



APP MODERNISATION DAY

09/04/2019 — MICROSOFT HOUSE, MILANO

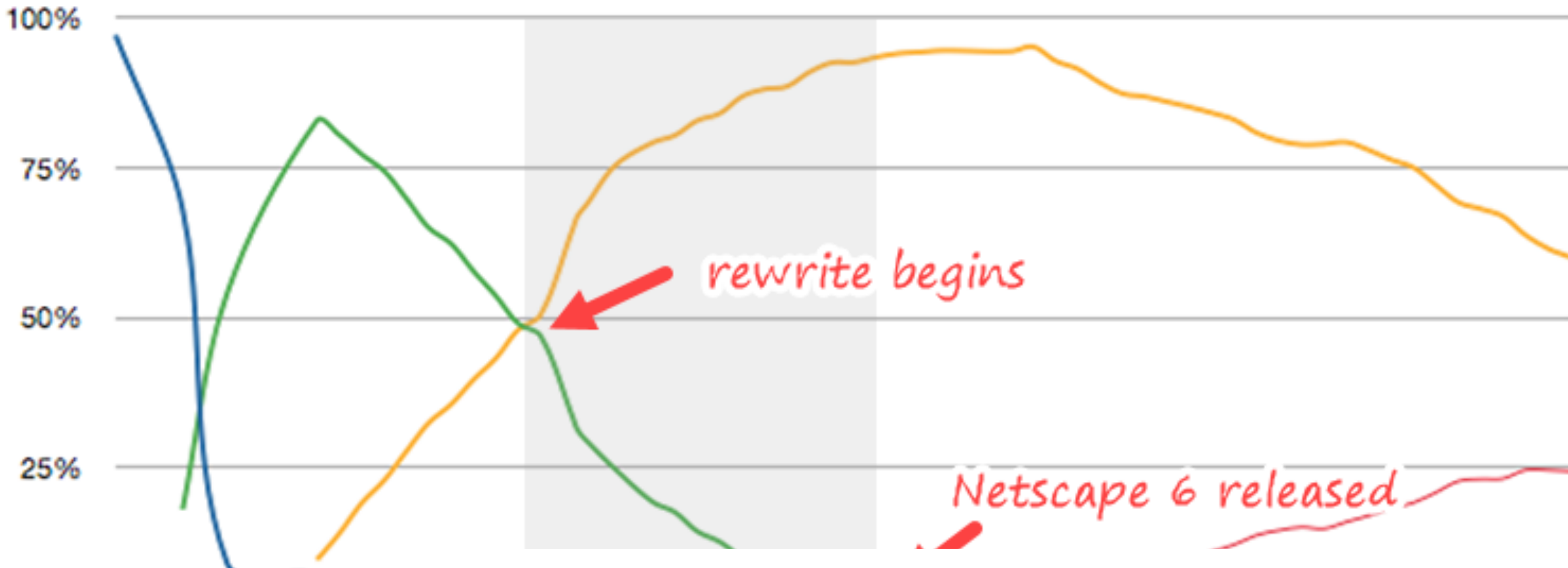
The path to the future...



Lorenzo Barbieri
Cloud Solutions Architect
@_geniodelmale
[linkedin.com/in/geniodelmale](https://www.linkedin.com/in/geniodelmale)

