

The background features a dark blue gradient with abstract, glowing shapes in shades of purple and pink. Two thin, light blue lines intersect to form a large 'A' shape. The text is positioned on the left side of the image.

# AWS re:Invent

DECEMBER 2 - 6, 2024 | LAS VEGAS, NV

DOP208

# Accelerate .NET porting from Windows to Linux with Amazon Q Developer

**Murtaza Chowdhury**

(He/Him)

Sr. Manager, PMT  
AWS Windows Migration &  
Modernization

**Raviprasad Mummidi**

(He/Him)

Principal Software Engineer,  
AWS Windows Migration &  
Modernization

**Nits Jeganathan**

(He/Him)

Sr. PMT,  
AWS Windows Migration &  
Modernization



# Agenda

- 01 Why make .NET applications Linux-ready?
- 02 Meet Amazon Q Developer transform for .NET
- 03 Demo and key features

# Customers are increasingly choosing Linux

*Customers have reduced their Windows usage by over 70% to adopt Linux*



*Staying on Windows increases costs, restricts scalability, and limits performance—Linux removes these barriers*



# Why is this the right strategy?



Cross-platform



High-performance



Microservice  
architecture



Community



Cost-effective

*Porting to Linux empowers .NET applications to leverage the efficiency, scalability, and flexibility of the cloud*

# The .NET porting process

Analyze code

Detect  
incompatibilities

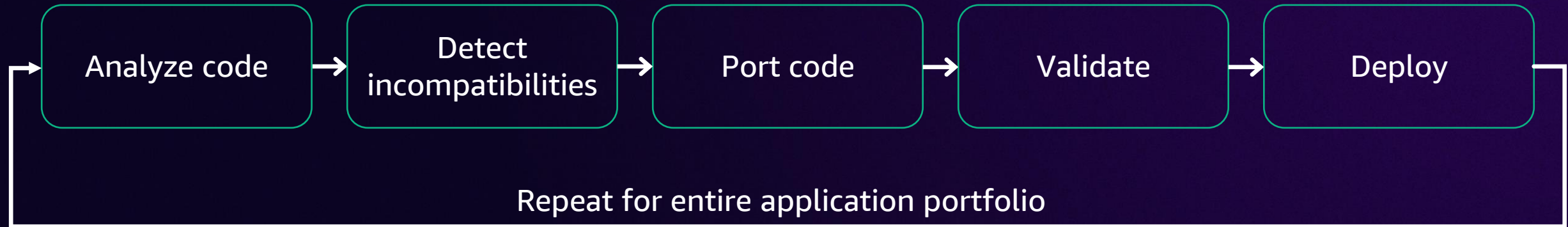
Port code

Validate

Deploy

*Porting to Linux requires careful planning, compatibility assessment, transformation, and testing*

# Porting to cross-platform .NET is hard & slow!



Labor intensive



Error prone



Collaboration  
bottlenecks

*Porting .NET applications to Linux is complex, requiring code rewrites, error handling, and team coordination*

# Transformation capabilities of Amazon Q Developer





# Demo: IDE experience for per-application porting

*The IDE integration provides real-time transformation support with code porting and comparison views*



```

1  using System.Diagnostics;
2  using Bookstore.Web.ViewModel;
3  using Bookstore.Domain.Books;
4  using System.Threading.Tasks;
5  using Bookstore.Web.ViewModel.Home;
6  using System.Web.Mvc;
7
8  namespace Bookstore.Web.Controllers
9  {
10     [AllowAnonymous]
11     public class HomeController : Controller
12     {
13         private readonly IBookService bookService;
14
15         public HomeController(IBookService bookService)
16         {
17             this.bookService = bookService;
18         }
19
20         public async Task<ActionResult> Index()
21         {
22             var books = await bookService.ListBestSellingBooksAsync(4);
23
24             return View(new HomeIndexViewModel(books));
25         }
26
27         public ActionResult Privacy()
28         {
29             return View();
30         }
31     }
32 }

```

Solution Explorer

Search Solution Explorer (Ctrl+;)

