



aws INNOVATE

MODERN APPLICATIONS EDITION

20 October, 2022

Rapidly modernize your Microsoft .NET applications on AWS

Sriwantha Attanayake

Senior Partner Solution Architect

Amazon Web Services



Agenda

- Market observation
- Porting Assistant for .NET
- AWS Microservice Extractor for .NET
- Database modernization with Babelfish for Aurora PostgreSQL

Market observation

Why now? .NET transition

.NET Full Framework

Only works on Windows

Microsoft officially announced .NET full framework 4.8 is the **last version**
No more new features

.NET Core

Works on both Linux and Windows

Microsoft officially announced .NET core will be the future and revealed it's roadmap

.NET Core road map



Real-life story



IT Operations
updated the browser
version



Chart components
started failing



Developer

Dear Vendor,
Do you have a
new update
for the new
browser?

Yes, we have a fix.
But it's only
available on .NET
core



Chart
component
vendor



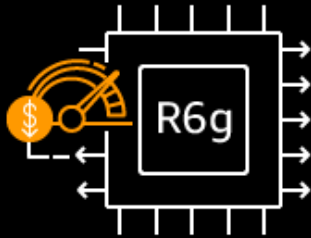
Upgraded the
entire app to
.NET core

.NET Core ♥ ARM processors

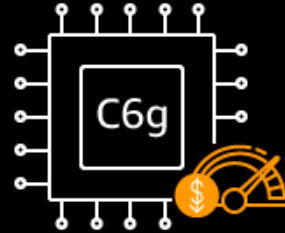
New 6th Generation Amazon EC2 instances



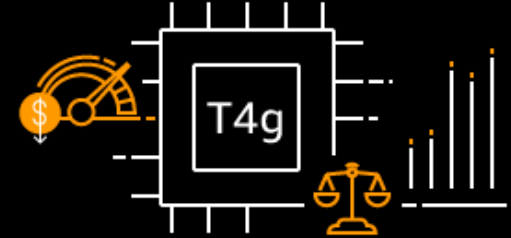
General purpose



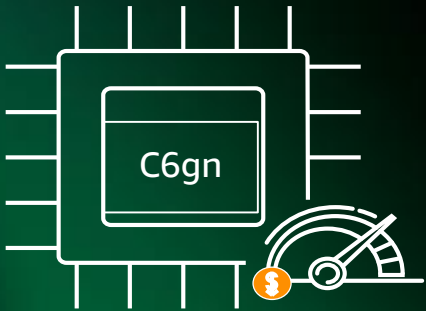
Memory optimized



Compute optimized



Burstable



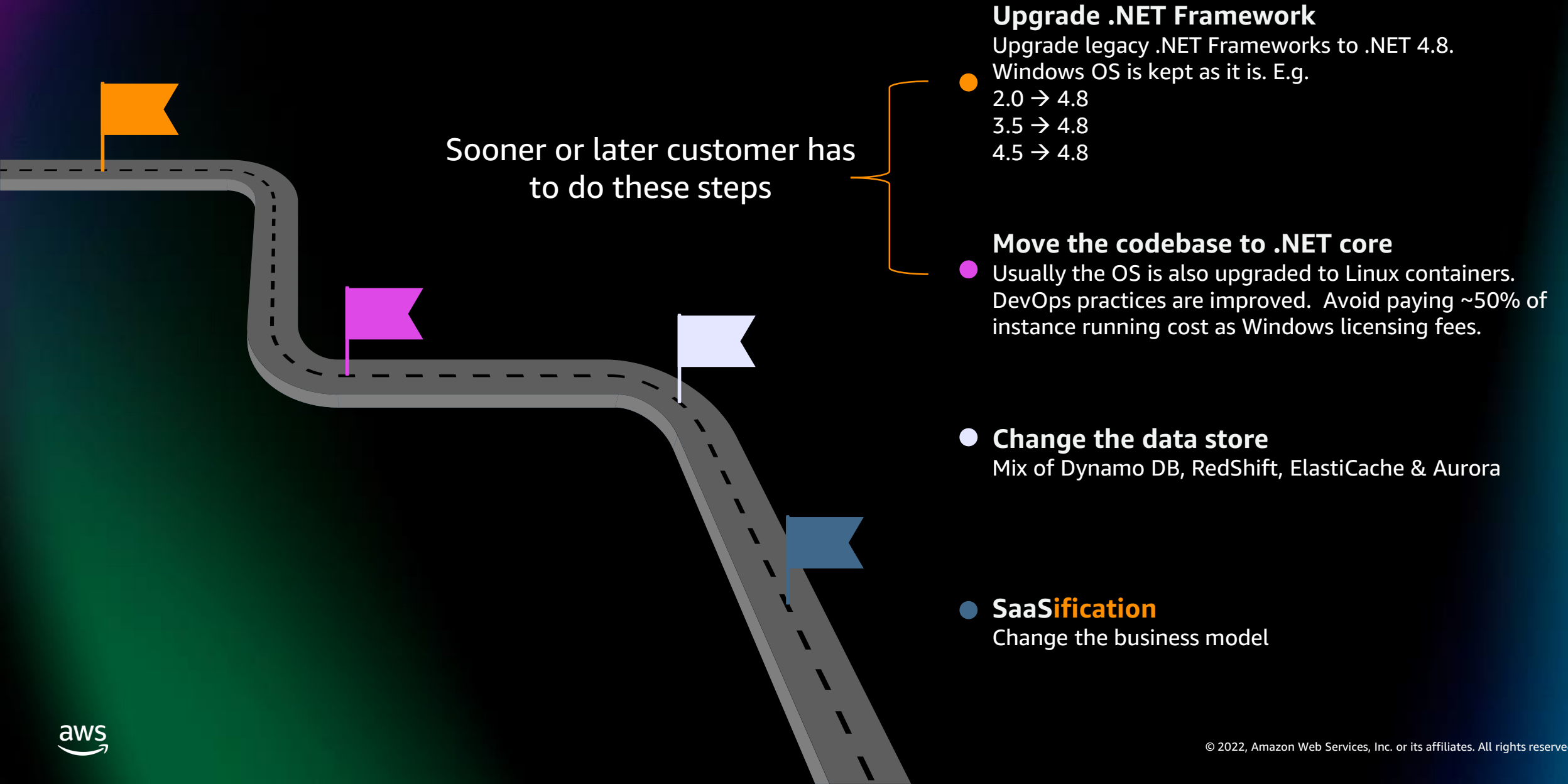
AWS Graviton2 Processors: Up to 40% better price performance over comparable current gen x86 instances

AWS Graviton3 processors: Up to 25% better compute performance, up to 2x higher floating-point performance, and up to 2x faster cryptographic workload performance compared to AWS Graviton2 processors.

ARM64 Performance in .NET 5



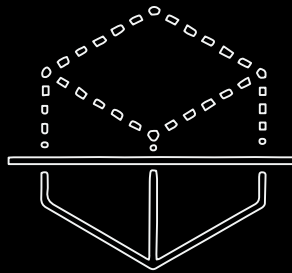
Modernization path we typically see



Porting Assistant for .NET

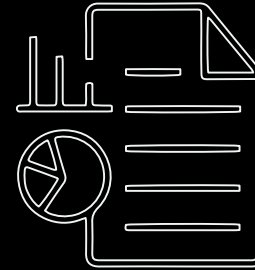
Porting Assistant for .NET

Insight and assistance
for porting from .NET
Framework to .NET Core



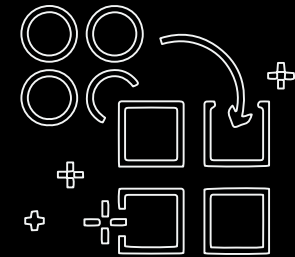
Scan

Scans .NET Framework
applications to find APIs and
NuGet Packages that are
incompatible with .NET Core



Assessment report

Generates compatibility
assessment and suggests
available replacements



Assisted porting

Updates packages and
changes project reference
files for you to start porting

Assessed solutions (4) [Info](#)[View details](#)[Actions](#) ▼[↻ Reassess solution](#)[Assess a new solution](#)

Porting Assistant for .NET has successfully assessed the following solutions for .NET Core compatibility. Improve the compatibility of your solutions by refactoring the code in your IDE.

