

# Microsoft Cloud Workshop

## Microservices architecture

Nick Ward

<http://linkedin.com/in/nickward13>

[nickward@microsoft.com](mailto:nickward@microsoft.com)



# Abstract and learning objectives

## Abstract

Help attendees gain a better understanding of microservices and serverless architectures. Attendees will help an online concert ticket vendor survive the first 5 minutes of crushing load by handling the client's scaling needs through microservices built on top of Service Fabric. Students will also apply smooth updates roll-back failing updates, and design a load testing implementation to optimize the architecture for handling spikes in traffic.

## Learning objectives

- Implement scale and resiliency with Service Fabric
- Enable serverless solutions with Azure Functions
- Control API access with API Management
- Provide query flexibility with Cosmos DB

# Step 1: Review the customer case study

## Outcome

Analyze your customer needs

## Timeframe

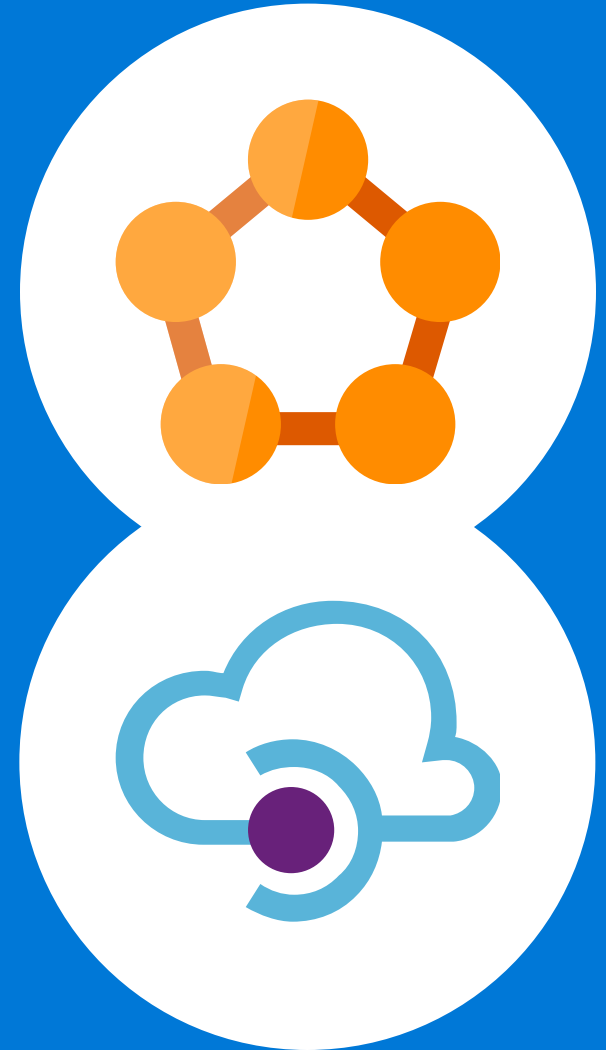
15 minutes

# Customer situation

- Contoso Events is an online ticket provider experiencing consistent growth
- Has plans to further growth demand
- Wants to extend customer reach through partners
- Plans to retire and replace existing customer solution

# Customer situation

- Concerned about performance, scale, and costs
- Desire a decoupled design
- Interested in microservices, Service Fabric, and serverless architectures
- Looking for strategy for exposing APIs to partners



# Customer needs

- Event tickets can be orders from multiple channels
- Customers must be registered/logged in to place orders
- Admin site for order management and reports
- Ability to rapidly release new features, while reducing downtime



# Customer needs

- Be able to handle unpredictable spikes in demand
- Improved operations management
- Migrate to Cosmos DB
- Secure API management
- Integration with third-party credit card processor



