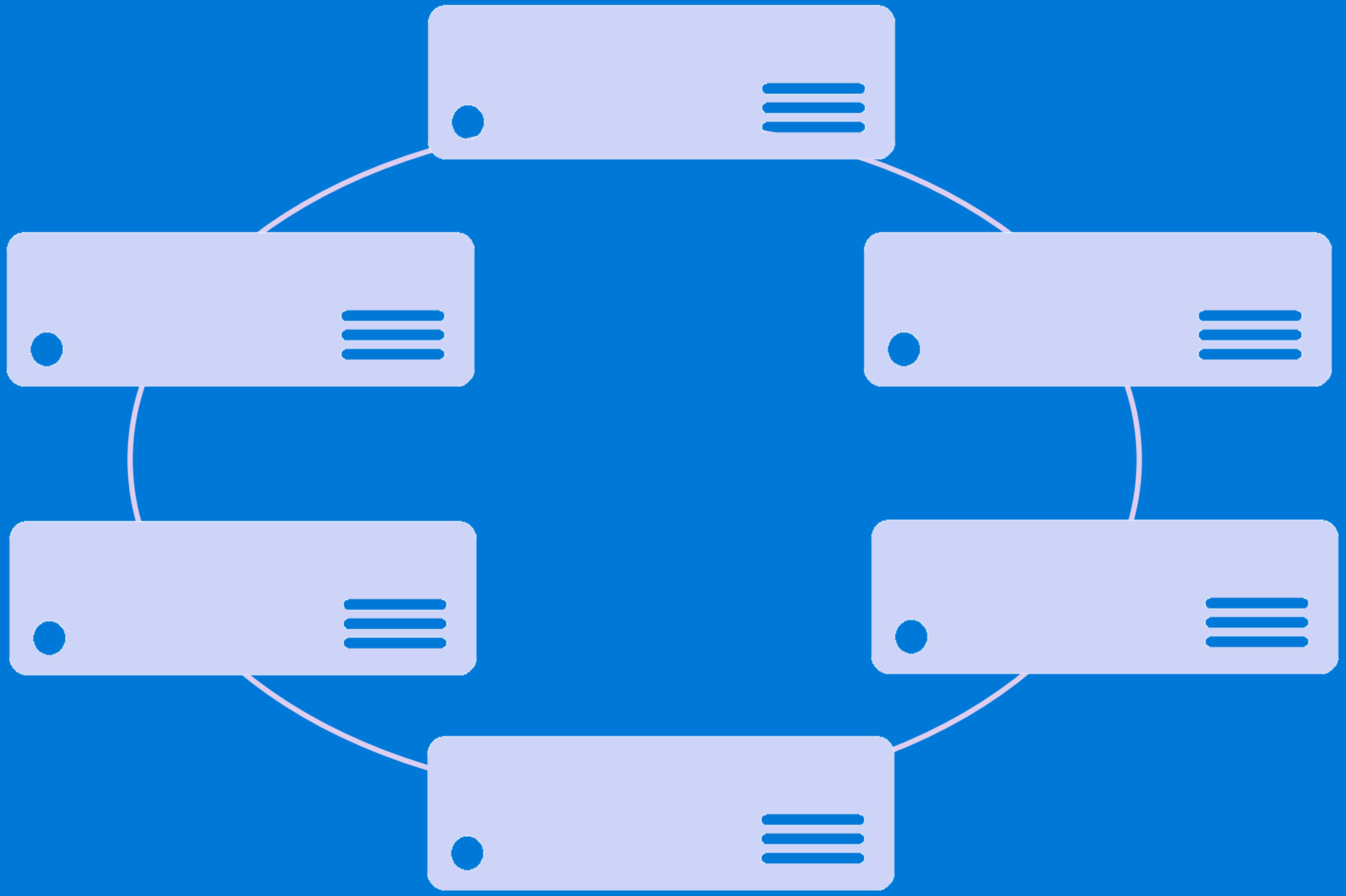
Service Fabric cluster with microservices



