







API-AZI219

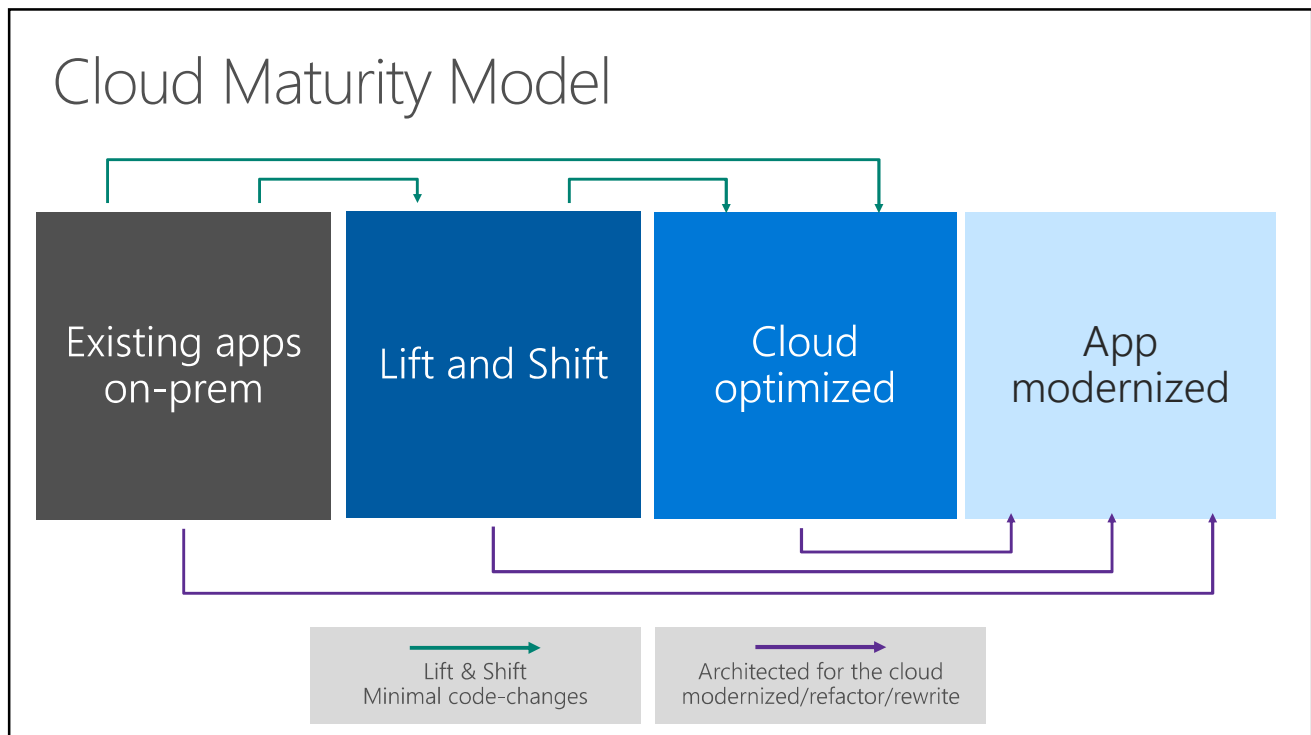
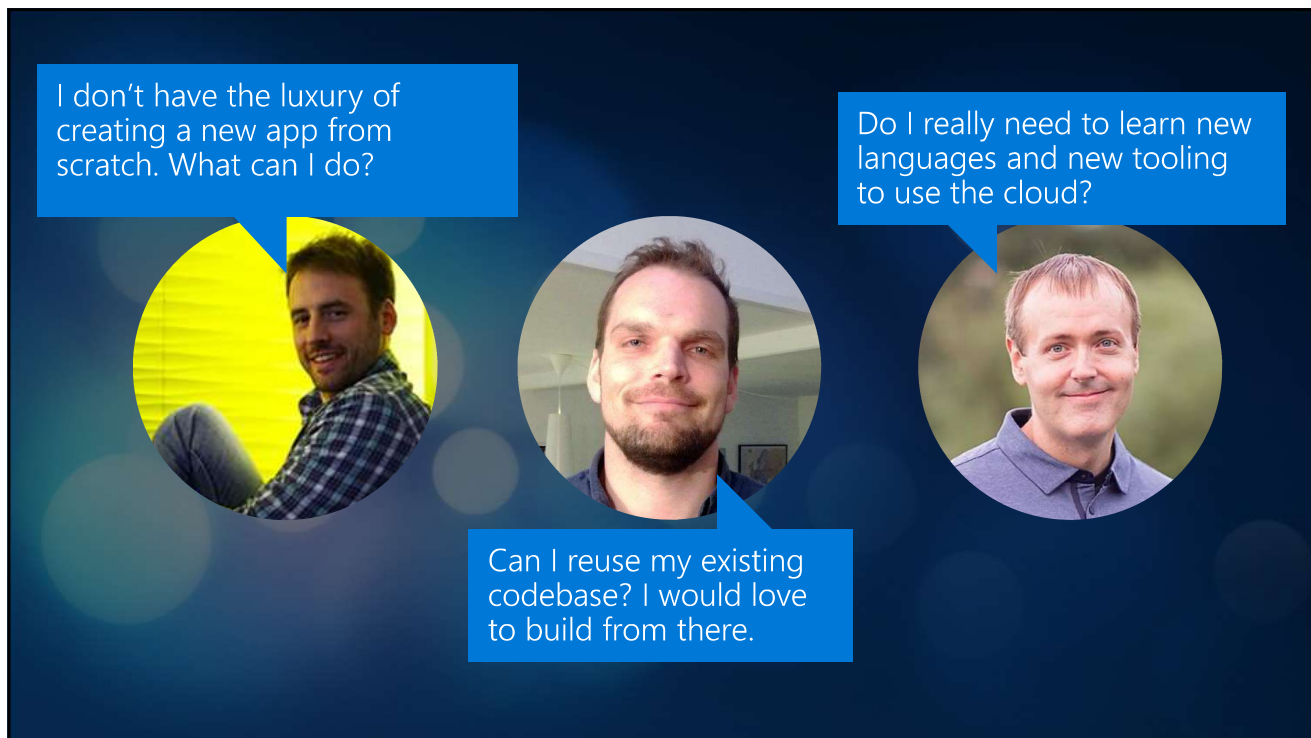
Application Modernization on Azure