# Customer objections

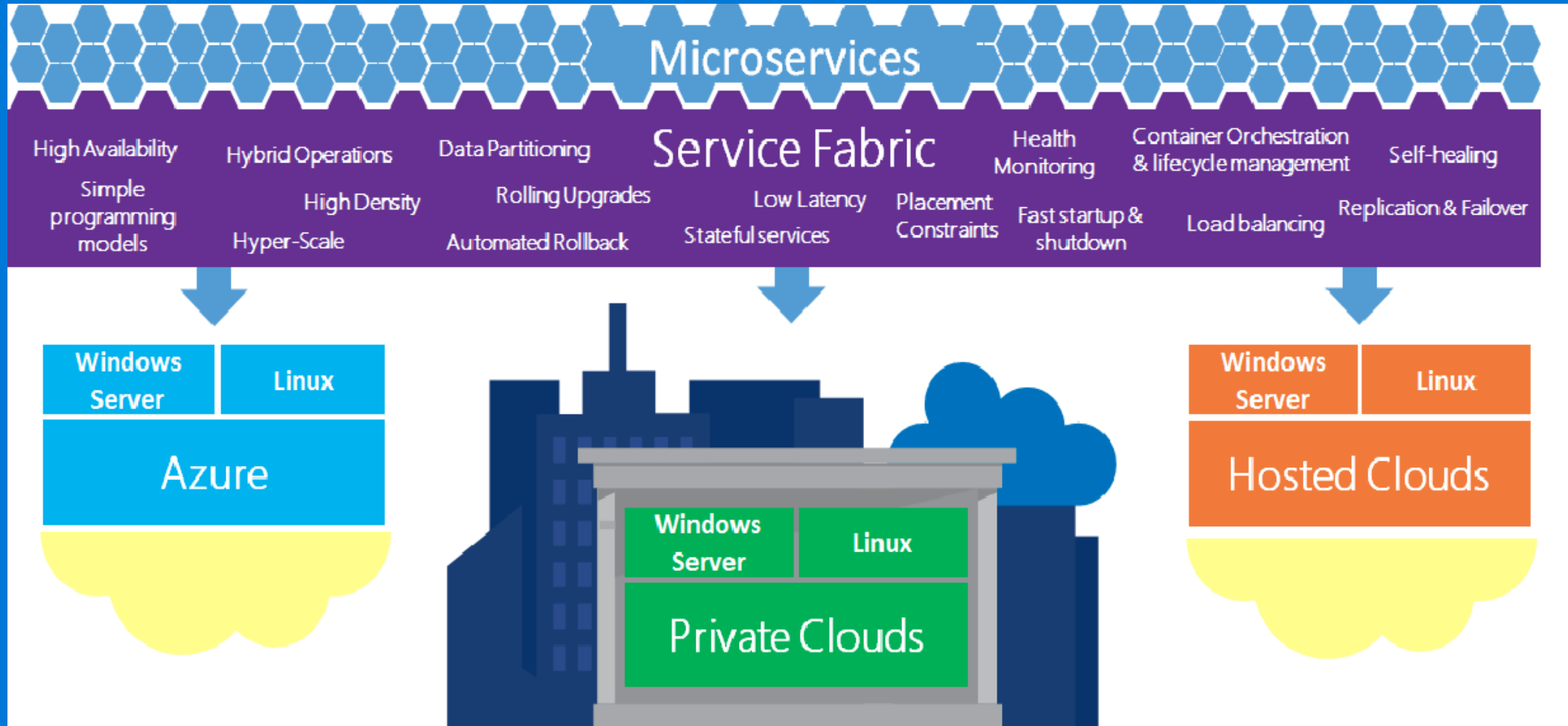
- Is Service Fabric the right solution?
- Which of our existing skills can be applied to microservices and Service Fabric?
- Can stateful services or actors help us with ticket ordering throughput?
- How and where can stateful services and actors help us?
- How can Azure Functions be leveraged?