#APPMODERNISATION

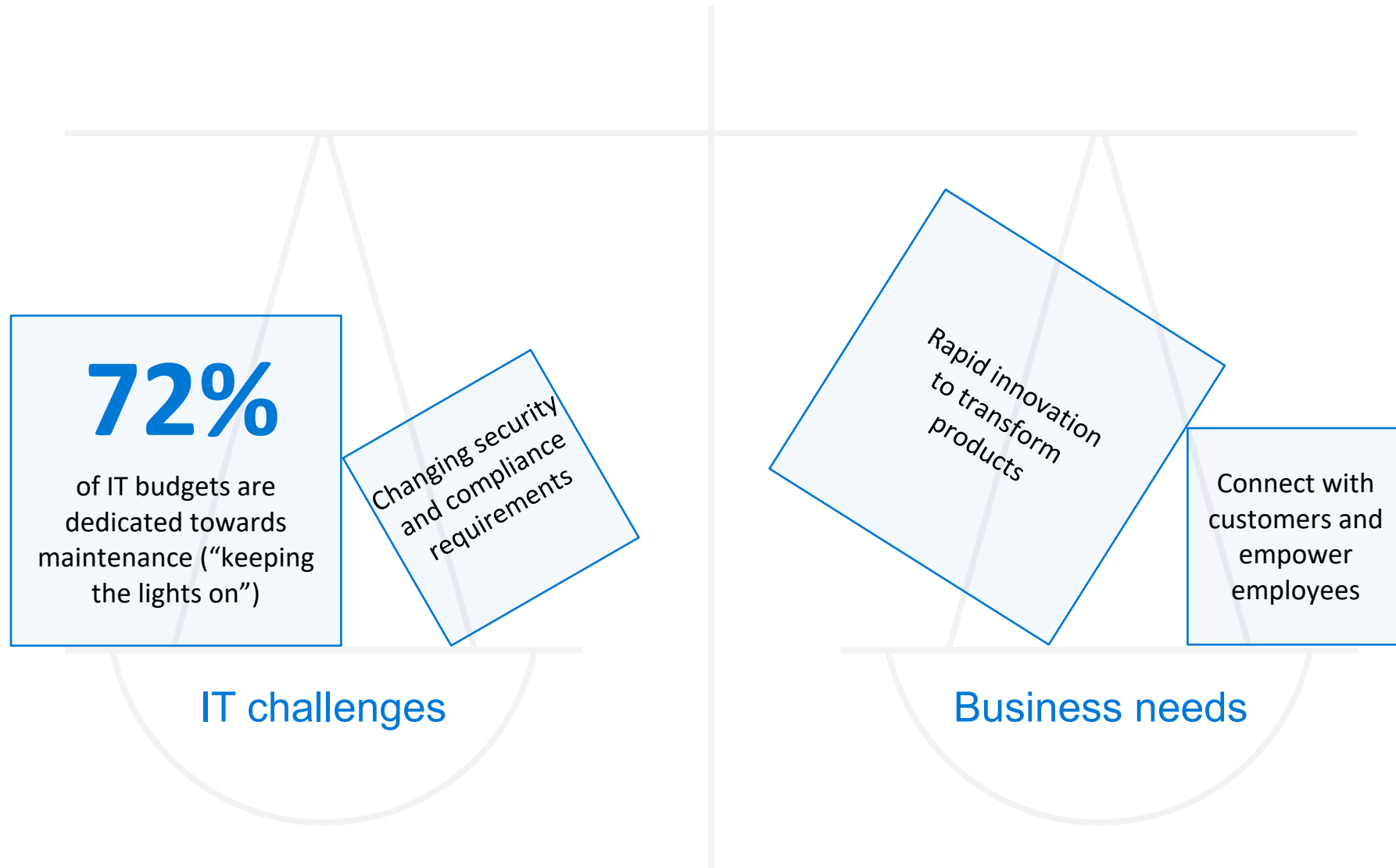
#UGIDOTNET



Never rewrite! (Almost...)

- <https://www.joelonsoftware.com/2000/04/06/things-you-should-never-do-part-i/>
- <https://medium.com/@herbcaudill/lessons-from-6-software-rewrite-stories-635e4c8f7c22>
- The history repeated some years later...

Balancing IT and business





App
Modernization
to the rescue!

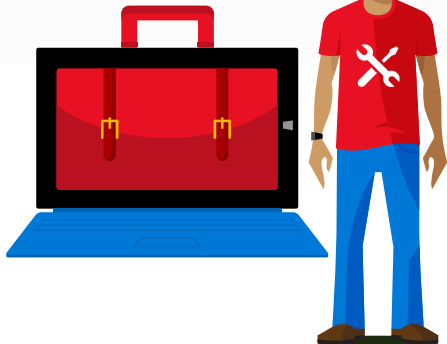
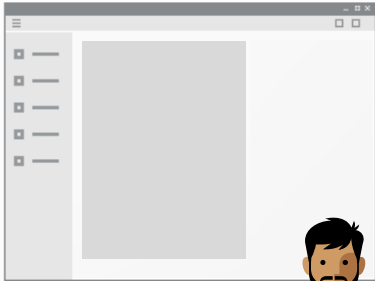


Web/API
or
Desktop?

