



Cloud Migration & Modernization

Microsoft Practice
Development
Playbook



www.yourstrategic.com



aka.ms/practiceplaybooks

About this Playbook

This playbook is intended for the business and technical leadership for new and existing Microsoft partners focused on migrating workloads to Microsoft Azure or modernizing legacy applications to the cloud.

Objectives

The goal of this playbook is to help partners accelerate and optimize their Azure-focused practice by teaching them the ins-and-outs of migrating workloads or modernizing applications and hosting them in Microsoft Azure.

For the business side, this playbook provides an understanding of the strategies for building a migration or modernization practice, including which workloads to focus on, how to price the services, and how to build the technical team.

For the technical side, this playbook provides detailed guidance on how to plan and execute the migration of an existing workload or application to Azure, including advice on different approaches, useful tools, and technical details on best practices with infrastructure migrations. Application modernization, optimization, and management are discussed, as are Azure best practices and advice on how to build unique IP to make a migration practice more successful.

How this playbook was made

This playbook is part of a series of guidance written by Microsoft Partner [Opsgility](#), in conjunction with the Microsoft One Commercial Partner group and 12 other successful Azure partners that have volunteered their time to provide input and best practices to share with the rest of the partner community.

To validate the guidance provided in these playbooks, we worked with MDC Research to conduct a survey of 364 global partners who are currently selling Azure migration services to customers and offering or actively working towards offering cloud services. In this survey, we gathered insights on a range of topics, including how partners hire, compensate and train resources; their business model, revenue and profitability; what practices and services they offer; and what skillsets they have in place to support their offers. The results of this survey are provided in line with the guidance found within this playbook.

CONTRIBUTING PARTNERS

[Attunix](#)

[BitTitan](#)

[CAPSiDE](#)

[Clear People](#)

[Cloud Direct](#)

[Daisy Group](#)

[Hanu](#)

[Sentia](#)

[Inframont](#)

[Intercept](#)

[Rackspace](#)

[TCS](#)



Using the playbook

Quickly read through the playbook to familiarize yourself with the layout and content. Each section includes an executive summary and key actions for that specific topic. Review these summaries first to decide which areas to focus on. Go over the content several times, if needed, then share with your team.

TO GET THE MOST VALUE OUT OF THIS PLAYBOOK:

- Get the team together and discuss which pieces of the strategy each person is responsible for.
- Share the playbook with sales, marketing, support, technical, and managed services teams.
- Leverage the resources available from Microsoft to help maximize profitability.
- Share feedback on how we can improve this and other playbooks by emailing playbookfeedback@microsoft.com.

“

“Every company is a software company. You have to start thinking and operating like a digital company. It’s no longer just about procuring one solution and deploying one. It’s not about one simple software solution. It’s really you yourself thinking of your own future as a digital company.”

SATYA NADELLA

CEO
Microsoft

Table of Contents

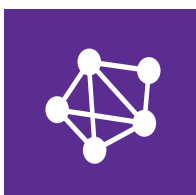
About this Playbook.....	2	Customized Virtual Machine Images	82
Digital Transformation	6	Migrating to Virtual Machines	83
The Cloud Migration Opportunity.....	10	Migrating Disks	84
Define Your Strategy.....	11	Migrating using Azure Site Recovery	85
The Big Picture of Migration	13	Migrating VMware Workloads.....	86
Identify Target Customers.....	14	Migration Tools.....	87
Define the Service Offering.....	16	Business Continuity	89
Defining the Pricing Strategy.....	20	Migrating Existing VMs to CSP	90
Apply for Azure Incentive Programs	22	Migrating Databases.....	91
Build the Migration Practice.....	24	Modernizing Apps.....	93
Landing a Migration Project.....	25	Modernizing Applications with Azure	95
Executing the First Migration Project.....	28	Cloud-Native Architecture and Design	97
Leverage Reusable IP.....	29	Cloud Design Considerations.....	99
Managed Applications.....	31	Cloud Design Patterns.....	101
Managing a Migration Program using a Migration Factory	32	Azure DevTest Labs.....	102
Guide: Optimize and Grow	33	Migrating Applications to Azure App Service	105
Hire & Train	34	Microservices and Containers	107
Building a Migration Team	36	What is Docker?	108
Job Descriptions for the Migration Team	37	Modern Data Platform	109
Reskilling Existing Technical Resources	44	Cognitive Services and AI	111
Hiring and Onboarding.....	47	Optimize & Manage	112
Training & Readiness	48	Cost Optimization.....	114
Competencies and Certifications	52	Azure Cost Management.....	115
Migration Assessment	55	Automatic Shutdown of VMs.....	126
Discovery.....	57	Optimized Architecture	128
Planning.....	61	Azure Management Best Practices	134
Evaluation	67	Azure Subscriptions.....	135
Lift & Shift.....	69	Resource Organization.....	138
Building Out the Network.....	71	Controlling access to Azure Resources.....	139
Network Appliances.....	75	Security and Compliance.....	140
The Virtual Data Center.....	76	Infrastructure as Code	141
Enabling Hybrid Identity.....	77	Azure Resource Manager Templates	142
Planning for Storage.....	78	Blueprints	143
File Shares	79	Automated Scripts.....	144
Choosing Virtual Machines.....	80	Azure SDKs and REST APIs.....	145
Availability Options.....	81	Playbook Summary	146

Digital Transformation

The path to unprecedented growth goes through the cloud, helping customers connect people, data, and processes in new ways to embrace the possibilities enabled by modern technologies. To succeed in a digital-first world, business leaders are bringing business and IT closer together and optimizing processes to create new value for customers.

The potential is huge. By 2019, IDC predicts \$1.7 trillion USD in spending worldwide to create new business models, operational efficiencies, and customer experiences. Digital transformation is now an executive mandate and partner development capabilities will take advantage of customer demand for custom and packaged software.

Three trends are helping shape this profitability opportunity:



DIGITAL PLATFORMS AND ECOSYSTEMS

By 2020, 60% of all enterprises will have fully articulated an organization-wide digital platform strategy and will be in the process of implementing that strategy as the new IT core for competing in the digital economy.



CLOUD

By 2021, spending on cloud services and cloud-enabling hardware, software and services will more than double to over \$530 billion, leveraging the diversifying cloud environment that is 20% at the edge, and over 90% multi-cloud.



















HYPER-AGILE APPLICATIONS

By 2021, enterprise apps will shift toward hyper-agile architectures, with 80% of application development on cloud platforms (PaaS) using microservices and cloud functions, and over 95% of new microservices deployed in containers.

Partners play a key role in helping businesses make the platform and cultural shifts needed, and such transformations are creating amazing partner multiples. In a recent IDC study, partners reported earning \$9.64 in revenue for every \$1 of Microsoft revenue generated in 2017. This is expected to continue through 2022 and include a mix of software (45%), services (50%), and hardware (5%), that are sold in relation to Microsoft solutions.

These changes affect all aspects of a modern business, both internal and external. Microsoft models these changes in four pillars:

ENGAGING CUSTOMERS	EMPOWERING EMPLOYEES	OPTIMIZING OPERATIONS	TRANSFORMING PRODUCTS
<p>Give them new personalized experiences that bolster acquisition and strengthen loyalty.</p>	<p>Boost productivity with flexible workstyles and mobile solutions that enable a data-driven culture.</p>	<p>Drive efficiencies with a cloud platform that accelerates agility.</p>	<p>Create new revenue opportunities using intelligent technology to innovate new products and processes.</p>
<p> Customer centricity integrated across the business</p>	<p> Intentional about people priorities and related strategies</p>	<p> Harnessing technology for next level of efficiency</p>	<p> Leveraging data to enter new markets</p>
<p> Creating fans & segment of one</p>	<p> Using more data to drive insights and decision making</p>	<p> Leveraging digital platforms to reduce delivery timeframes</p>	<p> Revising business models to prioritize agility and emerging trends</p>
<p> Data driven customer insights</p>	<p> Delivering self-service & simplifying processes</p>	<p> Testing new products and services at a fraction of the cost</p>	<p> Making customers business partners</p>
<p> Marketing leaders as technology decision makers</p>	<p> Enhancing HR employee skills</p>	<p> Anticipating and solving customer issues before they become issues</p>	<p> Connecting products to amplify and redefine their value</p>

FURTHER READING

- ➔ [Microsoft Digital Transformation eBook Series](#)
- ➔ [Designed to Disrupt: Reimagine your apps and transform your industry](#)

The Cloud Enables Digital Transformation

Cloud technologies are at the center of the digital transformation revolution. The cloud has changed more than the way we implement and manage IT; it's changing the very fabric of business. With ready access to data, and intelligent new ways to view, analyze and use the information, the cloud has engendered powerful new capabilities which are disrupting entire business models.

