



AZ-400.00

Learning Path 09:

Implement continuous feedback

Agenda



- Module 01: Implement tools to track usage and flow.
- Module 02: Develop monitor and status dashboards.
- Module 03: Share knowledge within teams.
- Module 04: Design processes to automate application analytics.
- Module 05: Manage alerts, Blameless retrospectives and a just culture.
- Labs & Learning Path review and takeaways.

Learning Path overview



Learning objectives

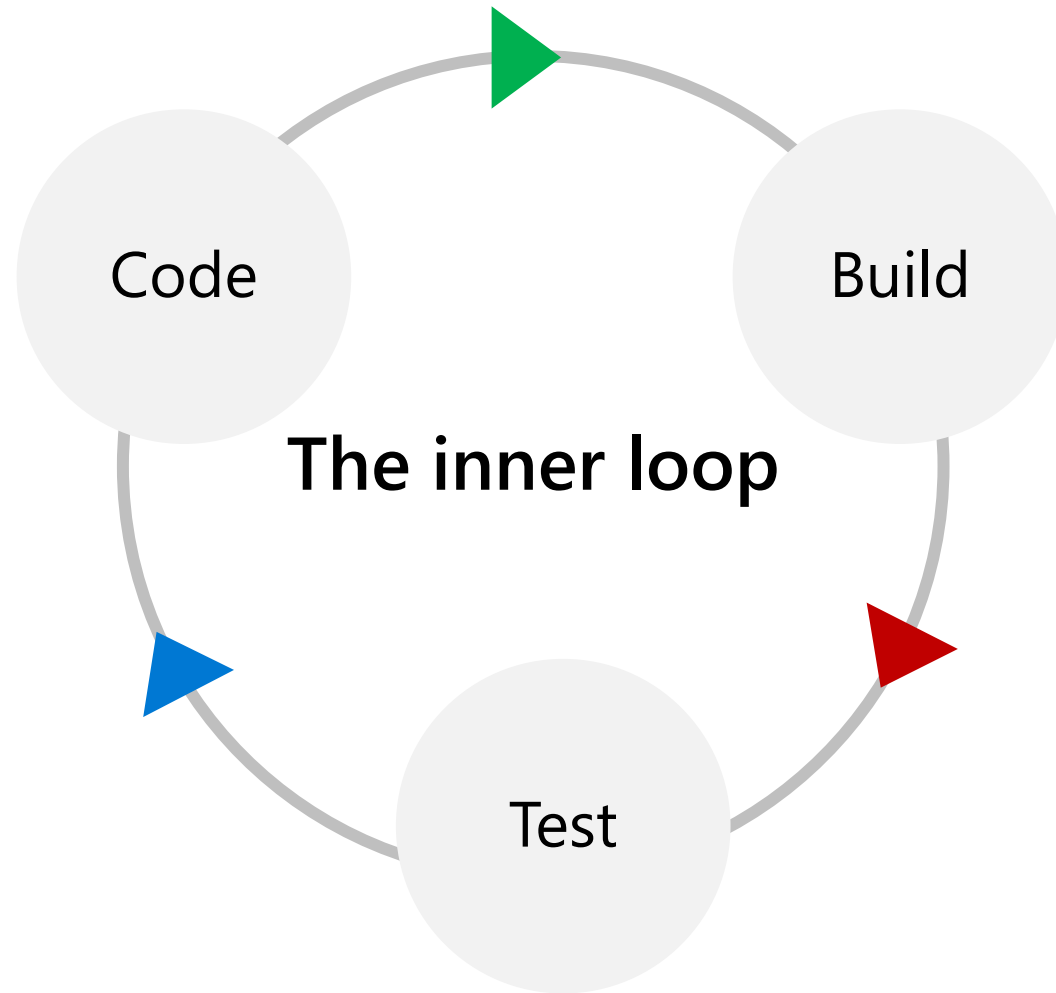
After completing this Learning Path, students will be able to:

- 1 Develop monitoring and status dashboards
- 2 Implement tools to track system usage, feature usage, and flow
- 3 Integrate and configure ticketing systems with development team's work management
- 4 Design processes to automate application analytics
- 5 Manage alerts and reduce meaningless and non-actionable alerts
- 6 Carry out blameless retrospectives and create a just culture