App1

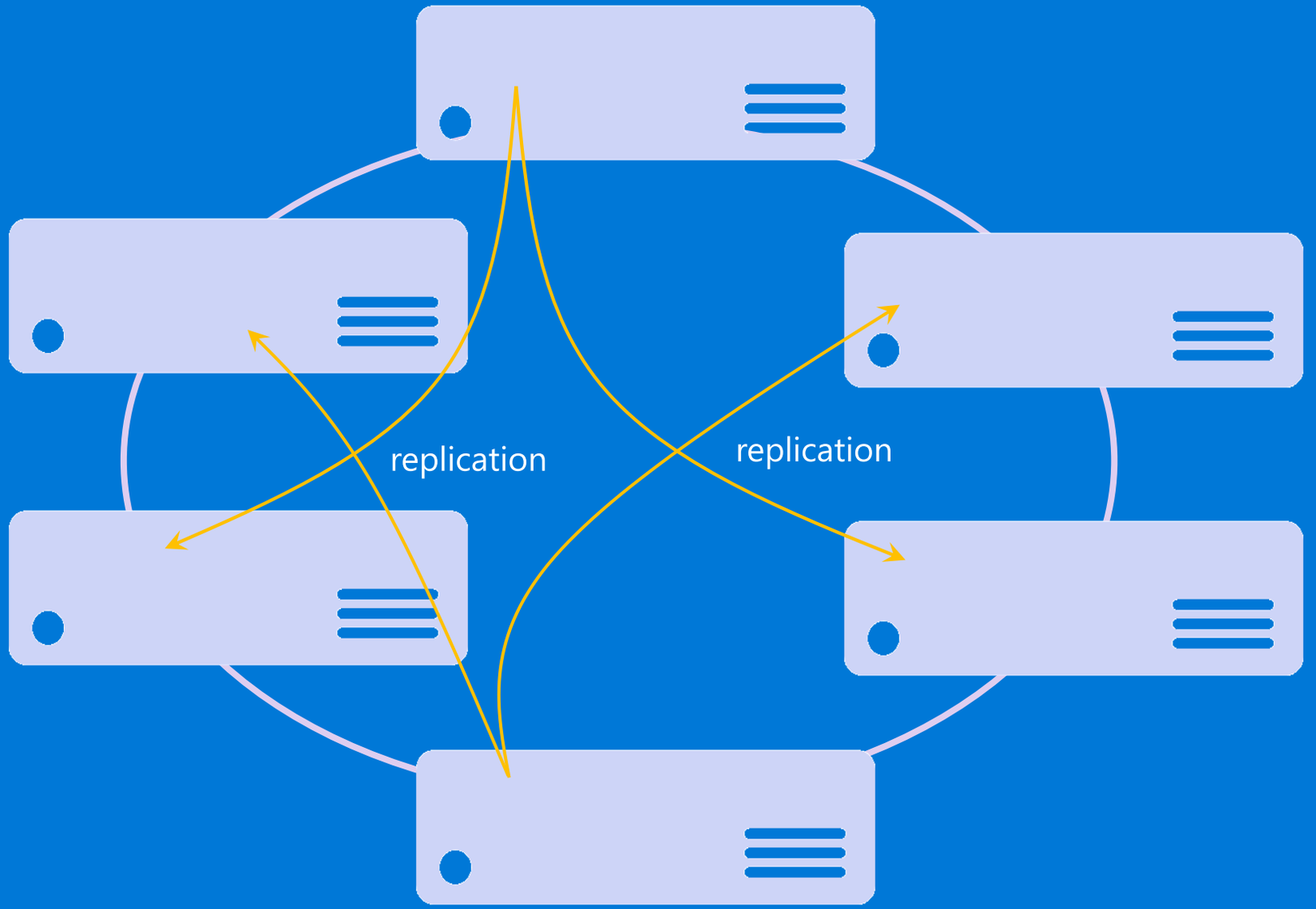


App2

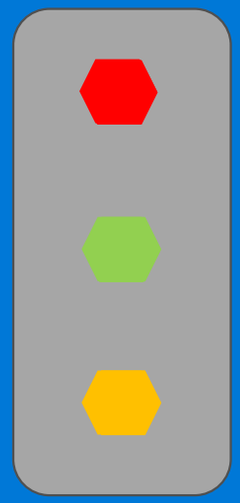


Stateful microservice

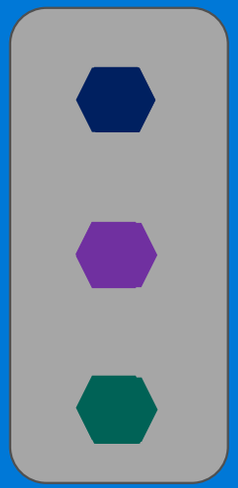
Application
Package



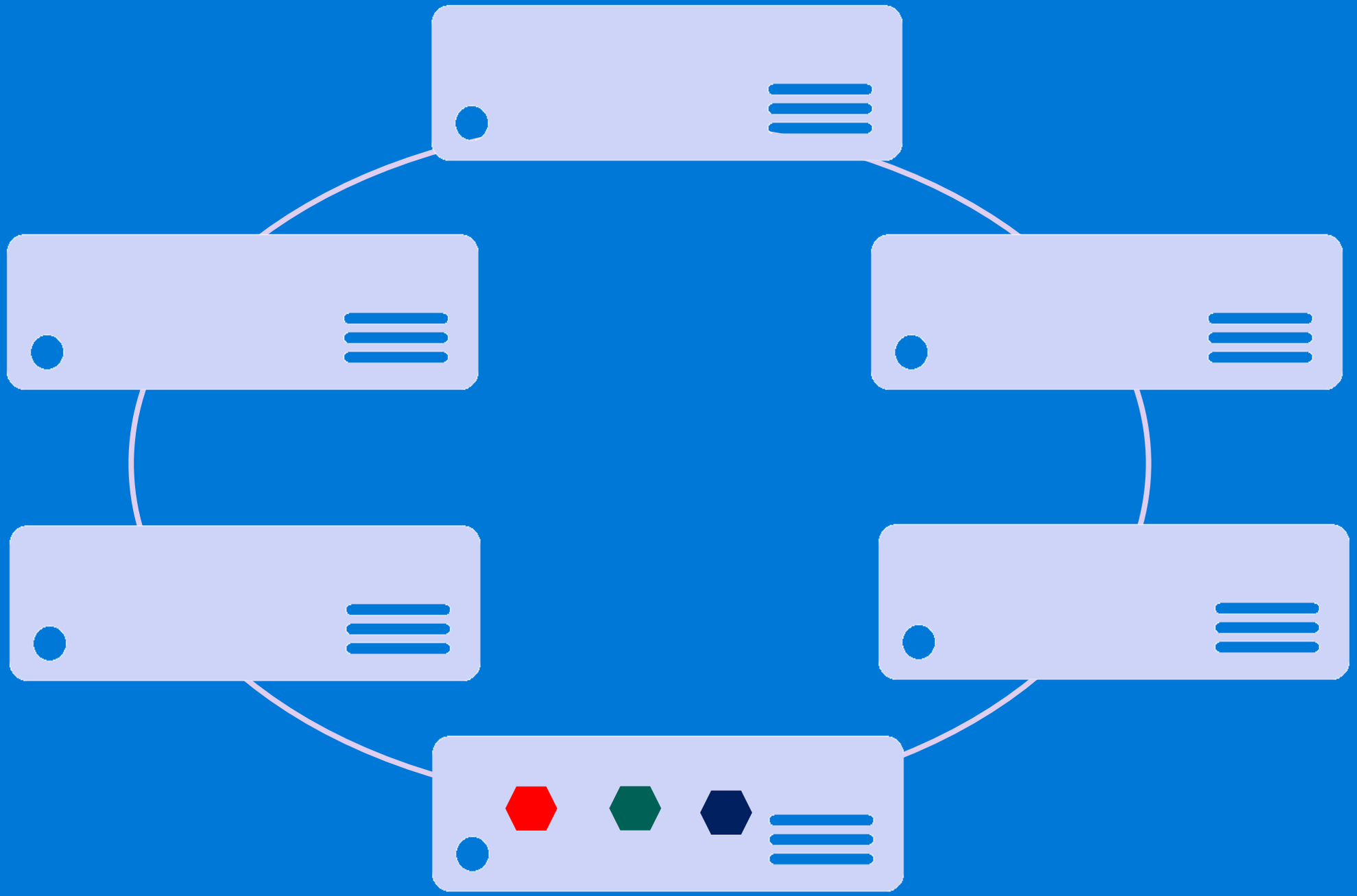