There are many advantages to adopting the cloud. Businesses moving to the cloud do so for a range of motivations, seeking a variety of benefits. These benefits fall into four categories: cost, agility, service quality, and new scenarios:

- **Cost:** Cloud computing offers significant potential cost-savings over on-premises infrastructure, especially considering the full cost of the latter. In addition, cloud computing enables organizations to move IT spending from capital expenditure (CapEx) to operational expenditure (OpEx). Since the fixed costs associated with shared infrastructure are avoided, the cloud also provides much greater visibility into the true cost of individual applications.
- **Agility:** Where traditional on-premises infrastructure can take weeks or even months to deploy, Azure offers near-instant provisioning of resources. This enables Azure projects to move much more quickly, without the need to over-provision resources in advance or spend considerable time on infrastructure planning. To take full advantage of this new flexibility, organizations are accelerating the adoption of new ways of working, such as by using agile software development methodologies, continuous integration and deployment (CI/CD), and modern PaaS-based application architectures.
- **Service quality:** Azure's infrastructure has been designed to support some of the world's most demanding workloads. These workloads continuously raise the bar on the quality of service Azure must provide. As a result, migration to Azure often offers significant improvements in performance, reliability, and security over on-premises infrastructure.
- **New scenarios:** Azure enables new application scenarios which are simply not possible, or would be prohibitively expensive to deliver, using on-premises infrastructure, such as big data storage and analytics, machine learning, and compliance with industry certifications such as ISO, PCI, HIPA and GDPR, where customers can leverage the certifications offered by cloud providers. These technologies are enabling new application scenarios, driving innovation and competitive advantages only available in the cloud.

Business Value of the Cloud



Costs

CapEx → OpEx
 Transparency
 Cost Savings



Agility

Instant Provisioning
 DevOps and CI/CD
 Modern Application Architectures
 Faster Time to Market



Service Quality

Performance
 Scalability
 Reliability
 Security and Compliance



New Scenarios

Big Data and IoT / Analytics
 Machine Learning
 Artificial Intelligence
 Digital Transformation

These benefits are all central to a successful digital transformation strategy.

Reduced costs and the shift from CapEx to OpEx dramatically lowers the cost of innovation, enabling a 'fail-fast' experimental approach.

This is supported by the increase in agility that lowers innovation cost and enables a faster time-to-market. The scale, performance, reliability, and global reach of the cloud enables small development teams to develop global services for global audiences.

Most of all, new technologies including big data, IoT, machine learning, and AI empower the insight and customer focus upon which digital transformation depends.

These technologies are often only available in the cloud or are prohibitively expensive on-premises. Moreover, competition between major cloud providers is driving a tidal wave of innovation within the cloud itself. New features and services are added on a weekly or even daily basis, providing an ever-richer platform and enabling business to continue to experiment, innovate, reduce cost and deliver increasing value.

Embracing the cloud is not simply the easiest, or cheapest, or fastest way to drive digital transformation—it is the only way. For many businesses, the first step on this journey is to migrate existing applications to the cloud.

CLOUD OPERATIONS	DIGITAL TRANSFORMATION VALUE
IT becomes an enabler to the business	Driving envisioning and agility
Security by design	Continuous regulatory compliance delivery expertise
Dynamic monitoring with anomaly detection	Proactive insight into end user experience
DevOps tools and processes, CI/CD skillsets	Scale up, scale down, and move to different geographies
Solution and application-based SLAs	Meet business outcomes and customer performance expectations
Decentralized operations and resources	Modernize operations
Software and cloud-based solutions	Automation and orchestration
Expertise consulting, designing, architecting, automating, and optimizing for the cloud	Increase agility and optimization

The Cloud Migration Opportunity

Many businesses will seek help on their cloud migration journey. Cloud migration represents an enormous business opportunity for partners.

As businesses of all sizes embrace digital transformation, traditional on-premises IT becomes increasingly seen as a costly, restrictive, and distracting burden. This creates pressure to reduce or even eliminate on-premises IT by moving existing applications and services to the cloud.

The business-critical nature of many existing applications means any change—especially one as fundamental and far-reaching as cloud migration—represents a business risk. Migration must be as seamless and safe as possible. Cloud migration is a highly technical endeavor and requires skills and experience that are lacking in traditional IT departments. Recognizing this, many businesses seek outside expertise to help them with their cloud migration journey.

This trend has created a rapidly growing business opportunity for specialist IT providers. The global market for cloud migration services is forecast to grow from \$2.4B

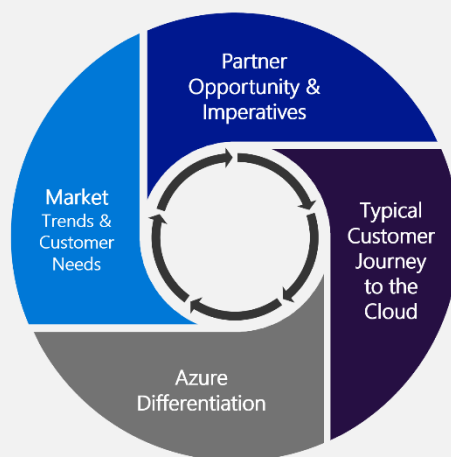
to \$7.1B between 2016 and 2021⁴. This opportunity does not end with cloud migration. Once moved to the cloud, applications must continue to be maintained and updated. Once again, this requires specialist cloud skills and expertise, and once again many businesses will outsource this ongoing maintenance to specialist managed service providers. Outsourcing this work also enables a business to focus on their core activities rather than IT.

This creates a three-fold business opportunity for migration partners. First, to provide application migration services. Second, to provide the ongoing maintenance, support, and related services for migrated applications. And third, to become a trusted, strategic partner in the customer's digital transformation journey, by leveraging the data generated by those applications to deliver insight, innovation and enhanced customer value.

Building a Migration Practice – The Why

- Public cloud is exploding - \$500B by 2020¹
- Most apps are on premise, but 42% will move to public cloud in 2 years²
- Modernization is the key driver - CTO/IT, EOL technologies, LOB Users/Shadow IT
- Cost reduction, On-demand capacity and IT as a strategic asset - motivators²
- Private/Hybrid cloud will be relevant

- 46% of customers prefer Azure over any other public cloud (AWS 42%)²
- Azure is Secure, Hybrid, Scalable
- Application footprint is heterogeneous – Azure has native support for Windows and Linux
- Great ecosystem of ISVs for tooling
- Managed services with CSP



- Migration – entry point for Cloud MSPs
- > 50% of customers will look to 3rd parties for help³
- Plan, Design, Build, Manage - immediate opportunities²
- Security and roadmap offers are critical
- Automation is key and a source of increased revenue

- Monitoring, security and compliance are top of mind
- Most start with IaaS 'Lift & Shift', IaaS optimization
- IaaS to PaaS, App modernization through architecting/re-platforming
- Customer size does not equal to migration complexity

¹IDC CloudView 2016 Survey, 2016
²PULSE Gatepoint Research, March 2016

³451 Research April 2016
⁴Markets and Markets 2016

Microsoft
Partner
Network

Define Your Strategy

Cloud Migration and Modernization

aka.ms/practiceplaybooks



Executive Summary

We start at the beginning, by looking at why customers are interested in migrating applications to the cloud. There are many benefits to the cloud, and not every customer has the same motivations. We present the various ways in which the cloud can deliver value for a business.

Understanding the value proposition is the foundation for building a successful practice in application migration and management.

With this understanding, we will then discuss the variety of services which a Managed Service Provider (MSP) can offer as part of a cloud migration and management practice. We also discuss ways in which an MSP can specialize their business to differentiate from the competition and provide greater value.

We then discuss a variety of revenue streams and pricing models for both application migration and on-going managed services, showing how partners can maximize

their returns by aligning their pricing to the value offered by their services. We also present the various Microsoft incentive programs to help partners build their businesses.

Finally, we provide a head-start on how to identify and close a deal for a migration project, including the common objections. And we will close this section with an outline of the implementation approach that will be the focus of the remainder of this playbook.

Throughout this playbook, we provide links to a variety of resources provided by both Microsoft and third parties to help build a migration practice and execute successful Cloud migrations. A good place to start is the [Azure Migration Center](#), which contains lots of useful guidance.

For further business guidance on building a successful Azure practice, see the [Cloud Infrastructure Practice Development Playbook](#).

Top 4 things to do

Define the business strategy.
Here are the top 4 things to do when defining the strategy for a migration practice.

- Understand the cloud migration value proposition
- Define the service offering
- Define the pricing strategy
- Apply for Azure incentive programs

The Big Picture of Migration

Prior to defining the practice strategy, it is helpful to understand the migration process. At a high level, it can be broken down into three key phases:



ASSESS

The assessment phase is where the team will use a mixture of software tools and consultancy best practices to discover what applications can be migrated, what their current configurations are, the people within the customer organization that will be impacted by the migration, and the dependencies of the application. The output of the assessment will include a comprehensive plan for what to do with the application and the expectations on availability and functionality.

This phase is discussed in detail in the [Migration Assessment section](#) of the playbook.

MIGRATE

The migration phase is when the recommendations in the assessment plan are put into place. The following steps are usually taken.