- Solution 'BobsBookstoreClassic' (5 of 5 projects)
  - Solution Files
  - Bookstore.Cdk
  - Bookstore.Common
  - Bookstore.Data
  - Bookstore.Domain
  - Bookstore.Web
    - Connected Services
    - Properties
    - References
    - App\_Data
    - App\_Start
    - Areas
    - Content
    - Controllers
      - AddressController.cs
      - AuthenticationController.cs
      - CheckoutController.cs
      - HomeController.cs
      - OrdersController.cs
      - ResaleController.cs
      - SearchController.cs
      - ShoppingCartController.cs
      - WishlistController.cs
    - Helpers
    - ILLink
    - Models
    - Scripts
    - Views
    - favicon.ico
    - Global.asax
    - packages.config
    - Startup.cs
    - Web.config

Output

Show output from: Amazon Web Services

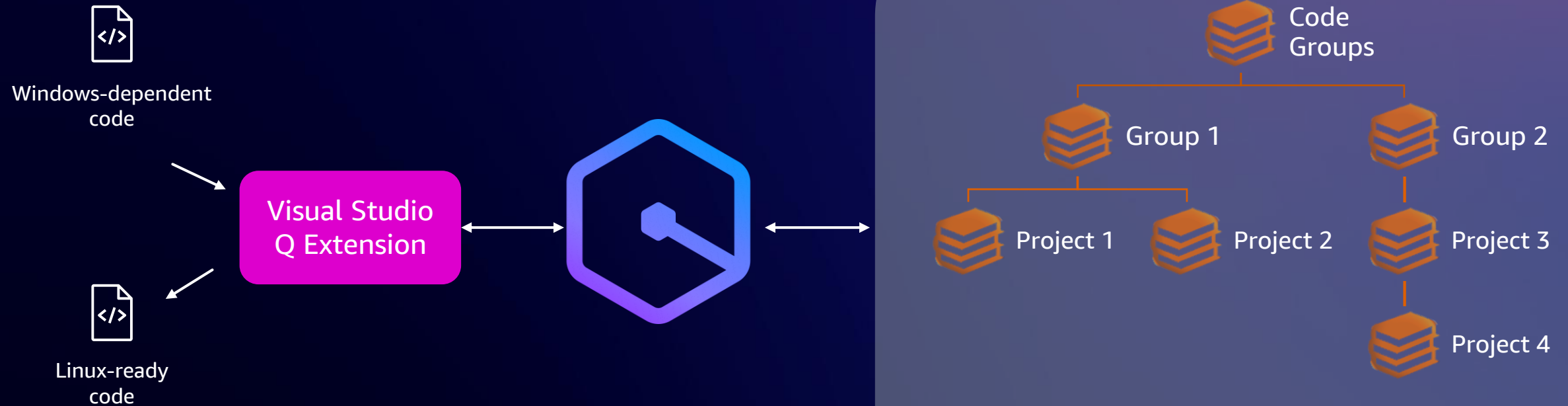
```

Launching Amazon Q Language Server v2.3.0 from local cache location: D:\Users\artrodri\AppData\Local\aws\toolkits\language-servers\AmazonQ\2.3.0
(Attribution notice for aws-lsp-codewhisperer can be found at: https://aws-language-servers.us-east-1.amazonaws.com/2d915c68-5138-4699-9d85-ddc4b05018d9/THIRD_PARTY_LICENSES)
Activating: Amazon Q Language Client
Initialized: Amazon Q Language Client
Reconnected to Amazon Q with Profile:AmazonQDev