Mayur Shintre
Principal Program Manager
Commercial Software Engineering
Microsoft

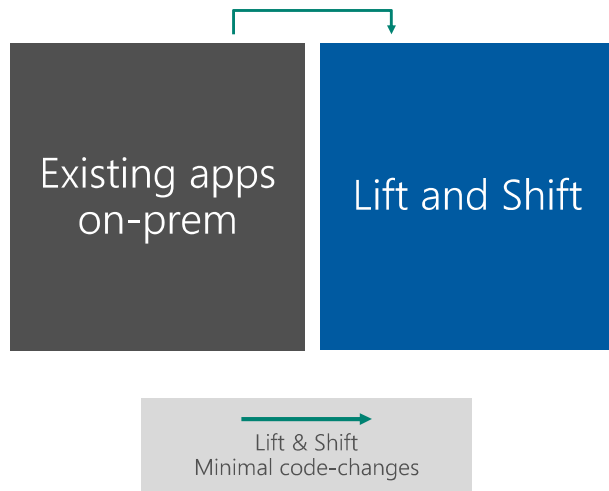
@mayurshintre

Value to Optimize and Modernize

Agility	Time to market	Total cost of ownership	IT simplification
Continuous delivery Containers	Scalability and HA Insights and Analytics	Infrastructure Cost Ongoing Maintenance	Standardization Simplification
			



Cloud Maturity Model



Lift and Shift

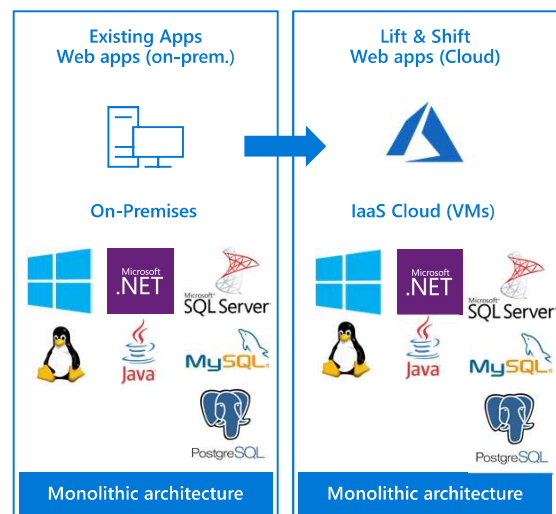
Migrate your on-premise application to IaaS on Azure

Pros

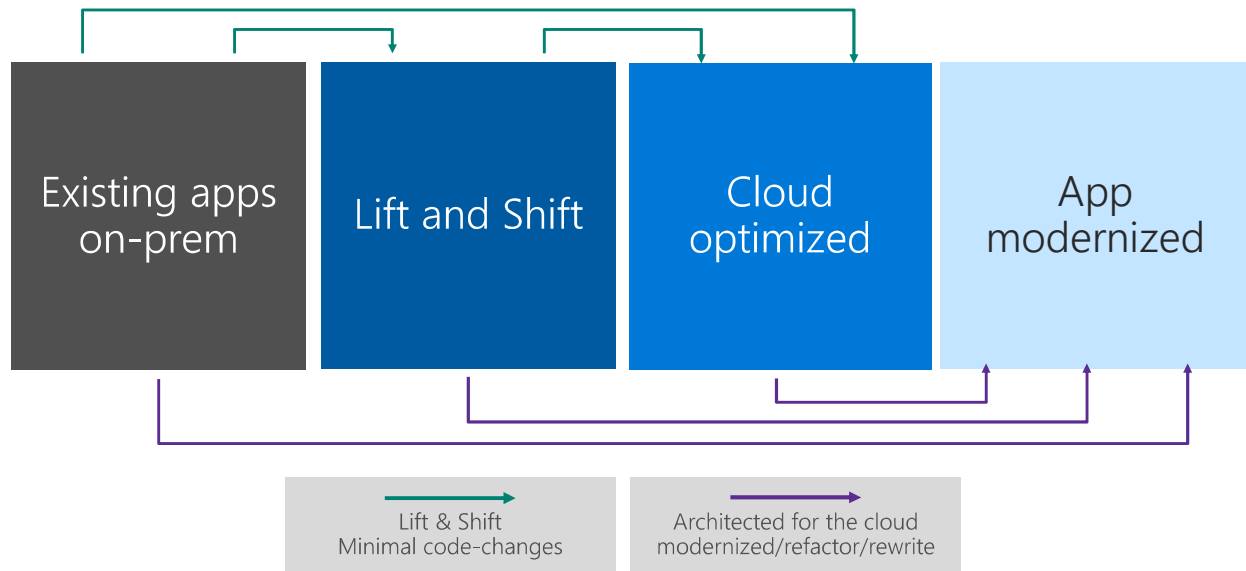
- No re-architect or new code
- Least effort for quick migration
- Supported on the least common denominator

