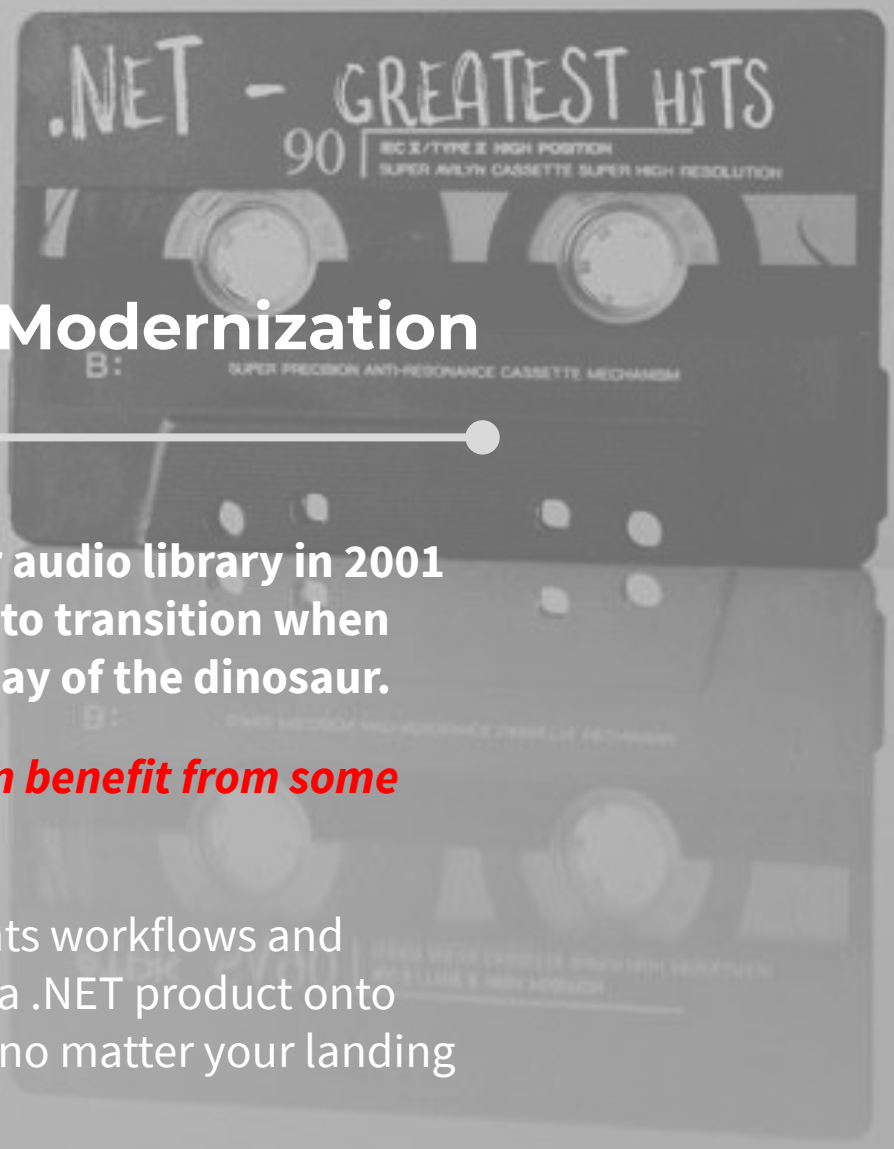


.NET Migration & Modernization

Just like you upgraded your audio library in 2001 with iTunes, or were forced to transition when cassette players went the way of the dinosaur.

Legacy .NET applications can benefit from some modernization.

While this case study highlights workflows and deep-dive details for moving a .NET product onto AWS, the insight is beneficial no matter your landing destination.

Microsoft .NET → aws

</SOURCEFUSE>

.NET Migration & Modernization

.NET is one of the most popular software frameworks in the world. **Designed to perform on Windows, which can become expensive and subject to Vendor lock-in. Making future development dependent on Microsoft.**

To mitigate these issues more and more .NET web applications have been moving to AWS-provided or self-managed IIS web servers.