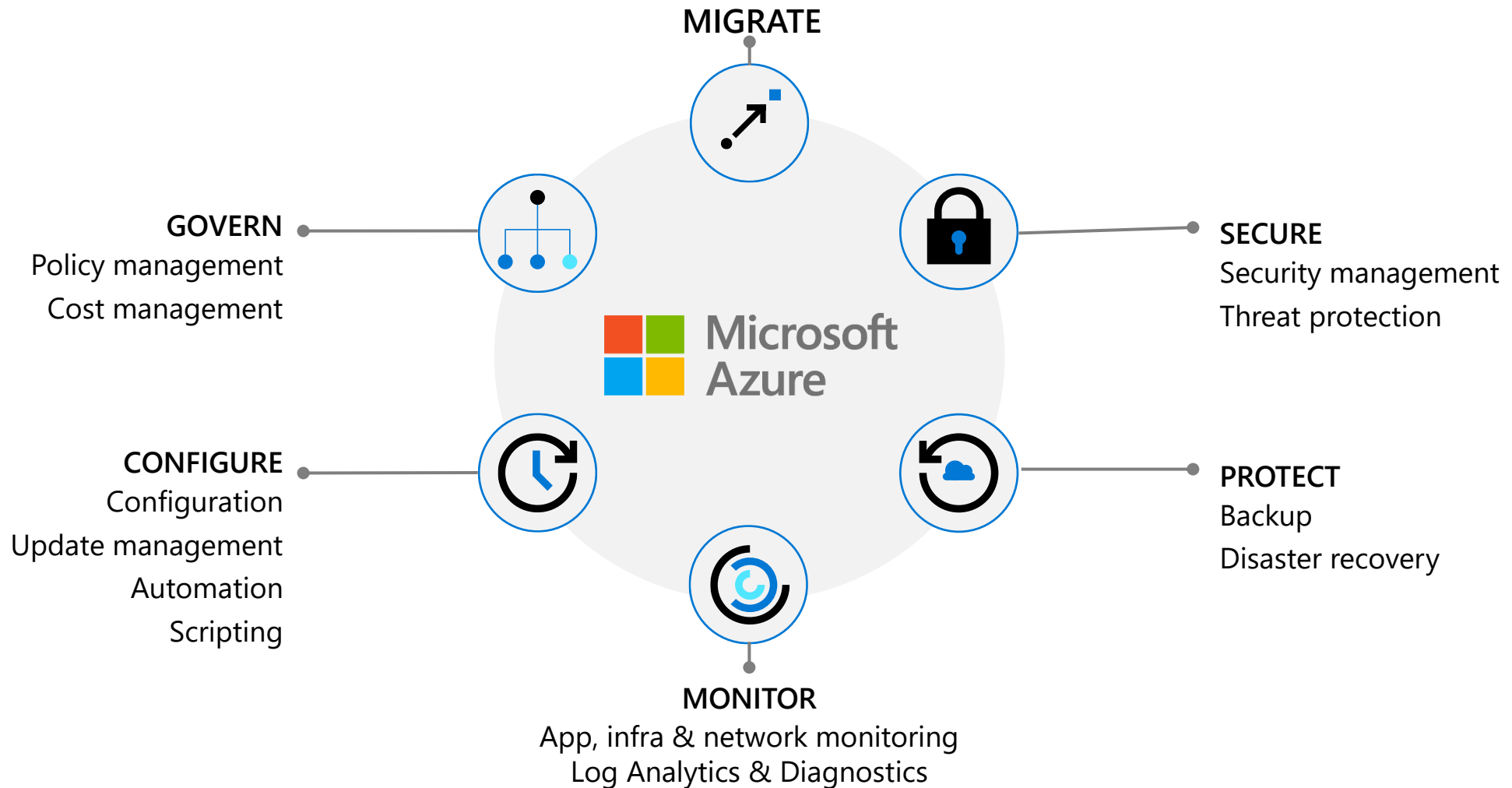
Module 01: Implement tools to track usage and flow