- Setup Azure subscriptions using best practices for security, connectivity, policies and general governance prior to migration to ensure that customers are using Azure correctly from the start.
- Perform the migration using the prescribed method identified in the assessment plan: rehost, retire, replace, rearchitect or retain.
- Evaluate and test to ensure the migrated application meets the criteria outlined in the assessment.

Learn more about rehosting applications in the [Lift and Shift section](#) of the playbook, and to learn more about rearchitecting applications for Azure see the [Modernizing Apps section](#).

OPTIMIZE

In the optimization phase, partners will use Azure security and management resources to govern, secure, and monitor the cloud applications in Azure. This is also the time to look for opportunities to optimize spending. Common tasks at this stage are:

- Review Azure Cost Management and Azure Advisor to track spending and identify areas for cost savings.
- Evaluate migrated applications for opportunities to right size over provisioned virtual machines and services.
- Implement automation to resize or stop based on a utilization schedule.
- Identify applications that could benefit from optimization with platform as a service (PaaS) services or containers.

Learn more about the optimization phase in the [Optimize and Manage](#) section of the playbook and for a more detailed view, look to the [Operations and Management playbook](#).

Identify Target Customers

Choose what type of organization the migration practice will target, and which verticals to focus on.

When defining the strategy for a cloud migration practice, a key consideration is the target customer. Choosing the right target customers will create focus and accelerate growth.

Our survey of Microsoft partners with a cloud migration practice studied two aspects of the partners' customer base: the size of customer, and their vertical market segment. In this section, we share our findings to help understand the opportunity within each segment and make the right choice for the practice.

INDUSTRY VERTICALS

Our survey considered which industry verticals had driven the most migration or modernization projects. We found that Professional Services was the top industry served with

Azure migration or modernization projects, followed by Technology, Manufacturing, Retail, and Financial Services.

ENTERPRISE VS SMB

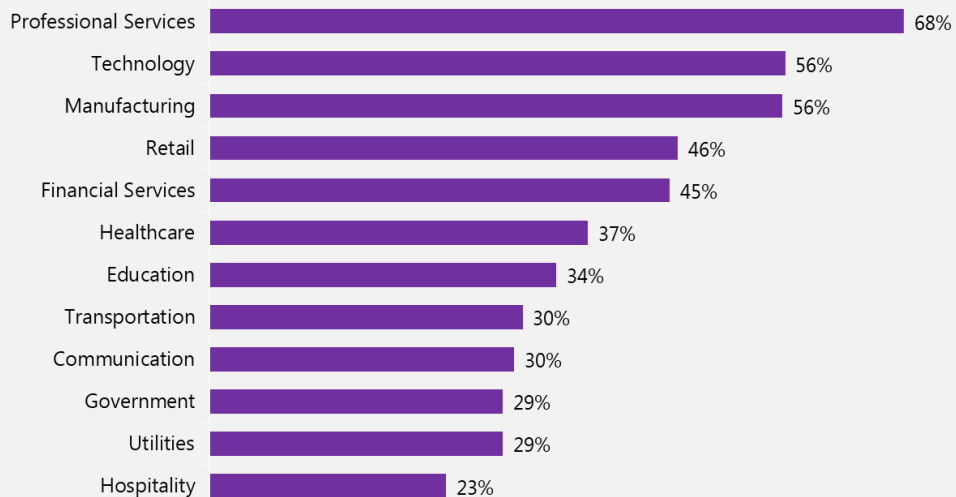
Some migration practices focused on the enterprise segment, where other focused on the small and mid-size business (SMB) segment. Key findings of our survey were:

- SMB customers generate nearly three-quarters of cloud migration or modernization business.
- Migration practices that focus on SMB customers tend to have shorter migration project durations, with nearly half of all projects taking 3 months or less.
- Practices with an enterprise focus tend to see higher follow-on revenue for additional services post-migration.

SURVEY DATA

In our survey of Microsoft partners with an Azure migration practice, Professional Services was the top industry served with Azure migration or modernization projects.

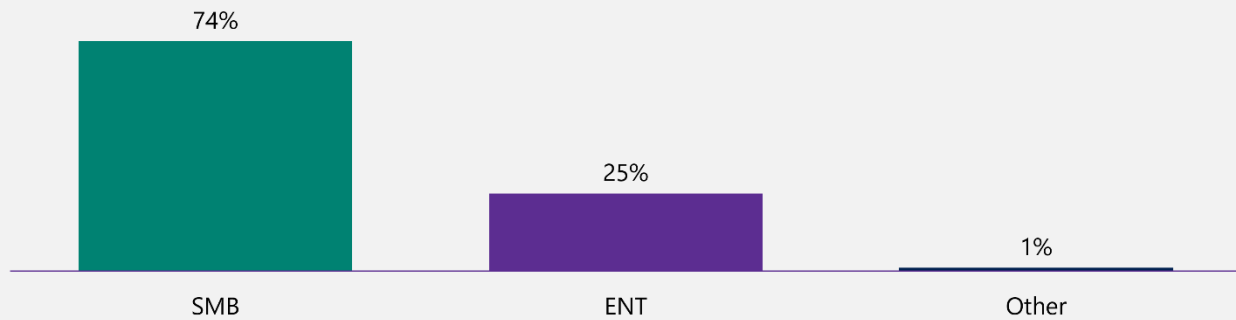
Top Industries Offered Azure Migration or Modernization Services
(n=364)



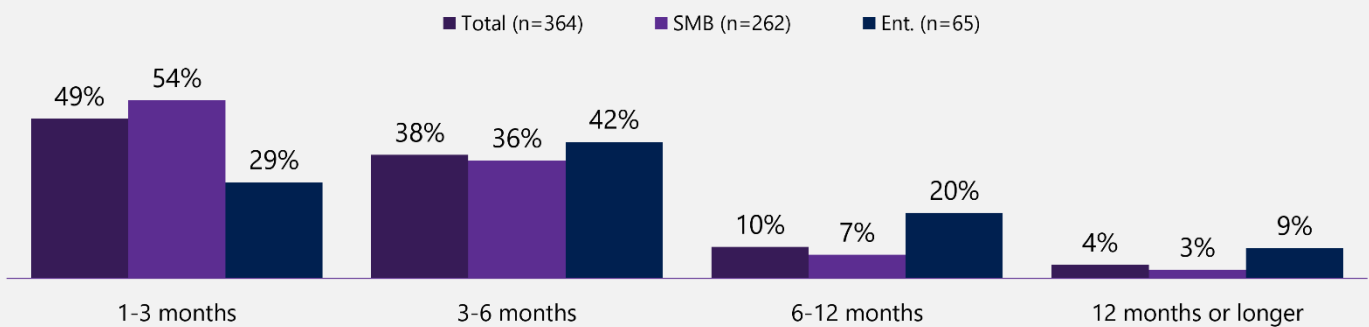
SURVEY DATA

Our survey of Microsoft partners with an Azure migration practice studied variations in business volume, project duration and follow-on revenue based on a practice focus on either enterprise or SMB customers.

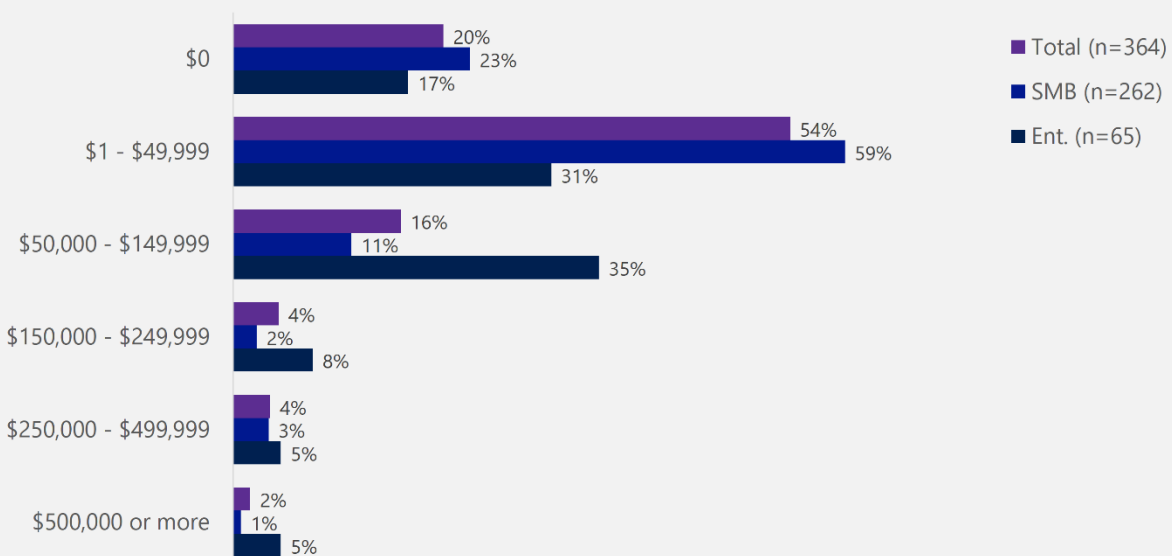
Mean Cloud Migration / Modernization Business Size (n=364)



Average Duration of a Migration Project by Customer Focus



Average Follow-on Revenue for Additional Services the Past 12 Months (n=364)



Define the Service Offering

Customers have different priorities when migrating to the cloud, and migration partners specialize in different types of migration. It is important to understand these variations and choose an area of focus.