Handling machine failures



App1



App2



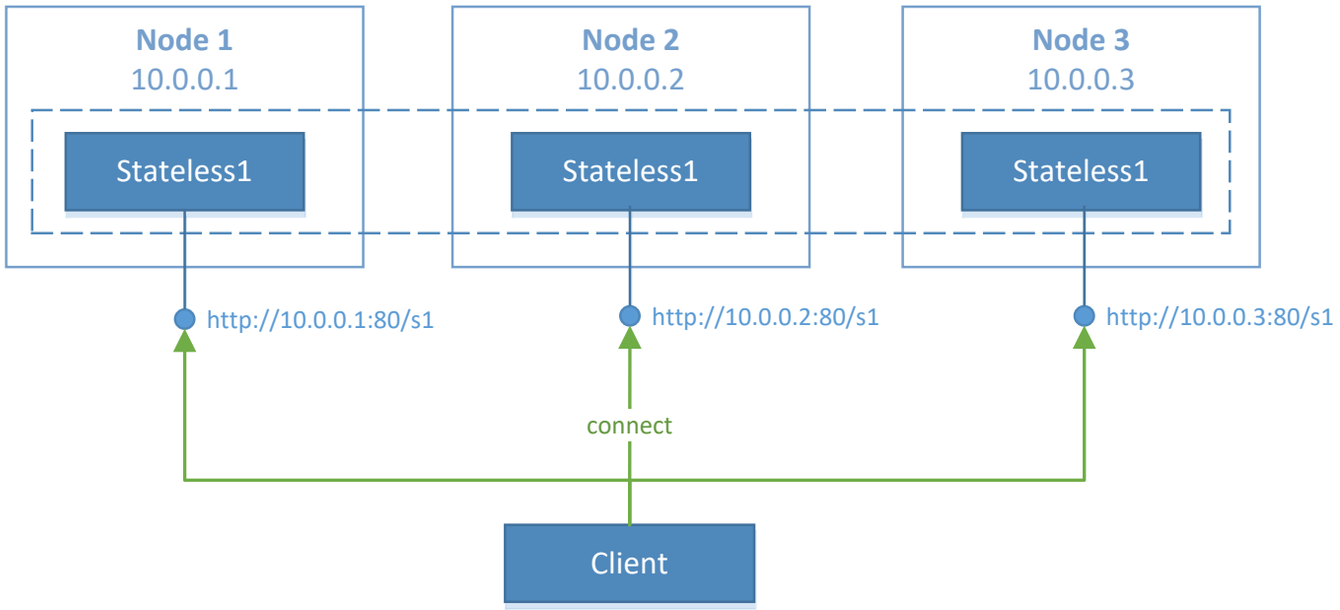
Stateless service communication

A stateless service is identical on all nodes.

Clients can connect directly to any instance.