```

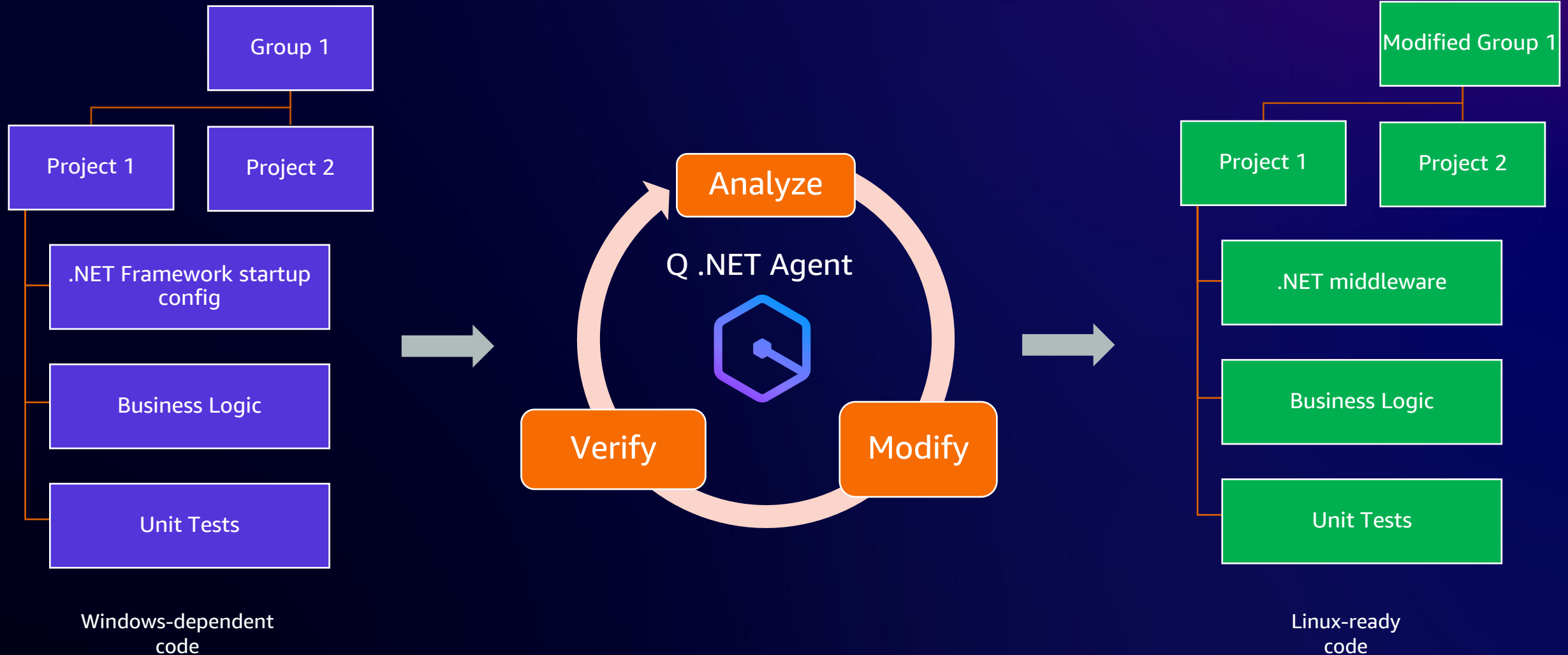
# How it works?

## IDENTIFICATION OF CODE GROUPS



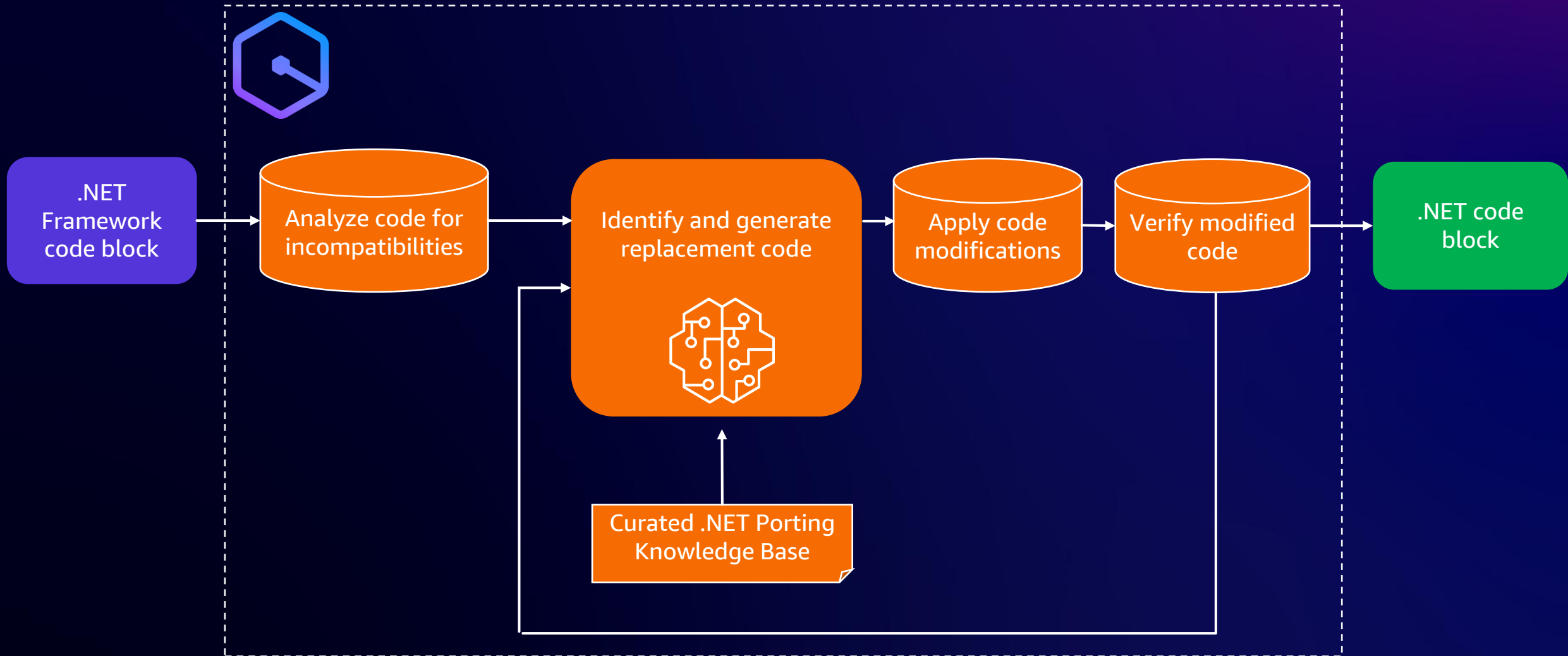
# How it works?