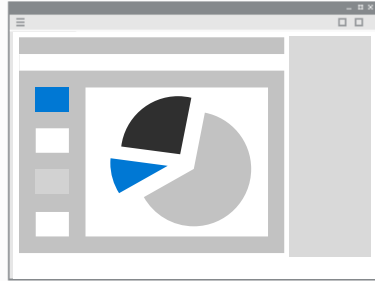
Current investments on Desktop Apps



WinForms



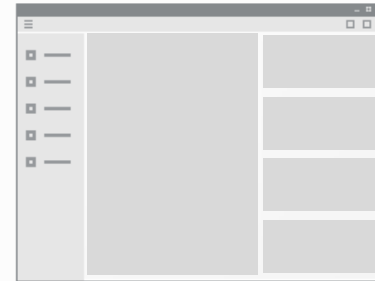
WPF



UWP



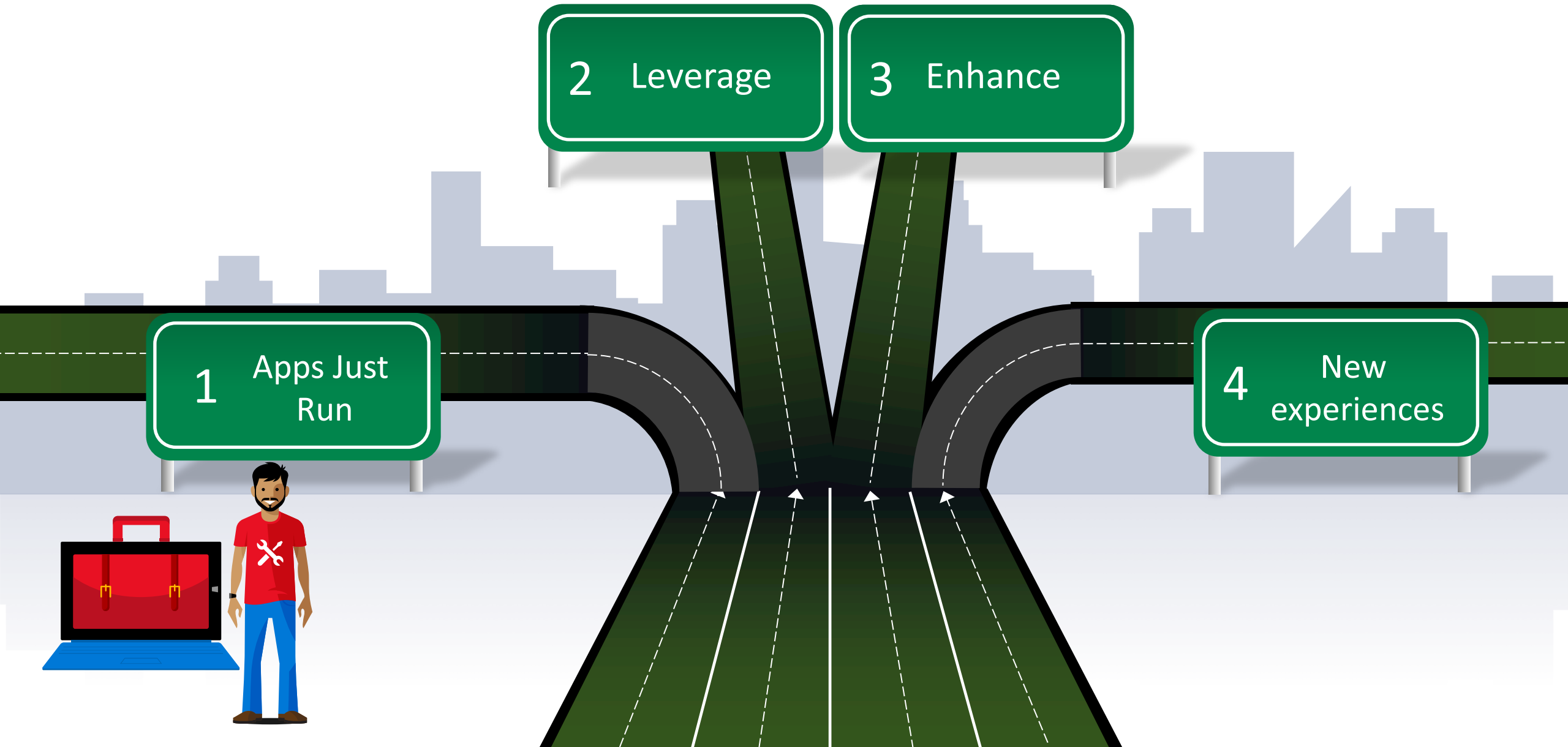
Web



Others



Multiple Journeys to help evolve Desktop Apps

