Cloud Day Athens

ATHENS CONSERVATOIRE | 10 OCTOBER 2023

MIC207

Migrate and modernize your Microsoft workloads

Dragoș Mădărășan
Solutions Architect Team Lead
AWS

Pavlos Kaimakis
Senior Solutions Architect
AWS

Agenda

Why Windows on AWS

AWS modernization journey

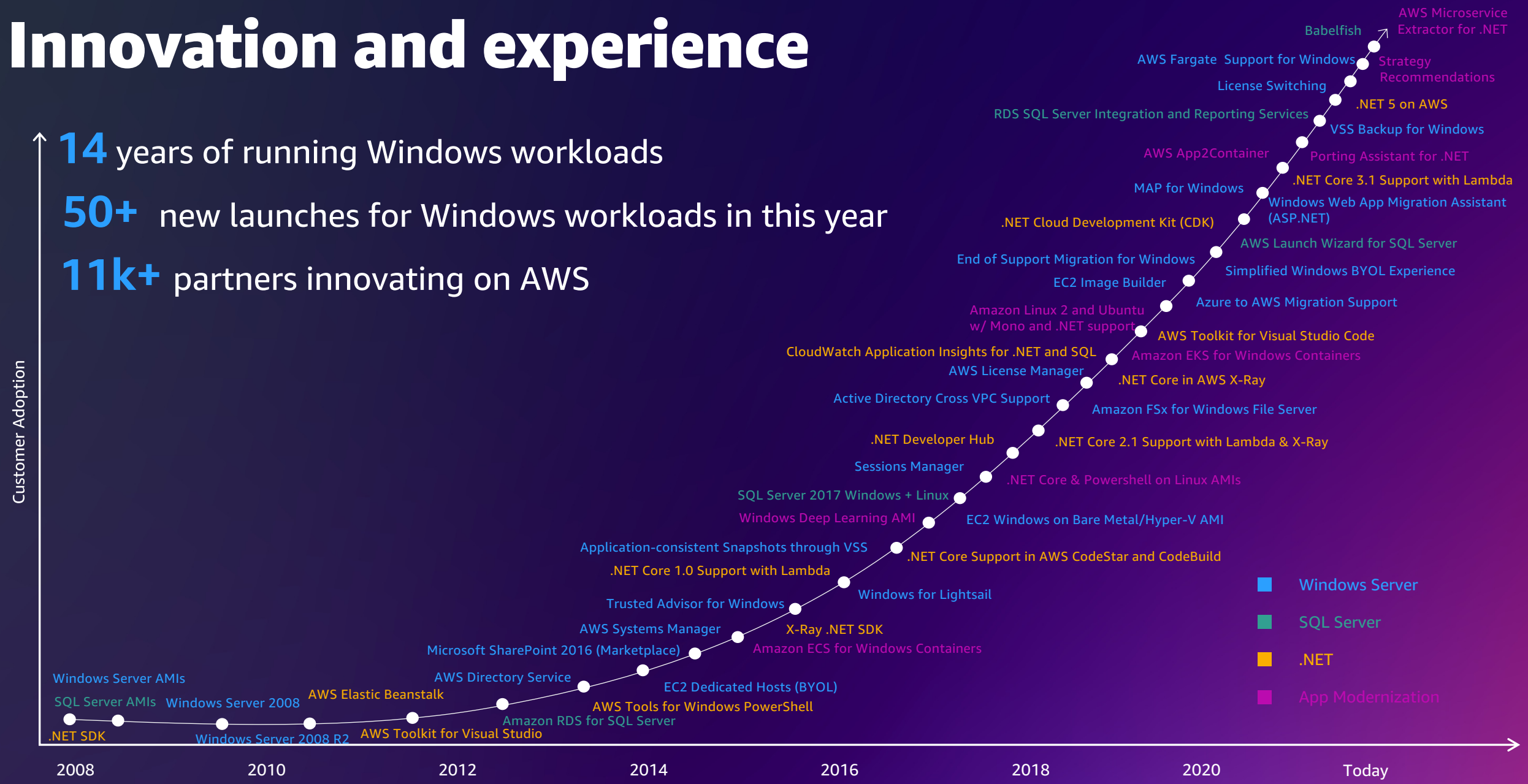