Has either no state or it can be retrieved from an external store

There can be N instances



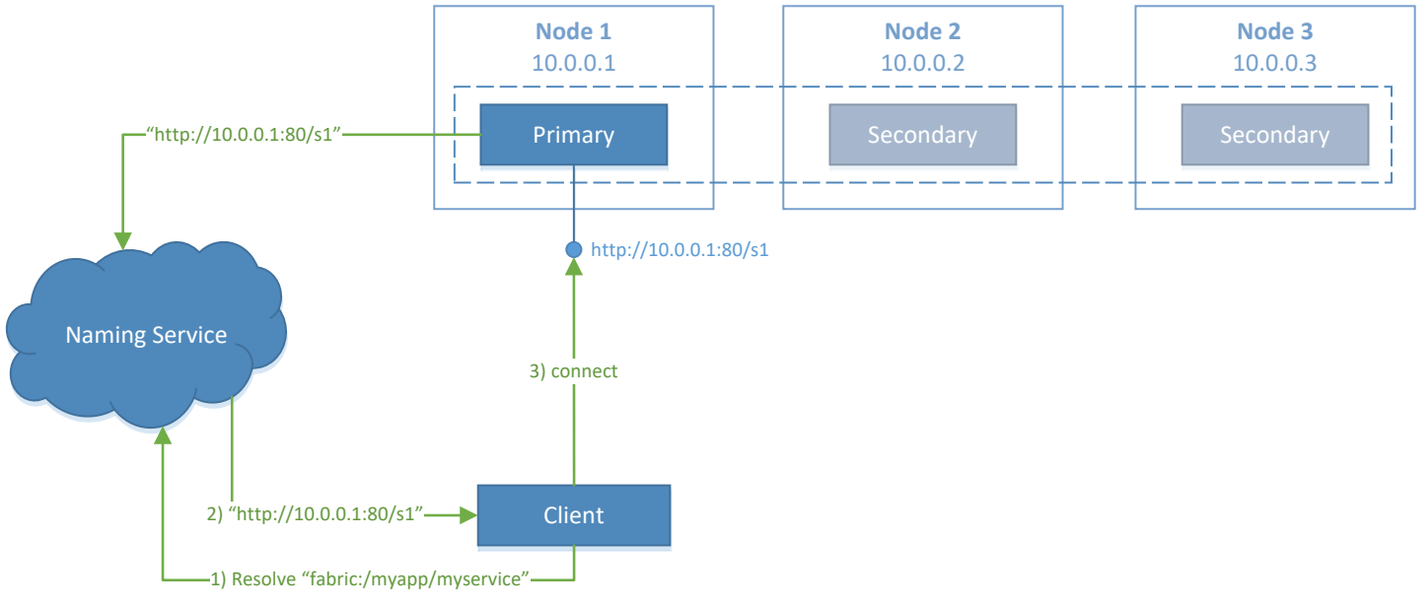
Stateful service communication

A stateful service is not identical on all nodes

Clients must find the right replica to connect to

Maintain hard, authoritative state

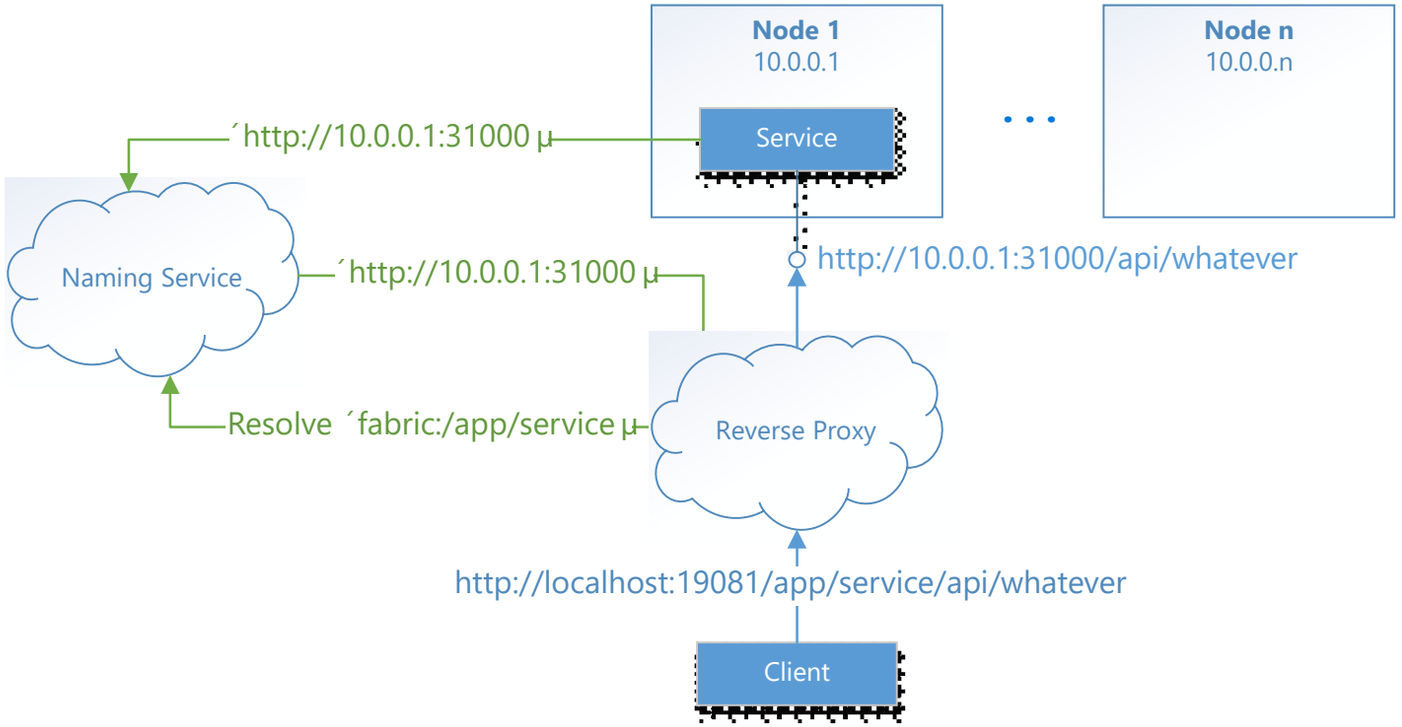
N consistent copies achieved through replication and local persistence