Cons

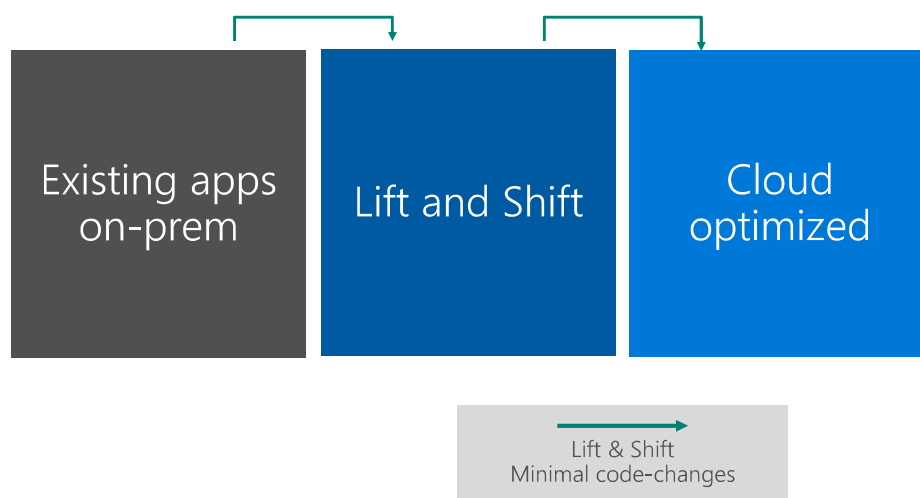
- Smaller cloud value
- Manual patching, upgrades
- No automated app scaling and high availability



Cloud Maturity Model



Cloud Maturity Model



Cloud Optimized

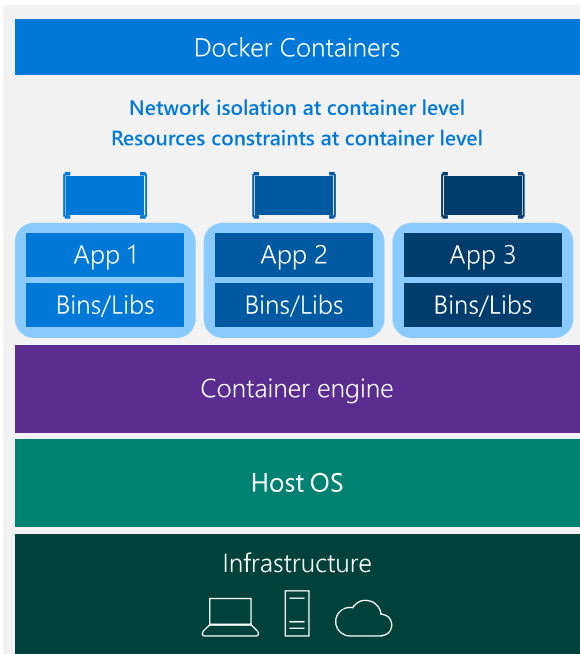
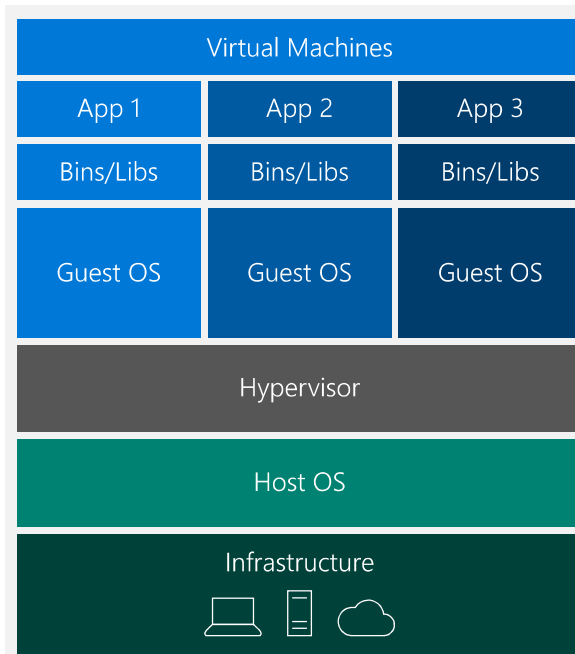
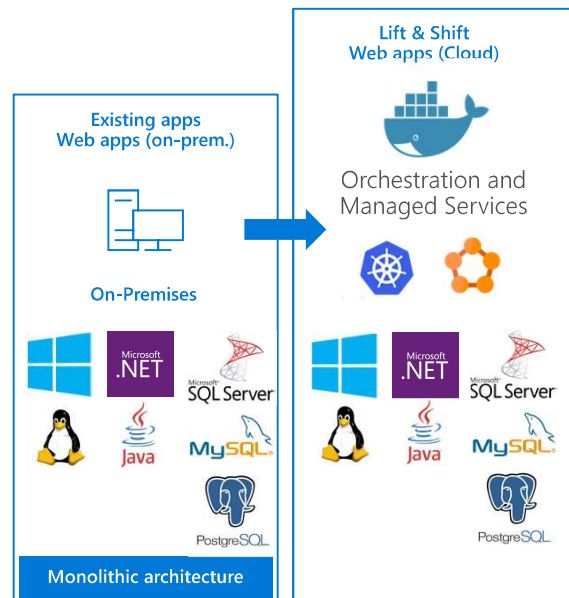
Get more Cloud benefit by Containerizing your app

Pros

- No re-architect or new code
- Increased density & lower deployment cost
- Improved productivity and DevOps agility
- Portability of apps and dependencies
- High availability with Orchestrators