< 1 >

	Name ▼	Ported projects ▼	Incompatible packages ▼	Incompatible APIs ▼	Portability score ▼
<input type="radio"/>	Orchard	0 of 90	58 of 76	778 of 1765	56%
<input type="radio"/>	NerdDinner	1 of 1	25 of 36	0 of 0	-
<input type="radio"/>	MyLegacyApp	0 of 1	10 of 18	11 of 11	0%
<input type="radio"/>	MyLegacyApp	0 of 1	10 of 18	11 of 11	0%

Assessment overview

The level of compatibility will affect the effort required to port your solution to .NET Core.

Portability score [Info](#)

56%

Filepath

C:\Temp\Orchard\Orchard-dev\src\Orchard.sln

Incompatible NuGet packages [Info](#)

58 of 76



☒ Incompatible
☐ Compatible

Incompatible APIs [Info](#)

778 of 1765



☒ Incompatible
☐ Compatible

Projects

Project references

NuGet packages

APIs

Source files

Projects (90) [Info](#)

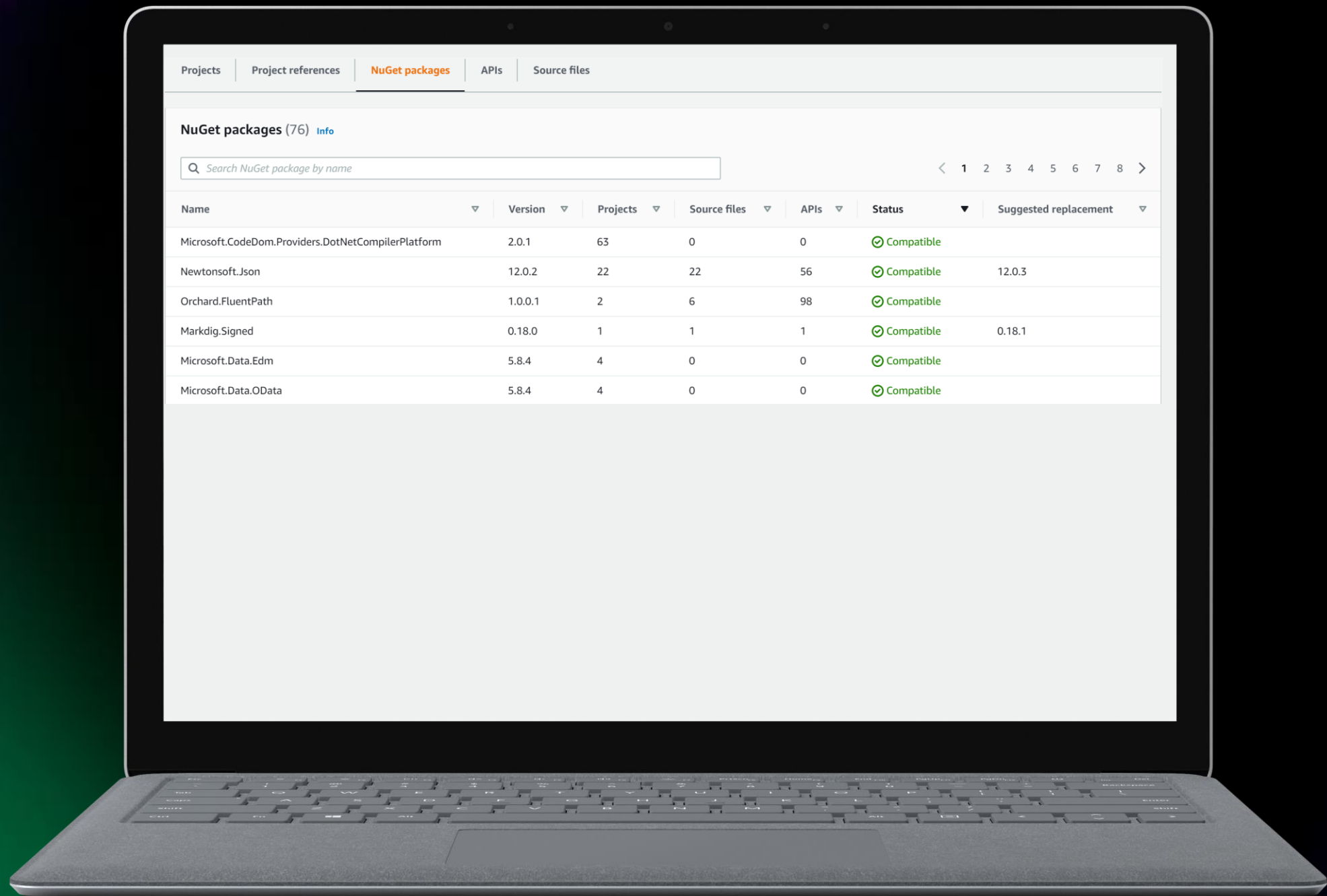
Port your project when you have reached your desired level of compatibility.

[View details](#)

[Port project](#)

< 1 2 3 4 5 6 7 8 9 >

<input type="checkbox"/>	Name ▲	Target framework ▼	Referenced projects ▼	Incompatible packages ▼	Incompatible APIs ▼	Portability score ▼	Port status ▼
<input type="checkbox"/>	Lucene	net461	1	6 of 7	33 of 65	49%	⊗ Not ported
<input type="checkbox"/>	MSBuild.Orchard.Tasks	net461	0	0 of 0	4 of 40	90%	⊗ Not ported
<input type="checkbox"/>	Markdown	net461	3	4 of 6	0 of 5	100%	⊗ Not ported
<input type="checkbox"/>	Orchard	net461	0	0 of 0	8 of 47	83%	⊗ Not ported
<input type="checkbox"/>	Orchard.Alias	net461	2	4 of 5	13 of 83	84%	⊗ Not ported