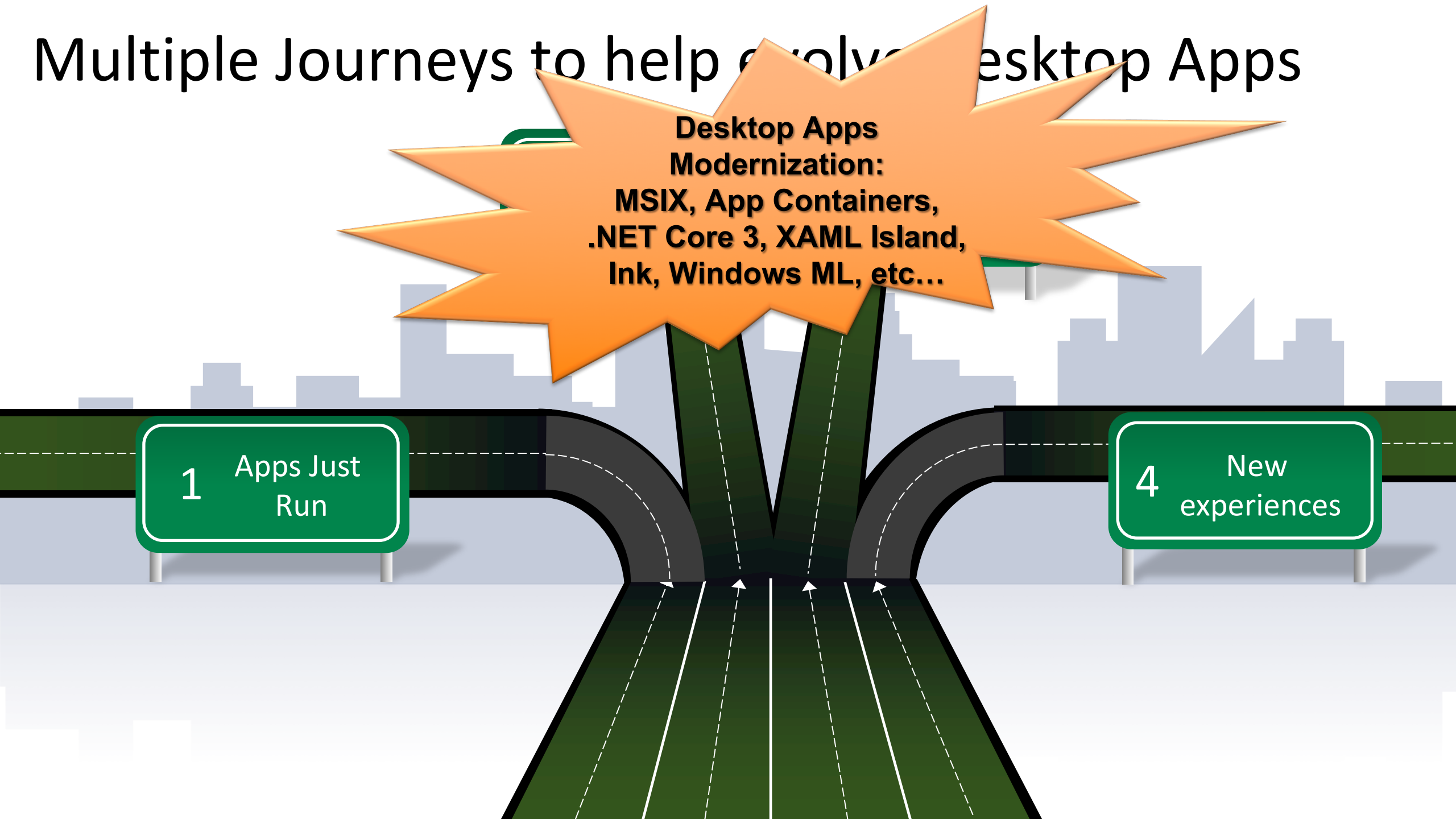


Multiple Journeys to help evolve Desktop Apps

**Desktop Apps
Modernization:
MSIX, App Containers,
.NET Core 3, XAML Island,
Ink, Windows ML, etc...**

1 Apps Just
Run

4 New
experiences