Cons

- Containerization is an additional step



Deploy containers everywhere in Azure



ACI



AKS



Service Fabric



Web Apps



Batch



Azure Container Instances

Deploy containers with a single command

Launch container instances in seconds

Cost effective per second billing

ACI Bridge for Kubernetes



No infrastructure to manage

Fast start and stop

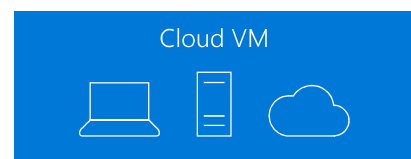
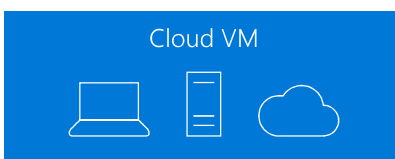
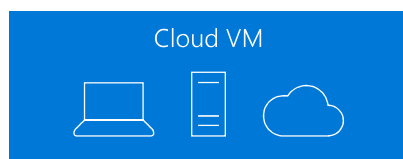
Fast scale without VM start-up



Controlled placement and HA

Effective resource utilization

Application co-location



Container orchestration with infinite scale



Kubernetes provides rich orchestration capabilities



Azure Container Instances provides infinite container-based scale



The ACI Connector for Kubernetes brings them together
(Newly released as the Virtual Kubelet)

Making Kubernetes Easy to Use



Open Service Broker for Azure

- Easily connect Kubernetes apps to Azure services
 - Azure Database for MySQL
 - Azure Database for PostgreSQL
 - Azure SQL
 - Azure CosmosDB
 - Azure Redis
 - Azure Container Instances
 - Azure Service Bus
 - Azure Storage
 - More to come...
- Built on an open standard
- Integrated with Helm



Service Fabric Products

 Service Fabric Standalone

Public preview (@May 2018)

- Simplified SF application model
- Multi Instance container support (Windows and Linux)
- Built in Service Mesh (Envoy)
- ^{On-premises} Service Fabric Storage volume driver

 Azure Service Fabric

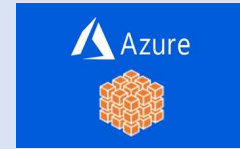


App, Cluster, Hardware Control
Customer manages hardware, OS patching, runtime upgrades, cluster capacity & application deployment

App & Cluster Control
Azure automates VMs, OS patching, runtime upgrades
Customer manages application deployment and cluster capacity

Announcing

 Service Fabric Applications



App Control

Azure manages infrastructure (VMs, runtime upgrades etc)
Customer manages application.
Micro billing (CPU/sec, Mem/sec, etc).

SF Applications 1st private preview open, signup @ <http://aka.ms/seabreezepreview>



Visual Studio 2017 + Containers

Integrated Docker tooling
Develop, debug, test, deploy
End-to-end DevOps support

The screenshot shows the Visual Studio 2017 interface with the Docker extension. The Dockerfile is visible on the left, and the Solution Explorer on the right shows the project structure. The Output window at the bottom displays the command to pull the Docker image.