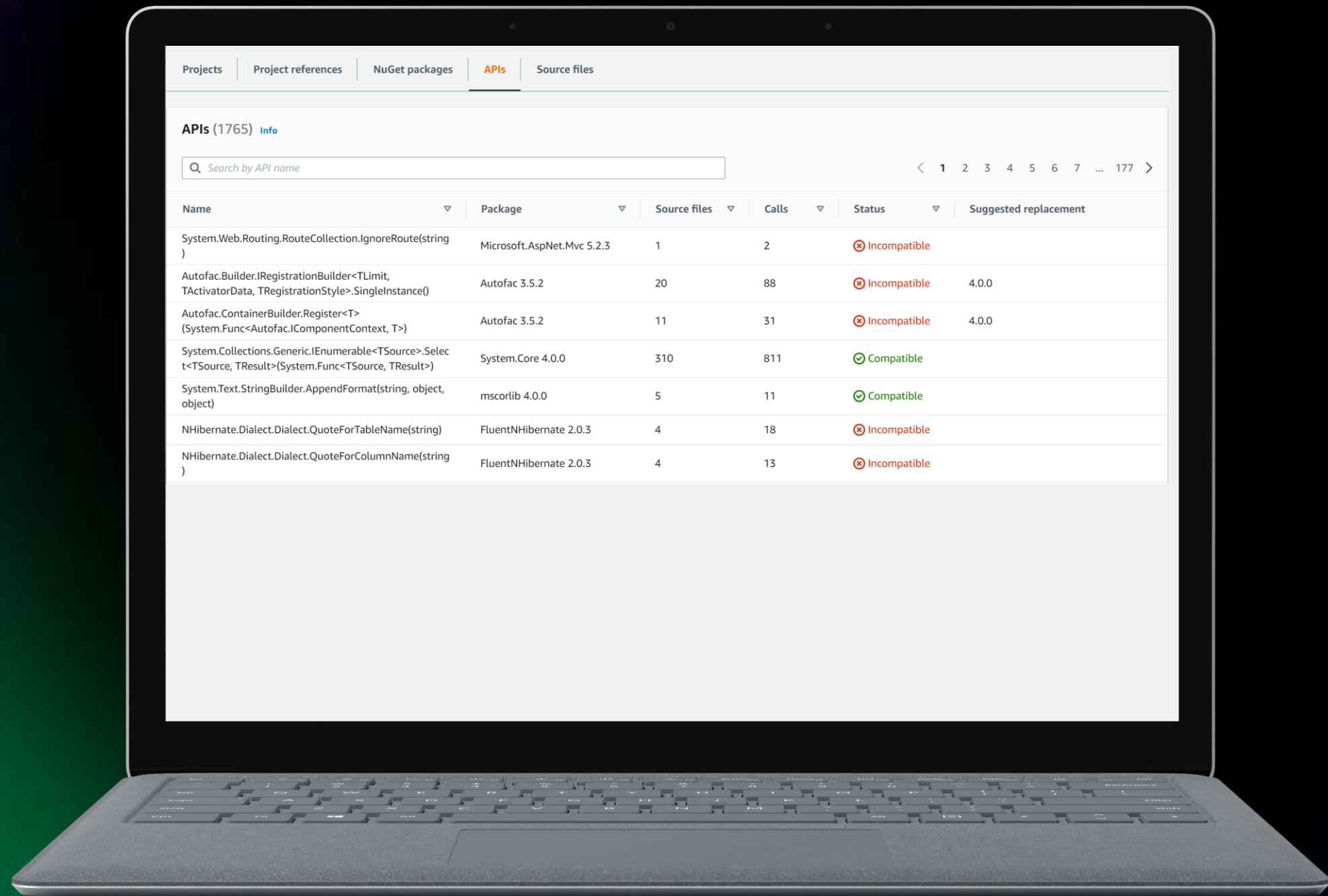


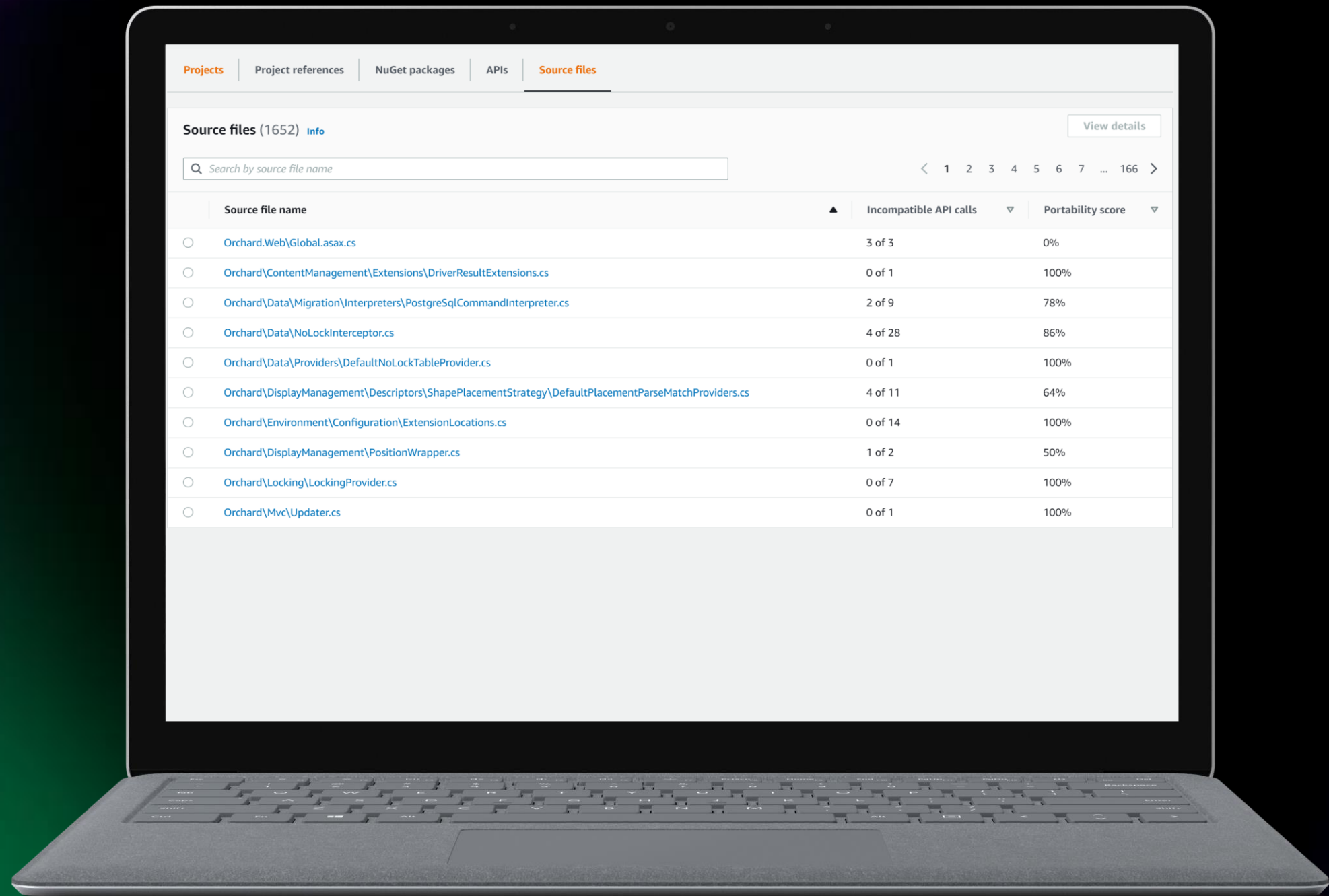
NuGet packages (76) [Info](#)

Search NuGet package by name

< 1 2 3 4 5 6 7 8 >

Name ▾	Version ▾	Projects ▾	Source files ▾	APIs ▾	Status ▾	Suggested replacement ▾
Glimpse.AspNet	1.9.2	2	0	0	⊗ Incompatible	
Glimpse.Mvc5	1.5.3	2	0	0	⊗ Incompatible	
log4net	2.0.3	6	4	50	⊗ Incompatible	2.0.6
Microsoft.AspNet.Mvc	5.2.3	74	198	1226	⊗ Incompatible	
Microsoft.AspNet.Razor	3.2.3	75	1	1	⊗ Incompatible	
Microsoft.AspNet.WebPages	3.2.3	75	1	1	⊗ Incompatible	
Microsoft.Owin	4.0.0	5	7	11	⊗ Incompatible	
Microsoft.Owin.Host.SystemWeb	4.0.0	3	2	4	⊗ Incompatible	
Microsoft.Web.Infrastructure	1.0.0	75	0	0	⊗ Incompatible	
MySQL.Data	6.7.9	1	0	0	⊗ Incompatible	6.10.0-alpha





DefaultSessionConfigurationEvents.cs [Info](#)

Orchard\Data\DefaultSessionConfigurationEvents.cs

Code

```
1 using FluentNHibernate.Automatic;
2 using FluentNHibernate.Cfg;
3
4 namespace Orchard.Data {
5     /// <summary>
6     /// Base class for session configuration
7     /// </summary>
8     public class DefaultSessionConfigurationEvents : SessionConfigurationEventsWithParameters {
9         /// <summary>
10        /// Called when an empty fluent configuration object has been created,
11        /// before applying any default Orchard config settings (alterations, conventions etc.).
12        /// </summary>
13        /// <param name="cfg">Empty fluent NH configuration object.</param>
14        /// <param name="defaultModel">Default persistence model that is about to be used.</param>
15        public override void Created(FluentConfiguration cfg, AutoPersistenceModel defaultModel) {
16            defaultModel.OverrideAll(map => {
```

Incompatible method invocation
FluentNHibernate.Automatic.AutoPersistenceModel.OverrideAll(System.Action<FluentNHibernate.Automatic.IPropertyIgnorer>)