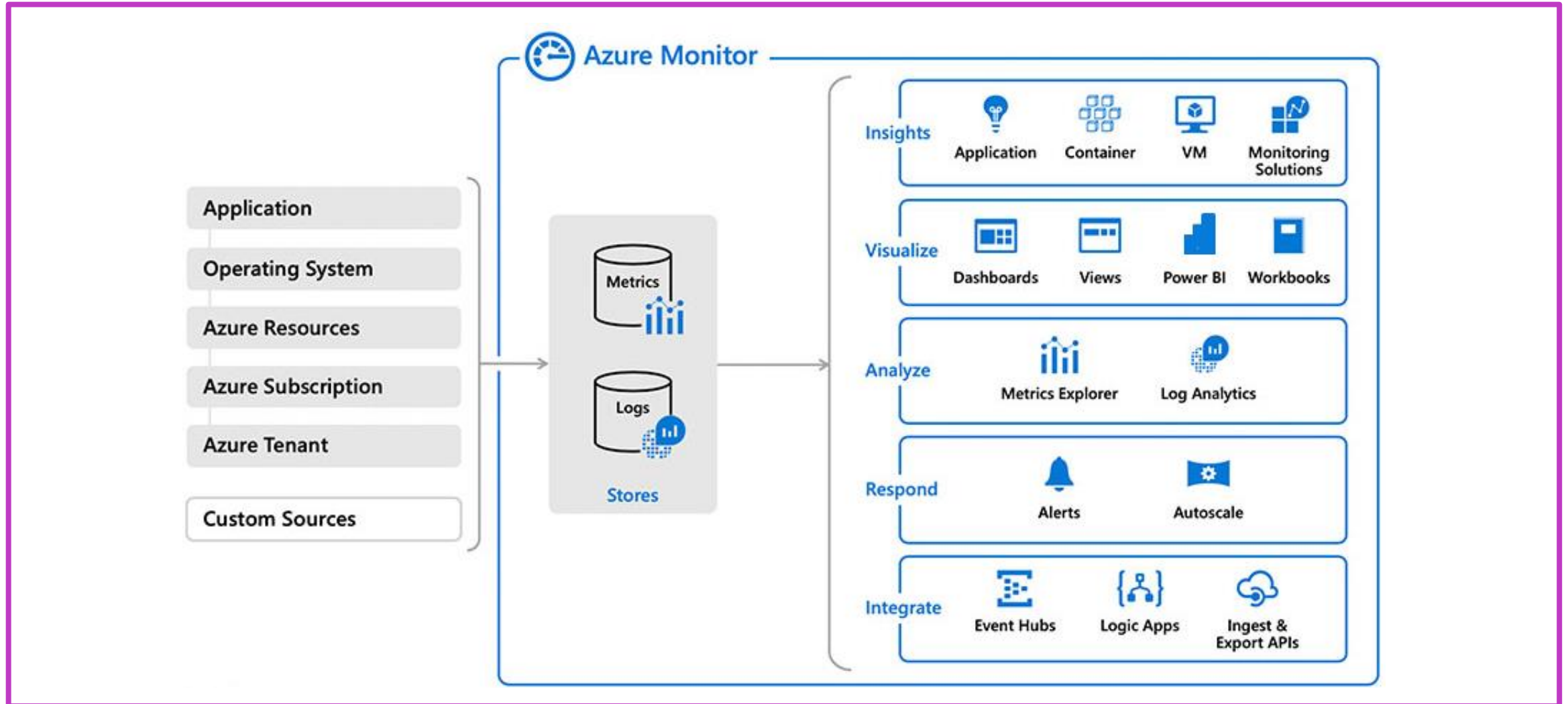
Understand the inner loop



Introduction to continuous monitoring



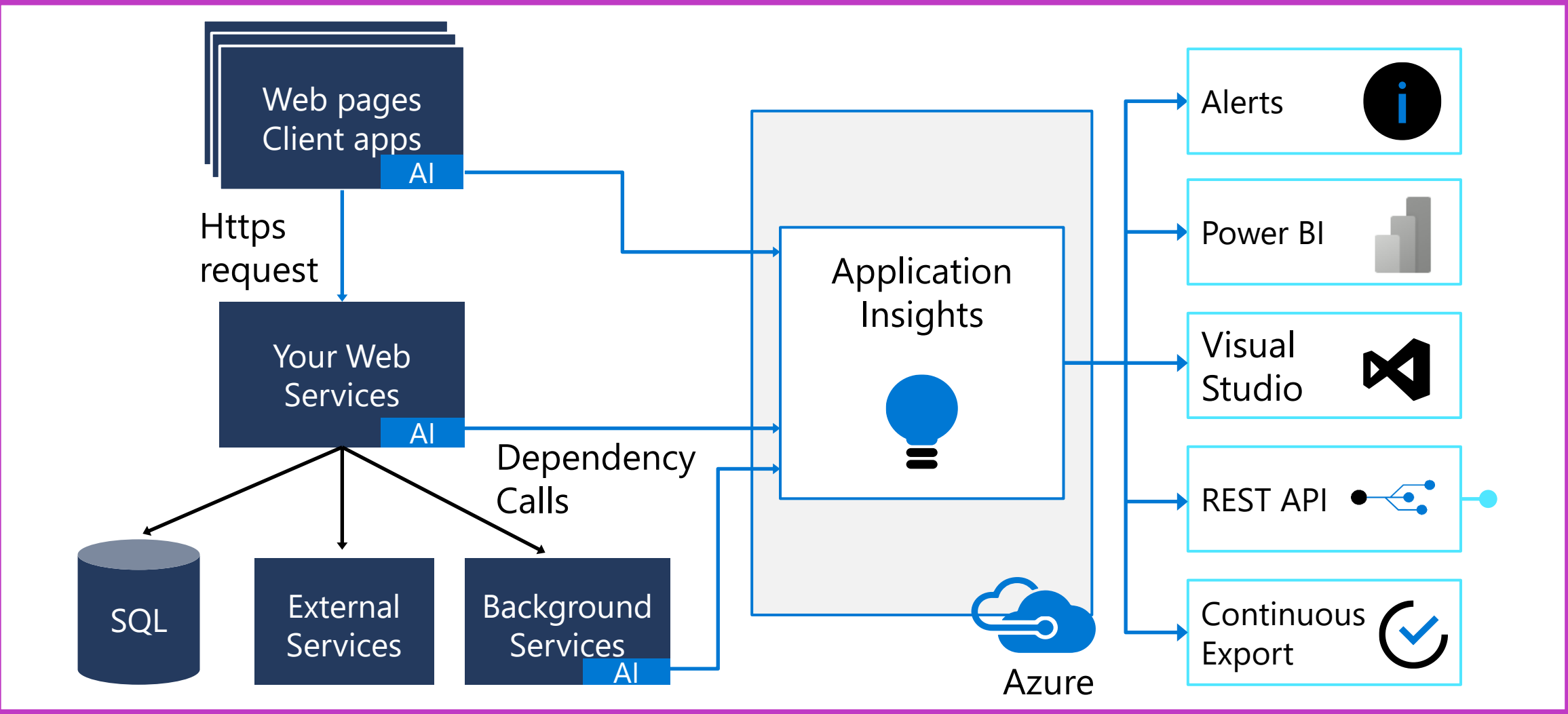
Explore Azure Monitor and Log Analytics



Examine Kusto Query Language (KQL)