Great Partner Solutions



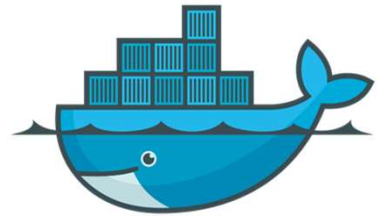
DC/OS



Pivotal

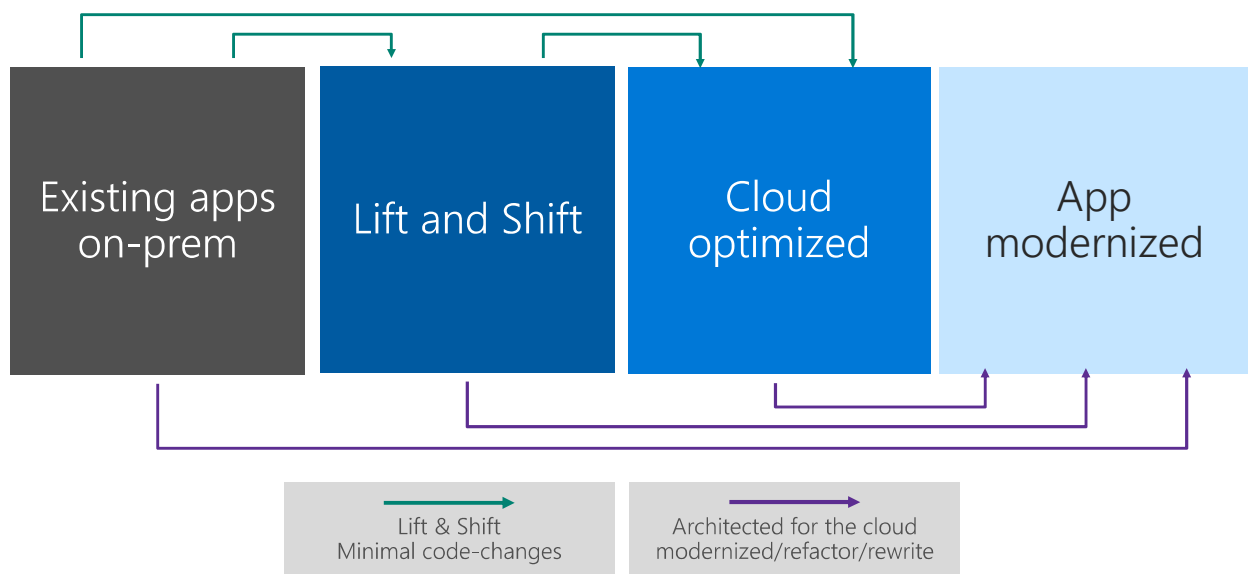


OpenShift

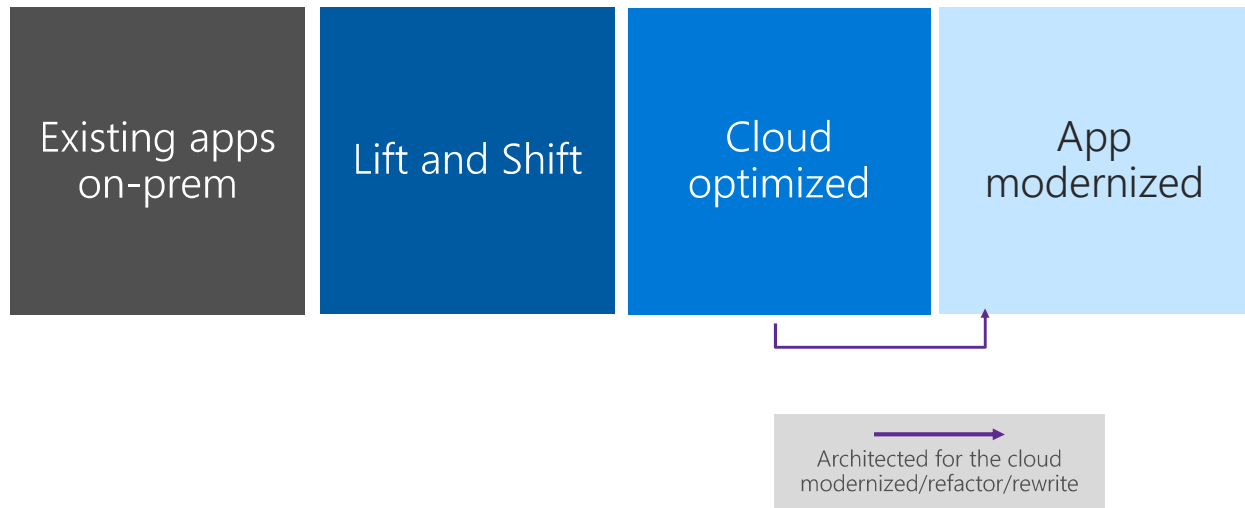


Docker Enterprise

Cloud Maturity Model



Cloud Maturity Model



App Modernized

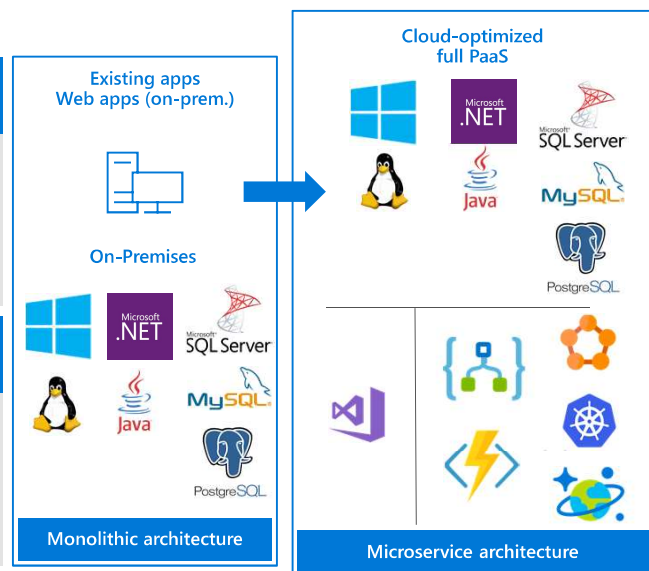
Extend your apps with new cloud native services including serverless, microservices and PaaS.

Pros

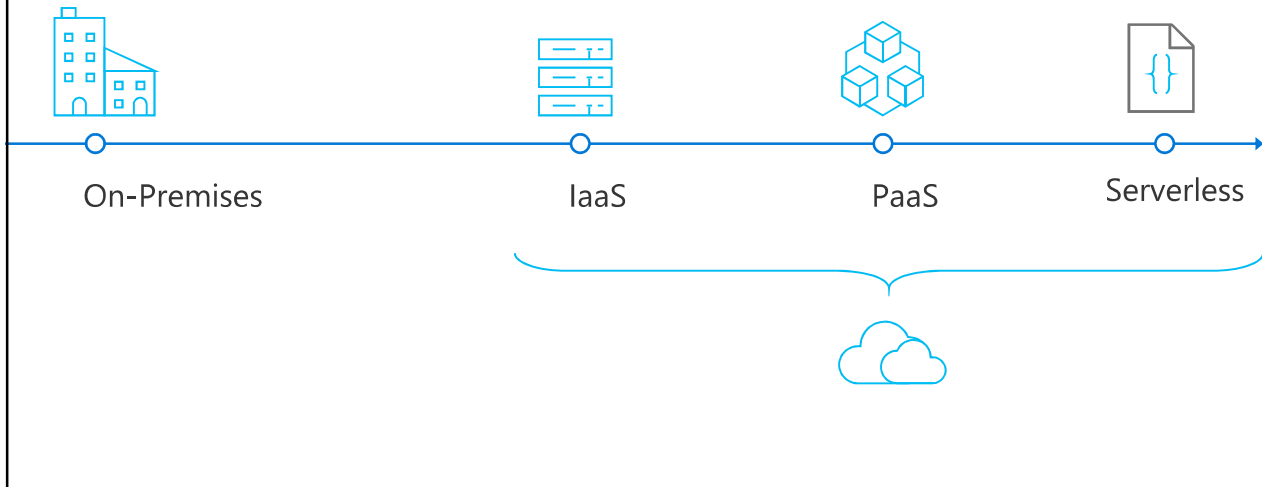
- Optimized for long term agility
- Optimized for scale and high availability
- Modern Architecture with Microservices