## PROCESSING OF CODE GROUP COMPONENTS



# How it works?

ANALYZE, MODIFY, AND VERIFY



# How it works?

## EVALUATION CRITERIA



- **Port a line of code with incompatibility?**
- **Repeatability of the process?**

# How it works?

## EVALUATION

**1. Evaluate modification for syntax errors**

**2. Test against different project types**

**Class Libraries, Console, MVC, Web API, WCF**

**3. Validate against thousands of projects**



# What's supported?

PORTING .NET APPLICATIONS TO BE LINUX-READY

- .NET projects written in C#
- .NET project types:
  - ✓ Console application
  - ✓ Class library
  - ✓ WCF Service
  - ✓ ASP.NET web API
  - ✓ Business logic layers of MVC and SPA
- Microsoft-authored NuGet packages
- Default Microsoft IIS configurations

## Source versions:

- .NET Framework versions 3.5+
- .NET Core 3.1
- .NET 5, .NET 7

## Target versions:

- .NET 8



# Large-scale porting of .NET applications



*Amazon Q Developer transformation capabilities automate .NET application porting to Linux, handling complexity and enhancing efficiency*

# Demo: Large-scale porting of .NET applications

*The web interface enables large-scale porting, real-time tracking, and easy management of application portfolios*

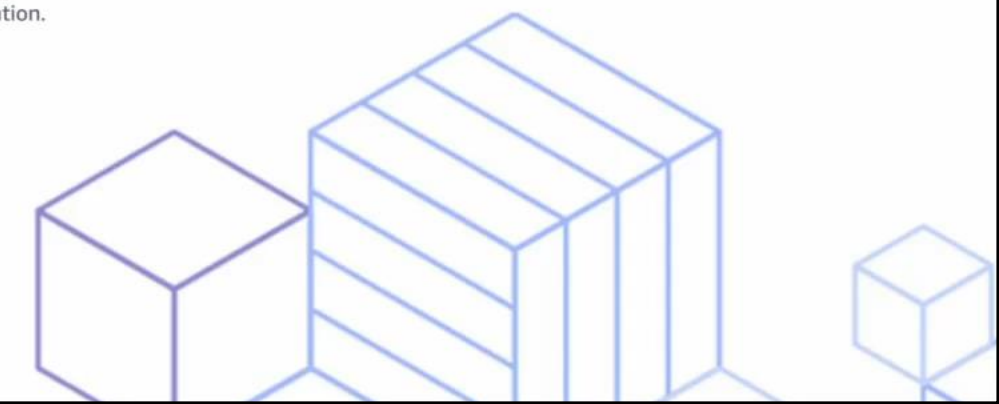
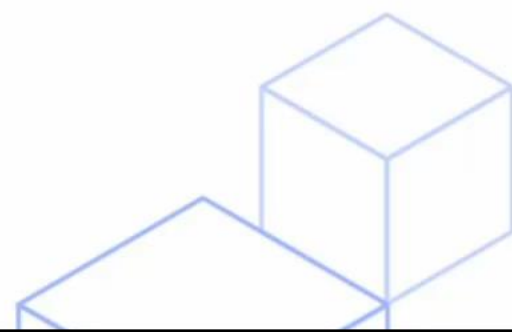
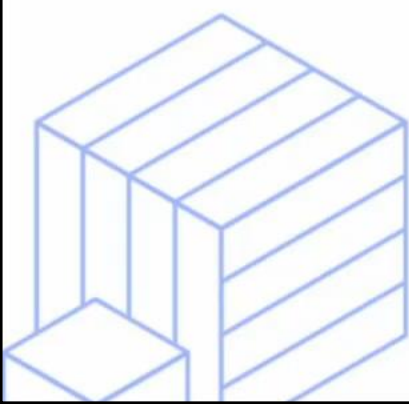