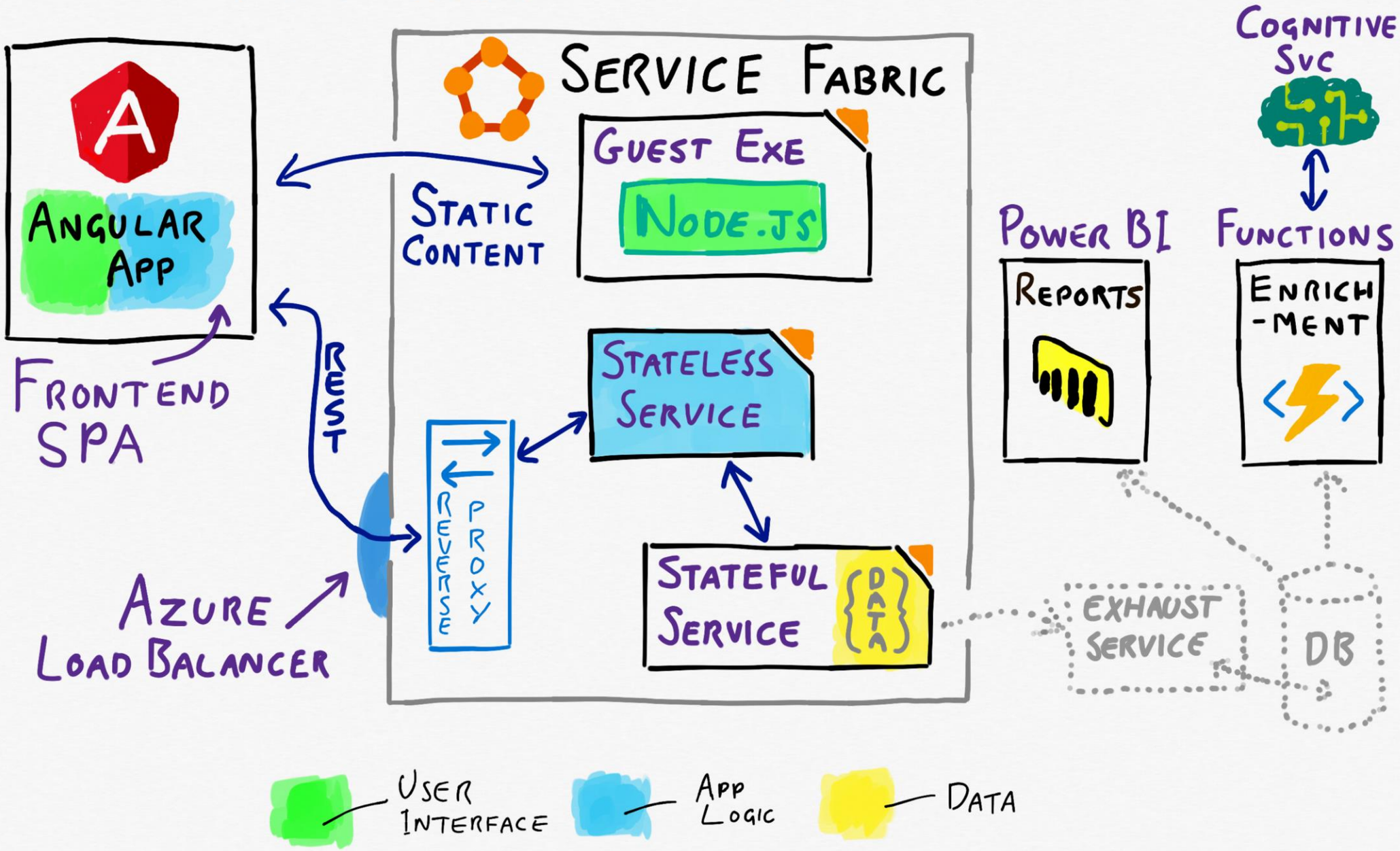
Reverse Proxy communication

Reverse Proxy: HTTP forwarding with Naming Service

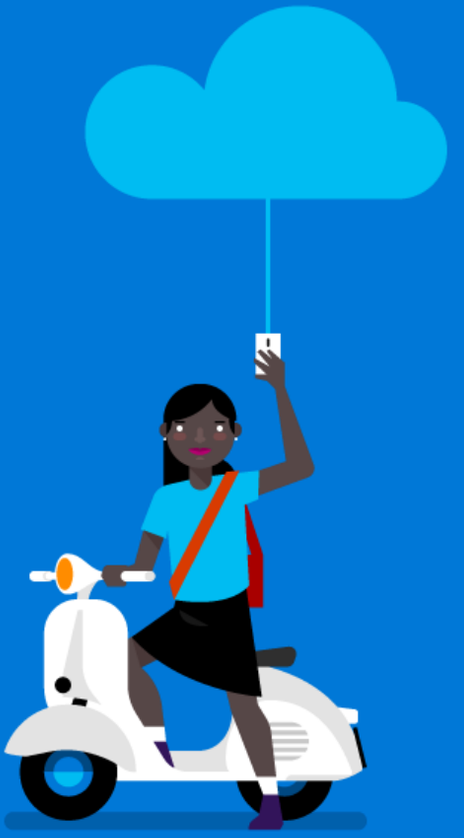
```
HttpClient httpClient = new HttpClient();
await httpClient.GetAsync("http://localhost:19081/app/service/api/whatever");
```