Different organizations will have different objectives and priorities for their cloud migration. For example, some may be strongly motivated by the new scenarios which the cloud enables, whereas others may be focused on reducing costs or increasing agility.

These different customer motivations in turn offer different potential business models for a migration partner.

MIGRATION SERVICES

The most common service offered is 'lift and shift' migrations to Azure infrastructure services (IaaS). This focuses on cost reduction by reducing or removing the dependency on on-premises infrastructure. Within this area, a range of complementary services can be offered, such as migration assessments and networking services.

In addition, some providers focus on application modernization—transforming existing applications to take advantage of Azure platform services (PaaS). While these are more complex and typically longer migration projects, they provide increased agility and manageability in addition to cost savings.

Of the partners we interviewed, a minority specialized in enabling new business scenarios, working with customers at the business rather than infrastructure level to re-define existing processes to take advantage of advanced cloud technologies such as machine learning and big data. These projects are the most complex, but also have the potential to deliver the greatest value by generating new revenue streams as well as reducing costs.

Offerings can vary in other ways. For example, ongoing application support can be offered at different levels, from

24-hour response times, down to 1-hour or even 15-minute response times as a premium service. Some providers focus on Azure-based services, while others provide a hybrid service spanning on-premises infrastructure, traditional hosting, and Azure.

It's not an either/or choice. For example, a common combination is for a provider to specialize in 'lift and shift' migrations, and to provide application modernization as an additional service once those applications are migrated. Another example is providers whose operations teams specialize in extracting business insight from application usage data once the application has been migrated.

Within each of the major service areas—migration assessment, migration execution, and (especially) ongoing operations—there are a wealth of opportunities for additional services offering additional value. For example, some customers choose to run their own operations, but will need guidance and training on how to transform and optimize their processes and roles.

Decide which services to provide directly, which to provide through partners, and which services not to offer. To choose your strategy, you will need to understand the intended customers, the potential for each approach in the target markets, and the capabilities on the technical team. Making the right choice is a critical step in defining a cloud practice.

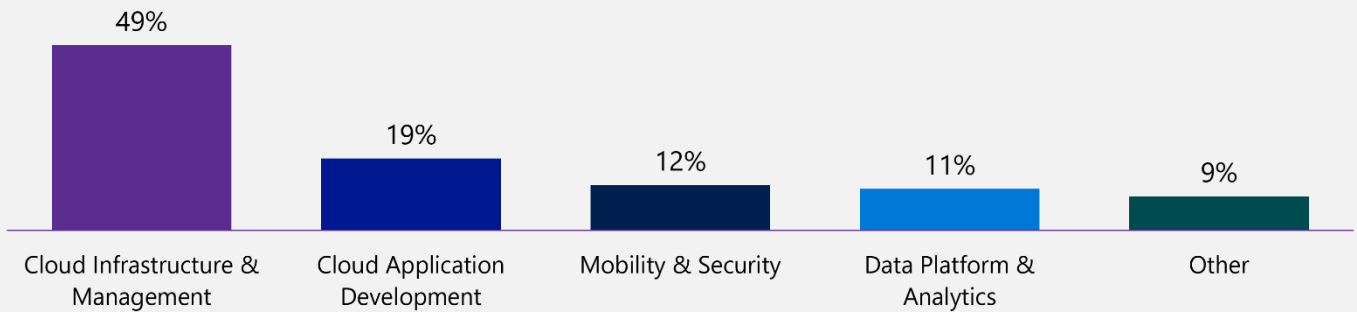
RESOURCES

For further information, including detailed information on the many services offered by Managed Service Provider partners, see the [Azure Managed Service Provider Playbook for CSP Partners](#).

SURVEY DATA

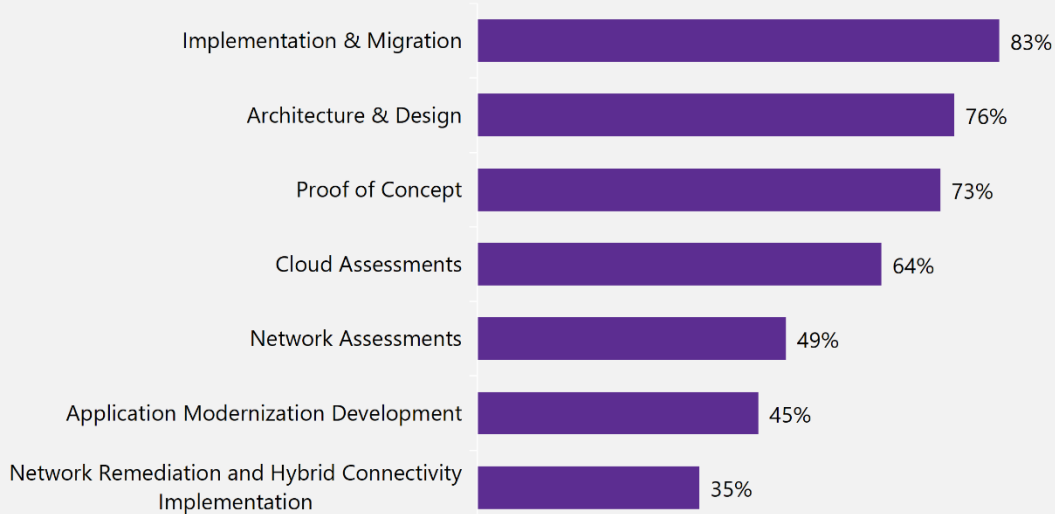
In our survey of Microsoft partners with an Azure migration practice, Cloud Infrastructure & Management services accounted for nearly 50% of reported revenue.

Mean Percentage of Cloud Practices Revenue
(n=364)



Our survey also showed that the most commonly offered service was implementation and migration. Partners with an enterprise focus were more likely to offer Architecture & Design (85%), Proofs of Concept (82%) and Application Modernization (55%) project services than those with an SMB focus.

Top Professional Project Services Provided When Migrating to Azure
(n=364)



Public Cloud, Hybrid Cloud, or Multi-Cloud

Should partners support multiple vendors' clouds, or specialize in Azure?

Microsoft offers a compelling vision for the cloud, coupled with a unique range of offerings, including Office365, Azure, Dynamics, SQL Server, Windows, and more. These products and services span all aspects of a modern digital business.

In some cases, however, customers may have a multi-cloud strategy. This may be motivated by a policy of using multiple vendors, or simply because different groups in large organizations may have made different purchasing decisions in the past.

Some partners are dedicated specialists offering only the Microsoft stack; others combine Microsoft technologies

with those from other vendors. In either case, when defining a strategy as a cloud migration practice, partners often need to decide whether to combine their support for Azure with support for other vendor's clouds, or with on-premises solutions such as VMware.

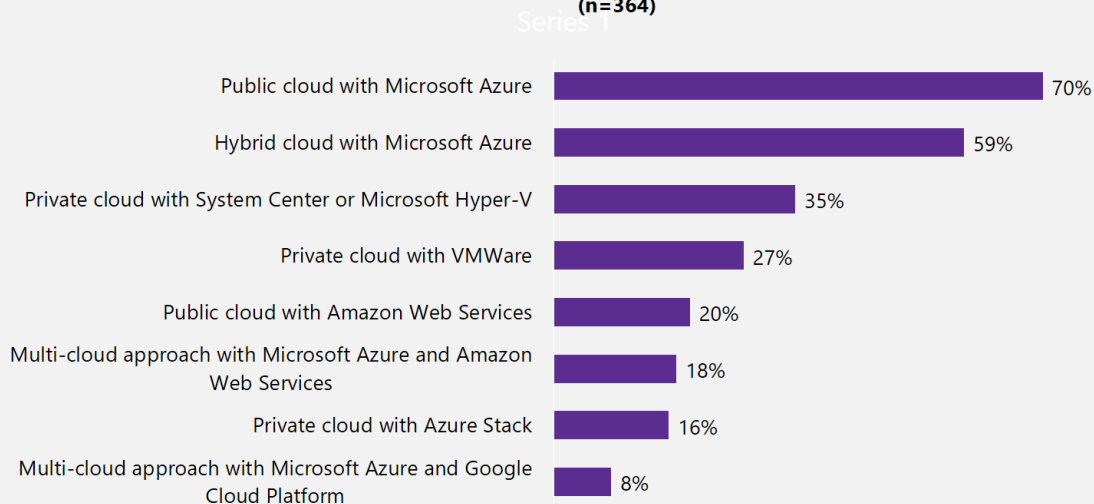
Such decisions become easier with data. Our survey of Microsoft partners with an established cloud migration practice looked at how many projects used other clouds in addition to Azure (see panel). Use this information together with an understanding of the local market when defining a strategy.

SURVEY DATA

In our survey of Microsoft partners with an Azure migration practice, only a small proportion of projects combined Azure with other vendors' public clouds.