The fundamentals around your migration and modernization process:

- Understand the benefits of your new system
- Asses gaps in your existing system
- Understand and plan migration patterns
- Migrate and validate

Some Background.

Currently there are over 700 different Windows ISV (Independent Software Vendor) listings in the AWS marketplace, more than 50 different Amazon Machine Images for Windows workloads, and over 57% of Windows workloads run on AWS.

Just in case you need further affirmation that Windows workloads and frameworks can perform outside of Microsoft environments. These stats should help you feel confident in external support and resources.

This guide is beneficial for everyone -- those looking for technical details on how to execute, as well as those searching for the value within the process.

The following sections assume basic knowledge of:

1. Amazon EC2
2. Microsoft Windows
3. .NET application development
4. Load balancing
5. Microsoft SQL server databases


aws partner network
Advanced
Consulting
Partner
DevOps Competency
Public Sector Partner
Digital transformation generates income.
$100M
Digitally transformed enterprises find or generate 100-million in additional income.
Keystone Strategy, Data & Analytics Maturity Model & Business Value: Milestone 5
</SOURCEFUSE>

Understand the benefits of your new system.

When looking to assess the benefits of your new system it's important to consider everything from the bottom-up.

Ask yourself the below questions when identifying benefits of your landing destination:

- Will you have to provision resources manually?
- Can it support multiple environments?
- How easy is it to share code with teammates, and test the code to ensure accuracy?
- What tools are available?
    - Configuration orchestration tools are designed to automate the deployment of infrastructure.
    - Configuration management tools are designed to help configure the software and systems on infrastructure that has been provisioned.

Outside of the above questions, you should be asking yourself a lot of the same questions you would be, when evaluating any cloud or infrastructure provider.

How easy is it to:

- Scale
- Save money
- Save time
- Monitor your assets
- Improve security practices and protocols

Understand the benefits of your new system.

The microservices evolution

Moving your .NET application to a cloud or hybrid infrastructure allows you to **start evolving to a Microservices Architecture,** with the ability to phase in Containers and a Serverless architecture for Microsoft .NET Applications.

Easily automate deployment

AWS Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with .NET and IIS to meet changing market conditions. **Elastic Beanstalk for .NET makes it easier to deploy, manage, and scale your ASP.NET web applications.**

Create a CI/CD pipeline

Automatically deploy a continuous integration / continuous delivery (CI/CD) pipeline using standard Microsoft Windows technologies.

Asses gaps in your existing system.

I'm sure any developer or DevOps practitioner can rattle off a dozen gaps of any current system without hesitation. But to truly identify current shortcomings it can be helpful to ask questions tailored to increasing your business agility and reduce costs or consolidating data resources.

- Would altering the architecture of you application improve usability?
- What current visibility do you have into solution architecture and corresponding performance?
- What workarounds are currently created?
    - Are you calling into libraries compiled for .NET Framework from .NET Core?
- How long is your current development/release cycle?
- Is anything currently automated, like administrative tasks?
- Do you currently have end-to-end encryption or multiple security layers?
- What is your current external support network?