## Sign in

Username

By continuing, you agree to the [AWS Customer Agreement](#) or other agreement for AWS services, and the [Privacy Notice](#). This site uses essential cookies. See our [Cookie Notice](#) for more information.



# Dashboard and worklog

### Job details

Target branch destination: qt-mybranch-10-20-2024 | Target version: .NET 8.0

### Transformation summary

Repositories transformed: 40 of 135 | Projects transformed: 300 of 374

#### Repository status

Status	Count	Percentage
In progress	40	40%
Pending	10	10%
Complete	40	40%
Failed	25	25%

#### Project status

Status	Count	Percentage
In progress	40	40%
Pending	10	10%
Complete	40	40%
Failed	25	25%

#### Repositories (135)

Search by name

Name	Status	Projects	Source branch	Lines
<a href="#">WebAppFramework</a>	In progress	100 of 100	Main	###, #
<a href="#">CloudInfrastructureScripts</a>	In progress	30 of 30	Main	###, #

Job complete | Time elapsed: 1 hour 19 minutes 57 seconds

Dashboard | Collaboration | **Worklog**

### Worklog

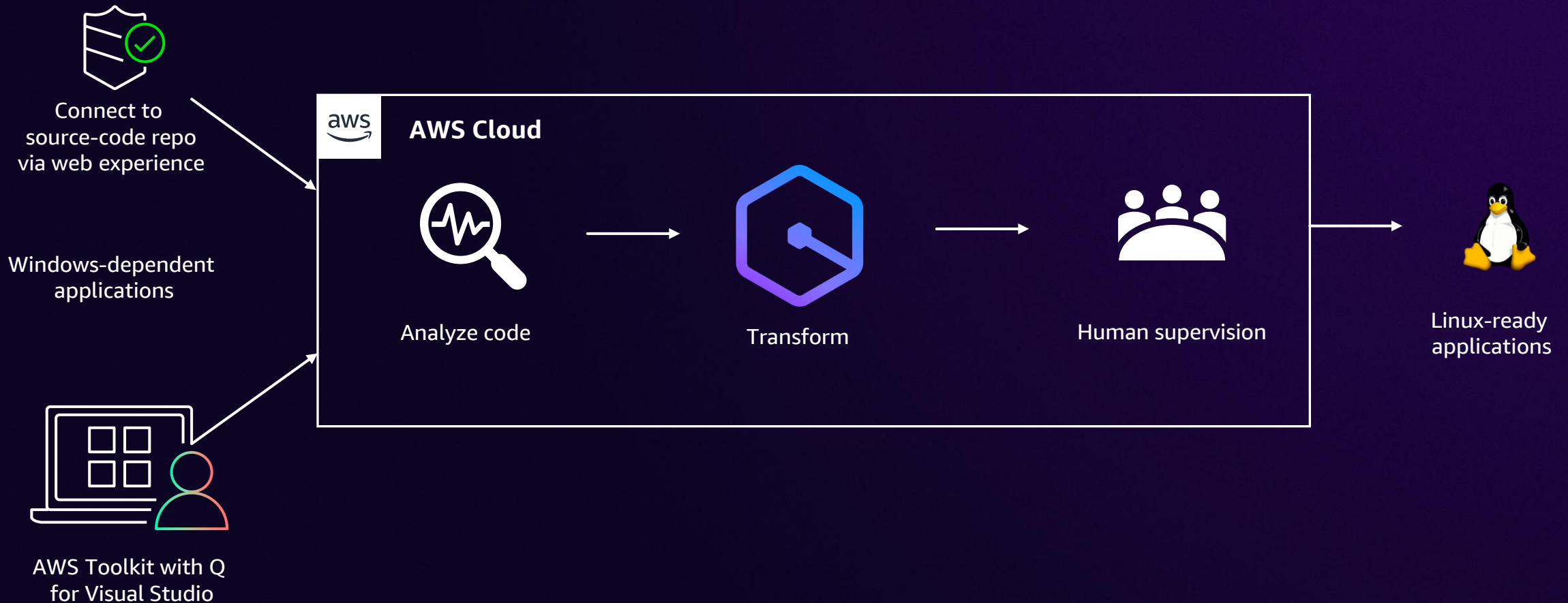
Timestamps reflect the Pacific Standard Time (PST) time zone.