# Common scenarios

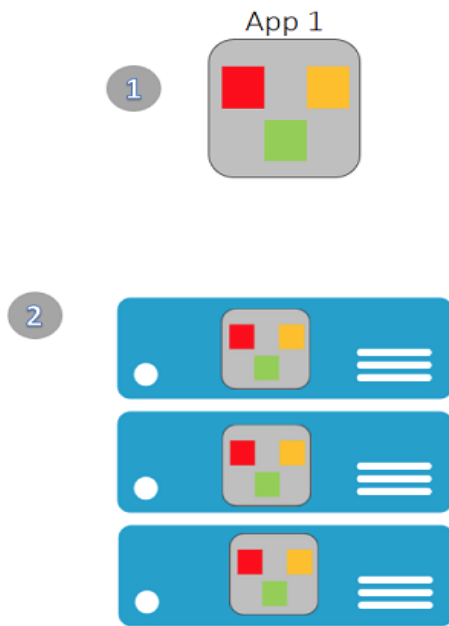
## Service Fabric overview



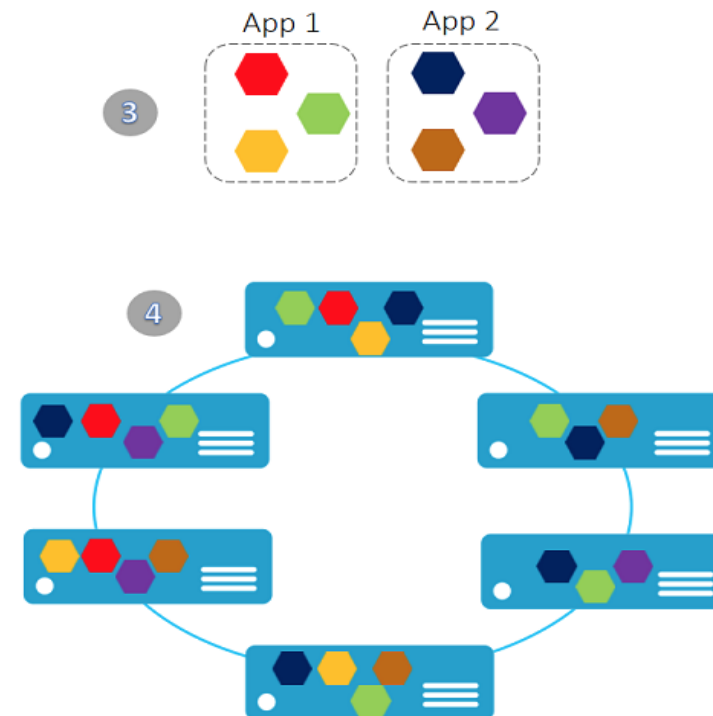
# Common scenarios

## Monolithic vs. Microservices approach

Monolithic application approach



Microservices application approach



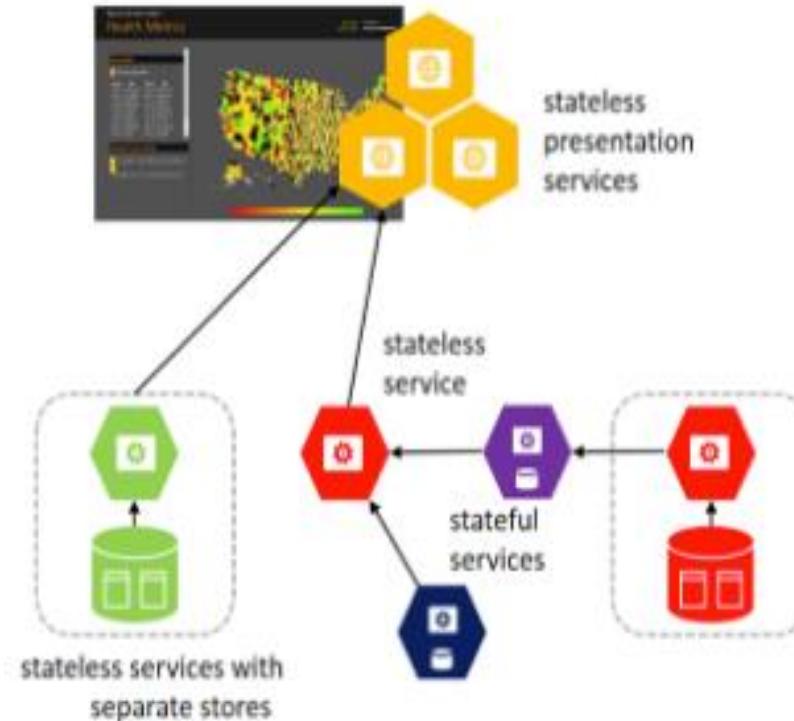
# Common scenarios

## Monolithic vs. Microservices – storage state

State in Monolithic approach



State in Microservices approach



# Step 2: Design the solution

## Outcome

Design a solution and prepare to present the solution to the target customer audience in a 15-minute chalk-talk format.