Service Fabric Architecture



Demo



Microsoft

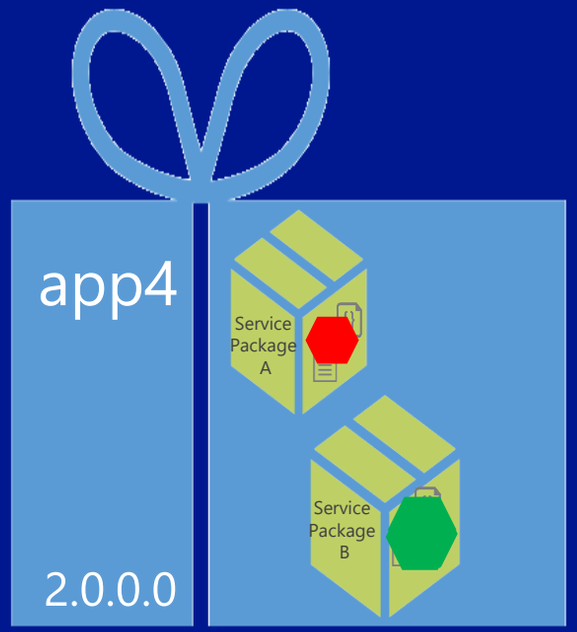
aka.ms/smilr

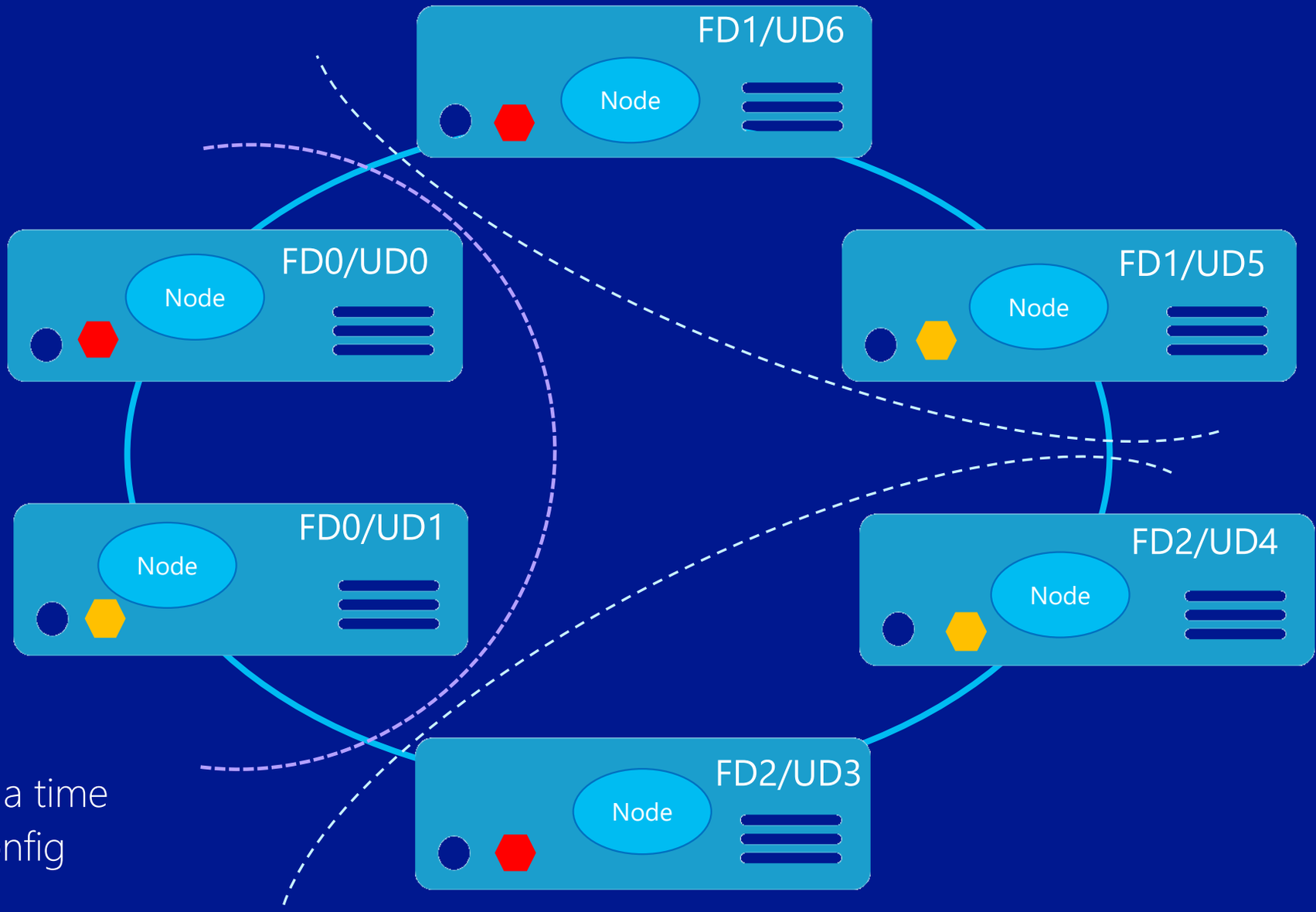
Microsoft

Service Fabric management capabilities

- Reliable optics into application health
- Automatic repair action based on policies you set
- Scales up/down based on service demand
- Integrated alerting and notification system
- Tools to effectively test a service for robustness
- Tools for easy deployments and config management
- Tools to perform service upgrades without downtime

Upgrading Services with zero downtime

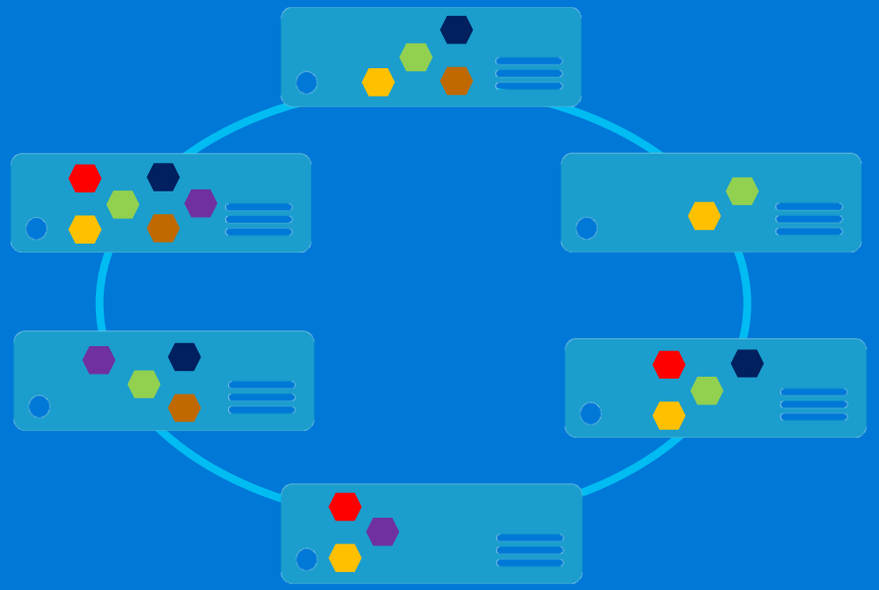




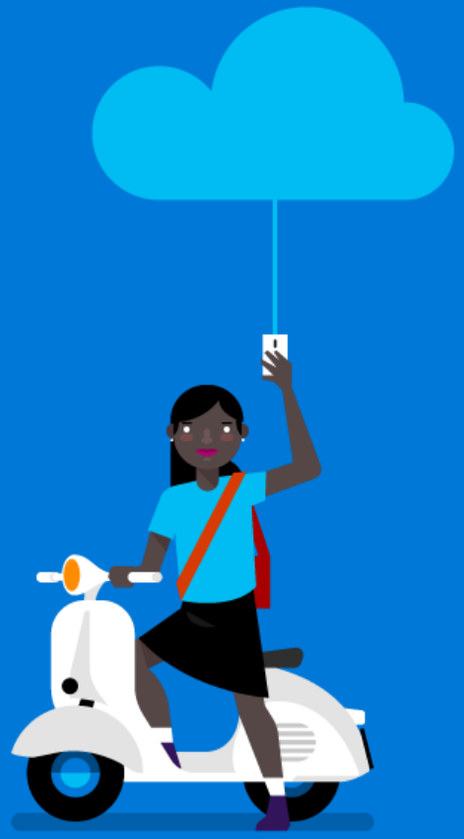
- Upgrade progresses one UD at a time
- Upgrade limited to the code/config package that changed

Other important topics in Service Fabric

- Cluster scale-up and down and different VM sizes
- Cluster placement constraints
- Application rolling upgrades and rollback
- Application health monitoring and reporting
- Application operational insights
- Advance application resource balancing and scheduling
- Chaos and scenario testing in production



Example Customer Solutions



Microsoft

TalkTalk, a UK video-on-demand service delivering TV and movie content across multiple-devices

Benefits

Microservices workflow for content encoding and resolution

Agility - Ability to upgrade microservices independently and without downtime. No need to coordinate DB schema with app upgrades

Programming API - Using actors and reliable collections to easily orchestrate the encoding and resolution of the on-demand content

Scalability - Real time résolution for 30K titles, designed to scale for growth of users, devices and content