- 1** KQL is the language used to query Log Analytics
- 2** Supports queries and control commands
- 3** Supported within Azure Data Explorer and Azure Data Studio (with notebook support)

Explore Application Insights



Implement Application Insights

- 1** Monitor
- 2** Detect, diagnose
- 3** Build, measure, learn

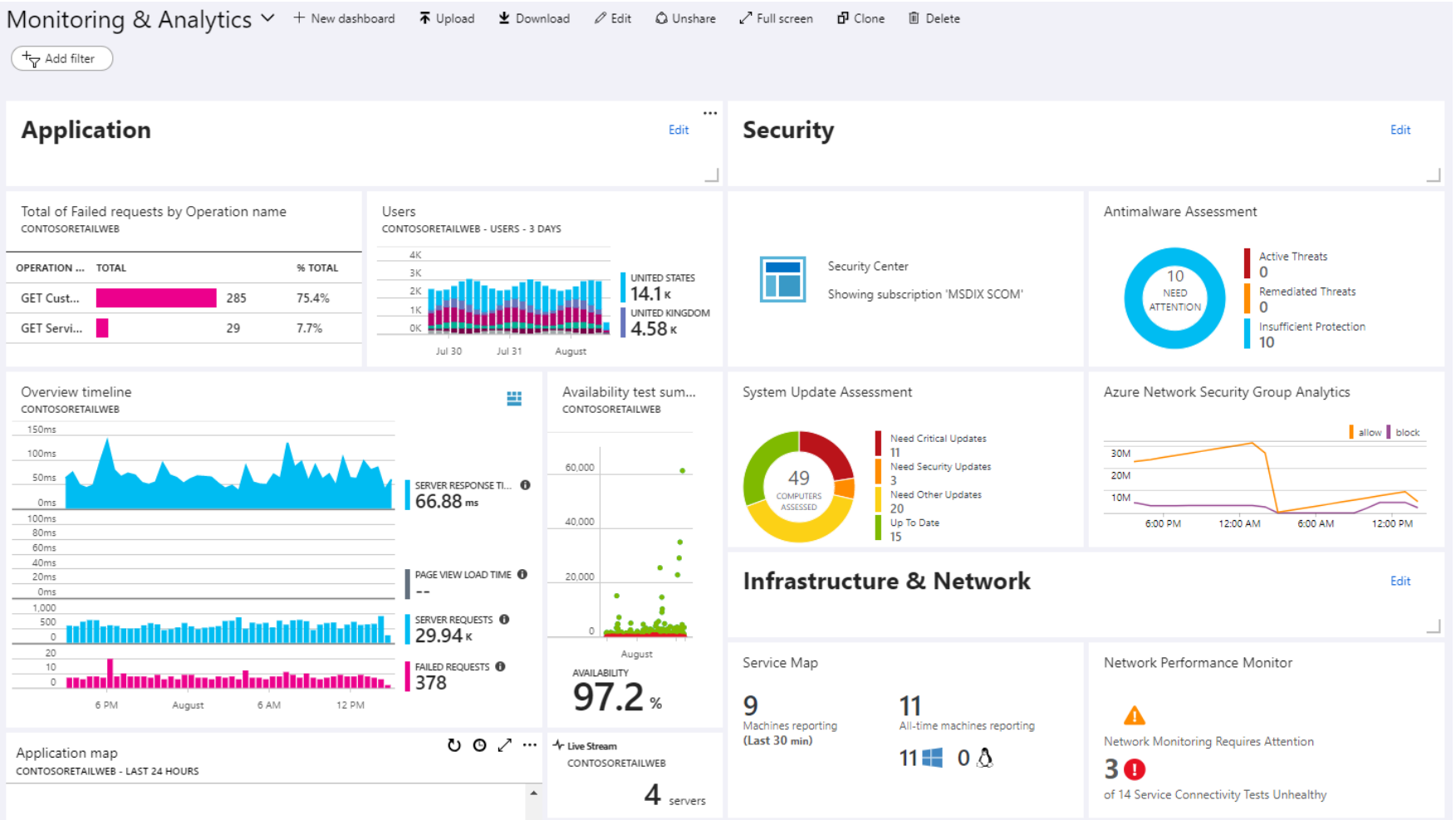
Demonstration: Add Application Insights to an ASP.NET core application