NuGet package
FluentNHibernate 2.0.3

Replacement Strategy
2.1.1

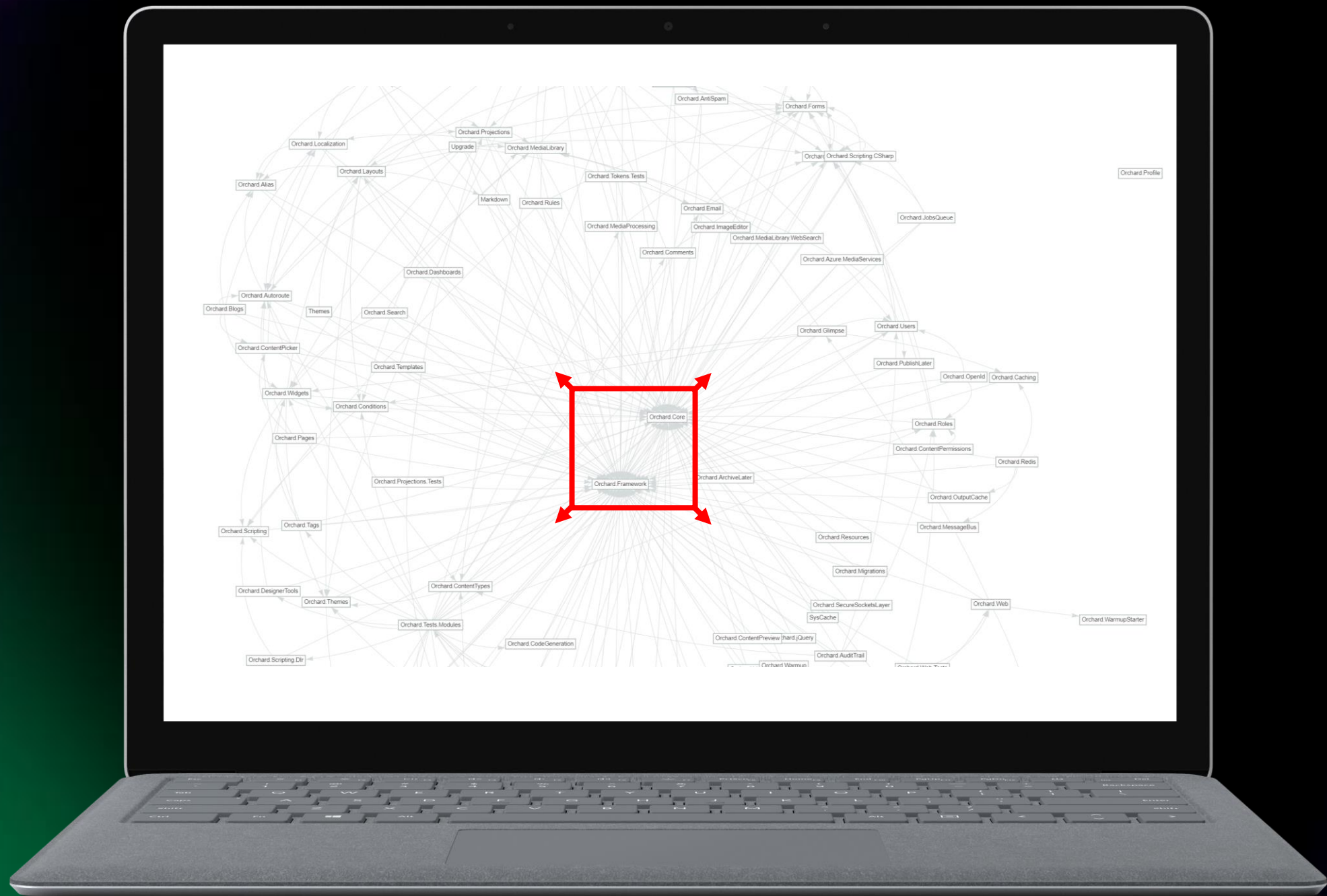
```
17         map.IgnoreProperties(x => x.MemberInfo.IsDefined(typeof(DoNotMapAttribute), false));
```

Incompatible method invocation
FluentNHibernate.Automatic.IPropertyIgnorer.IgnoreProperties(System.Func<FluentNHibernate.Member, bool>)

NuGet package
FluentNHibernate 2.0.3

Replacement Strategy
2.1.1

```
18     });
19 }
```



Monolithic vs Microservices Architecture

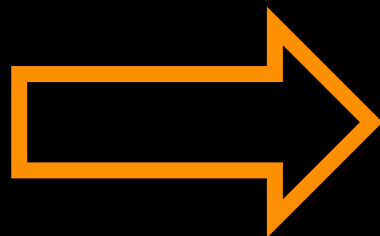
Monolithic application



Limits experimentation: feature changes require coordination among all teams involved

Impacts scalability: a spike in demand may require architectural changes to the entire app

Impacts availability: a failure in one part of the app may cause the entire app to crash



Refactored microservices



Independent teams: can build, test, and release new functionality independently

Granular scaling: without replicating the entire application

Localized failures: don't risk uptime for the whole application

Challenges when moving to microservices



Difficult to identify parts of the application that need to be extracted as separate services



Difficult to group functionality based on business domains and related processes



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

AWS Microservice Extractor for .NET

Demo

ec2-13-239-73-100.ap-southeast-2.compute.amazonaws.com

aws.amazon.com/microservice-extractor/

aws

ProductsSolutionsPricingDocumentationLearnPartner NetworkAWS MarketplaceCustomer EnablementEventsExplore More

Contact UsSupportEnglishMy AccountSign InCreate an AWS Account

AWS Microservice Extractor for .NETOverviewFeaturesGetting StartedFAQsPartners

AWS Microservice Extractor for .NET


Simplify refactoring .NET applications

Get started with Microservice Extractor for .NETDownload Microservice Extractor for .NET

Free AWS Training | Advance your career with AWS Cloud Practitioner Essentials—a free, six-hour, foundational course »

AWS Microservice Extractor for .NET simplifies the process of refactoring older monolithic applications into smaller code projects to build a microservices-based architecture. Modernize and transform your applications with an assistive tool that analyzes source code and runtime metrics to create a visual representation of your application and its dependencies. With Microservice Extractor providing automated recommendations, developers get guided experience to refactor legacy applications.

Microservice Extractor now identifies common extraction candidates using heuristics based techniques, and highlights those in visualization. These recommendations can be used as is or used as a starting point to extract microservices off of monolithic codebase. Thus, automated recommendations from Microservice Extractor helps to speed up refactoring large applications even if the developer is unfamiliar with the codebase. You can also extract the codebase into separate projects that teams can develop, build, and operate independently to improve agility, uptime, and scalability.



Introducing AWS Microservice Extractor for .NET | Amazon Web Services (1:01)

Windows taskbar icons

2:44 AM 8/30/2022

Visualization canvas (1)

Select canvas Default canvas

Actions

Extract group

Alternate views Reset

Filter or search resources by property or value

These three classes depend on Inventory

We are going to extract Inventory as a microservice

Inventory depends on these three classes

Node details

General

Selected node (1)

< 1 > ⚙

Source node	Edges	Dependency node
GadgetsOnline.Services.Inventory	←	GadgetsOnline.Controllers.HomeController
GadgetsOnline.Services.Inventory	←	GadgetsOnline.Controllers.ShoppingCartController
GadgetsOnline.Services.Inventory	←	GadgetsOnline.Controllers.StoreController
GadgetsOnline.Services.Inventory	→	GadgetsOnline.Models.GadgetsOnlineEntities
GadgetsOnline.Services.Inventory	→	GadgetsOnline.Models.Product
GadgetsOnline.Services.Inventory	→	GadgetsOnline.Models.Category

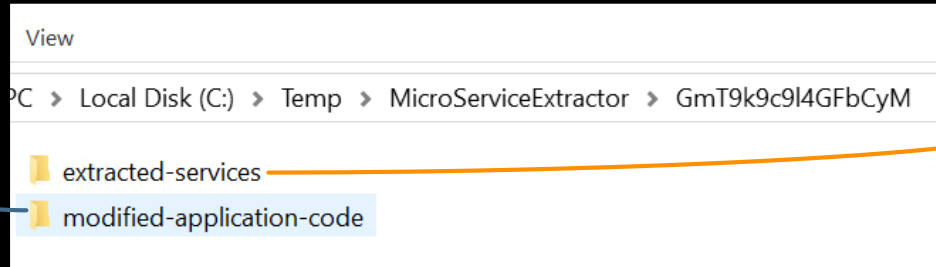
Node Group Starting point Non extractable Node Types POCO User Interface Data Access Service Access Multi-type Group Collapsed group Incoming dependency Outgoing dependency