- Replacing existing IaaS/DB backed system with microservices solution
- 1.5 PB of for streaming content delivered to millions of customers using Azure Media Services

TalkTalk

Microservices workflow for content encoding

TalkTalk

Service Fabric Cluster

Content Master App

Content Profile App

Encoding Workflow App

Azure Media Services Job Management App

KeyStore App

Resource Resolution App

Encryption Job Management App

Management API

DocumentDB

(Ad-hoc searching, long term data storage)

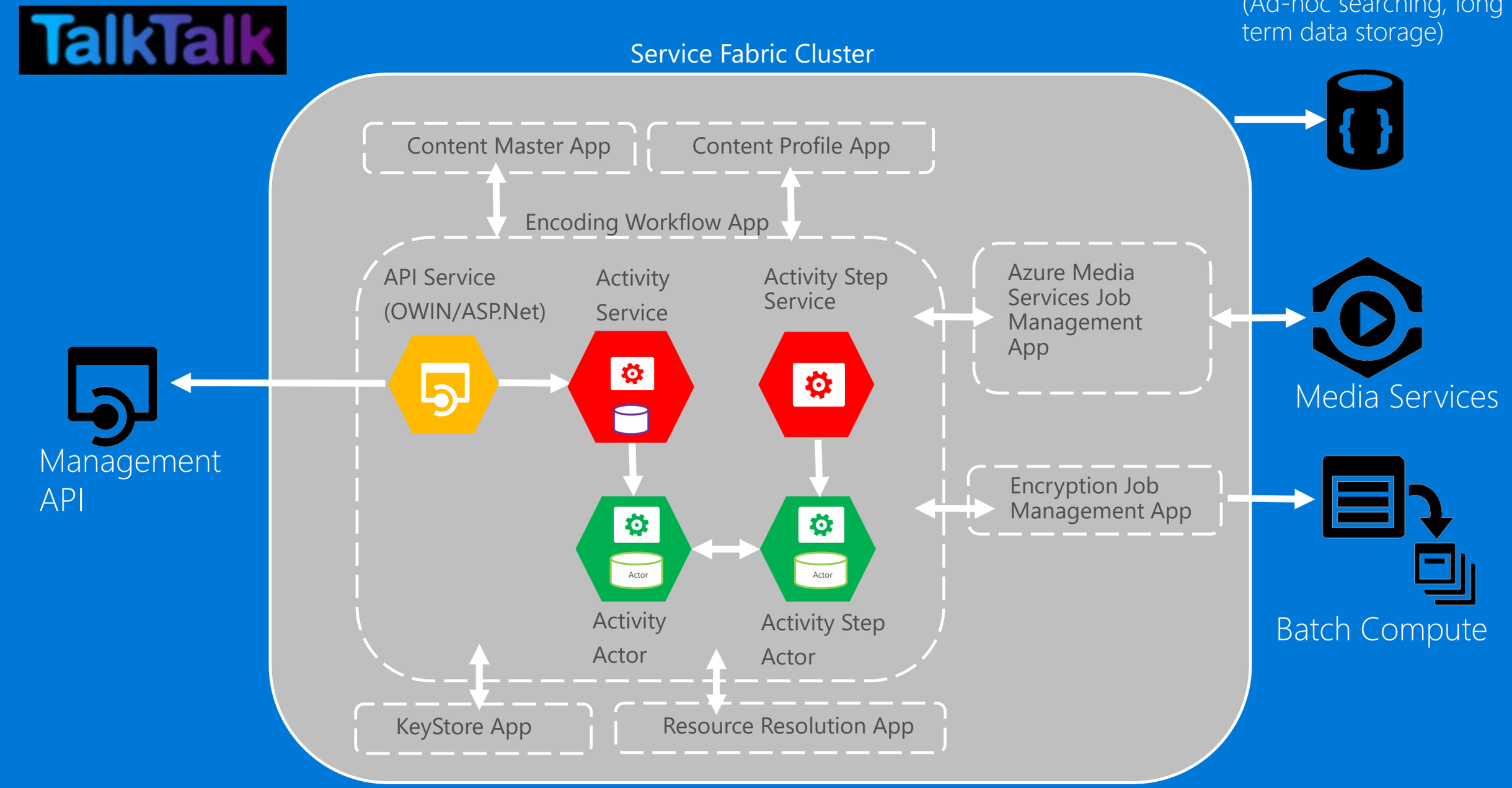
Media Services

Batch Compute

Microservices workflow for content encoding

TalkTalk

DocumentDB
(Ad-hoc searching, long term data storage)



Schneider Electric develops connected technologies and solutions to manage energy and process in ways that are safe, reliable, efficient and sustainable

Benefits

Microservices IoT solution to manage uninterruptable power supplies (UPS)

Scale – Service Fabric simplifies scale. With millions of devices we need partitioning and resource balancing that makes this transparent

Actor Programming API – Service Fabric has the simplest-to-use actor model implementation in the market

Density and Availability – VM utilization enables managing millions of devices with automatic failover

- Management & operation of devices. Query and execute commands, send commands from device to LOB apps
- Communicate with devices securely and in multiple protocols
- Processing and latency need to be sub-second
- Integration Azure services such as Event Hubs and storage.

Schneider Electric

Microservices IoT solution to manage devices