- Nov. 27, 2024, 11:58:14 Step 2 - Transforming project bookstore.domain to net8.0. Q will transform your project to net8.0, ...It will update packages and refactor code.
- Nov. 27, 2024, 11:58:15 ...It will update packages and refactor code.
- Nov. 27, 2024, 11:58:16 Step 3 - Transforming project bookstore.data to net8.0. Q will transform your project to net8.0, it ...will update packages and refactor code.
- Nov. 27, 2024, 11:58:17 ...will update packages and refactor code.
- Nov. 27, 2024, 11:58:18 Step 4 - Transforming project bookstore.web to net8.0. Q will transform your project to net8.0, it ...will update packages and refactor code.
- Nov. 27, 2024, 11:58:19 ...will update packages and refactor code.
- Nov. 27, 2024, 11:58:20 Step 5 - Transforming project bookstore.cdk to net8.0. Q will transform your project to net8.0, it ...will update packages and refactor code.
- Nov. 27, 2024, 11:58:21 ...will update packages and refactor code.
- Nov. 27, 2024, 11:58:22 Step 6 - Transforming project bookstore.domain.tests to net8.0. Q will transform your project to ...net8.0, it will update packages and refactor code.
- Nov. 27, 2024, 11:58:23 ...net8.0, it will update packages and refactor code.
- Nov. 27, 2024, 11:58:24 Step 7 - Generate transformation summary and perform Linux readiness assessment. Q will generate a ...summary of the changes applied to your code and a report detailing how much effort remains to be ...100% Linux compatible.
- Nov. 27, 2024, 11:58:25 ...summary of the changes applied to your code and a report detailing how much effort remains to be ...100% Linux compatible.
- Nov. 27, 2024, 11:58:26 ...100% Linux compatible.
- Nov. 27, 2024, 11:58:27 Q is transforming BobsBookstore.sln in BobsBookstore: completed 0 lines of code
- Nov. 27, 2024, 11:58:28 Q is transforming BSTTiming.sln in BSTTiming: completed 0 lines of code
- Nov. 27, 2024, 11:59:14 Q is transforming BobsBookstore.sln in BobsBookstore: completed 0 lines of code
- Nov. 27, 2024, 11:59:23 Q has transformed BSTTiming.sln in BSTTiming to .NET 8.0
- Nov. 27, 2024, 11:59:26 Q committed the transformed code for BSTTiming.sln
- Nov. 27, 2024, 11:59:27 Results by project in qtBenchmark1/BSTTiming: success 1, partial 0, failed 0, skipped 0.
- Nov. 27, 2024, 11:59:28 Q finished transformation of the .NET solutions in qtBenchmark1/BSTTiming
- Nov. 27, 2024, 12:00:18 Q is transforming BobsBookstore.sln in BobsBookstore: completed 0 lines of code
- Nov. 27, 2024, 12:01:22 Q is transforming BobsBookstore.sln in BobsBookstore: completed 0 lines of code
- Nov. 27, 2024, 12:02:26 Q is transforming BobsBookstore.sln in BobsBookstore: completed 0 lines of code
- Nov. 27, 2024, 12:03:29 Q is transforming BobsBookstore.sln in BobsBookstore: completed 5982 lines of code
- Nov. 27, 2024, 12:04:33 Q has transformed BobsBookstore.sln in BobsBookstore to .NET 8.0
- Nov. 27, 2024, 12:04:39 Q committed the transformed code for BobsBookstore.sln
- Nov. 27, 2024, 12:04:40 Results by project in qtBenchmark1/BobsBookstore: success 5, partial 0, failed 0, skipped 0.
- Nov. 27, 2024, 12:04:41 Q finished transformation of the .NET solutions in qtBenchmark1/BobsBookstore
- Nov. 27, 2024, 12:04:41 **Job lifecycle** Q job has completed
- Nov. 27, 2024, 12:04:42 Q has completed transformation of .NET solutions
- Nov. 27, 2024, 12:05:39 **Job lifecycle** Q could not complete the job. Try again, omitting the steps that completed successfully.