Cons

- Requires significant code refactoring or rewriting



The “evolution” of application platforms



Serverless: Build Apps Faster



Manage apps
not servers

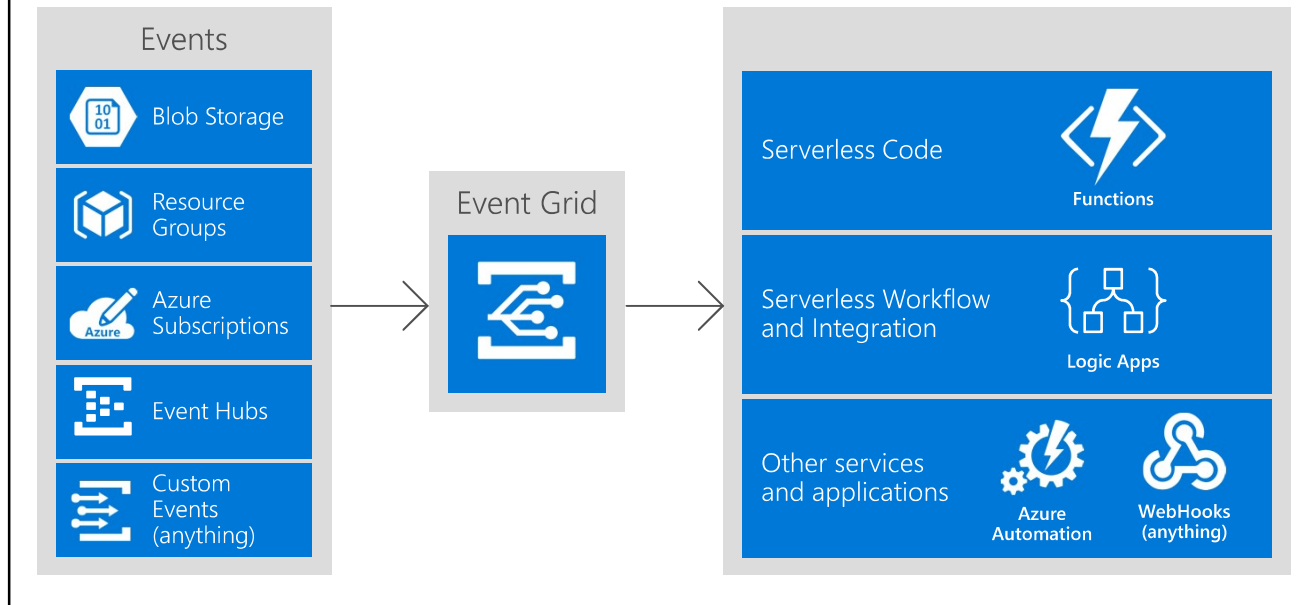


Reduced
DevOps

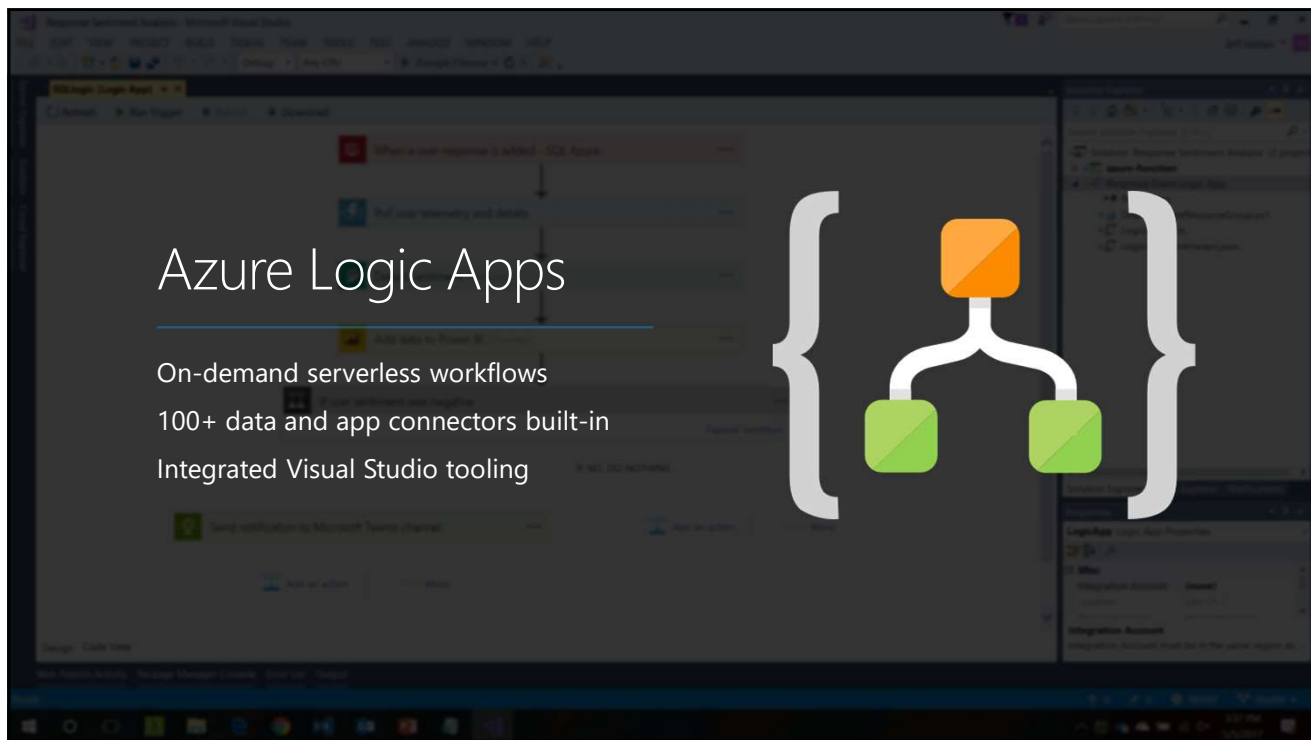


Faster time-
to-market

Microsoft Azure Serverless Platform

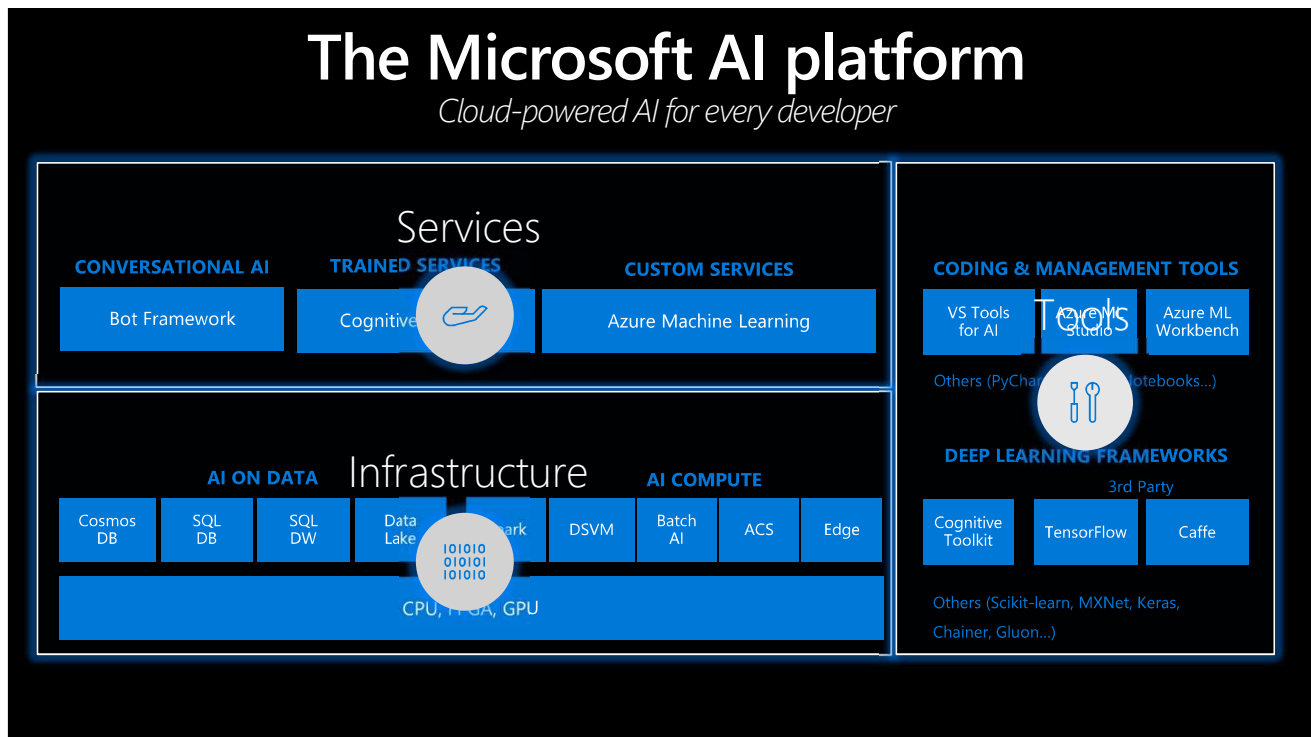


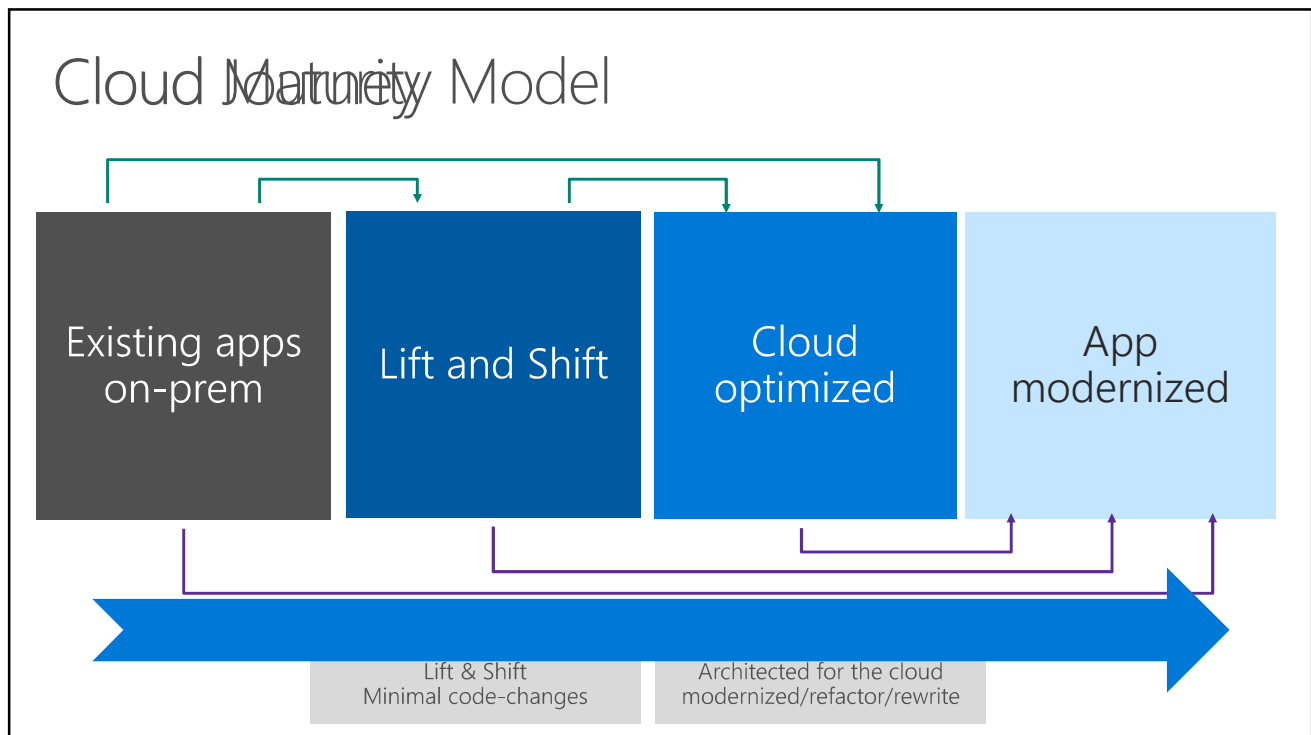
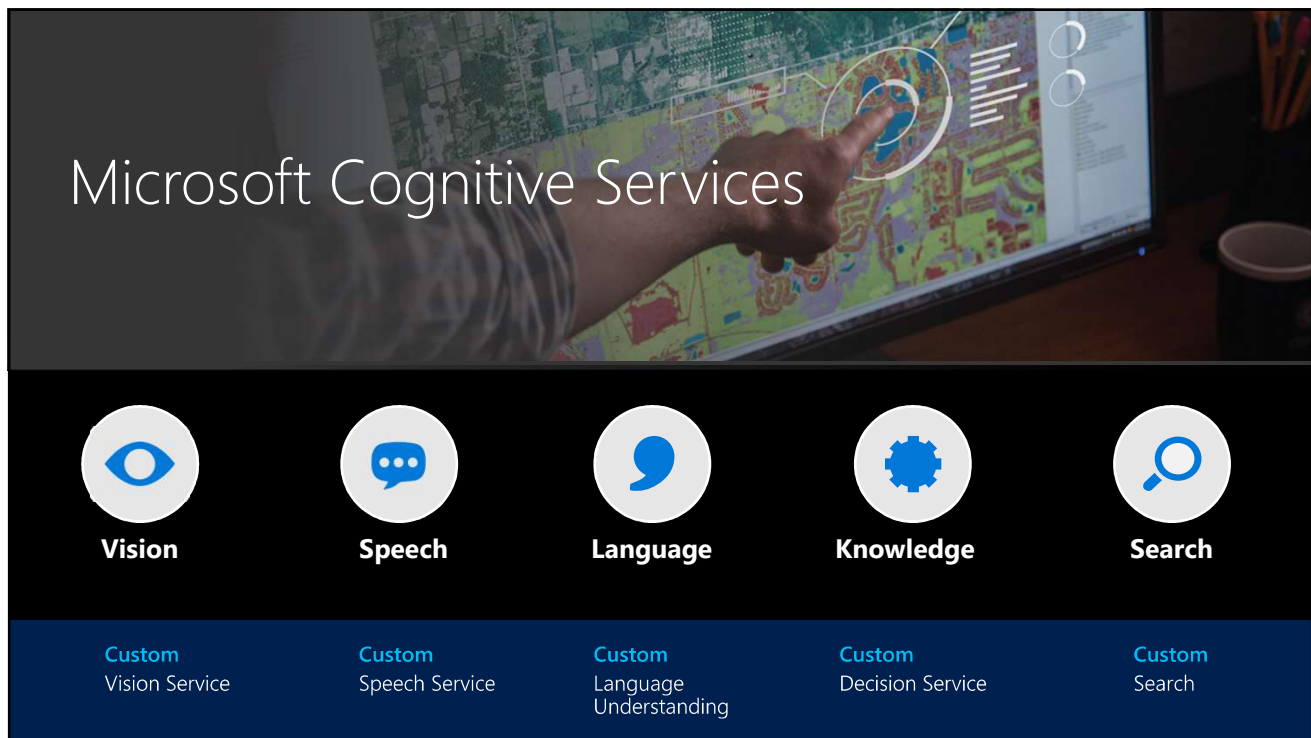