## Timeframe

60 minutes

<b><i>Business needs</i></b> (10 minutes)	<ul style="list-style-type: none"><li>• Respond to questions outlined in your guide and list the answers on a flipchart.</li></ul>
<b><i>Design</i></b> (35 minutes)	<ul style="list-style-type: none"><li>• Design a solution for as many of the stated requirements as time allows. Show the solution on a flipchart.</li></ul>
<b><i>Prepare</i></b> (15 minutes)	<ul style="list-style-type: none"><li>• Identify any customer needs that are not addressed with the proposed solution.</li><li>• Identify the benefits of your solution.</li><li>• Determine how you will respond to the customer's objections.</li><li>• Prepare for a 15-minute presentation to the customer.</li></ul>

# Step 3: Present the solution

## Outcome

Present a solution to the target customer in a 15-minute chalk-talk format

## Timeframe

30 minutes (15 minutes for each team to present and receive feedback)

## Directions

- Pair with another table
- One table is the Microsoft team and the other table is the customer
- The Microsoft team presents their proposed solution to the customer
- The customer asks one of the objections from the list of objections in the case study
- The Microsoft team responds to the objection
- The customer team gives feedback to the Microsoft team

# Wrap-up

## Outcome

- Identify the preferred solution for the case study
- Identify solutions designed by other teams