# Amazon Q Developer transformation capabilities for .NET

PREVIEW



*Transform for .NET automates .NET application porting to Linux, handling complexity and enhancing efficiency*

# Human supervision and guardrails

SELECTION, CONFIGURATION, NOTIFICATION, VALIDATION, APPROVAL, AND ERROR CORRECTION



Selection



Approval



Notification



Error correction



Validation



Dashboard

# Benefits of large-scale .NET application porting



Accelerated transformation  
(4x faster)



At-scale visibility



Human supervision

*Transform for .NET automates code porting with 4x speed, combines human oversight for accuracy, and offers full visibility with a centralized dashboard*



*We started transforming our components responsible for license management, which would have been slated for a 6–8 month project timeline, but Amazon Q Developer’s capabilities for .NET porting helped us complete this project in a few days. Without Amazon Q Developer, this application modernization initiative, whose outcome is very valuable to us, would most likely have been be delayed or de-prioritized.*

**Jordi Martin, CTO**  
Signaturit Group





*Simplification and reducing the risks of migrations are paramount to unlocking modernization initiatives. While many organizations are advancing in application modernization, Windows-based monolithic architectures often lag due to the significant effort required for manual code transformation. This impacts our ability to support customers, as migrations from monolithic legacy databases are often tied to broader application modernization efforts. This is where Amazon Q Developer can be a game-changer. Its gen AI-powered code conversion can dramatically accelerate development cycles and reduce the resource demands of large-scale application and database modernization projects. We're excited to collaborate with the Amazon Q Developer team and look forward to empowering our customers on their transformative journey to distributed modern architectures.*

**Allen Terleto, VP Global Partners and Ecosystem**



# Takeaways

PORT YOUR WINDOWS .NET APPLICATIONS TO BECOME LINUX-READY

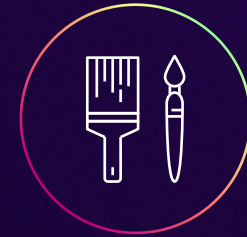
- Port .NET applications at scale from your source-code repositories
- Leverage guided porting directly in Microsoft Visual Studio
- Provide human supervision for key decision points

# Get started



## Amazon Q Developer: Transform Features

[https://aws.amazon.com/q/  
developer/features/](https://aws.amazon.com/q/developer/features/)



## Amazon Q Developer: Transform

[https://aws.amazon.com/q/  
developer/transform/](https://aws.amazon.com/q/developer/transform/)

# Related content

## **XNT401-R1: Port .NET Framework applications to Linux with Amazon Q Developer**

Thursday, Dec 5 | 12:00 PM – 2:00 PM PST | MGM Grand | Level 1 | Grand 123

# Thank you!



**Murtaza Chowdhury**

 [linkedin.com/in/MurtazaChowdhury/](https://www.linkedin.com/in/MurtazaChowdhury/)

**Raviprasad Mummidi**

 [linkedin.com/in/ravi-mummidi/](https://www.linkedin.com/in/ravi-mummidi/)

**Nits Jeganathan**

 [linkedin.com/in/nithyanandaj](https://www.linkedin.com/in/nithyanandaj)  
 @NSivaJ



Please complete the session survey in the mobile app