View Group classification

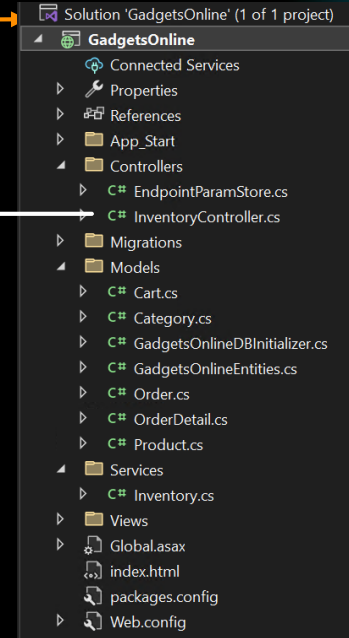
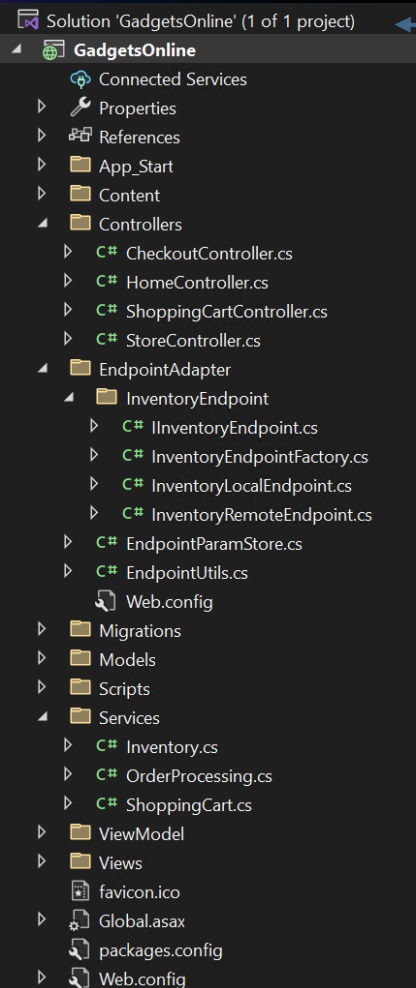
Service extraction

Rest of the modified application

Inventory service



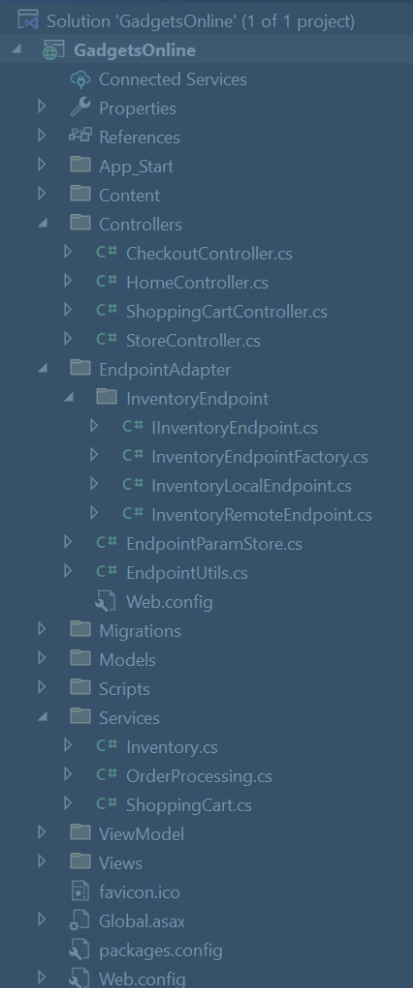
Service
API



Service extraction

Rest of the modified application

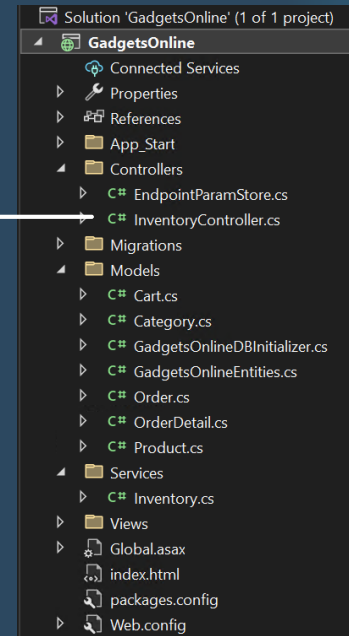
Inventory service



```
namespace GadgetsOnline.Controllers
{
    [RoutePrefix ( "api/Inventory" )]
    public class InventoryController : ApiController
    {
        [Route("GetBestSellers_6da88c34")]
        [ResponseType(typeof(List<Product>))]
        [HttpPost]
        public IHttpActionResult GetBestSellers_6da88c34Wrapper(dynamic endpointContainer)
        {
            try
            {
                dynamic ctorContainer = EndpointParamStore.GetConstructorContainer(endpointContainer);
                dynamic methodContainer = EndpointParamStore.GetMethodContainer(endpointContainer);
                Inventory myInstance = null;
                string ctorParamHash = EndpointParamStore.GetConstructorParamHash(ctorContainer);
                // Initialize the right constructor
                if (ctorParamHash.Equals("e3b0c442"))
                {
                    myInstance = new Inventory();
                }

                // Retrieve Method parameters
                int count = methodContainer.count;
                return Ok(myInstance.GetBestSellers(count));
            }
            catch (Exception e)
            {
                Console.WriteLine(e.ToString());
                return InternalServerError(e);
            }
        }

        [Route("GetAllCategories_e3b0c442")]
        [ResponseType(typeof(List<Category>))]
        [HttpPost]
        public IHttpActionResult GetAllCategories_e3b0c442Wrapper(dynamic endpointContainer)
        {
            try
            {
                dynamic ctorContainer = EndpointParamStore.GetConstructorContainer(endpointContainer);
                dynamic methodContainer = EndpointParamStore.GetMethodContainer(endpointContainer);
                Inventory myInstance = null;
                string ctorParamHash = EndpointParamStore.GetConstructorParamHash(ctorContainer);
                // Initialize the right constructor
                if (ctorParamHash.Equals("e3b0c442"))
                {
                    myInstance = new Inventory();
                }
            }
        }
    }
}
```