The screenshot shows the Visual Studio IDE with a C# code file for an Azure Function. The code defines a function that reverses a string. A large yellow lightning bolt icon is overlaid on the code. To the right of the code, the text "Azure Functions" is displayed. Below this, the text "On-demand serverless code" is shown. Further down, the text "Develop, debug, deploy in Visual Studio & Visual Studio for Mac" and "CI/CD support in Visual Studio Team Services and Github" are listed. On the right side of the screenshot, a console window is visible, showing logs for the function's execution, including the function name, trigger, and execution time.



Azure Logic Apps

On-demand serverless workflows
100+ data and app connectors built-in
Integrated Visual Studio tooling

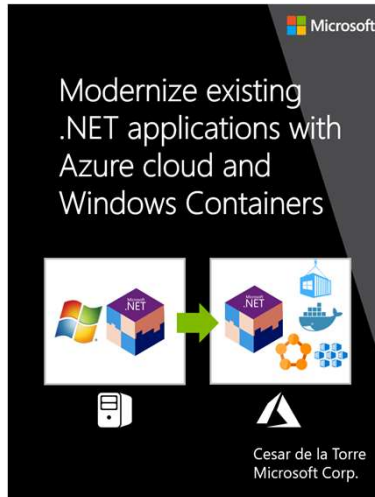




Free eBooks



<https://aka.ms/azuredevebook>



<https://aka.ms/modernizeappebook>



<https://aka.ms/microservicesebook>

Thank you

MICROSOFT CONFIDENTIAL – INTERNAL ONLY