## Timeframe

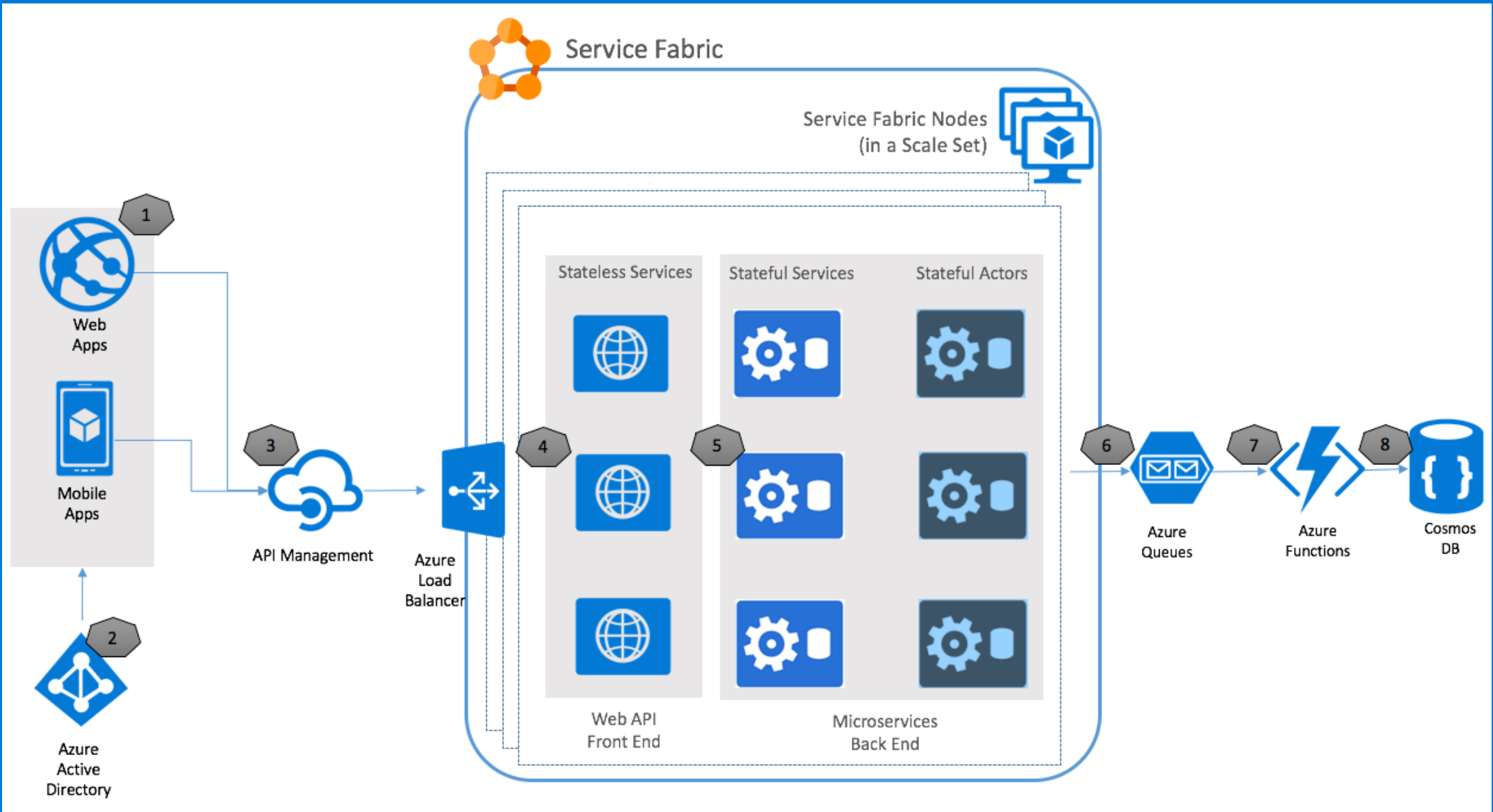
15 minutes

# Preferred target audience

- Steve Dormer, CIO at Contoso Events
- Primary audience is business and technology decision makers
- Usually talk to key architects, developers, and Infrastructure Managers who report to the CIO, or to application sponsors or their representatives