Common Technical Scenarios Used in Migration the Past 12 Months

(n=364)



Hosting Azure Stack

Azure Stack combines the agility and flexibility of Azure with the geo-location, isolation, and security of on-premises infrastructure. Hosting Azure Stack offers a unique business opportunity for Managed Service Providers to differentiate their offerings.

Azure Stack is an extension of Azure, bringing the agility and fast-paced innovation of cloud computing to on-premises environments.

Why would customers choose Azure Stack instead of Azure? There are three main reasons. The first is disconnected environments, such as a cruise ship, where connectivity to the public cloud is not available or prohibitively slow or expensive. The second, and most common reason, is regulatory or policy compliance, where a workload cannot be placed in the public Azure cloud. For example, some countries require banking data to remain within national borders. Alternatively, a high-security application might not be permitted to run in a multi-tenant public cloud environment (despite the security and tenant-isolation features Azure provides). The third reason is latency, where demanding performance requirements require that the application is physically located close to the end user, and no suitable Azure data center location suffices.

Of course, such applications could be delivered using conventional infrastructure. However, this approach comes with all the disadvantages that infrastructure brings.

Azure Stack offers an alternative approach. By replicating the core functionality of Azure in an on-premises environment, Azure Stack offers the agility associated with cloud development, with the isolation and security of an on-premises deployment.

Agility is key. Azure Stack allows partners to use the same set of tools, APIs, DevOps processes, and other technologies for both Cloud and on-premises development. And Azure Stack is not just limited to infrastructure services—it includes many fully-managed Azure platform services, including serverless computing, distributed microservice architectures, and containers. By

offering a consistent experience with Public Cloud Azure, partners can benefit from full application portability between Public Cloud Azure and Azure Stack.

Azure Stack can be purchased in two ways:

- **As a managed system:** Typically, on-premises, partners purchase the hardware from a hardware vendor and license the Azure Stack software from Microsoft. They manage the system and contact Microsoft for support.
- **As a managed service:** Typically, at a service provider premises, partners purchase the service from the service provider who purchases and manages the hardware and software. Partners call the service provider for support.

Azure Stack therefore creates an exciting new business opportunity for managed service providers, by offering Azure Stack as a fully managed service.

Hosting Azure Stack is supported through the CSP program:

- **Direct CSP providers** can purchase, host and manage Azure Stack, offering Azure Stack services to their customers.
- **Indirect CSP providers** and **resellers** can also provide Azure Stack services and have the same responsibilities for billing and support as they do in Public Cloud Azure. Either party can take responsibility for owning and administering the Azure Stack hardware.

RESOURCES

- [Azure Stack Overview](#)
- [How to Buy Azure Stack](#)
- [Service Provider Licensing Guide](#)

Defining the Pricing Strategy

How do partners make money as Managed Service Providers (MSPs)? A variety of pricing models are available. Choose the right model for the service offerings.

Migration practices can perform many functions for their customers in the cloud. In each of the major areas—migration assessment, migration execution, and operations—there are a wealth of different services to offer, as discussed earlier in this section. These services can be priced in different ways and offer different potential margins.

CLOUD CONSUMPTION RESALE

Microsoft offers two major schemes enabling partners to receive a share of the revenue generated by Azure spend:

Cloud Solution Provider (CSP)

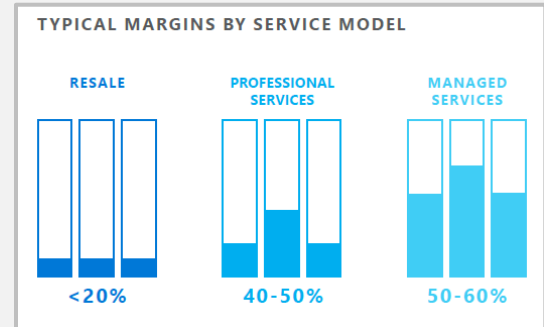
In this program, the partner resells the Azure subscription to the customer. Microsoft charges the CSP partner for subscription usage, and the partner passes on these charges to the customer, making a margin in doing so. In return, the partner takes on certain responsibilities from Microsoft for managing the subscription, such as support and billing.

There are two CSP models—*direct* and *indirect*. In the direct model, the partner works directly with the customer and with Microsoft, becoming the customer’s only point of contact for their Azure services. In the indirect model, two types of partner are involved—the indirect provider (distributor) works with Microsoft, taking responsibility for support and billing, and reaches customers through their partner channel, the indirect reseller, who manages the customer relationship. Further details are given in the next section, on Azure incentive programs.

Digital Partner of Record (DPOR)

In this program, the customer obtains their Azure subscription directly from Microsoft, for example via an Enterprise Agreement. By assisting the customer with Azure usage, the MSP is eligible to be registered by the customer as the *digital partner of record* for the subscription. The MSP then receives a percentage of the Azure usage on the subscription as payment from Microsoft.

aka.ms/practiceplaybooks



PROFESSIONAL SERVICES

Charging for professional services can be an effective way to generate revenue from cloud migrations, and typically offers higher margins than cloud consumption. The MSP can potentially charge for any of their services offered, the most common being migration assessments and migration execution. These can be charged at a daily rate or on a fixed-price project basis.

In some cases, MSPs choose not to charge for migration assessments, instead choosing to fund them internally as ‘pre-sales’ activities in the hope of greater future revenue from migration execution or on-going managed services. This choice is typically made on a case-by-case basis, depending on the assessment costs and size of the potential opportunity.

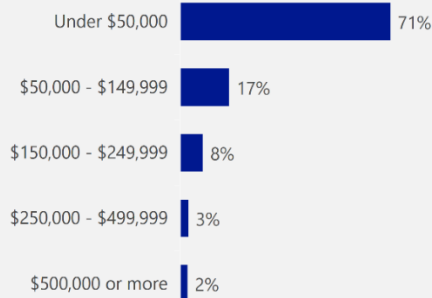
MANAGED SERVICES

Many migration partners focus on charging for the ongoing management of migrated applications as managed services. These typically offer the greatest margins, especially once the number of services under management is high enough for the economies of scale relating to 24x7 support to apply and for custom-built tools to show strong return on investment.

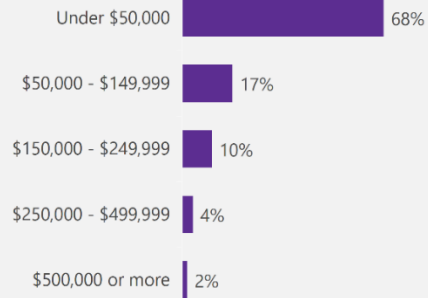
SURVEY DATA

In our survey of Microsoft partners with an Azure migration practice, typical margins for a cloud migration project were in the 25-30% range, with most individual projects generating under \$50,000 in revenue, although some generate substantially more. These figures did not vary significantly between migration and modernization projects, nor between customers with an enterprise or SMB focus.

Average Initial Migration Revenue in the Past 12 Months (n=326)



Average Modernization Project Revenue in the Past 12 Months (n=250)



Mean Migration Project Margin in Past 12 Months

Total Mean (n=326)	27.8%
SMB (n=233)	28.0%
Enterprise (n=59)	26.3%

Mean Modernization Project Margin in Past 12 Months

Total Mean (n=250)	29.1%
SMB (n=173)	28.1%
Enterprise (n=45)	30.0%

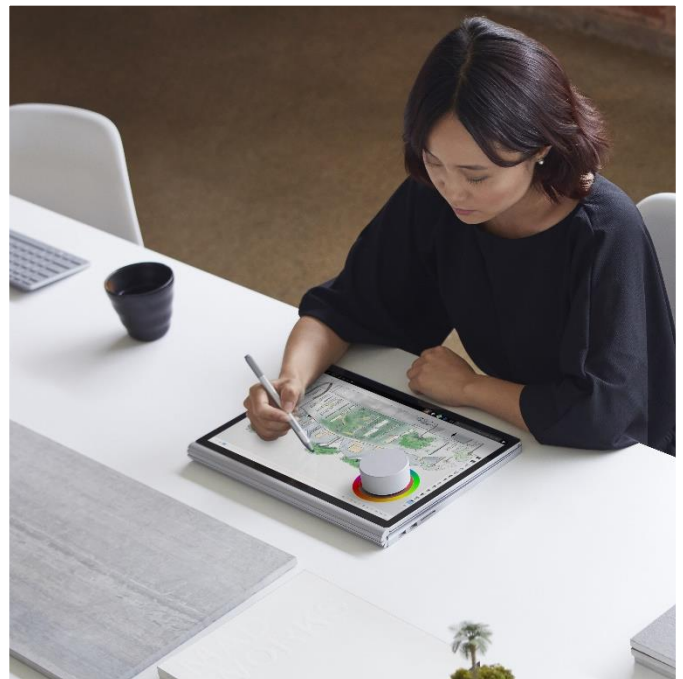
There are several managed service pricing models available. For example, customers can be billed per application, per VM, per user, or per connected device. The right choice will depend on the application’s operating cost and usage model, as well as customer preferences.