DEMO

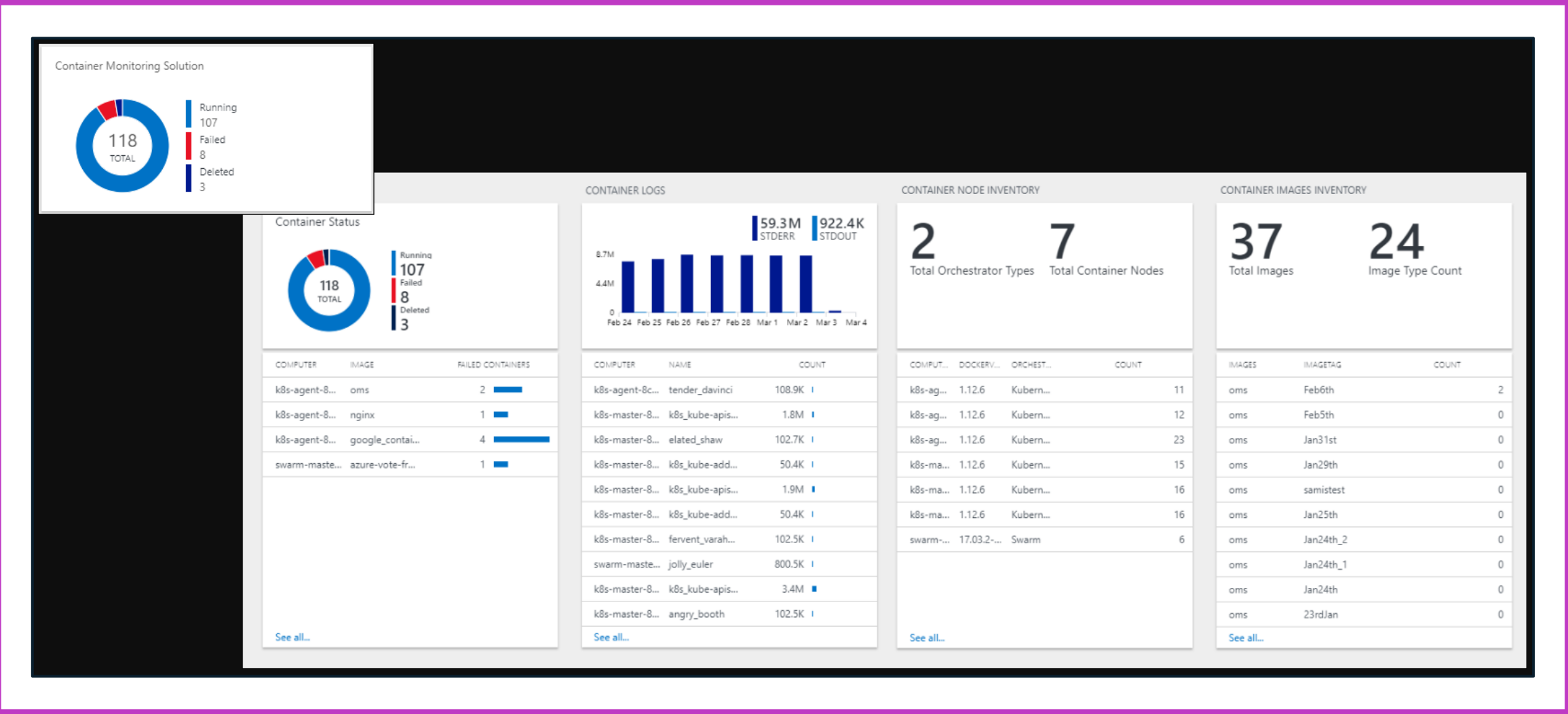
Module 02: Develop monitor and status dashboards



Explore Azure Dashboards



Examine view designer in Azure Monitor



Explore Azure Monitor workbooks

Analysis of Page Views

Page views correspond to user activity in your app. Understanding how your users interact with your pages will give you good insights into what is working in your app and what aspects need improvements.

This report will help

- Usage
- Time spent on p
- Time to first int
- Exit rates

If your telemetry do

Pages: Home Page, D

OptimizeCalculations

Usage

This section helps you understand how

Page N...	Unique Users	As % of
Overall	8813	
Home P...	8811	
Create	8792	
Details	8780	

- The As % of App Users/Sessions/V

Time Spent on Page

This report helps you understand the time spent by customers in your pages. Longer time spent pages usually indicates good engagement and is generally the desired behavior.

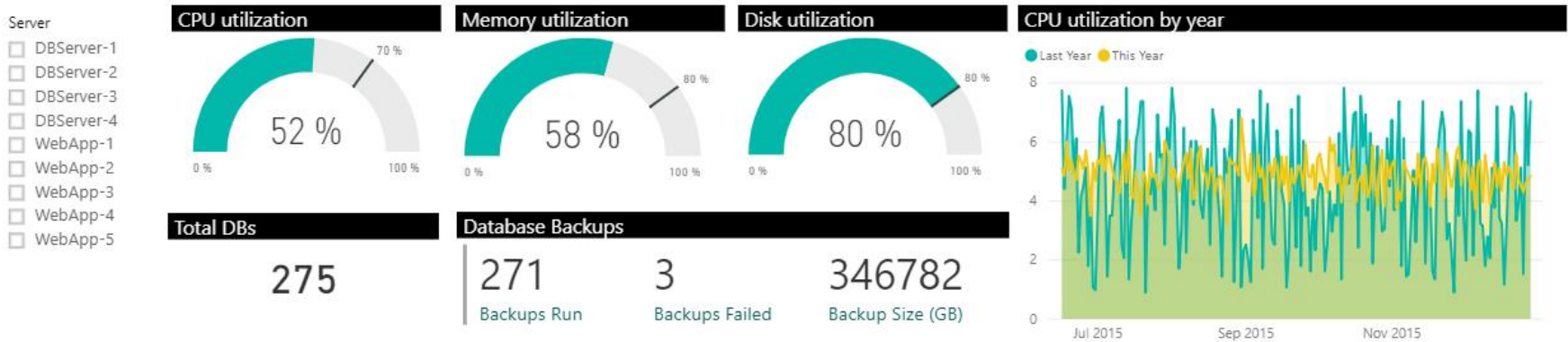
Ignore Durations Over: 3600s

Page Name	Sampled Page ...	Median (second...	75th Percentile (seconds)	90th Percentile ...	Mean (seconds)
Overall	107150	149.7	375	750.5	294.3
Create	29522	174.4	540	896.5	348.1
Details	33086	169.2	480	900	351.5
Home Page	44542	105.6	300	500	216.2