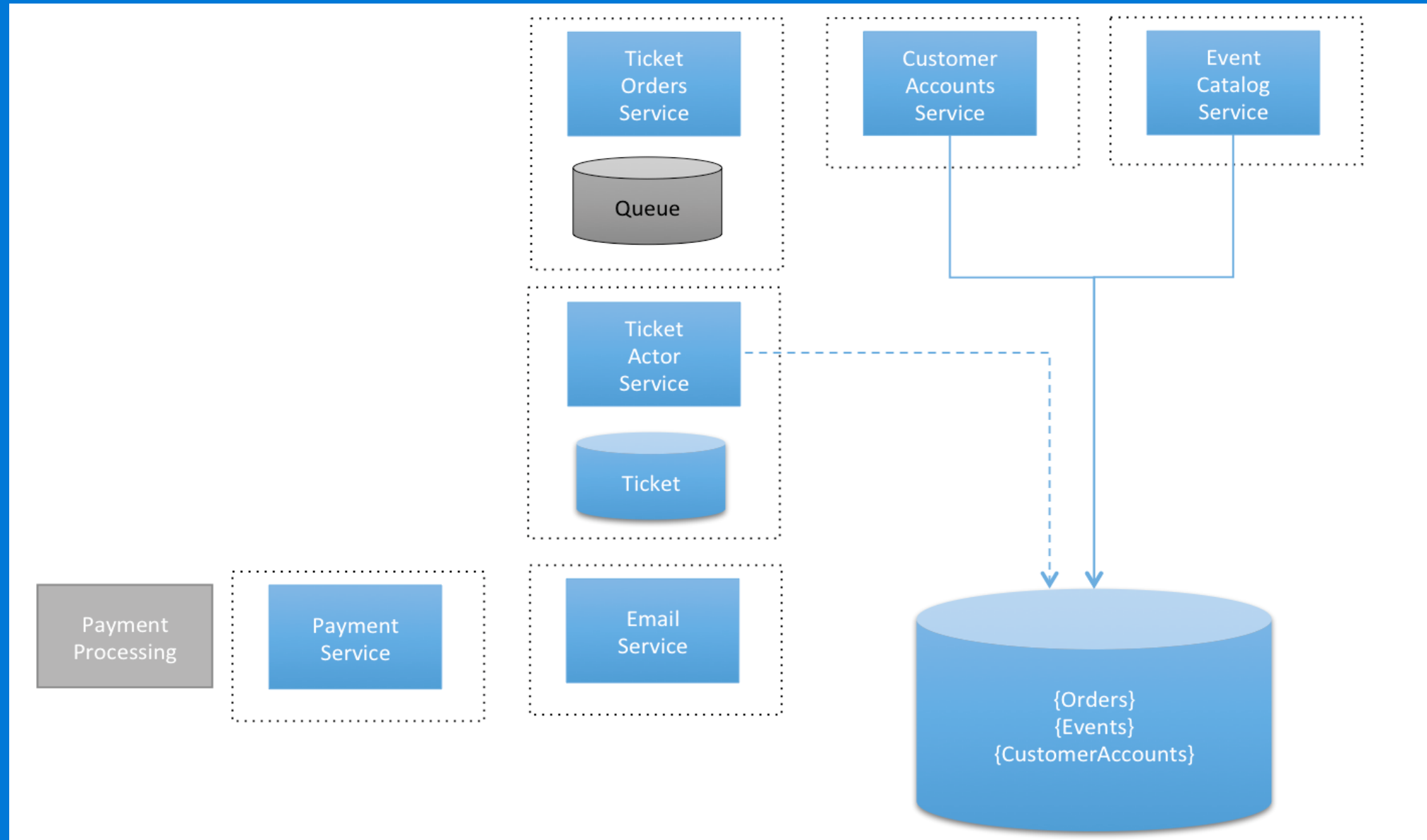


# Preferred solution



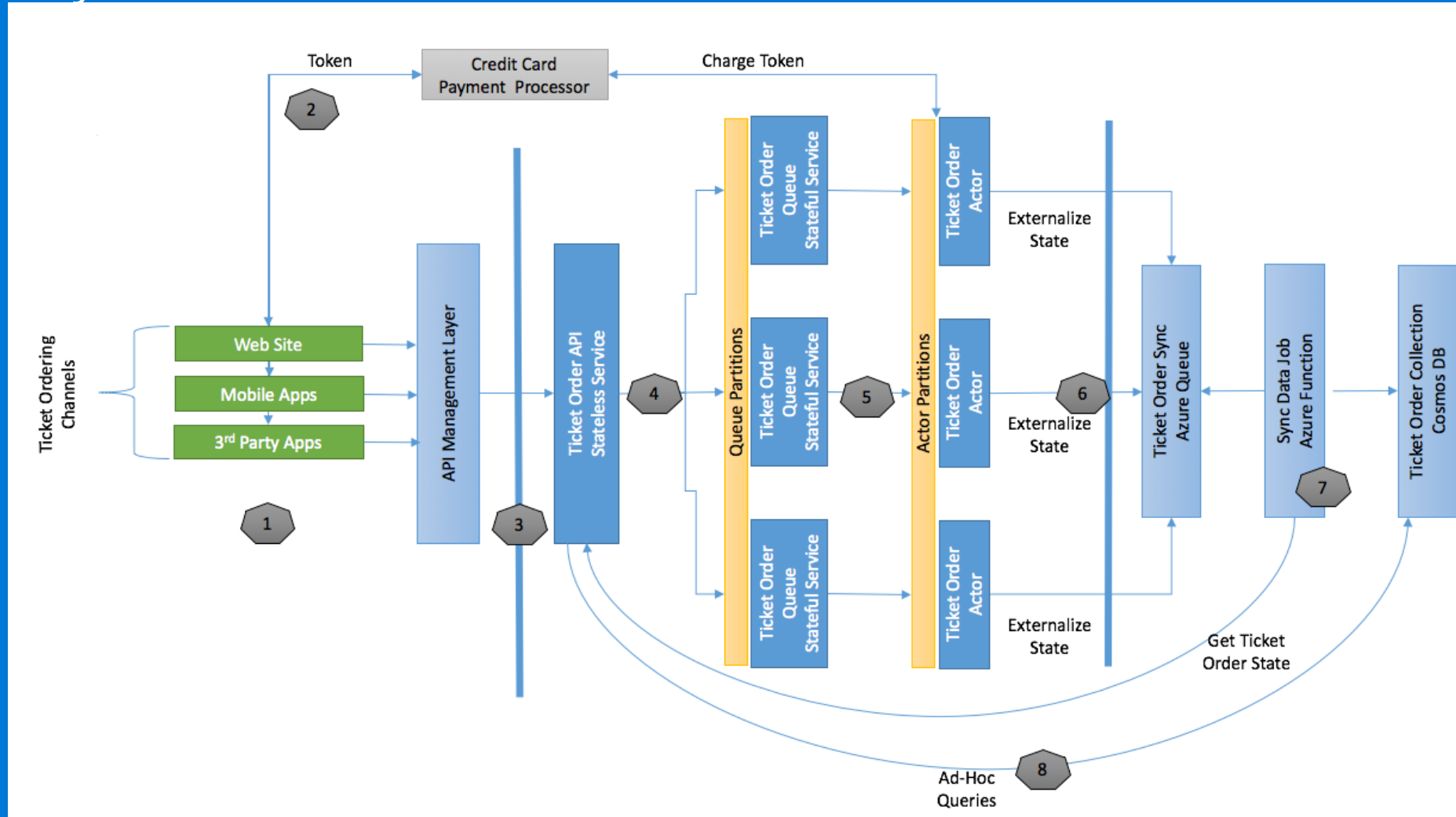


# Preferred solution



# Preferred solution

## Scalability of ticket orders



# Preferred solution

## Scalability of ticket orders

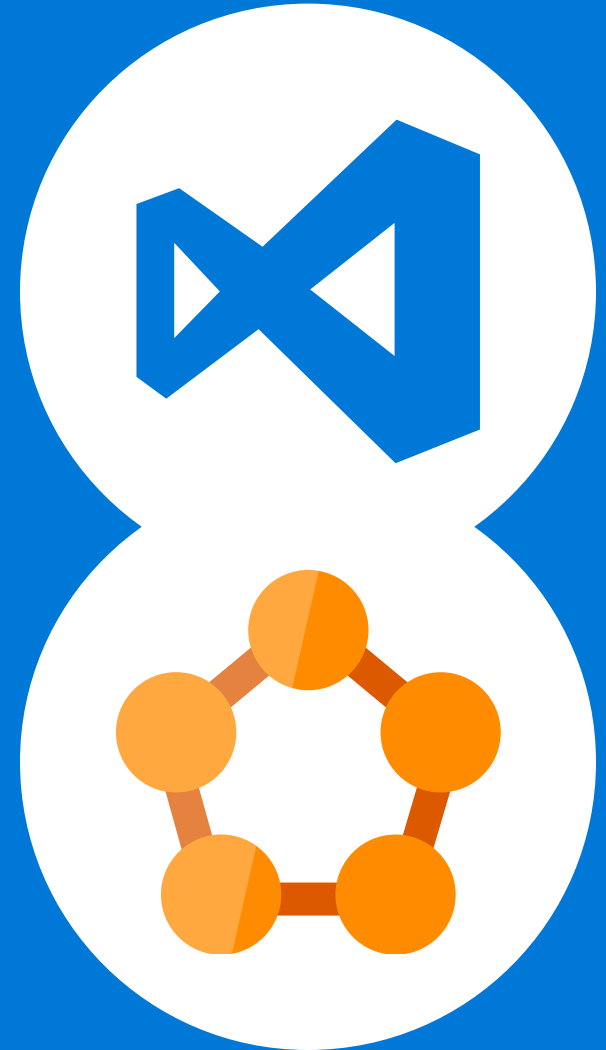
- API Management used to meet demand and high-availability requirements
- Ticker Order API offloads requests to Ticket Order Queue using Service Fabric
- Ticket Order Actor handles processing
- Azure Function persists orders in Cosmos DB