Pricing for on-going application support is often divided into ‘tiers’, for example ‘Silver’, ‘Gold’ and ‘Platinum’. The offerings within each tier vary according to the services offered and the service-level agreements included. For example, a ‘Silver’ tier might offer office hours support with 24-hour response time and 30-day backups, whereas the ‘Platinum’ tier might offer 24x7 support with 15-minute response time, together with a business continuity SLA of 15 mins RPO and 1-hour RTO.

FURTHER READING

For further information, including detailed information on the many services offered by Managed Service Provider partners, see the [Azure Managed Service Provider Playbook for CSP Partners](#).

aka.ms/practiceplaybooks



Apply for Azure Incentive Programs

Microsoft offers several incentive programs for Azure usage. Take advantage of these programs to boost a practice.

Over recent years, Microsoft has transitioned from a company focused primarily on software licensing, to a provider of online services. This is a fundamental shift and creates new opportunities for Microsoft’s partner community.

This focus on services places the Microsoft partner front-and-center in the relationship between Microsoft and its customers. The partner role has expanded far beyond reselling licenses, to helping the customer in their use of Microsoft services throughout the customer lifecycle.

The revenue model has also changed. The shift from software to services has moved revenue from one-time license sales to monthly billing. For partners, this change is reflected in new incentive programs to share these new revenue streams.

In this section, we will review the incentives Microsoft provides to partners who help drive business in Azure.

CLOUD SOLUTION PROVIDER

The primary incentive program for Managed Service Providers is the [Cloud Solution Provider \(CSP\)](#) program. This program supports not only Azure, but all Microsoft cloud services including Office365, Enterprise Mobility + Security, and Dynamics CRM Online.

The CSP program enables a partner to own the customer lifecycle and relationship for their consumption of Azure service. They set the price, bill customers directly, and directly provision and manage subscriptions. The CSP also acts as the first point of contact for customer support.

There are two CSP models: direct and indirect. It is important to understand the difference, and to choose carefully where in this ecosystem the practice will thrive.

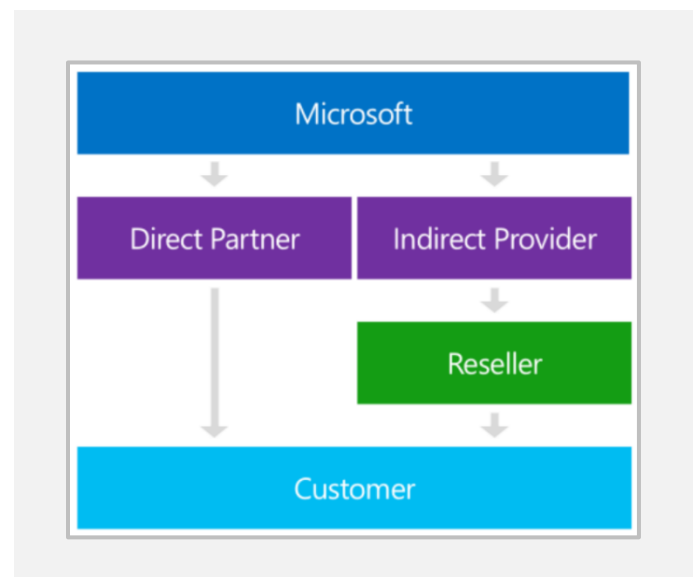
Direct Partners

This model is designed for resellers or Managed Service Providers who have the in-house capability to bill and support their customers at scale.

In this model, partners work with both their customers and with Microsoft directly. They take on the entire customer relationship, including support, billing, and invoicing. They become a customer's only point of contact for their Azure services. This provides continuity in the customer experience and helps build strong business relationships.

Azure CSP direct partners are responsible for customer support. Microsoft does not provide support for Azure CSP customers and relies on Azure CSP partners to manage their Azure workloads and resolve technical problems.

Azure CSP direct partners are also responsible for customer pricing, billing, and invoicing. Microsoft provides partner-facing billing capabilities to Azure CSP direct partners through the Partner Center portal and APIs.



The Azure CSP direct program requires that partners invest in the support and billing practices for the Microsoft cloud products that they want to deliver to their customers.

Indirect Providers and Resellers

The Azure CSP indirect model defines two types of partners: Azure CSP indirect providers (distributors) and Azure CSP indirect resellers. Azure CSP indirect providers work with Microsoft directly, but reach customers indirectly through their partner channel — Azure CSP resellers.

Azure CSP indirect reseller is a good choice for partners who don't want to manage as much infrastructure as an Azure CSP direct partner, so they team up with an indirect provider to handle their support, billing, and invoicing needs. They still build strong relationships with the customer and get many of the benefits of the Azure CSP program, but they offload support and billing to Azure CSP indirect providers.

To learn more about the Azure CSP program, start with the [Azure CSP Overview](#).

DIGITAL PARTNER OF RECORD

In some cases, customers may prefer to use their own Azure subscriptions rather than an Azure subscription provided by partners under the CSP program. For example, the customer may be receiving discounted Azure consumption via an Enterprise Agreement.

This does not prevent partners from managing services hosted within these subscriptions, nor does it prevent them from benefiting from the Azure consumption which they help to enable.

The [Digital Partner of Record](#) program enables Microsoft partners to benefit financially from the revenue they enable for Microsoft. As with the CSP program, this program is eligible across Office 365, Dynamics CRM online, Enterprise Mobility + Security, and other online services, in addition to Azure.

For further details, and to learn how to register, see [Digital Partner of Record](#).



Build the Migration Practice



After studying the market, identifying the customer base, and defining the offering and pricing, partners are ready to start their first migration projects, and begin building their practice.

In this section, we will look at how to build a migration practice, from the first sale to executing large-scale migration programs.

First, we will look in detail at how to generate the first leads and land the first deals. By building on the experiences shared by partners with existing migration practices who participated in our survey and interviews, we will show which techniques are most effective, according to those who have already been successful.

Next, we will give a brief summary of what is required to execute the first migration project, which will be detailed in subsequent chapters of this playbook.

As partners build their businesses, they land larger contracts and more migration projects. Large enterprise customers working on entire data center migrations have thousands of servers to migrate. To help scale the business, we will close this section by discussing how to build a migration factory, with specialized teams focused on each stage of the migration process, for greater productivity.

Landing a Migration Project

Finding a lead is the first step. Once the marketing and sales efforts have identified promising leads it will be up to the technical team to help close the deal.

FINDING LEADS

In our interviews with existing Microsoft partners, we identified two different approaches to finding leads. Some partners were established IT providers with a significant existing managed services customer base. These partners prioritized working with their existing customers on their cloud migration. We also interviewed partners who were relatively new to the market. Naturally, these partners tended to be chasing new business elsewhere.

Our survey of Microsoft partners with an Azure migration practice also looked at how leads were generated. For new customers, the most effective method was customer referrals; for existing customers, most leads came from account manager relationships (see panel).

UNDERSTAND THE CUSTOMER

Simply moving existing infrastructure to the cloud may generate some savings, but it will not by itself deliver the full benefits the cloud promises. Getting maximum advantage from the agility and new scenarios available in the cloud requires deeper changes, to roles, processes, organizational structures, and even culture. Not all customers are ready to embrace this level of change.

As with any sales process, it is essential to understand the customer. Do not assume that every potential customer already understands the value of the cloud. The value proposition is much broader than many people realize.

SURVEY DATA

Our survey of Microsoft partners with an Azure migration practice found that customer referrals were the best way to generate leads with new customers, and account manager relationships the best way to generate leads with existing customers.

Top Sales Activities that Generate New Customer Migration Leads
(n=364)



Top Sales Activities that Generate Existing Customer Migration Leads
(n=233)



Many customers will have specific goals in mind. These will typically map into the four pillars of cloud business value we presented earlier (cost, agility, service quality, and new scenarios). These benefits ‘pull’ the customer towards the cloud. In some cases, external factors such as expiring co-location contracts or end-of-life of an existing software package may ‘push’ the customer towards the need for change, in some cases with a hard deadline that must be met.

Where the customer isn’t clear, start with the basics. Focus on availability and cost savings, since these are fundamental concerns shared by all businesses. The full cost savings of the Cloud may not be realized by an initial project, since staff responsible for infrastructure maintenance may only be redeployed once more workloads have been migrated. Gains may be limited to improvements in uptime and agility.

Minimize the risk and impact of any change. More ambitious projects can wait until the customer has built greater confidence.