Web/API

Application layers

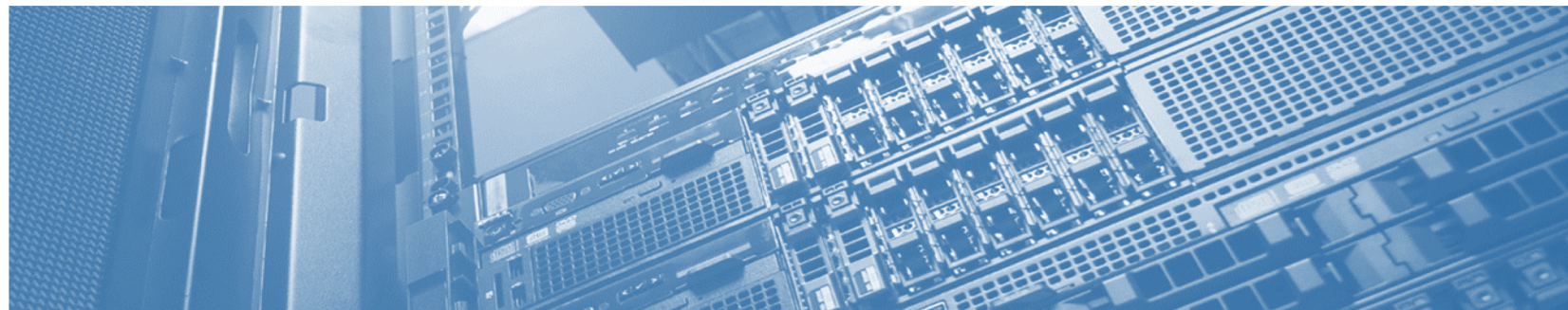
Application



Data + Intelligence

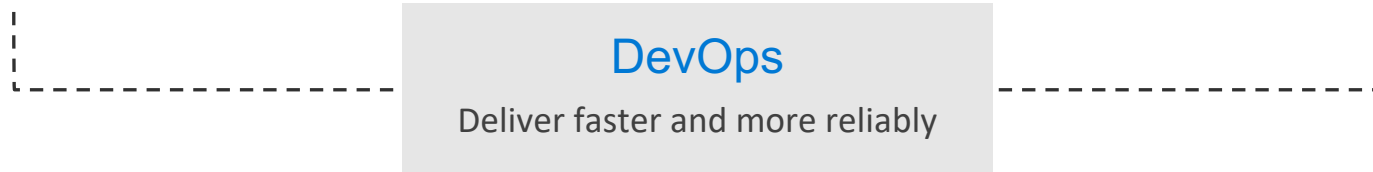
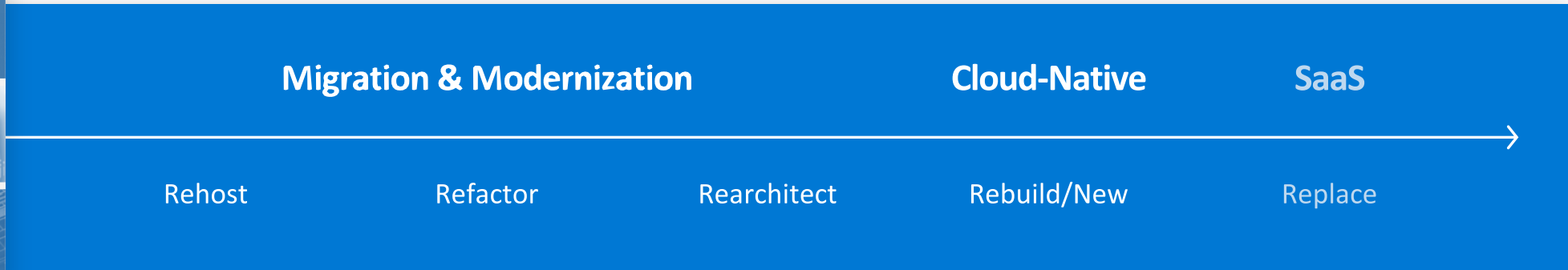
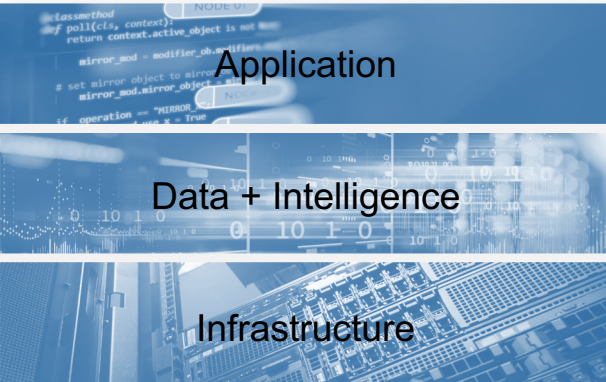