# Preferred solution

## Improving DevOps workflows

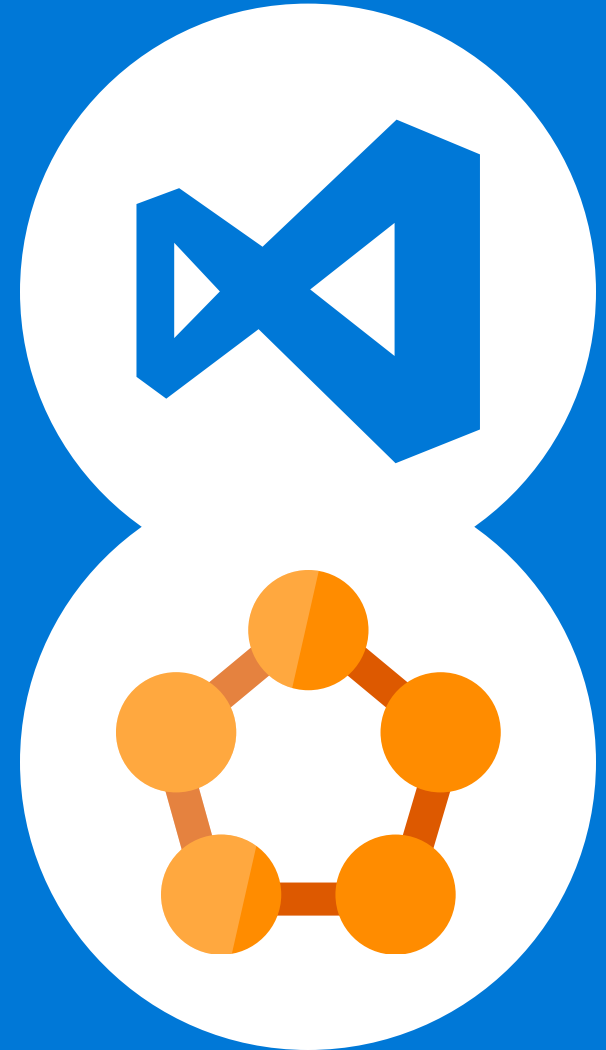
- Visual Studio Service Fabric solution
- Upgrade application to preserve state
- Service Fabric performance counters drive auto-scaling
- Service Fabric inherently provides HA



# Preferred solution

## Improving DevOps workflows

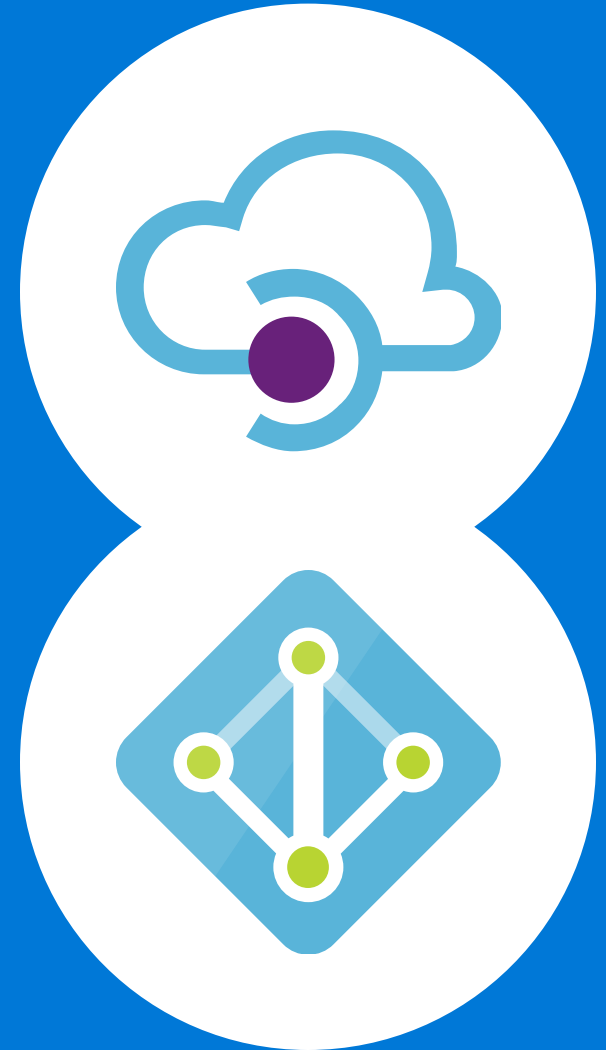
- Problems and failures reported in Service Fabric health manager
- Cluster security provisioned up front
- Service Fabric updates handled by Microsoft



# Preferred solution

## Controlling Access to APIs

- API publishing tools and Swagger
- Leverage API Management features
- All API consumers issued a key
- Employ Azure AD B2C for customer login



# Preferred objections handling

- Is Service Fabric the right solution?
- Which of our existing skills can be applied to microservices and Service Fabric?
- Can stateful services or actors help us with ticket ordering throughput?
- How and where can stateful services and actors help us?
- How can Azure Functions be leveraged?



# Customer quote

*"With Service Fabric we are able to move to microservices architecture without the DevOps headache. Service Fabric provides so much to support deployment, compute utilization, health monitoring and recovery – we could leverage the same team while increasing the size of our solution and feature set!"*

—Steve Dormer, CIO at Contoso Events