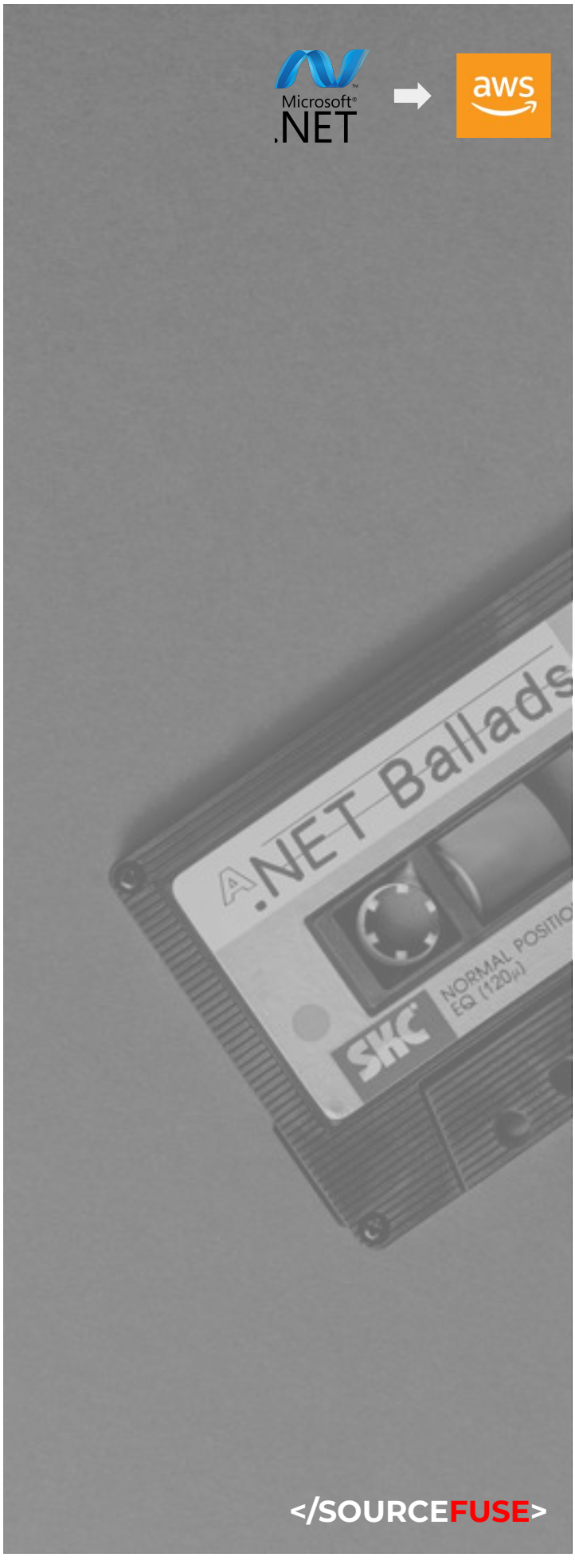
Understand and plan migration patterns.

Consider where your migration and modernization journey fits into your organization's larger business strategy and find opportunities for alignment.

When planning to deploy a .NET web application outside of your on-premises or Microsoft environment it can be unclear and a bit confusing. Especially when deciding on which method or migration pattern will allow you to get up and running quickly.

The 6 most common application migration strategies are:

1. Retain (re-visit)
2. Retire
3. Re-host ("lift and shift.")
4. Re-platform ("lift, tinker, and shift.")
5. Re-purchase
6. Re-factor / Re-architect



Understand and plan migration patterns.

Although there are 6 common migration patterns, 70% of initial cloud migrations fall into 2 patterns -- Re-host and Re-platform.

1. Retain (re-visit)
2. Retire
3. Re-host (“lift and shift”)
4. Re-platform (“lift, tinker, and shift”)
5. Re-purchase
6. Re-factor / Re-architect

70% of initial cloud migration strategies.

Re-host “lift and shift”

Re-hosting means that you are moving an existing application in its current state from its current infrastructure and placing it “as is” in a new infrastructure environment, away from co-located data centers and into a hybrid, public or private cloud.

In large-scale, legacy migrations, organizations can see immediate cost savings of up to 30% without any optimizations.

Re-hosting can be automated or done manually and allows you to move quickly in order to meet business objectives.

Re-platform “lift, tinker, and shift”

Not changing the core architecture of the application, but “tinkering” with certain aspects to take advantage of your new landing zone. Leveraging cloud native services like managed databases, containers, and storage to save even more, while adding elasticity to your infrastructure and business.

For example, reduce the amount of time you spend managing database instances by migrating to a database-as-a-service platform.