Modernization tools and services

Innovation and experience

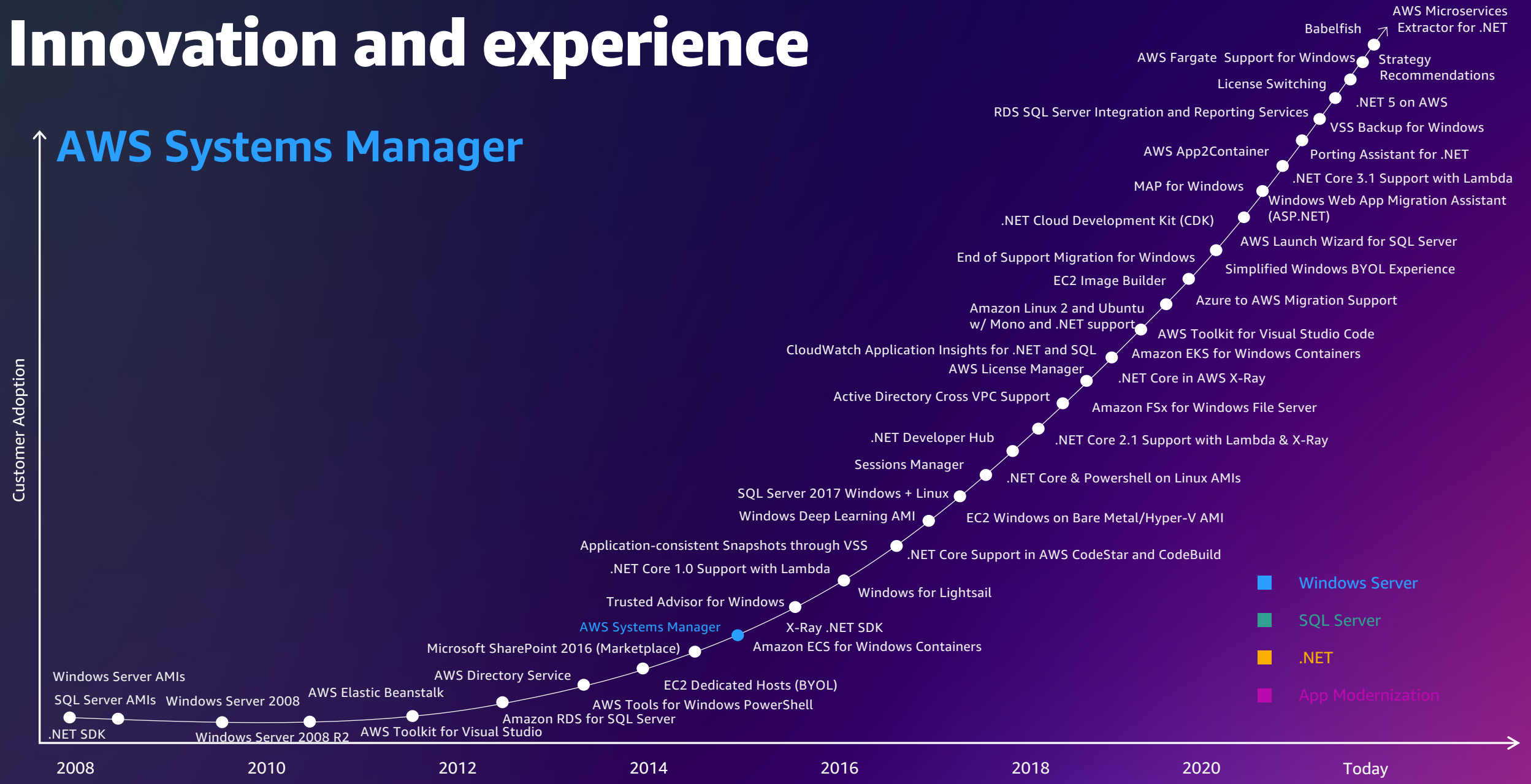
14 years of running Windows workloads

50+ new launches for Windows workloads in this year

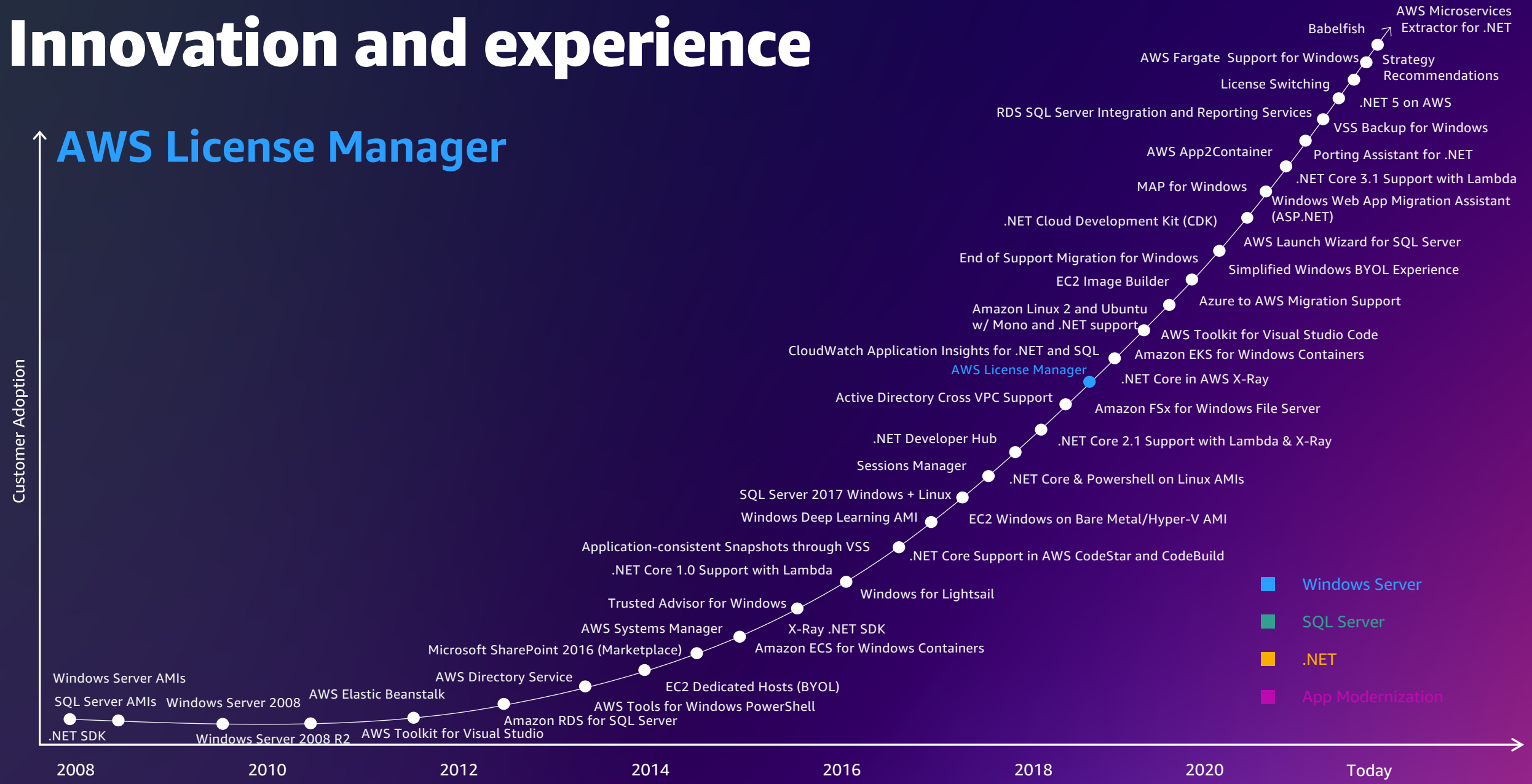
11k+ partners innovating on AWS



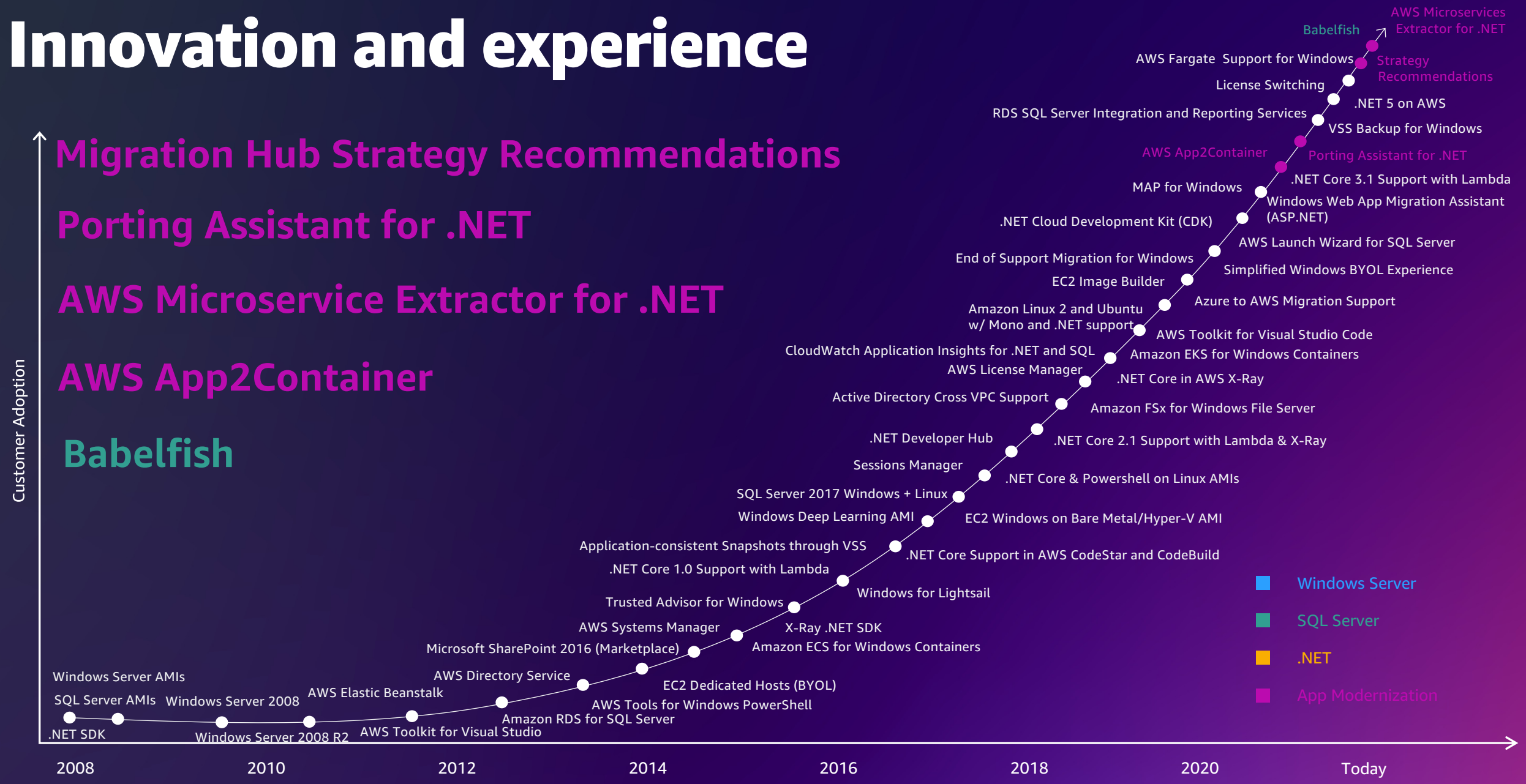