Infrastructure

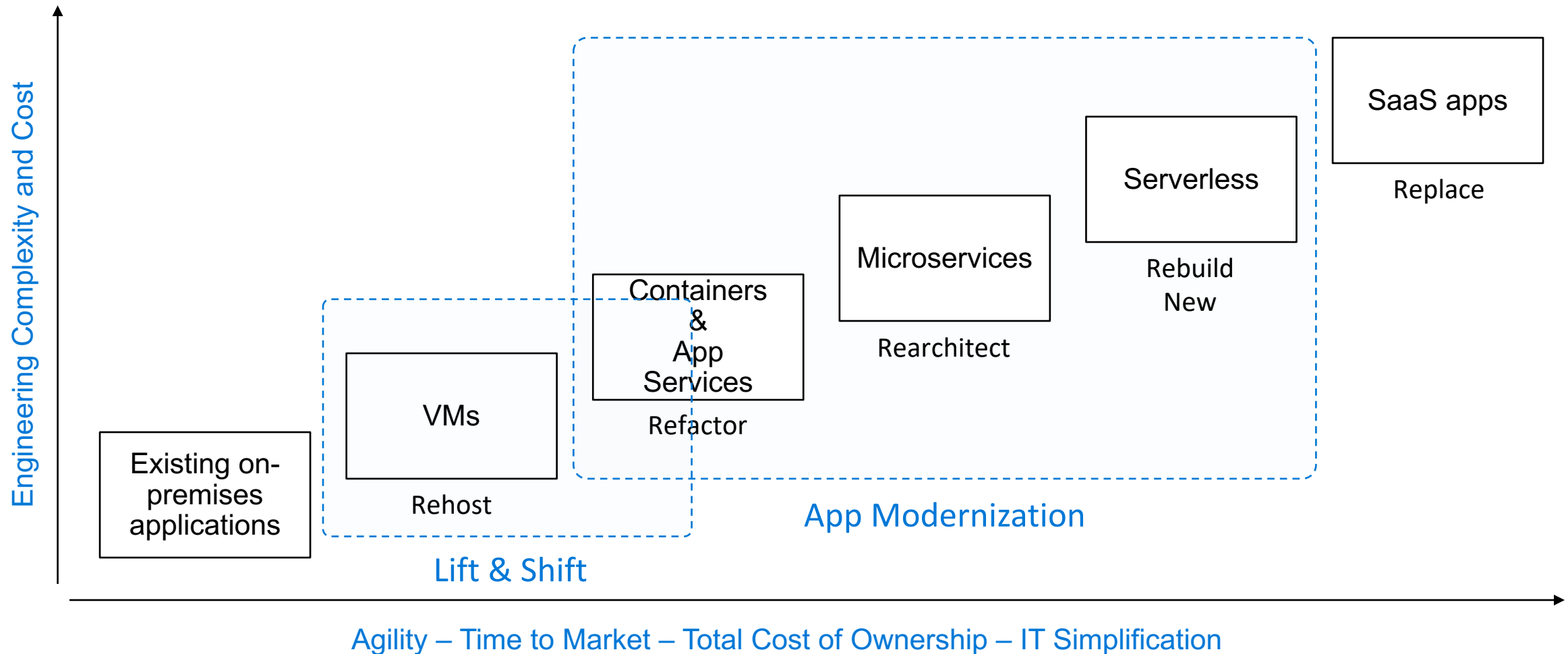


The journey to the (hybrid) cloud

On-Premises



(Hybrid) Cloud app **continuum**



Modernizing web apps with Azure



Challenges



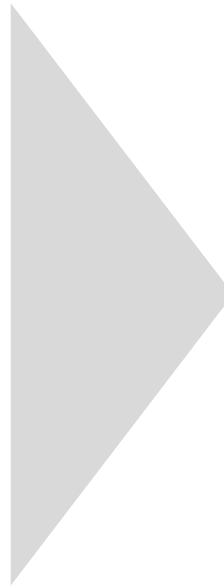
No consistent service level/uptime.
No scaling if unexpected traffic.



Reduced agility/lack of flexibility.
Monolithic applications, legacy code.



Lack of resources for new initiatives.
Limited technical support.



App Service benefits



Built-in no-configuration auto-scaling, load balancing and failover.

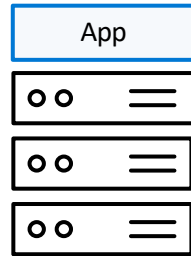


CI/CD/DevOps out of the box.
Serverless options for all app scenarios.



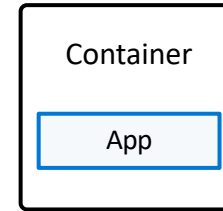
Frees IT to focus on innovation.
Use existing skills, tools & frameworks.

What is a **container**?



Virtual machines

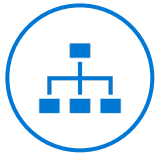
Virtualize the hardware
VMs as units of scaling



Containers

Virtualize the operating system
Applications as units of scaling

What are **microservices**?



A Software Architectural Style

Applications are composed of small, independent modules that communicate with each other using well-defined APIs. Not platform specific.



Decoupled

These service modules are highly decoupled building blocks that are small enough to implement a single functionality but together can form larger systems



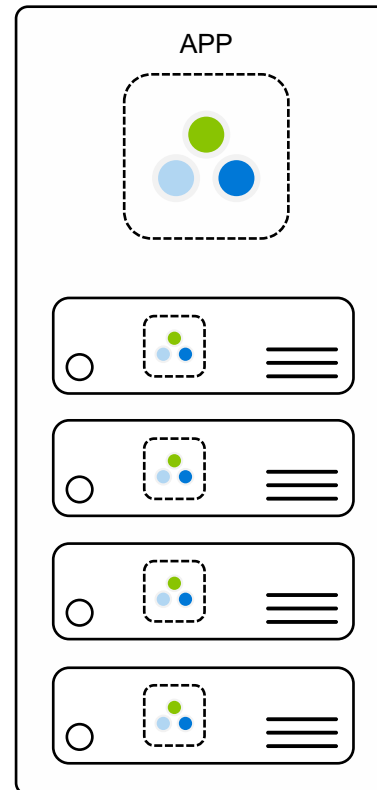
Independently versioned, deployed & scaled

With a microservices architecture, developers can create, manage and improve application services independently, even using different languages

Containers provide the consistent format and isolation desired by microservices.

