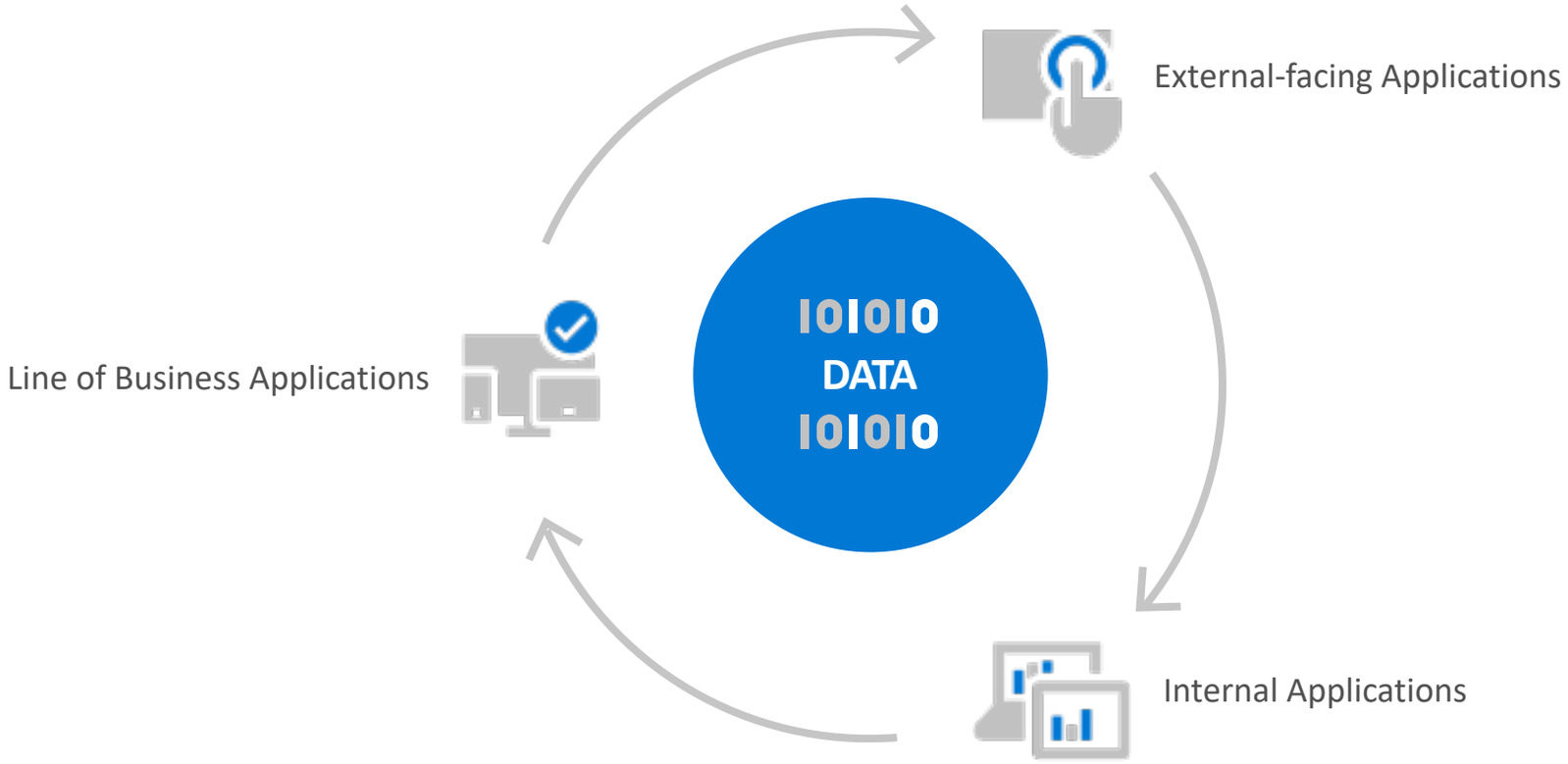
App Modernization

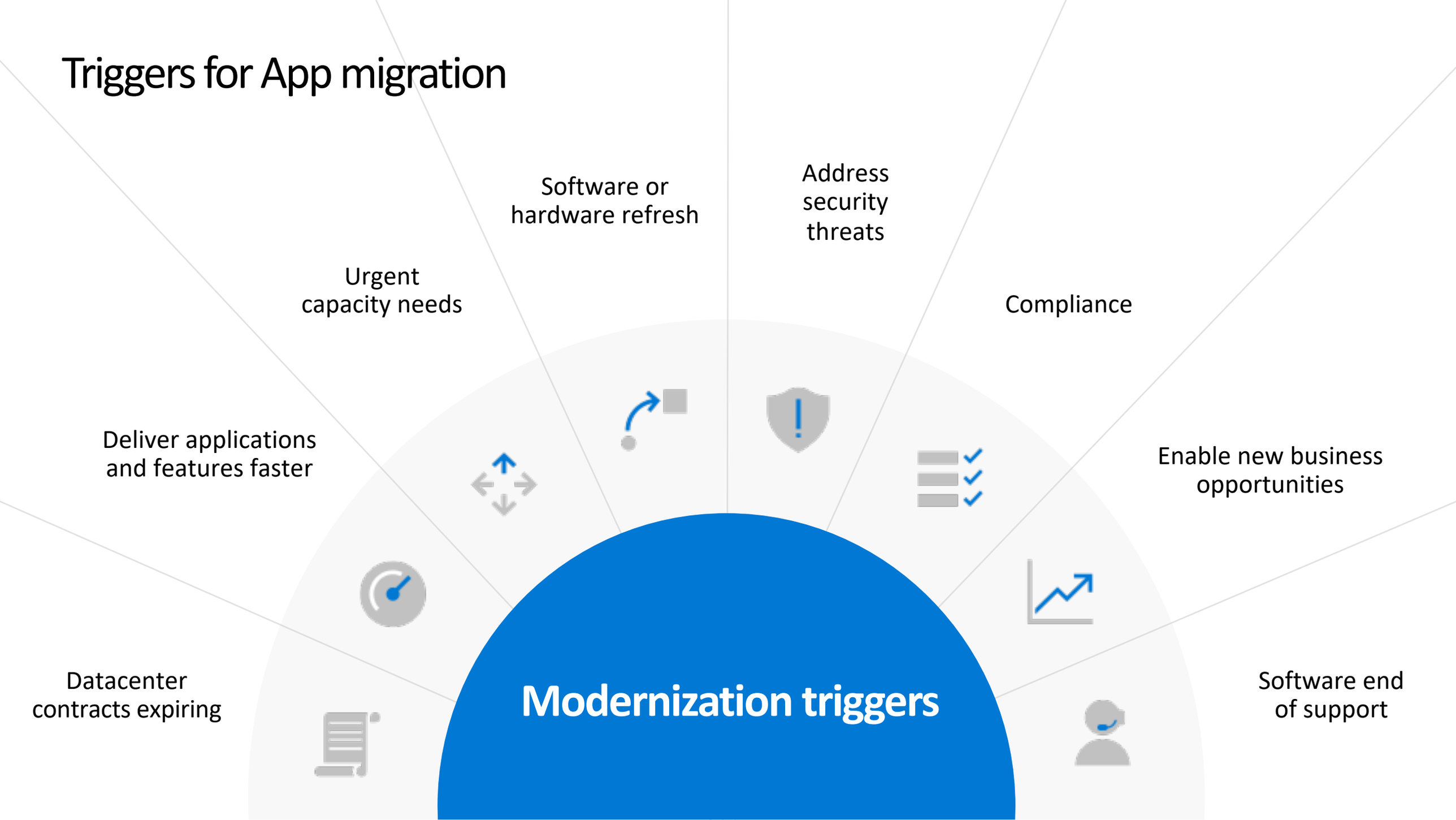
Future-proof the applications
that power your business today.