Innovation and experience



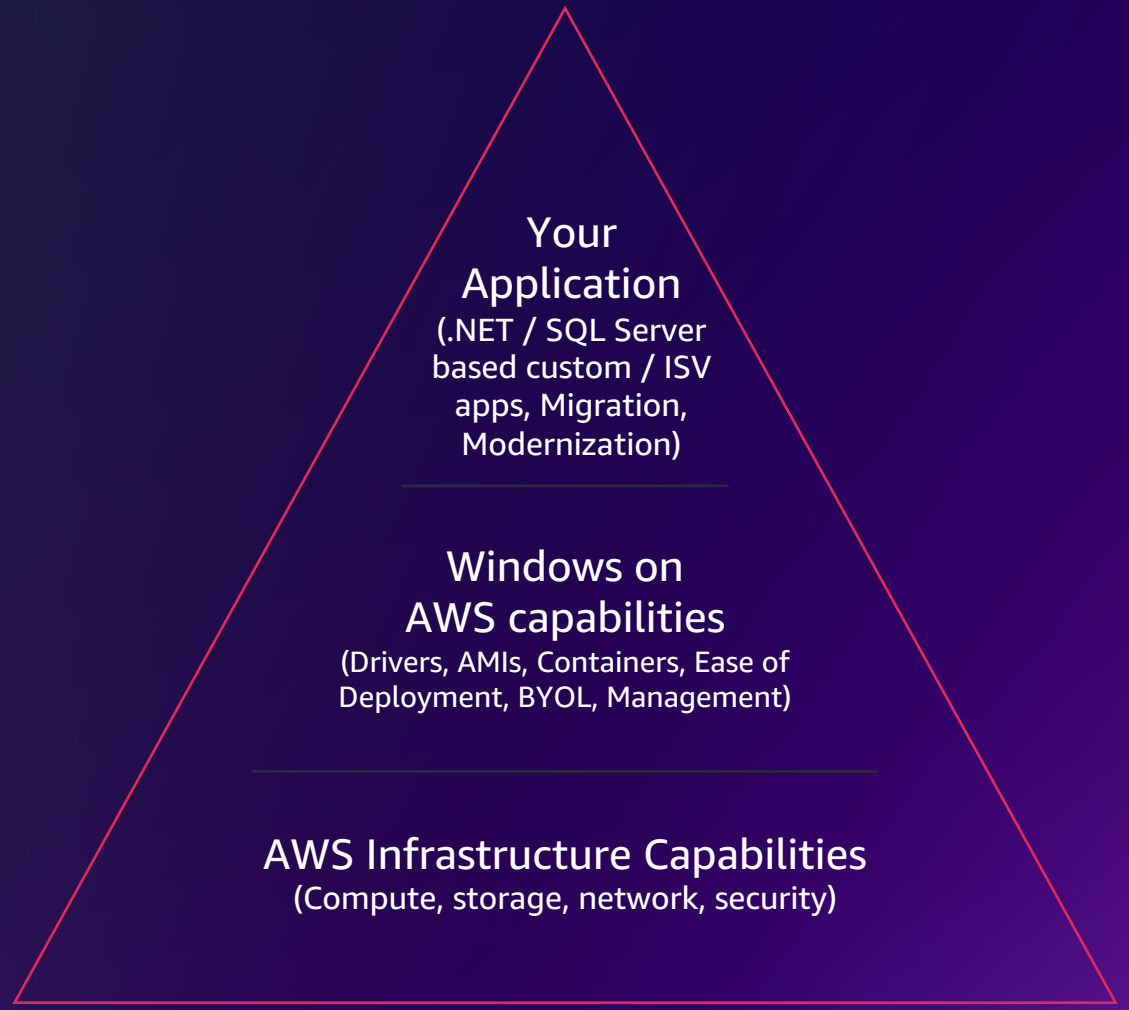
Innovation and experience



Innovation and experience



Building blocks



Infrastructure capabilities

Broadest and deepest platform choice

Categories

General purpose

Burstable

Compute intensive

Memory intensive

Storage
(high i/o, dense)

GPU compute

Graphics intensive

Capabilities

Choice of processor
(AWS, Intel, AMD)

Fast processors
(up to 4.5 GHz)

High memory footprint
(up to 24 TiB)

Instance storage
(HDD and SSD)

Accelerated computing
(GPU, FPGA, and ASIC)

Networking
(up to 400 Gbps)

Bare metal

Size
(Nano to 32xlarge)