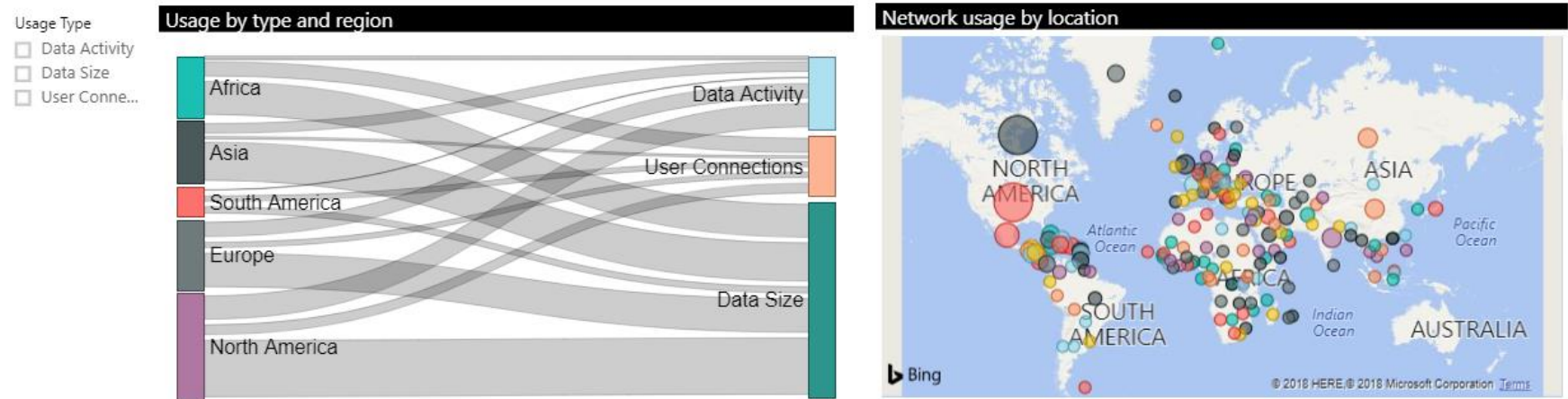
- The calculations may use sampling based on the `optimizeCalculationsFor` parameter.
- Time Spent on Page does not consider exit pages (last page of the session) in this calculations. The `Sampled Page Views` column may be fewer than the sampling count of 100000 because of this.

Explore Power BI

IT Operations



Resource usage and availability



Build your own custom application

1 Advantages:

- Complete flexibility
- Combine metrics and log data

2 Disadvantages:

- Significant engineering required

Module 03: Share knowledge within teams



Share acquired knowledge within development teams

- 1 Organizational knowledge is acquired over time
- 2 Organizational knowledge can easily be lost through staff turnover
- 3 Relearning old lessons is wasteful and expensive