Introduction + Challenge

Business is powered by applications



Triggers for App migration



Challenges

Keeping up with the speed of business

Code

More resilient and scalable applications

Adding new features and functionality without taking applications down

Future-proof applications

Infrastructure

Heterogeneous workloads

Hybrid deployments

Security and management

Continuous monitoring

Cost management

Data

Data growth and data silos

Incongruent data types

Performance constraints

Complexity of solutions

Rising data maintenance costs

Security issues and data breaches

Application Delivery

Shorter release cycles

Improved software quality

Responding faster to bugs and security incidents

Learning from real usage to continuously improve applications

Are your applications ready?

Today

Application silos, built in isolation

Limited set of platforms and form factors

Overabundance of data

Servers and infrastructure to manage

Upfront capacity planning, fixed scale

Future

Multi-channel applications, covering all touchpoints

Many platforms, devices and form factors

Data-driven intelligence in applications

Focus on application functionality, not infrastructure

Elastic, unlimited scale


How can I make sure that my existing applications can take maximum advantage of cloud capabilities?
I'm worried that I need to start from zero and rebuild for the cloud?
How do I get started, I have so many apps!

The application journey to the cloud

The journey to the cloud can be through various routes

IaaS/VM/Compute

Own your home

Platform as a Service

Bed and breakfast

Serverless

Hotel

Modernizing with managed services

Challenges

Infrastructure management slows down business processes

Inefficient resource management

Lock-in to a limited (legacy) stack. Lack of portability across clouds

Deployment not automated, slow, wasted time due to manual tasks

Production infrastructure can not be replicated on developer machines

Azure Benefits

Managed services let you focus on apps, not admin and speed up deployments

Smaller instances increase packing density and improve resource utilization

Managed services support all stacks. Containers run on any cloud

Fast and agile app deployment with built-in DevOps and instant startup

Environments are consistent across development, test and production

Closing

Lower your TCO by moving to Azure

Azure offers many ways to save money

78%

savings by migrating datacenters to Azure instead of staying on-premises

68%

savings by rehosting applications on Azure PaaS

63%

savings by rearchitecting applications for Azure

5x

lower cost for Microsoft workloads compared to AWS

Source

Azure TCO Calculator at www.azure.com/tco

How we can help

Let’s have a deeper discussion about ***your*** journey to Azure.