Options

Windows, Linux, Unix, macOS

Amazon EBS

Amazon Elastic Inference

Elastic Fabric Adapter

475

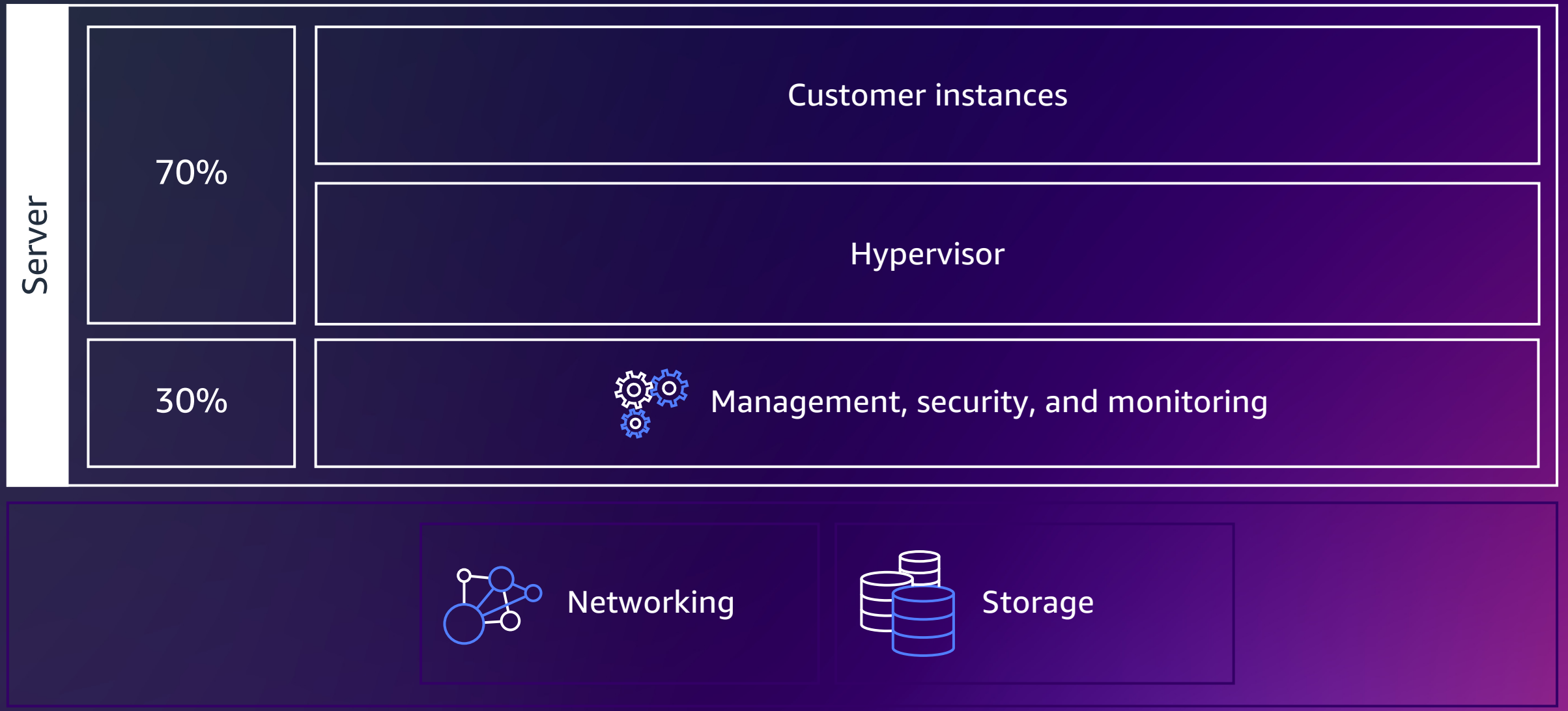
INSTANCE TYPES

for virtually every workload and business need

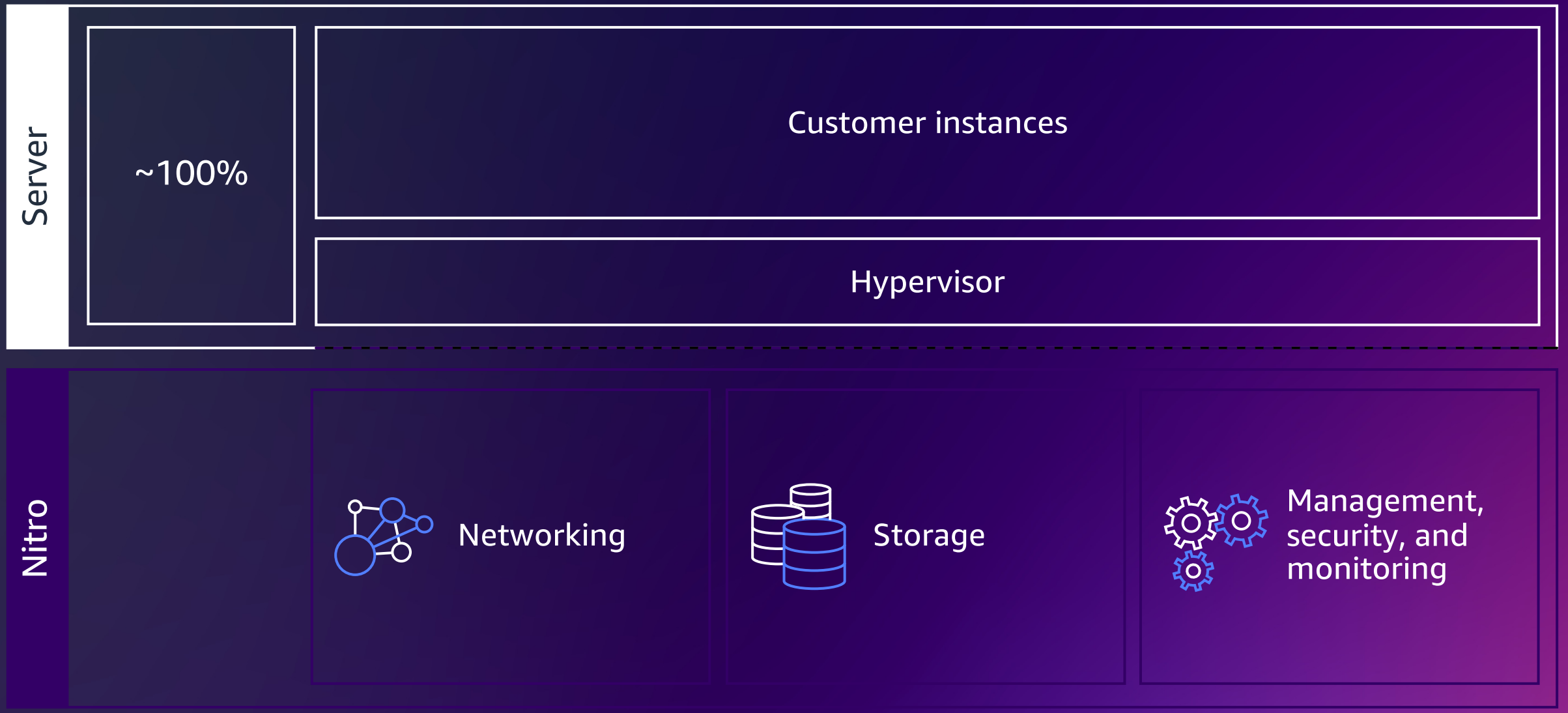
NITRO SYSTEM

Reimagined virtualization infrastructure

Pre-Nitro EC2 instance host architecture



The AWS Nitro architecture



Fleet Manager

CONSISTENT ADMINISTRATION ACROSS WINDOWS AND LINUX SERVERS

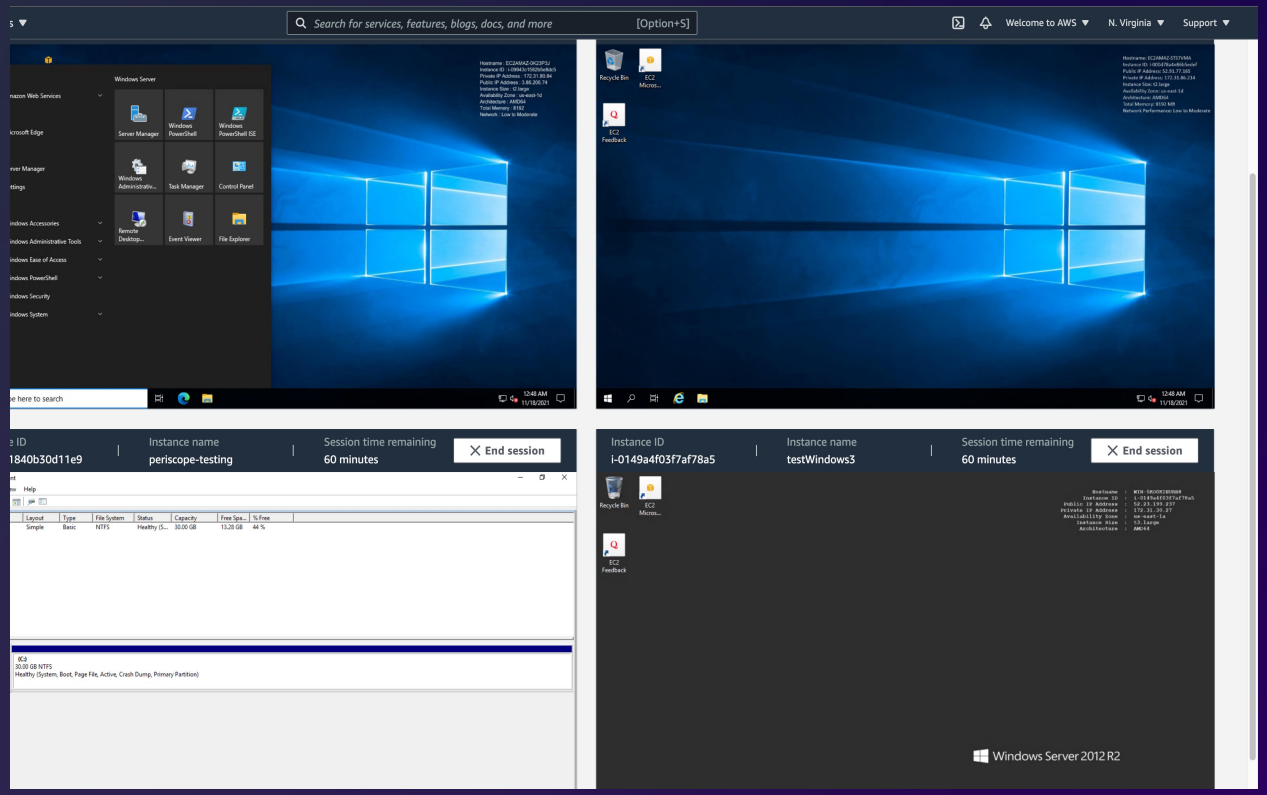