START SMALL

Many of the partners we spoke to advised us that one way to convince customers who are not yet committed to the cloud is to start with low-risk, high value workloads that

can easily be rolled back in the event of project failure. Examples include:

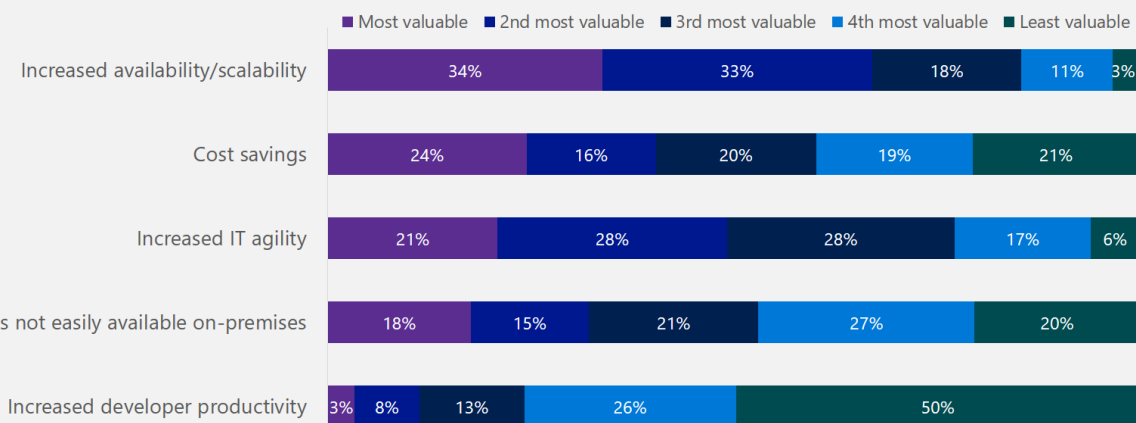
- Adopting Azure DevTest Labs for development/test environments.
- Migrating single instance virtual machines to Azure.
- Replacing existing on-premises backup solutions with Azure Backup.
- Implementing Azure Site Recovery for on-premises to cloud failover in environments that do not have an existing disaster recovery solution.
- Moving websites with minimal dependencies to Azure Web Apps (e.g. marketing or informational websites).
- Using the Azure Files service or Azure StorSimple to replace retiring file share servers.
- Replacing end-of-life hardware on a standalone, non-critical workload without complex dependencies.

As partners move forward, they can develop a long-term strategy, with the cloud as the future of IT and mapping out an incremental roadmap to get there. The customer doesn’t want to be left behind—their competitors will embrace the cloud, and benefit from the competitive edge it gives them.

SURVEY DATA

In our survey of Microsoft partners with an Azure migration practice, increased availability and scalability were reported as the most valuable benefit when convincing a customer to move to the cloud. Next most valuable were cost savings and increased IT agility.

Value of Benefits When Convincing Customer to Migrate to Azure
(n=364)



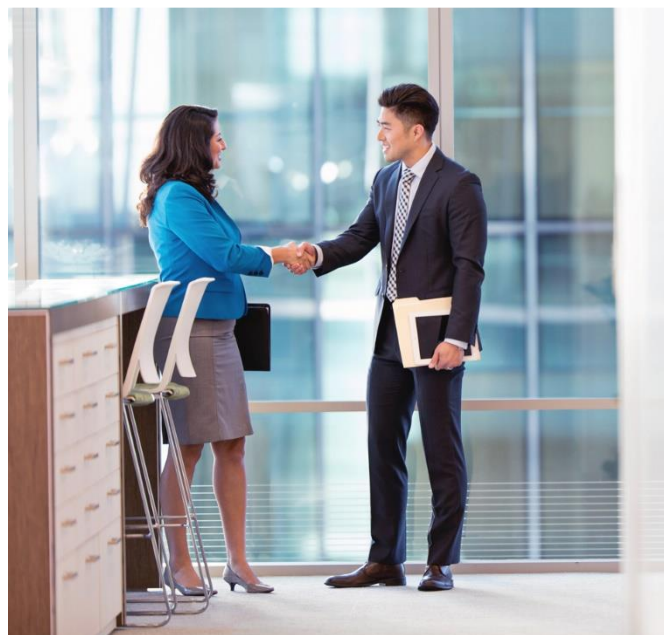
OVERCOMING CUSTOMER OBJECTIONS

Partners should be prepared to address their customer’s concerns and fears regarding cloud adoption. Here are the most common concerns and questions that customers are likely to ask. Be ready to answer—use the content and references in this playbook to prepare yourself and your team.

- What are the cost savings / total cost of ownership (TCO) if I move to Azure?
- Which applications should I move – and what is the recommended sequence?
- Will you build my entire environment on Azure or can part of it remain on-premises/private cloud?
- Will you take care of architecture changes to meet reliability, scalability, and availability requirements?
- What are the impacts to business continuity and to my customer relationships?
- Will you ensure that my data and processes comply with regulations?
- Can you show me how you plan to manage and monitor my application in the cloud?
- What are my risks?

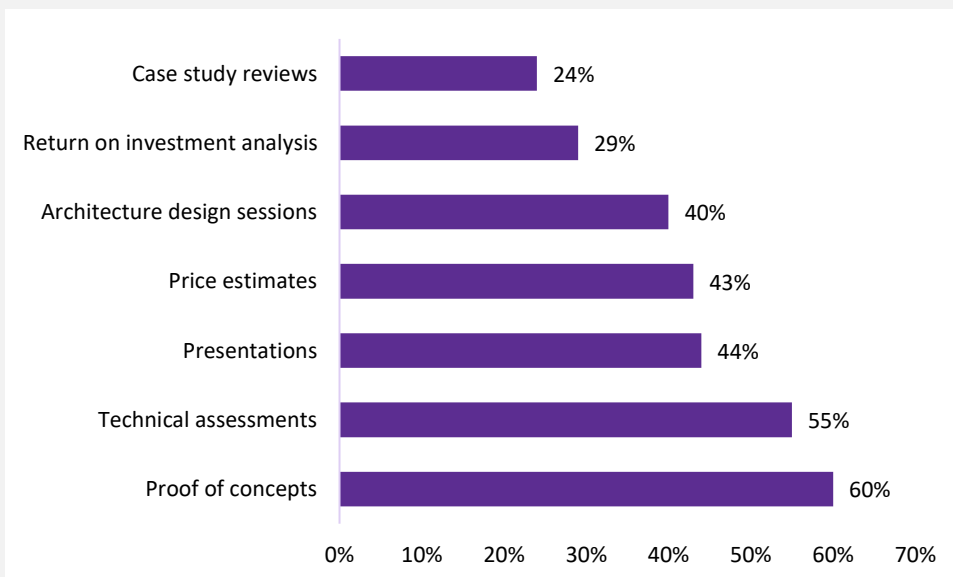
CLOSING THE DEAL

With the right solution, for the right price, and with objections addressed, partners are ready to close the deal. The panel below shows the most effective closing activities, as reported in our survey.



SURVEY DATA

In our survey of Microsoft partners with an Azure migration practice, proof of concepts were reported to be the most effective method of closing a migration deal.



Executing the First Migration Project

There is not a one-size fits all approach to migrating to the cloud. However, there are some general approaches that are tested and proven.



Assess



Migrate



Optimize

There is a wide variation between cloud migration projects, depending on the size, complexity, and technology of the application or group of applications involved. However, all projects typically follow a common migration framework, comprising the following three phases. Leverage the details in this playbook to build competence and capability for delivering each stage effectively and efficiently. Keep the first projects small and learn while building experience.

ASSESSMENTS

In this assessment phase, use a mix of interviews and technology to identify the current environment. but do not limit the assessment to just versions of software deployed, use this as an opportunity to understand the customer’s business in-depth. That means working with project managers, IT professionals, end users, and more to gain valuable insight and potential opportunities.

One of the outputs of the assessment phase is a detailed understanding of costs—existing application running costs, a forecast of cloud running costs post-migration, and the cost of migration itself. These costings, together with the other benefits of migration, are essential for making a go/no-go decision on the migration project. Typically, customers will therefore procure the migration assessment as a stand-alone work item, before deciding whether to proceed with the application migration itself.

Partners may choose to absorb this work as part of a pre-sales engagement or choose to charge as paid consultancy work. Most partners interviewed for this playbook deliver the assessment as a paid engagement. However, the choice will depend on your business model and the size of the opportunity.

MIGRATION & MODERNIZATION

This phase is where technical experts start migrating the first workloads into Azure, identified from the assessment phase of modernizing the application to take advantage of native cloud services. This involves laying the foundation of setting up the network, ensuring identity and security, and creating the resources in Azure. Additional considerations such as user acceptance testing and implementing a failback plan will be detailed in the Cloud Migration and Modernization chapter.



OPTIMIZE & MANAGE

The optimize and manage phase is where the managed services team takes over, and the focus is on monitoring, preventative maintenance, and optimization. After the workloads or applications are stabilized there will be a substantial amount of data to review that can open up opportunities for optimization.



Leverage Reusable IP

Reusable IP can drive efficiency and competitive advantage in every step of the migration cycle.

As partners perform more migrations, and manage more Azure-based services, they will identify common problems and tasks. These can occur at every stage of the migration cycle, from assessment, through migration, and to operations.

Buying or building their own repeatable processes or technology to automate these tasks gives their practice a distinct offering and competitive edge. Specialized software or service offerings can accelerate a practice and provide capabilities, insights, and even direct assistance services to migration projects.