Service extraction

Rest of the modified application

Inventory service

Two implementations of the same interface

Service API

```
namespace GadgetsOnline.EndpointAdapter
{
    public interface IInventoryEndpoint
    {
        List<Product> GetBestSellers(int count);
        List<Category> GetAllCategories();
        List<Product> GetAllProductsInCategory(string category);
        Product GetProductById(int id);
        string GetProductNameById(int id);
    }
}
```

```
namespace GadgetsOnline.EndpointAdapter
{
    public class InventoryLocalEndpoint : IInventoryEndpoint
    {
        private readonly Inventory originalInstance;
        public InventoryLocalEndpoint()
        {
            originalInstance = new Inventory();
        }

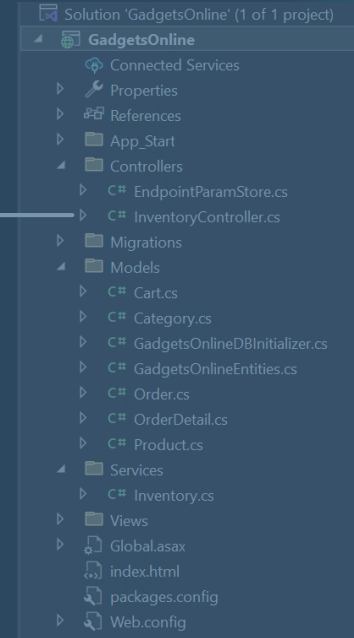
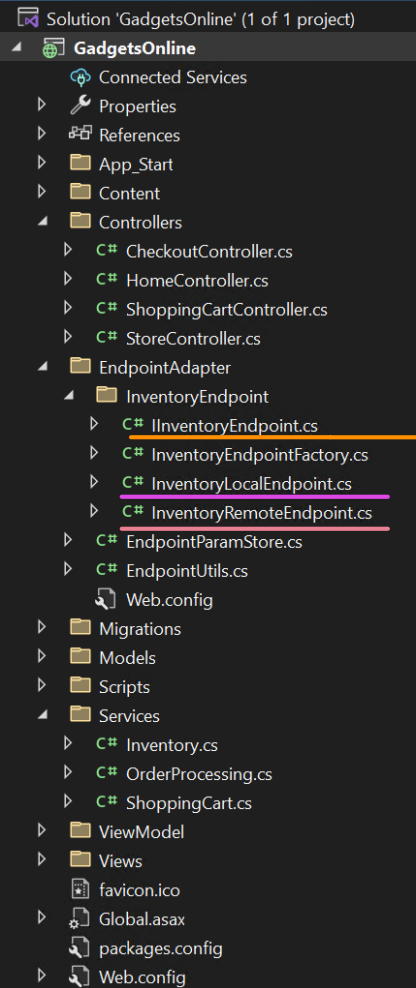
        public List<Product> GetBestSellers(int count)
        {
            return originalInstance.GetBestSellers(count);
        }

        public List<Category> GetAllCategories()
        {
            return originalInstance.GetAllCategories();
        }
    }
}
```

```
namespace GadgetsOnline.EndpointAdapter
{
    public class InventoryRemoteEndpoint : IInventoryEndpoint
    {
        private string endpointAddress;
        private static readonly string endpointPrefix = "api/Inventory";
        private readonly dynamic ctorParams;
        public InventoryRemoteEndpoint(string remoteEndpoint)
        {
            endpointAddress = remoteEndpoint;
            ctorParams = EndpointParamStore.NewContainer();
            EndpointParamStore.SetConstructorParamHash(ctorParams, "e3b0c442");
        }

        public List<Product> GetBestSellers(int count)
        {
            return GetBestSellersRemote("GetBestSellers_6da88c34", count);
        }

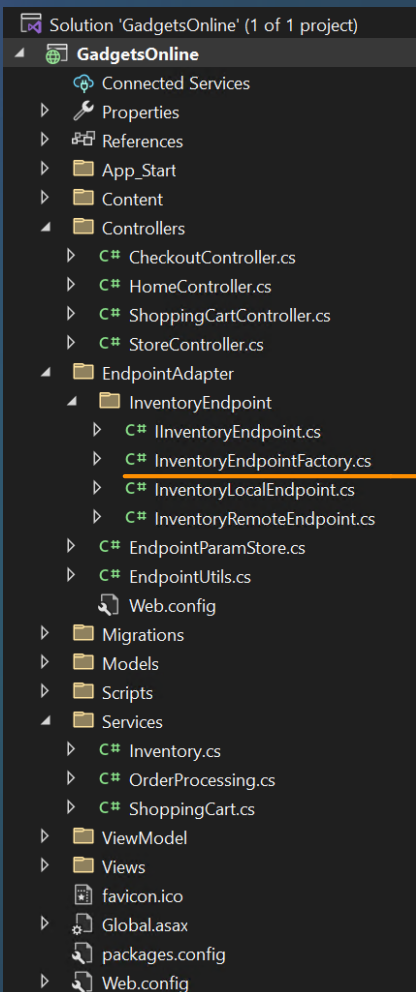
        public List<Product> GetBestSellersRemote(string endpointSuffix, int count)
        {
            try
            {
                dynamic methodParams = EndpointParamStore.NewContainer();
                methodParams.count = count;
                HttpClient client = EndpointUtils.InitializeClientWithJsonMedia(endpointAddress);
                HttpResponseMessage response = EndpointUtils.PostAndGetResponse<dynamic>(client, endpointPrefix + endpointSuffix, methodParams);
                return EndpointUtils.ReadResponse<List<Product>>(response);
            }
            catch (Exception e)
            {
                throw e;
            }
        }
    }
}
```



Service extraction

Rest of the modified application

Inventory service



```
namespace GadgetsOnline.EndpointAdapter
{
    public class InventoryEndpointFactory
    {
        // Knob that controls branch by abstraction (local / remote call)
        private static bool useRemoteEndpoint = false;

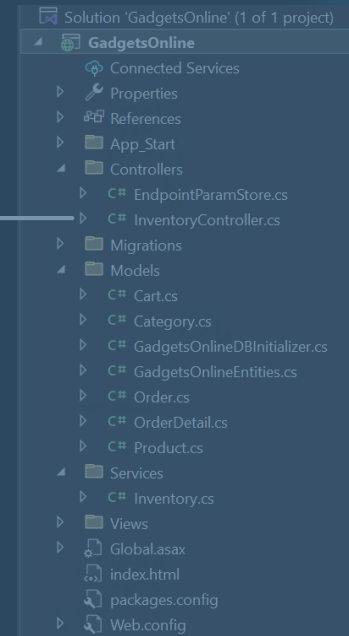
        // Endpoint address of remote endpoint
        private static string remoteEndpoint = "http://localhost:8080";

        private static bool initialized = false;

        // Initiate uses config file to read the value of this knob and endpoint
        public static void initEndpointFactory()
        {
            if (initialized)
            {
                return;
            }
            remoteEndpoint = EndpointUtils.getRemoteEndpoint();
            useRemoteEndpoint = EndpointUtils.isRemoteRouting();
            initialized = true;
        }

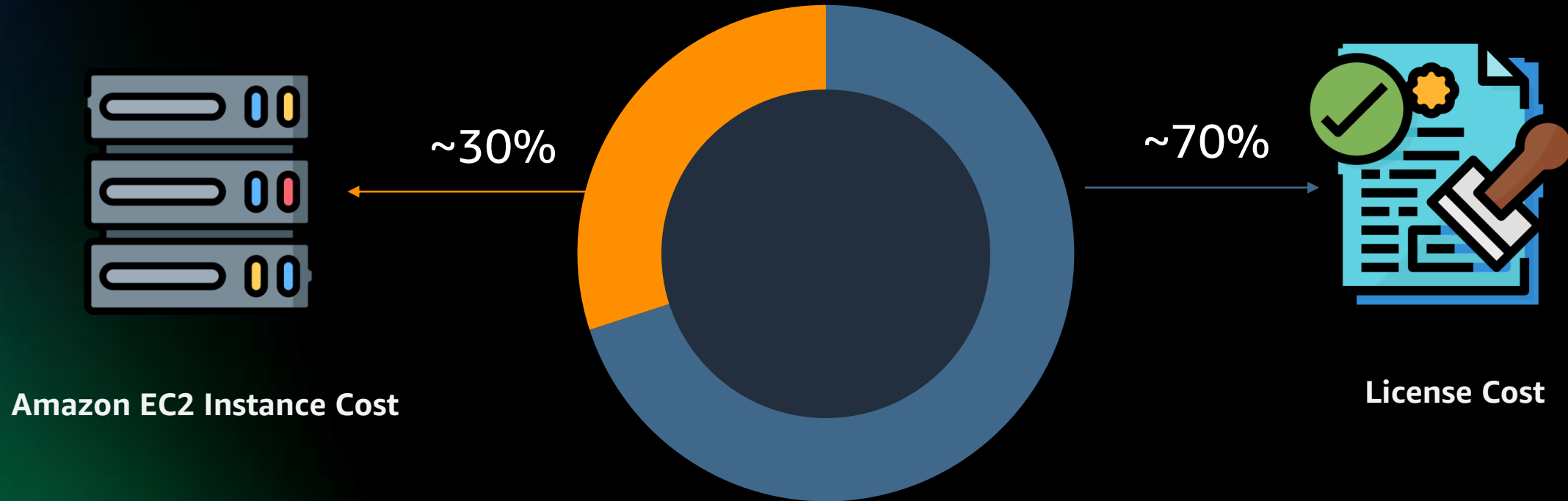
        public static IInventoryEndpoint GetEndpointAdapter()
        {
            initEndpointFactory();
            if (useRemoteEndpoint)
            {
                return new InventoryRemoteEndpoint(remoteEndpoint);
            }
            else
            {
                return new InventoryLocalEndpoint();
            }
        }
    }
}
```