Manage and troubleshoot your Windows and Linux servers running on AWS and on premises

Administer VM fleet without needing to remotely connect with RDP or SSH

Perform common operations such as browsing file systems, monitoring CPU metrics, etc. from a single integrated console

Secure point-and-click Windows management

CONSOLE-BASED MANAGEMENT FOR WINDOWS IN AWS SYSTEMS MANAGER



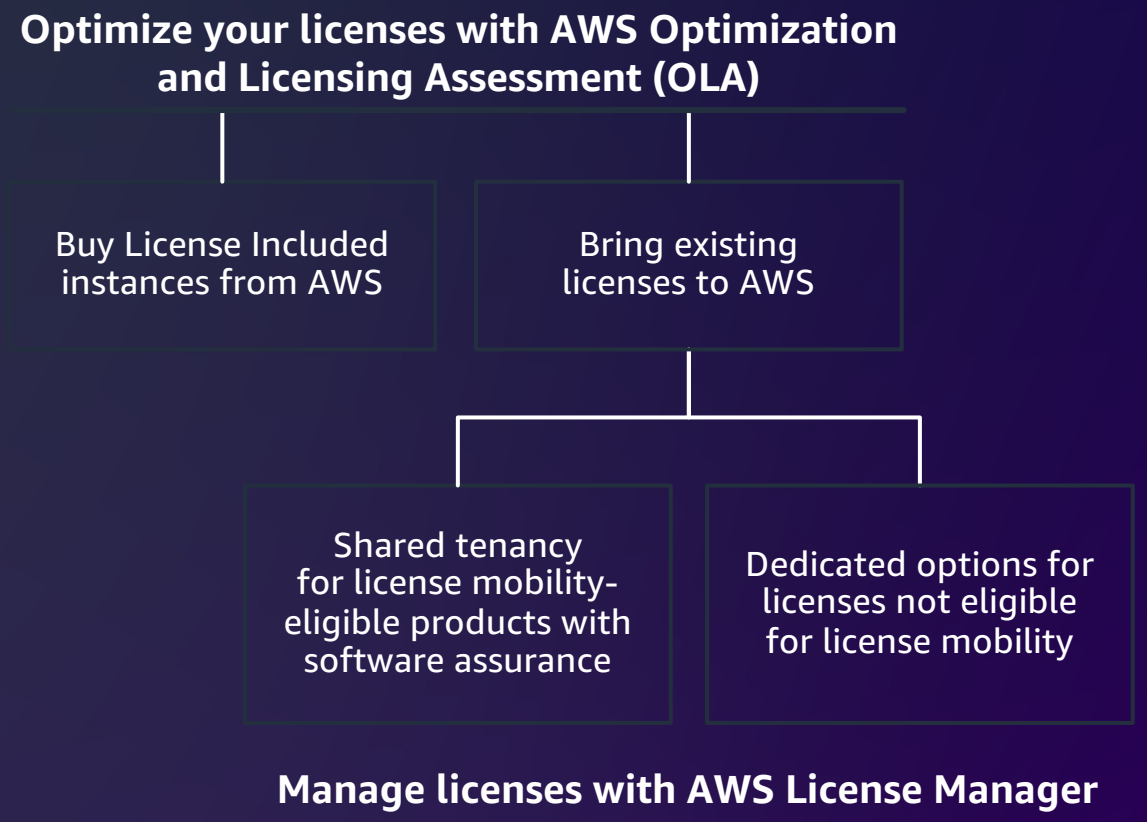
Graphical user interface built using high-performance NICE DCV protocol and Session Manager