Datacenter Migration

Windows Server on Azure

Linux on Azure

SAP on Azure

Azure Stack

Security and Management

Business Continuity Disaster Recovery

High-performance Computing

Application Modernization

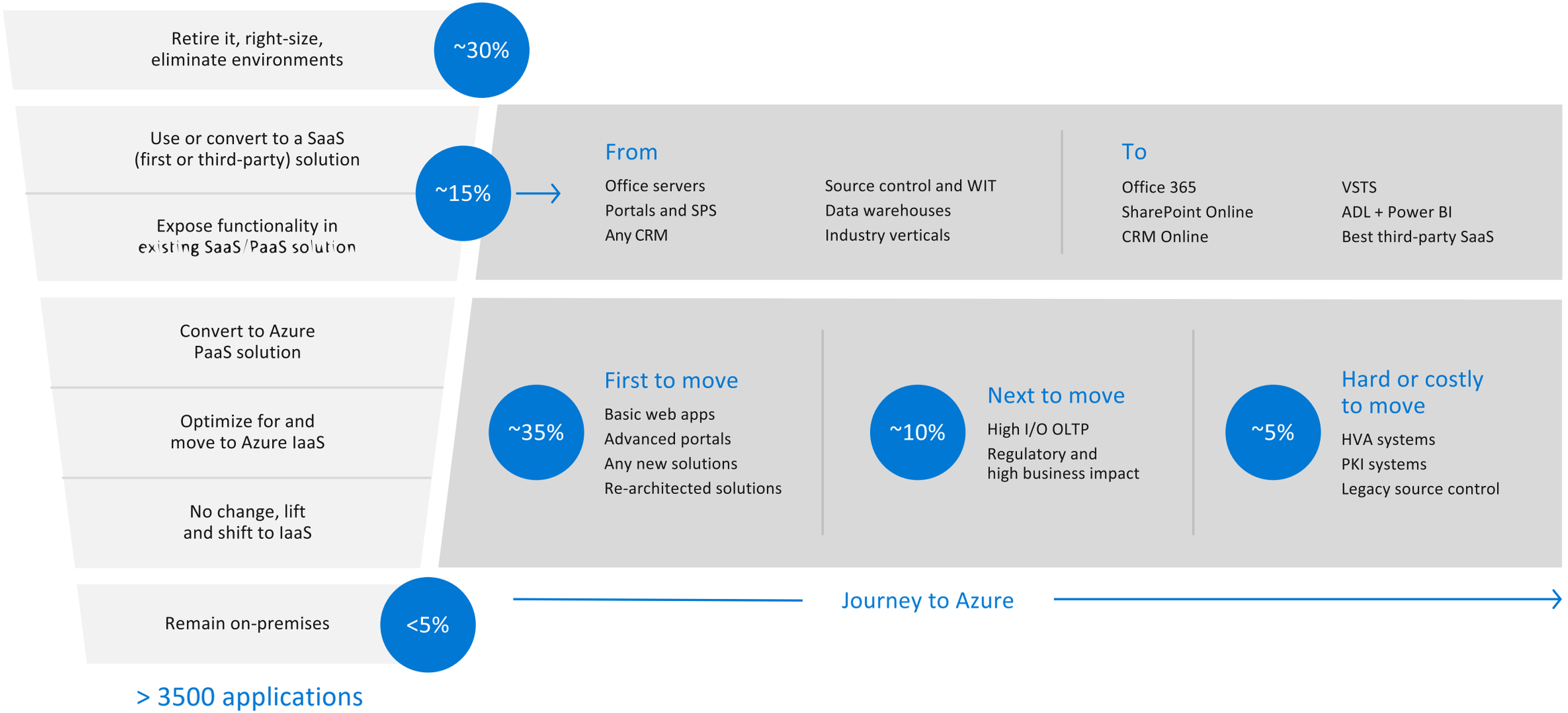
DevOps

Microsoft

Thank you

Journey to the cloud

Example demonstrating modernization a Fortune 500 application portfolio


Retire it, right-size, eliminate environments
~30%
Use or convert to a SaaS (first or third-party) solution
Expose functionality in existing SaaS/PaaS solution
~15%
From
Office servers
Portals and SPS
Any CRM
Source control and WIT
Data warehouses
Industry verticals
To
Office 365
SharePoint Online
CRM Online
VSTS
ADL + Power BI
Best third-party SaaS
Convert to Azure PaaS solution
Optimize for and move to Azure IaaS
No change, lift and shift to IaaS
~35%
First to move
Basic web apps
Advanced portals
Any new solutions
Re-architected solutions
~10%
Next to move
High I/O OLTP
Regulatory and high business impact
~5%
Hard or costly to move
HVA systems
PKI systems
Legacy source control
Remain on-premises
<5%
Journey to Azure
> 3500 applications

Our top 3 learnings

1. It’s a great opportunity to ‘clean house’ - retire applications that are no longer used or consolidate and transition functionality to off-the-shelf solutions where possible.

2. No cloud hosting model (IaaS, PaaS, SaaS) should be taken off the table too early.

3. Move custom applications in stages:

    - Move simple workloads directly to PaaS: web sites, static portals, standard three-tier applications
    - Migrate complex solutions to IaaS first and gradually modernize later to unlock immediate benefits
    - Keep legacy/undocumented code as-is and surround it with serverless functions to add features