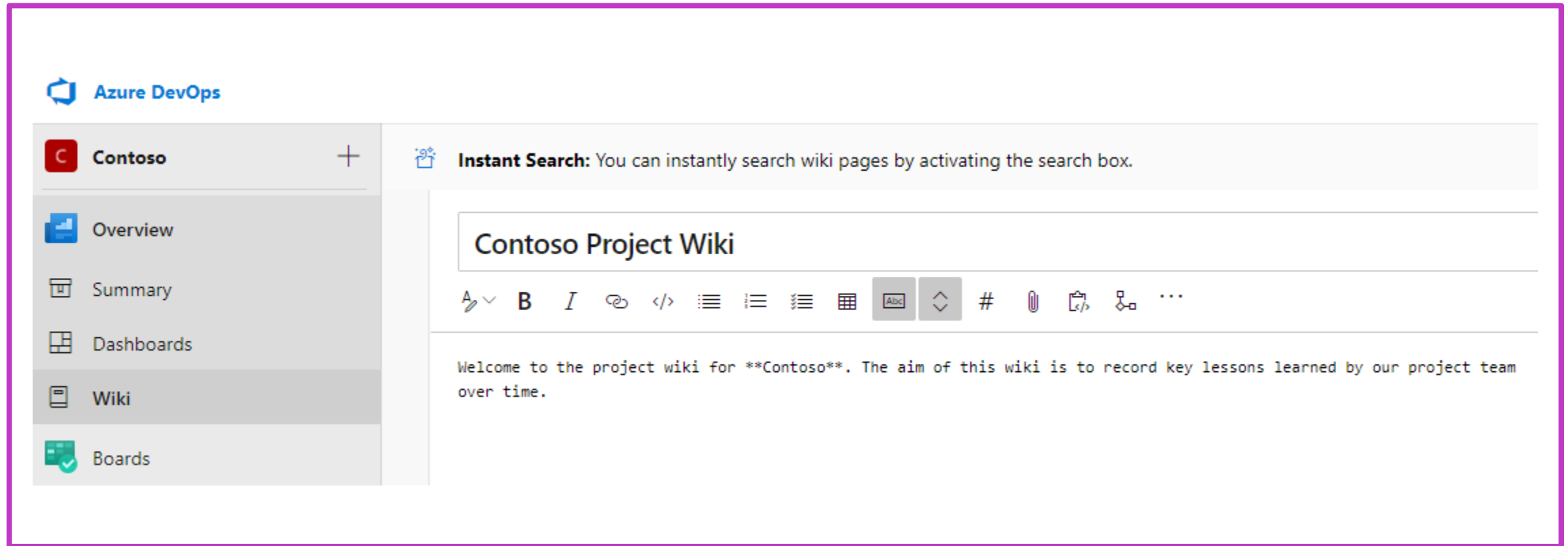
Discussion: Tools for knowledge sharing

Which knowledge sharing tools do you currently use?

What do you or don't you like about the tools?

Introduction to Azure DevOps project wikis

Created in Azure DevOps projects, stored in a repository



Wiki contents

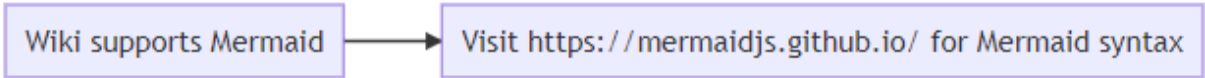
Written in markdown and can contain file attachments, including videos

Supports insertion of Mermaid diagrams

```
Welcome to the project wiki for Contoso. The aim of this wiki is to record key lessons learned by our project team over time.
```

```
::: mermaid
graph LR;
A[Wiki supports Mermaid] --> B[Visit https://mermaidjs.github.io/ for Mermaid syntax];
:::
```

Welcome to the project wiki for **Contoso**. The aim of this wiki is to record key lessons learned by our project team over time.



```
graph LR; A[Wiki supports Mermaid] --> B[Visit https://mermaidjs.github.io/ for Mermaid syntax];
```

Module 04: Design processes to automate application analytics



Explore rapid responses and augmented search

- 1** In Agile teams, issues that “slip through the cracks” directly impact end-users
- 2** Speedy resolution required – even if root cause determined later
- 3** Large volume of infrastructure and application logs to search
- 4** Augmented search uses semantic processing, statistical models, and ML

Integrate telemetry

Benefits:

- Provides detailed accurate insights on usage
- Monitors an object while physically far removed from it

Challenges:

- Do end users want it enabled?
- Users might not be comfortable with being monitored

Examine monitoring tools and technologies

- 1** Synthetic monitoring
- 2** Alert management
- 3** Deployment automation
- 4** Analytics

Explore IT Service Management Connector

Advantages:

- Create an incident or alert in your service desk solution based on alerts from Azure
- Sync incident data from service desk solution to Azure Log Analytics
- Correlate service desk data with Log Analytics data