Secure and operationally efficient

Single Sign-On using AWS SSO identities and third-party identity providers

License management & optimization

Flexible licensing options for Windows on AWS



1 **Bring your licenses to AWS (BYOL)**

Save costs with Dedicated Hosts

2 **Buy licenses included (LI) from AWS**

Pay as you go with no upfront costs

3 **AWS License Manager**

Manage, discover, and report software license usage

AWS Optimization and License Assessment (OLA)

Collect

Determine prospective workloads to optimize, and provide existing utilization data or collect utilization data for the underlying workloads using native AWS or third-party tooling

Analyze

Analyze the data to model cost and licensing optimization scenarios

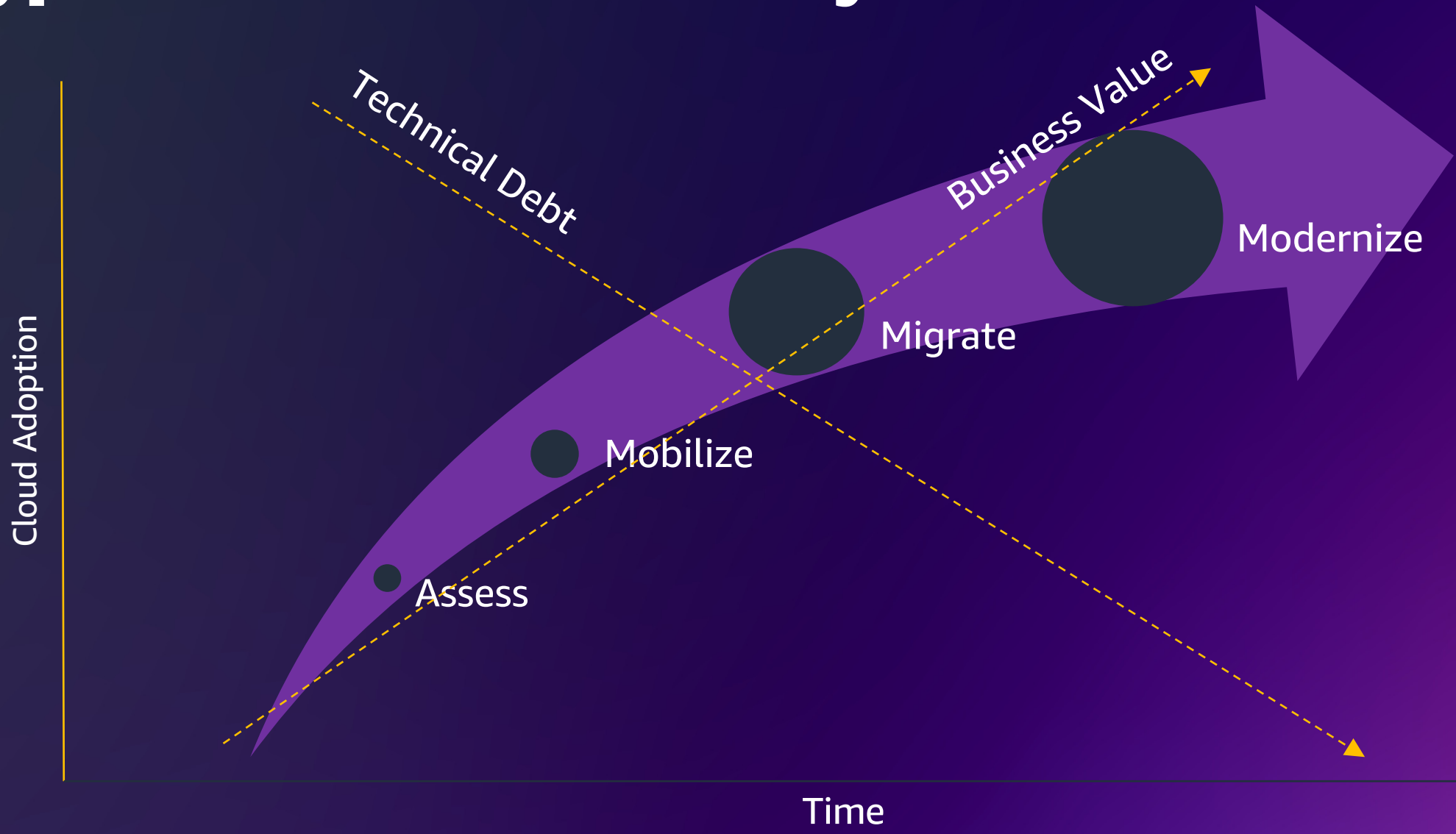
Plan

Review the results and build your business case or start a migration proof of concept

62.4% average cost savings

Modernization

Typical Customer Journey

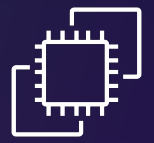


Why are customers modernizing applications on AWS?