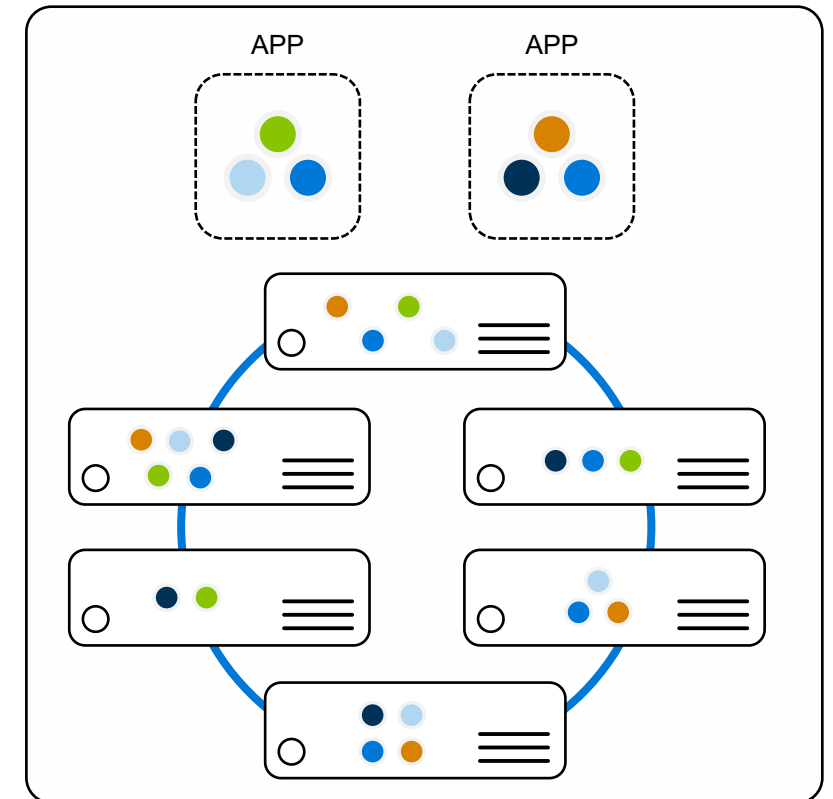
Monolithic

Large, all-inclusive app



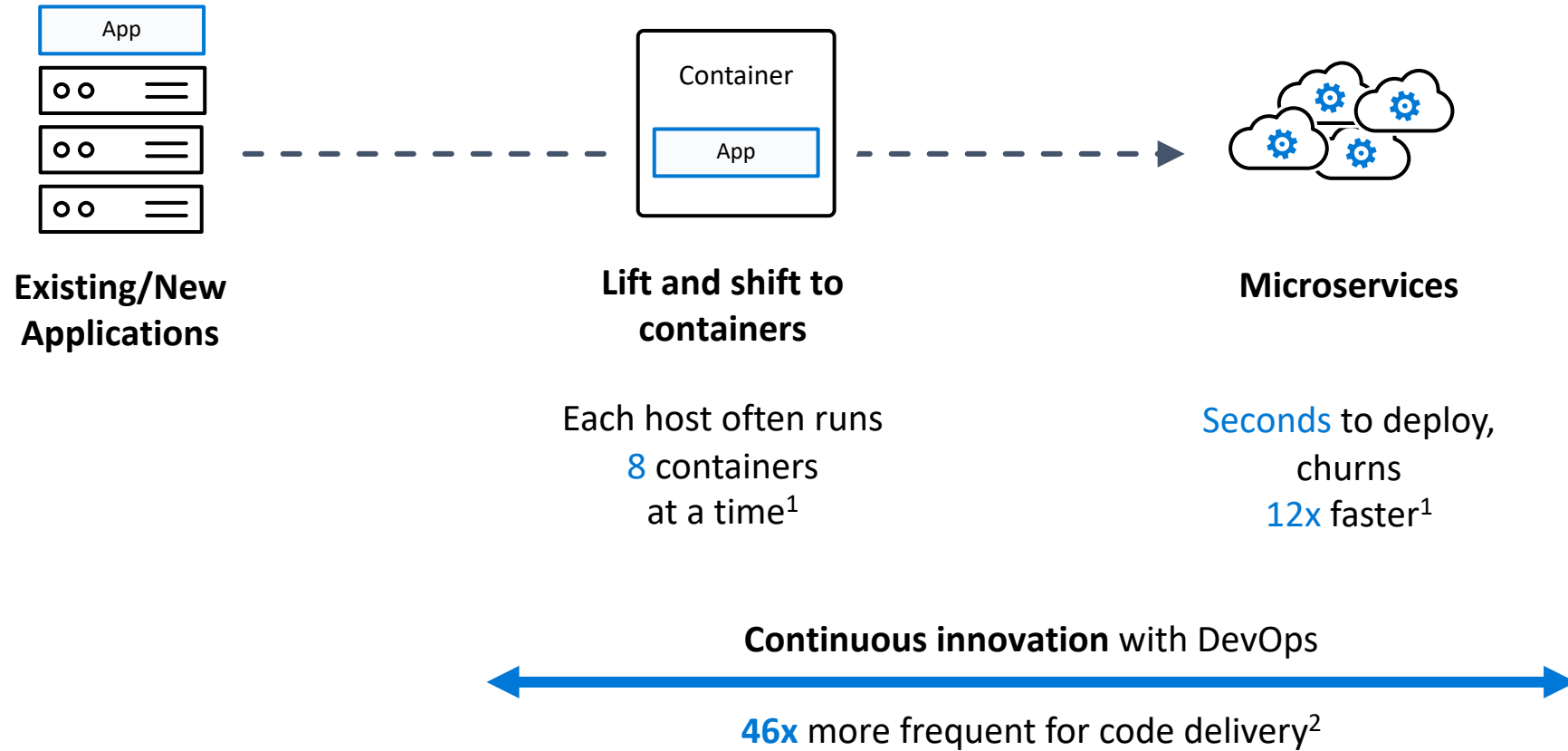
Microservices

Small, independent services



How can containers help your app modernization journey?

From **traditional systems** to a **portfolio of modern apps**



Containers in Azure



App Service

Deploy web apps or APIs using containers in a PaaS environment



Service Fabric

Modernize .NET applications to microservices using Windows Server containers



Kubernetes Service

Scale and orchestrate Linux containers using Kubernetes



Container Instance

Elastically burst from your Azure Kubernetes Service (AKS) cluster



Ecosystem

Bring your Partner solutions that run great on Azure



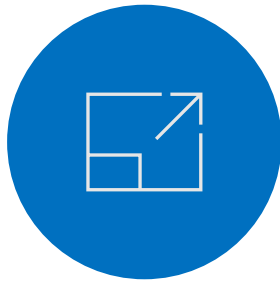
Azure Container Registry



Docker Hub

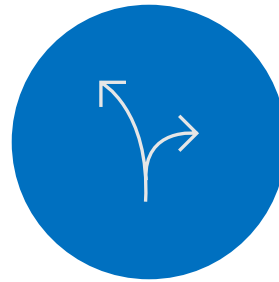
----- Choice of developer tools and clients -----

Azure Serverless



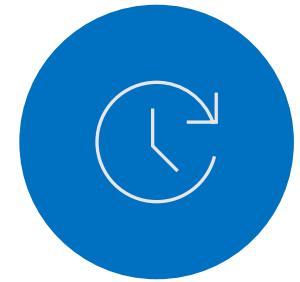
Focus

Auto-scale based on workload
No infrastructure management
No wasted resources



Flexibility

Hosting options
Multiple languages
Development environments