Module 05: Manage alerts, Blameless retrospectives and a just culture



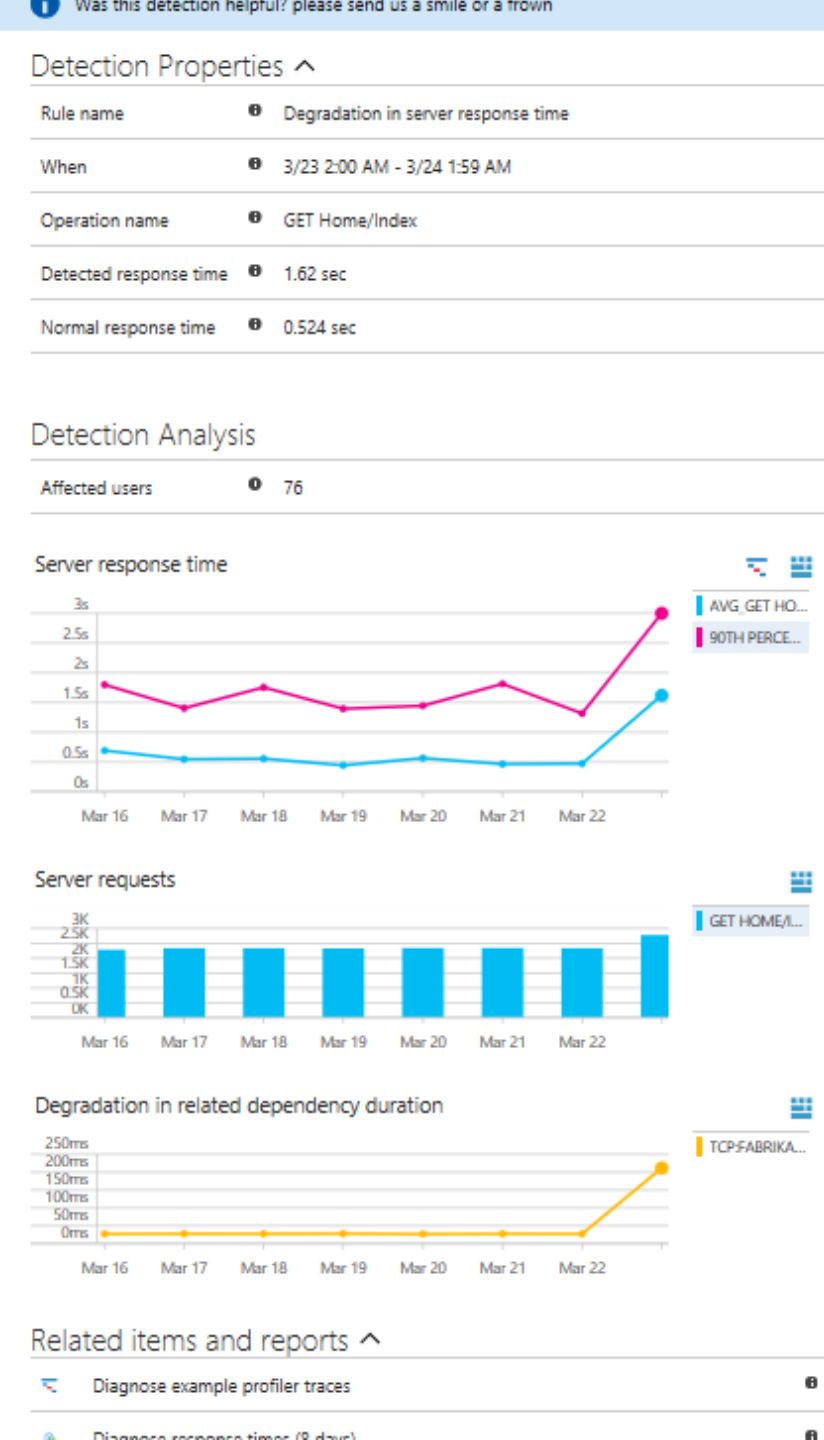
Examine when to get notifications



[Application Insights](#) automatically analyzes the performance of your web application and can warn you about potential problems

Explore how to fix it

1. Triage (how many users/ops are affected?)
2. Scope (all traffic or some pages?)
3. Diagnose (often will suggest the issue)



Explore smart detection notifications

fabrikamprod - Smart Detection settings

Application Insights

Search (Ctrl+/)

Overview

Activity log

Access control (IAM)

CONFIGURE

Getting started

Properties

Alerts

Smart Detection settings

Features + pricing

Refresh

	NAME	SEVERITY
	Slow page load time	Information
	Slow server response time	Information
	Long dependency duration	Information
	Degradation in server response time	Information
	Azure cloud service issues	Information
	Degradation in dependency duration	Information
	Failure Anomalies - fabrikamprod	Alert

© Copyright Microsoft Corporation. All rights reserved.

Improve performance

- 1** Triage
- 2** Diagnose slow page loads
- 3** Improve slow pages

Example: Understand server response time degradation

- 1 Response time compared to normal response time
- 2 Number of affected users
- 3 Average response time
- 4 Count of this operation requests on the day of detection and 7 days before
- 5 Correlation between degradation in this operation and degradations in related dependencies
- 6 Links to help diagnose the problem

Reduce meaningless and non-actionable alerts



Monitoring and alerting enables a system to tell us when it's broken, or, potentially, what's about to break



Alerts requesting immediate action should be urgent, important, actionable, and real

Discussion: Blameless retrospective

What does it mean to have a blameless retrospective?

Labs



Lab: Monitoring application performance with Application Insights



Lab overview:

In this lab, you'll learn about how you can add Application Insights to an existing web application, as well as how to monitor the application via the Azure portal.

Objectives:

- Deploy Azure App Service web apps
- Generate and monitor Azure web app application traffic by using Application Insights
- Investigate Azure web app performance by using Application Insights
- Track Azure web app usage by using Application Insights
- Create Azure web app alerts by using Application Insights

Duration:



Lab: Sharing team knowledge using Azure project wikis



Lab overview:

In this lab, you will create and configure wiki in an Azure DevOps, including managing markdown content and creating a Mermaid diagram.

Objectives:

- Create a wiki in an Azure Project
- Add and edit markdown
- Create a Mermaid diagram

Duration:



Learning Path review and takeaways



What did you learn?

- 1** Develop monitoring and status dashboards
- 2** Implement tools to track system usage, feature usage, and flow
- 3** Integrate and configure ticketing systems with development team's work management
- 4** Design processes to automate application analytics
- 5** Manage alerts and reduce meaningless and non-actionable alerts
- 6** Carry out blameless retrospectives and create a just culture

Learning Path review questions

- 1 Does Azure Monitor allow you to create alerts from log queries?
- 2 What features are provided by Azure Monitor?
- 3 What query language can you use to query Azure Log Analytics?
- 4 What platform integrations does Azure Monitor provide to visualize your logs in real time?
- 5 True or False: Application Insights analyses the traffic from your website against historic trends and sends you smart detection notifications on degradation.