Service
API



Microsoft SQL Server modernization

Microsoft SQL Server standard cost on Amazon EC2

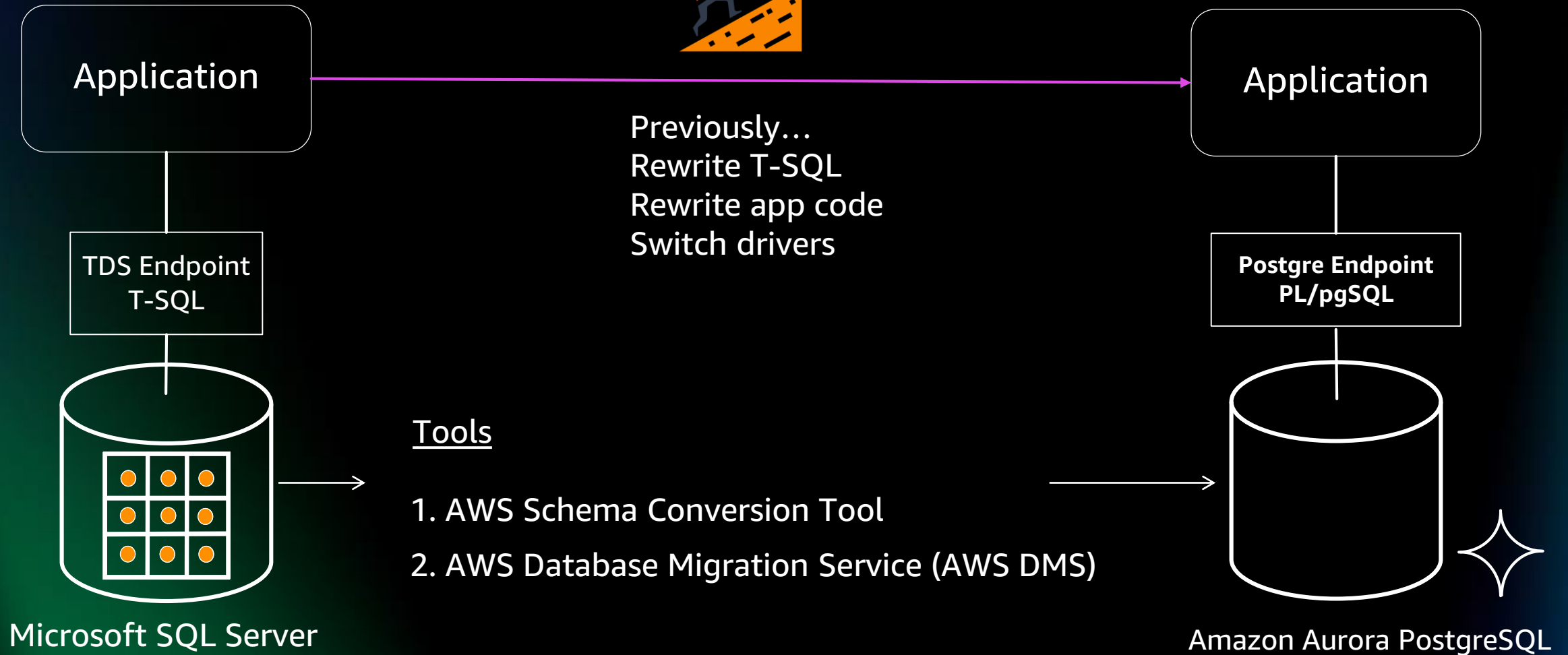


Amazon EC2 Instance Cost

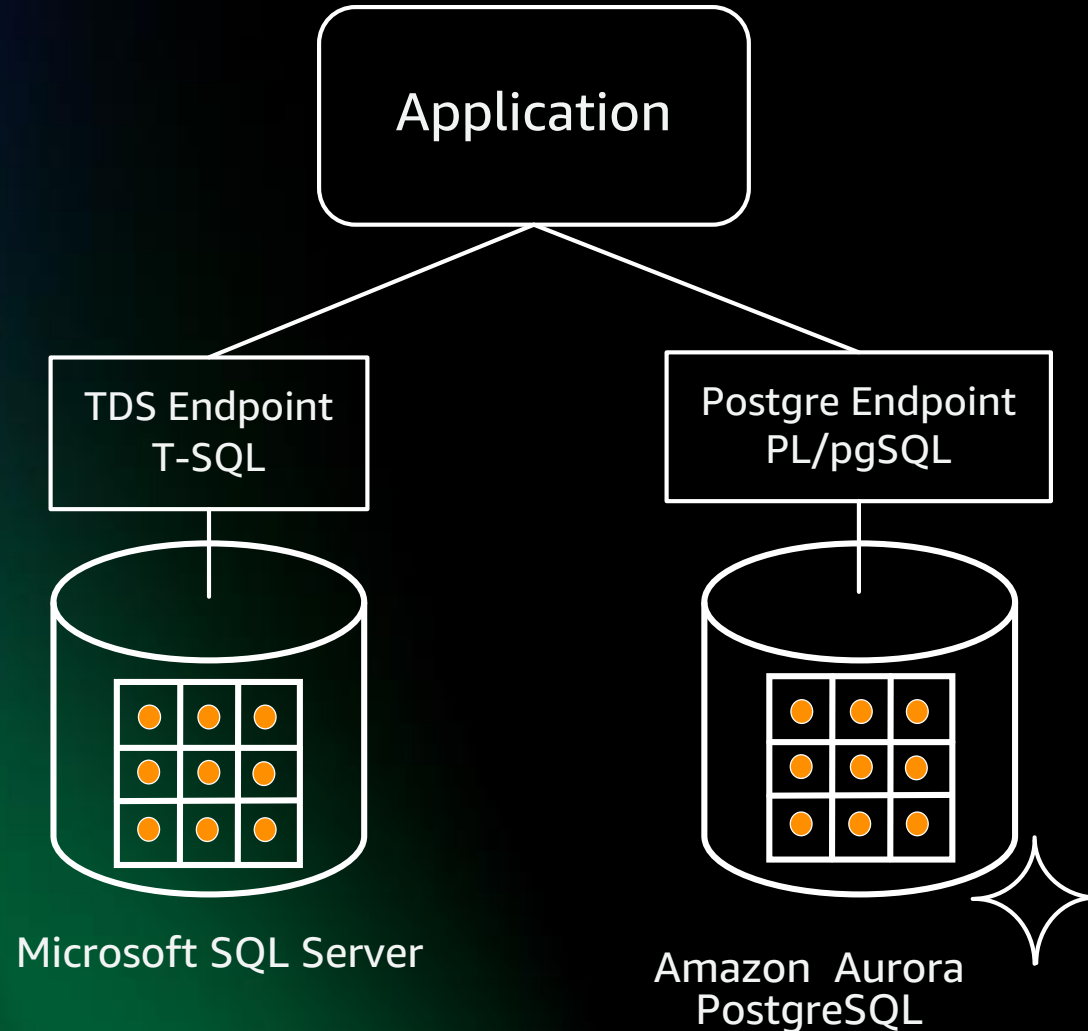
License Cost

R5.2xlarge instance + 100GB EBS Running on Linux + Sydney

Challenges in migrating from commercial to open source



Imagine if you could . . .



- Legacy application code remains written for SQL Server
- Client drivers do not need to change
- New application code written directly to PostgreSQL

Babelfish

Run Microsoft SQL Server applications on PostgreSQL with little to no code changes

Keep existing queries



Translation layer enables Aurora PostgreSQL to understand Microsoft SQL Server's proprietary T-SQL

Accelerate migrations



Lower risk and complete migrations faster, saving you months to years of work

Freedom to innovate



Run T-SQL code side-by-side with new open source functionality and continue developing with familiar tools

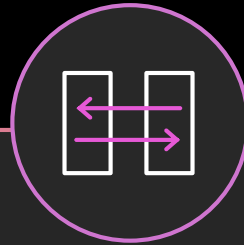
Babelfish for PostgreSQL (Open-Source Project)

**Customize and add
new features**



Contribute to help steer the
direction of Babelfish

**Apache 2.0
license**



Use it for any purpose, innovate
and distribute your modifications
with confidence

Available on GitHub



Is community-driven and
provides transparency into
the feature roadmap

Call to action

Run the free Babelfish App Modernization Workshop

- ~3 to 4 Hours
- Self serviced & self paced
- We will provide free AWS accounts to try the labs