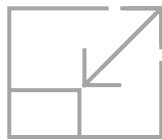
Efficiency

Shorter time to market
Event-driven programming model
End-to-end dev experience

Why is it a great fit for Microservices?

Problems of traditional microservices

- 1 Scaling of compute resources
- 2 Operations dependency
- 3 Pay per hosting nodes
- 4 Services discovery and managing services integration



Serverless solution

- ✓ Automatic scaling based on workload
- ✓ No infrastructure management
- ✓ Pay per execution
- ✓ Event-based programming model (triggers + bindings)



App Modernization continuum



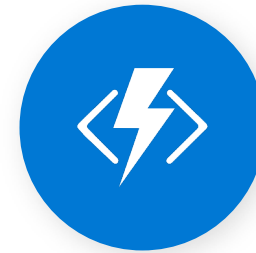
Orchestration
(Kubernetes)



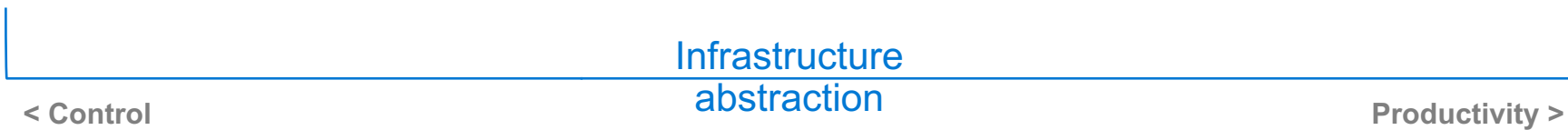
Microservices



Web Apps



Event-driven
Functions





APP MODERNISATION DAY

09/04/2019 — MICROSOFT HOUSE, MILANO

THANKS!!!



Lorenzo Barbieri
Cloud Solutions Architect
@_geniodelmale
[linkedin.com/in/geniodelmale](https://www.linkedin.com/in/geniodelmale)

#APPMODERNISATION

#UGIDOTNET



APP MODERNISATION DAY

09/04/2019 — MICROSOFT HOUSE, MILANO

L'EVENTO È STATO POSSIBILE GRAZIE A:

COMPUTER
GROSS

managed/designs