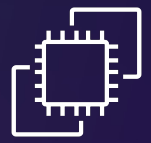
- **Agility**
  - Decomposing monoliths to microservices brings freedom to adapt and experiment quickly
  - Port and containerize .NET applications and deploy using cloud-native tooling
- **Lower costs**
  - No longer pay for expensive Windows and/or SQL Server licensing, saving more than 30% costs
- **Improve price/performance**
  - Amazon Aurora offers performance and availability of commercial-grade databases at 1/10th the cost
  - .NET 5/6 with AWS Graviton2 offers 40% better price/performance

Take full advantage of the cloud

Rehost

	MIGRATE
Windows applications	Self-managed apps and databases run on virtual machines (VMs) No code changes  Amazon EC2 Windows
SQL databases	 Microsoft SQL Server on Amazon EC2 Windows

Take full advantage of the cloud

	Rehost	Replatform
Windows applications	**MIGRATE** Self-managed apps and databases run on virtual machines (VMs) No code changes  Amazon EC2 Windows	**CONTAINERIZE** Automation and CI/CD Portability and predictability Minimal code changes  Amazon ECS  AWS Fargate  Amazon EKS
SQL databases	 Microsoft SQL Server on Amazon EC2 Windows	**MOVE TO MANAGED** Managed provisioning, backups, patching, monitoring, and scaling No code changes  Amazon RDS for SQL Server

Take full advantage of the cloud

	Rehost	Replatform	Refactor
Windows applications	**MIGRATE** Self-managed apps and databases run on virtual machines (VMs) No code changes Amazon EC2 Windows	**CONTAINERIZE** Automation and CI/CD Portability and predictability Minimal code changes Amazon ECS, AWS Fargate, Amazon EKS	**.NET 5+** License freedom Performance and security Cross-platform .NET Core **SERVERLESS** Pay as you go Build fast Scale effortlessly AWS Lambda
SQL databases	Microsoft SQL Server on Amazon EC2 Windows	**MOVE TO MANAGED** Managed provisioning, backups, patching, monitoring, and scaling No code changes Amazon RDS for SQL Server	**CLOUD NATIVE** High performance and scalability Saves on licensing costs Amazon Aurora, Amazon DynamoDB, Amazon Neptune, Amazon Redshift

Windows modernization toolbox

Modernize within Windows

App2Container

End-of-support migration program for Windows

Modernize from Windows

Porting Assistant

Microservice Extractor for .NET

Move to purpose-built databases

Schema Conversion Tool

Database Migration Service

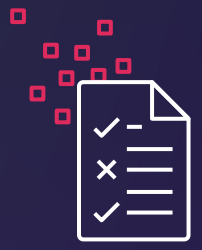
In-place replacement for SQL Server

Babelfish for Aurora PostgreSQL

Strategy Recommendations, Refactor Spaces

AWS Migration Hub

AWS Migration Hub Strategy Recommendation



Application transformation outcomes

Rehosting applications and databases to Amazon EC2

Moving to managed environment, including RDS and Elastic Beanstalk

Replatforming to containers

Operating system upgrades for Windows EOS versions

Refactoring .NET applications to .NET Core on Linux

Refactoring licensed enterprise databases (SQL Server, Oracle) to open source

How it works

Data collection

- Server inventory and application metadata
- Business objective and constraints
- Source code repo*
- Database*

Analysis

- Static environment
- Application process & runtime environment
- Source code*
- Database (SQL Server)*

Recommendation+

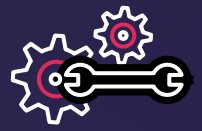
- Migration strategy
- Modernization destination
- Tools to use
- Application and database incompatibilities

* Optional
+ TCO is not a supported recommendation

Challenges of transforming .NET monolithic applications to microservices

Difficulty identifying parts of the application to be extracted as separate services

Challenges with grouping functionality based on business domains/process they relate to



Need to use multiple tools to co-relate source code and runtime metrics

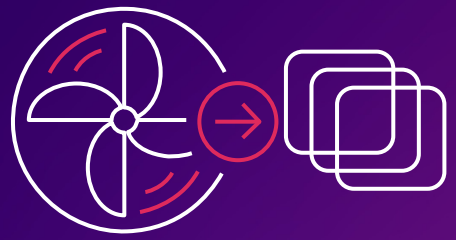
Manual work to carve out functionality and deploy it

AWS Microservice Extractor for .NET

Assistive tool that reduces effort to transform monolithic applications into microservices that teams can develop and operate independently

Faster identification of parts of the application to carve out as separate services

Facilitates planning of refactoring based on Domain-Driven Design



Assisted extraction of code into separate repositories

The manual refactoring process

…with AWS Microservices Extractor for .NET

Reduced manual effort to transform the apps

How it works

AWS Microservice Extractor for .NET

Download & install
Download and install on your server

Scan application
Scan your application by providing source code and runtime metrics

Group components
Identify components to be extracted from application map produced by the service

Prepare for extraction
Modify code to prepare application for extraction

Extract
Use the tool to extract parts of the application's codebase as separate code repositories

Build & Deploy
User may build and deploy the extracted code as independent services

Thank you!

Dragoș Mădărășan

Pavlos Kaimakis

Please complete the session survey in the Events App

App Store

Google Play Store