Run a PoC with Babelfish Compass

- ~ Takes about 15min
- Do an assessment & check the complexity



Babelfish App Modernization Workshop

Babelfish Compass reports are comprehensive and detailed

Table Of Contents

- [Applications Analyzed](#)
- [Assessment Summary](#)
- [Object Count](#)
- [Summary of SQL Features 'Not Supported'](#)
- [Summary of SQL Features 'Review Manually'](#)
- [Summary of SQL Features 'Review Semantics'](#)
- [Summary of SQL Features 'Review Performance'](#)
- [Summary of SQL Features 'Ignored'](#)
- [Summary of SQL Features 'Supported'](#)
- [X-ref: 'Not Supported' by SQL feature](#)
- [X-ref: 'Review Manually' by SQL feature](#)
- [X-ref: 'Review Semantics' by SQL feature](#)
- [X-ref: 'Review Performance' by SQL feature](#)
- [X-ref: 'Ignored' by SQL feature](#)
- [X-ref: 'Supported' by SQL feature](#)
- [X-ref: 'Not Supported' by object](#)
- [X-ref: 'Review Manually' by object](#)
- [X-ref: 'Review Semantics' by object](#)
- [X-ref: 'Review Performance' by object](#)
- [X-ref: 'Ignored' by object](#)
- [X-ref: 'Supported' by object](#)

Assessment Summary

[Back to Table of Contents](#)

#applications	:	1
#input files	:	1
#SQL batches	:	78
#lines SQL/DDL processed	:	333
#lines SQL in objects	:	0
total #SQL features	:	255
Supported	:	145
Not Supported	:	12
Review Semantics	:	14
Ignored	:	84

SQL Features Report

SQL features 'Not Supported' in Babelfish v.1.0.0

[Back to Table of Contents](#)

DDL (12/1)

① ALTER TABLE..CHECK CONSTRAINT : 12

SQL features 'Review Manually' in Babelfish v.1.0.0

[Back to Table of Contents](#)

-no items to report-

SQL features 'Review Semantics' in Babelfish v.1.0.0

[Back to Table of Contents](#)

DDL (14/1)

① Constraint PRIMARY KEY/UNIQUE, CLUSTERED, in CREATE TABLE : created as NONCLUS

SQL features 'Review Performance' in Babelfish v.1.0.0

[Back to Table of Contents](#)

DDL (14/1)

① Constraint PRIMARY KEY/UNIQUE, CLUSTERED, in CREATE TABLE : created as NONCLUSTERED, no physical row

SQL features 'Review Performance' in Babelfish v.1.0.0

[Back to Table of Contents](#)

-no items to report-



You can find where exactly the issues exist

--- X-ref: 'Not Supported' by SQL feature -----

[Back to Table of Contents](#)

ALTER TABLE..CHECK CONSTRAINT: dbo.MySite_AliasPartIndex (DDL, 1)
TABLE dbo.MySite_AliasPartIndex, line [277](#)

ALTER TABLE..CHECK CONSTRAINT: dbo.MySite_AutoroutePartIndex (DDL, 1)
TABLE dbo.MySite_AutoroutePartIndex, line [282](#)

ALTER TABLE..CHECK CONSTRAINT: dbo.MySite_ContainedPartIndex (DDL, 1)
TABLE dbo.MySite_ContainedPartIndex, line [287](#)

ALTER TABLE..CHECK CONSTRAINT: dbo.MySite_ContentItemIndex (DDL, 1)
TABLE dbo.MySite_ContentItemIndex, line [292](#)

ALTER TABLE..CHECK CONSTRAINT: dbo.MySite_DeploymentPlanIndex (DDL, 1)
TABLE dbo.MySite_DeploymentPlanIndex, line [297](#)

ALTER TABLE..CHECK CONSTRAINT: dbo.MySite_LayerMetadataIndex (DDL, 1)
TABLE dbo.MySite_LayerMetadataIndex, line [302](#)

ALTER TABLE..CHECK CONSTRAINT: dbo.MySite_TaxonomyIndex (DDL, 1)

```
262 ALTER TABLE [dbo].[MySite_Document] ADD DEFAULT ((0)) FOR [Version]
263 GO
264 ALTER TABLE [dbo].[MySite_TaxonomyIndex] ADD DEFAULT ((1)) FOR [Published]
265 GO
266 ALTER TABLE [dbo].[MySite_TaxonomyIndex] ADD DEFAULT ((0)) FOR [Latest]
267 GO
268 ALTER TABLE [dbo].[MySite_UserIndex] ADD DEFAULT ((1)) FOR [IsEnabled]
269 GO
270 ALTER TABLE [dbo].[MySite_UserIndex] ADD DEFAULT ((0)) FOR [IsLockoutEnabled]
271 GO
272 ALTER TABLE [dbo].[MySite_UserIndex] ADD DEFAULT ((0)) FOR [AccessFailedCount]
273 GO
274 ALTER TABLE [dbo].[MySite_AliasPartIndex] WITH CHECK ADD CONSTRAINT [MySite_FK_AliasPartIndex] FOREIGN KEY([Doc
275 REFERENCES [dbo].[MySite_Document] ([Id])
276 GO
277 ALTER TABLE [dbo].[MySite_AliasPartIndex] CHECK CONSTRAINT [MySite_FK_AliasPartIndex]
278 GO
279 ALTER TABLE [dbo].[MySite_AutoroutePartIndex] WITH CHECK ADD CONSTRAINT [MySite_FK_AutoroutePartIndex] FOREIGN
280 REFERENCES [dbo].[MySite_Document] ([Id])
281 GO
282 ALTER TABLE [dbo].[MySite_AutoroutePartIndex] CHECK CONSTRAINT [MySite_FK_AutoroutePartIndex]
283 GO
284 ALTER TABLE [dbo].[MySite_ContainedPartIndex] WITH CHECK ADD CONSTRAINT [MySite_FK_ContainedPartIndex] FOREIGN
285 REFERENCES [dbo].[MySite_Document] ([Id])
```

Call to action

Run the free Babelfish App Modernization Workshop

- Self-service & self-paced
- We will provide free AWS accounts to try the labs

Run a PoC with Babelfish Compass

- Generate an assessment report based on your schema
- Check the compatibility & complexity

Plug in a Partner

We have capable partners with proven experience to modernize databases

Access AWS PoC Funding

Build your business case faster



Babelfish App Modernization Workshop

How to get started with Babelfish Workshop

- Share your **use case** and let us know your interest for the Babelfish App Modernization Workshop in the **feedback form**
- We will organize a training for your team, and provide complimentary AWS accounts to try the labs
- Get a chance to receive USD\$25 worth of AWS credits



Babelfish App Modernization Workshop

Dear AWS Team,

I attended Rapidly Modernize your Microsoft .NET applications on AWS session at AWS Innovate Modern Application edition.

I have an HR application running on Microsoft SQL Server. I would like to check whether Babelfish will work with that and would like to do a PoC.

Can you organize the Babelfish App Modernization Workshop for my team.

Thanks,
Bob Smith
Tech Lead – XYZ company

Recap and resources

- Market observation
- Porting Assistant for .NET
- AWS Microservice Extractor for .NET
- Database modernization with Babelfish for Aurora PostgreSQL



AWS Microservice Extractor for .NET Labs



Babelfish App Modernization Workshop

Visit the Modern Applications resource hub

Dive deeper with these resources to help you develop an effective plan for your modernization journey.

- Build modern applications on AWS
- Business value of cloud modernization
- An introduction to event-driven architectures
- Accelerate full-stack web and mobile app development
- Determining the total cost of ownership: Comparing serverless and server-based technologies
- Building event-driven architectures with AWS
- Continuous learning, continuous modernization



<https://tinyurl.com/modern-apps-aws>

Visit resource hub

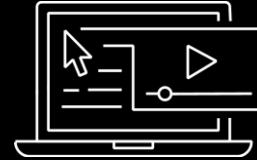


AWS Training and Certification

Get started with Free Digital Training for you and your team today



Achieve key milestones and plan your next steps with the AWS Modern Application skills training



Access 500+ free digital courses with [AWS Skill Builder](#)



Earn an industry-recognized credential:
[AWS Certified Developer – Associate](#)
[AWS Certified DevOps – Professional](#)



Create a self-paced learning roadmap
[AWS ramp-up guide - Developer](#)
[AWS ramp-up guide - DevOps](#)

Thank you for attending AWS Innovate Modern Applications Edition

We hope you found it interesting! A kind reminder to **complete the survey**.
Let us know what you thought of today's event and how we can improve the event
experience for you in the future.



aws-apj-marketing@amazon.com



twitter.com/AWSCloud



facebook.com/AmazonWebServices



youtube.com/user/AmazonWebServices



slideshare.net/AmazonWebServices



twitch.tv/aws

Thank you!

Sriwantha Attanayake
www.linkedin.com/in/sriwantha