Understand and plan migration patterns.

Your migration strategy should guide your teams to move quickly and independently. Applying project management best practices that include clear budgets, timelines, and business outcomes supports this goal.

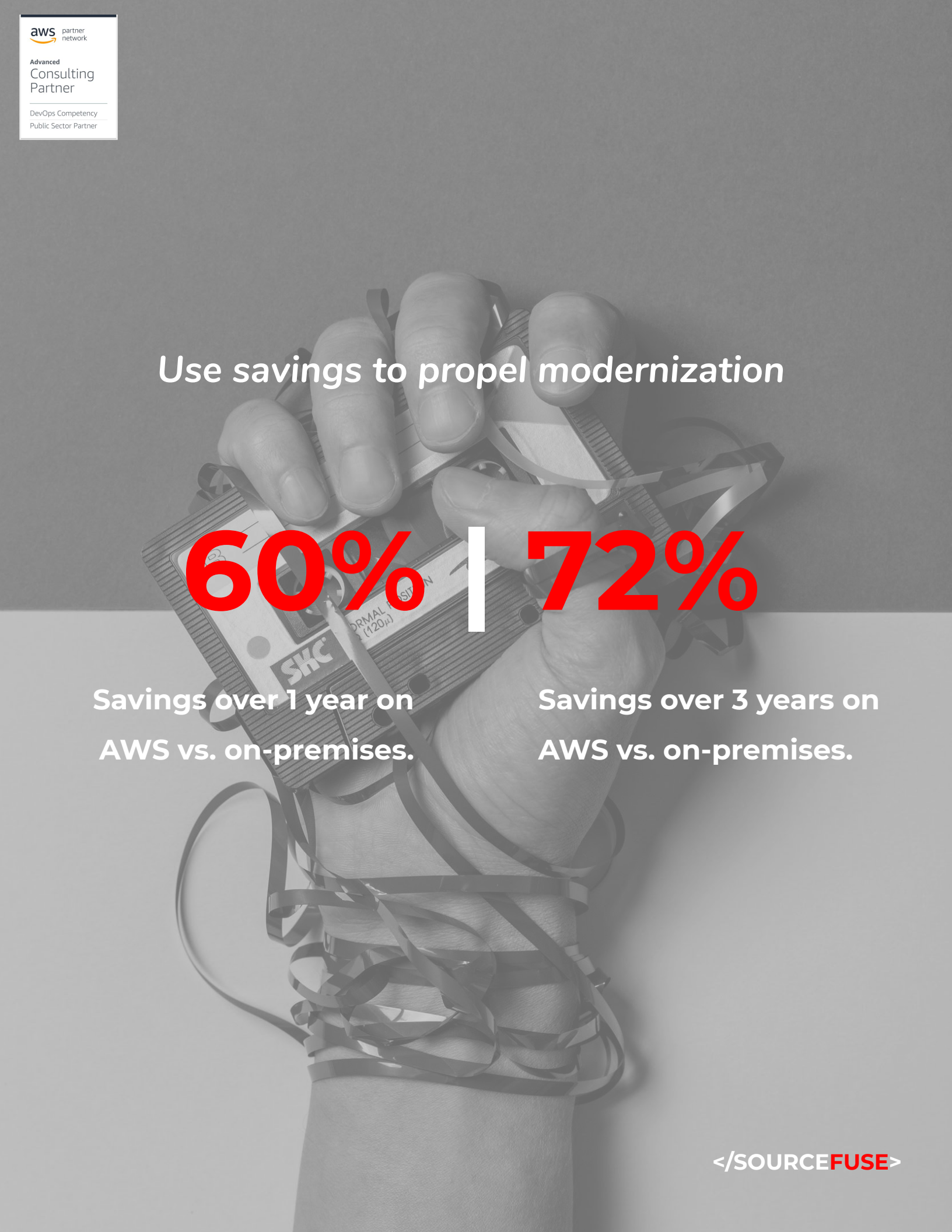
Your strategy should address the following questions:

- Is there a time sensitivity to the business case or business driver, for example, a data center shutdown or contract expiration?
- Who will operate your environment and your applications? Do you use an outsourced provider today?
    - What operating model would you like to have long-term?
- What standards are critical to impose on all applications that you migrate?
- What automation requirements will you impose on applications as a starting point for cloud operations, flexibility, and speed?
    - Will these requirements be imposed on all applications or a defined subset?
    - How will you impose these standards?

aws partner network

Advanced
Consulting
Partner

DevOps Competency
Public Sector Partner



Use savings to propel modernization

60% | 72%

Savings over 1 year on AWS vs. on-premises.

Savings over 3 years on AWS vs. on-premises.

</SOURCEFUSE>

Migrate and validate.

Being ready for a large migration requires preparation across several key areas.

For instance, a technical program manager could map the current hardware configuration of physical servers to equivalent EC2 instance types and estimate the combined storage and bandwidth requirements.

There are several proven frameworks to follow when executing a migration. The AWS Cloud Adoption Framework is a validated process for analyzing IT environments.

Items to consider:

1. Have you clearly defined the scope and the business case?
2. Have you evaluated the environment and applications in scope through the lenses of your landing zone?
3. Have your operations and employee skills been reviewed and updated to accommodate the change?
4. Do you (or does a partner) have the experience necessary to move the tech-stacks that are in scope?

It is important to develop migration skills and experience early to help you make informed choices about your workload patterns.

Migrate and validate.

Validation means that each application goes through a series of specific tests before being finalized and released for the cutover.

Your teams should evaluate release management, verify rollout, rollback plans, and evaluate performance baselines, while **ensuring protection, change management and monitoring tools and practices are in place.**

Infrastructure Protection

Infrastructure protection starts with creating separate Virtual Private Clouds for different projects. This includes defining the infrastructure topology—including gateways, routing tables, and public and private subnets.

Create a NACL (Network Access Control List) and SGs (security groups) to add additional layers of protection.

Data Protection

Use HTTPS endpoints for request calls to AWS services like S3 URLs and TLS/SSL for RDS (Relational Database Services) to encrypt data in transit.

Change Management

A tool like AWS CloudTrail records AWS API calls for your account and delivers log files for auditing while also providing the ability to alert on defined metrics.

Monitoring

After implementing your new architecture, you'll want to make sure proper monitoring tools are in place. Proper monitoring tools will give you the ability to record and act on any performance issues before your customers are aware of any system hiccups.

- CloudWatch provides metrics, alarms, and notifications
- Trusted Advisor will monitor infrastructure resource performance

.NET Migration & Modernization

Migrating to the cloud has moved from asking "why" to asking "when".

Building an effective migration strategy and plan will change your response to "NOW!"

So, instead of rewinding your cassette and searching for a player, to listen to Led Zeppelin, why not just ask Alexa or visit YouTube?

</SOURCEFUSE>

Microsoft® .NET → aws

Microsoft Workloads on AWS.

Bring your Windows licenses to AWS and migrate your Windows Server 2003/2008 32-bit operating systems and .NET applications.

Run Microsoft applications like SharePoint, Dynamics and Exchange in a more secure, easily managed, high performance approach.

- .NET Core 2.0 Support with Lambda & X-Ray
- Hyper-V support in SMS
- Windows Server & SQL Server 2016
- X-Ray .NET SDK
- Trusted Advisor checks for Windows
- Windows for Lightsail
- .NET on Lambda & Codebuild
- Application-consistent Snapshots through VSS
- Windows Deep Learning AMI

</SOURCEFUSE>

Building digital ideas, processes and products.

Rapid Migration POCs

Migrate your first Windows workload in a week

Rapid Migration offers customers a partner-led migration of one qualifying application build in a one-week engagement.

aws partner network

Advanced
Consulting Partner

DevOps Competency
Public Sector Partner