Examples of areas of investment include:

- A repeatable discovery, planning, and evaluation methodology that streamlines the assessment process
- Tools to more accurately forecast prices based on designs or usage data
- Software and services that organize, manage, and scan application workloads and generate planning and cost models that can be implemented in an automated manner to migrate.
- Software that can analyze workloads and their components and recommend alternative topologies

and Azure Services that can be used to modernize applications as part of their automated migration.

- An in-house library of Azure Resource Manager blueprints, templates, or scripts to assist with building proofs-of-concept or even production environments.
- A test framework that speeds up the testing phase of the migration process, while also improving test quality and reducing migration risks.
- An analytics system that helps identify cost savings and optimize running systems.

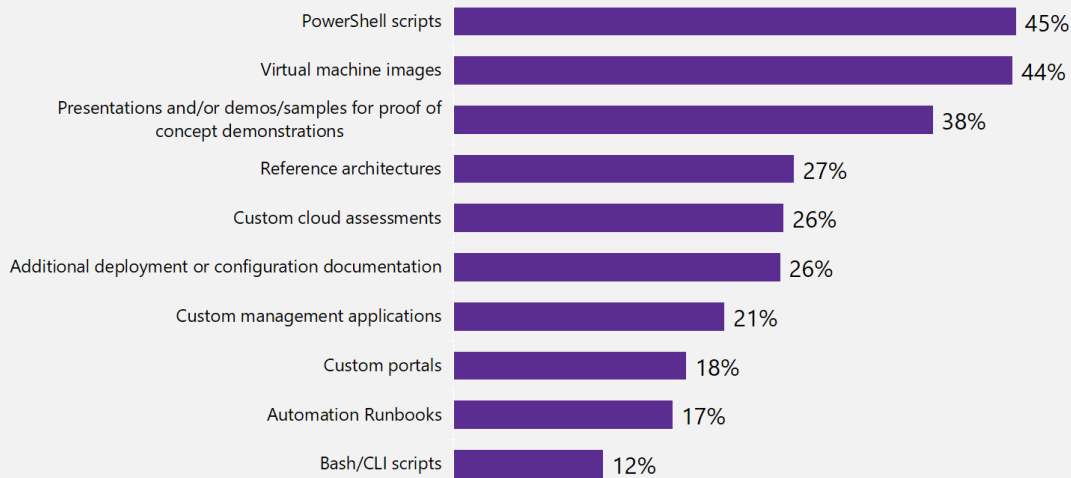
The possibilities are almost endless. Buying (or using as SaaS) migration software, or building repeatable processes and tools enables partners to work faster, with fewer mistakes, and higher quality. It drives down costs, shortens delivery schedules, and improves the customer experience.

In some cases, a low-tech approach suffices. A re-usable migration assessment process could start with something as simple as a Word template. A pricing tool could start as an Excel spreadsheet. In other cases, the technical team will need to code custom tools for specific assessment, migration, or operational tasks.

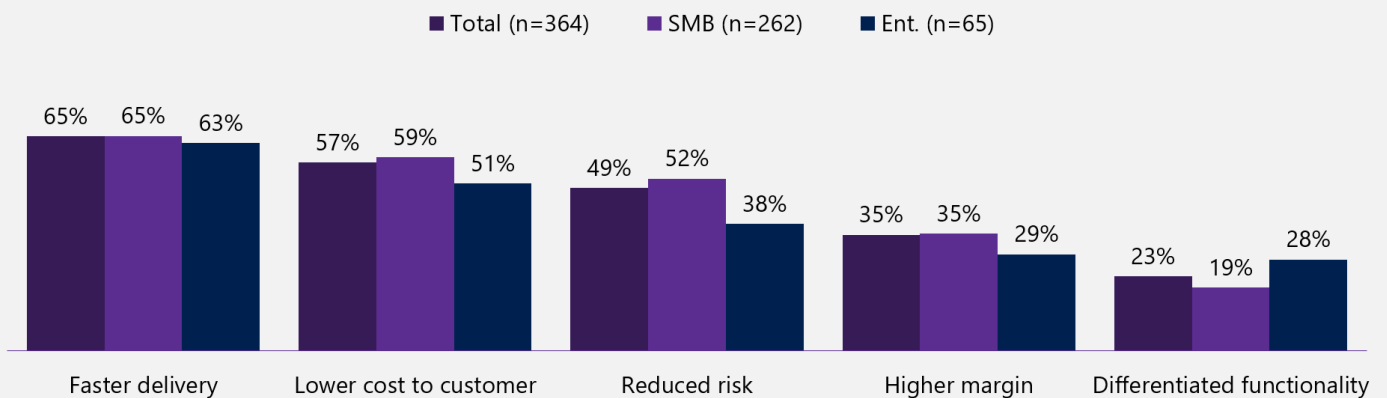
SURVEY DATA

Our survey of Microsoft partners with an Azure migration practice found that PowerShell scripts and virtual machine images were the most common types of reusable IP. Faster delivery and lower cost to the customer were the main benefits that IP delivered.

Top Types of Repeatable IP in Migration and Modernization
(n=364)



Main Value of Repeatable IP



Managed Applications

Cloud solutions that are easy for customers to deploy, and for partners to manage.

As partners develop a migration practice, they may develop a reusable application or tool that is of potential value to broad range of customers. These can be released and monetized as [Azure Managed Applications](#).

Managed Applications enable partners to offer cloud solutions that are easy for customers to deploy, and easy for partners to provide on-going management services. Customers do not need expertise in cloud infrastructure to use the solution and have limited access to the critical resources. They do not need to worry about making a mistake when managing it.

With Managed Applications, partners establish an ongoing relationship with their customers. They define terms and fees for managing the application, and all charges are handled through Azure billing.

A managed application is like a solution template in the marketplace, with one key difference: in a managed application, the resources are provisioned to a resource group that is managed by the publisher of the app. The resource group is present in the consumer's subscription, but an identity in the publisher's tenant has access to the resource group. As the publisher, you specify the cost for ongoing support of the solution.

To make Managed Applications available to all customers, they can be published to the [Azure Marketplace](#). To make Managed Applications available to only users within an organization, they can instead be published to an internal service catalog.



Managing a Migration Program using a Migration Factory

Large customers have hundreds of applications and servers to migrate. A programmatic approach is needed to manage the complexity and scale.

Enterprise customers have large IT footprints, with thousands of servers hosted on-premises or in dedicated data centers. Those embracing the cloud are seeking to dramatically reduce their server footprint, with a goal of significant cost savings.

Projects to eliminate entire data centers are a significant challenge even for an established migration practice. They are also a huge commercial opportunity.

Migrating thousands of servers, with hundreds of applications, requires a systematic, programmatic approach. Each individual migration project needs to follow a defined structure, with processes in place for roll-up progress tracking and status reporting.

A common approach to manage large-scale migration programs is to build a migration factory. A migration factory works just like a manufacturing production line,

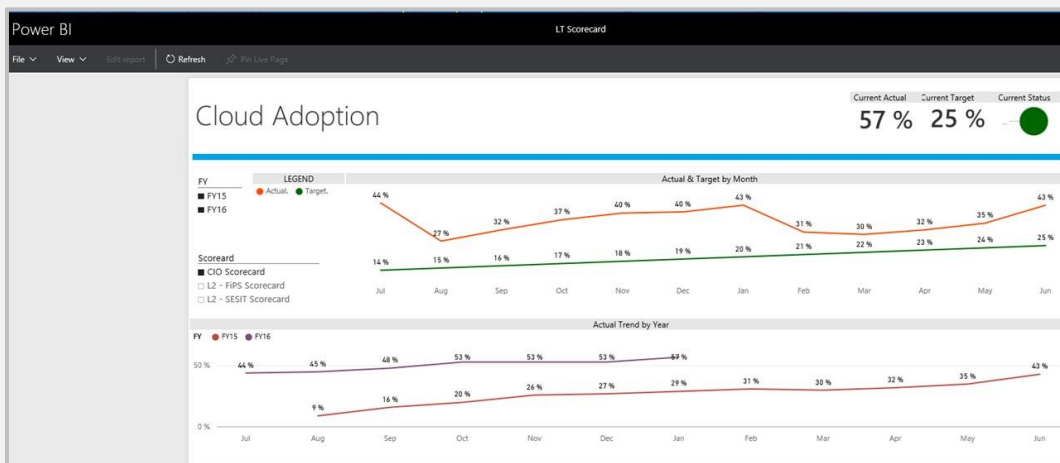
with dedicated, specialized teams focused on each stage of the migration. Just as in a production line, this focus and specialization enables a significant increase in productivity, efficiency, and quality, resulting in faster, cheaper, and more reliable migrations.

To be successful, each stage must follow a well-defined process, using dedicated, optimized tools. This consistency enables smooth handovers between teams as each migration project progresses.

A consistent approach also enables program-level reporting. Each individual project should report status and progress using a common format. For example, projects may maintain a scorecard for key migration criteria. Roll-up scores can be used to dashboard an overview of progress to senior management and quickly identify problem migrations for additional attention.

MIGRATION DASHBOARD

A migration dashboard enables a roll-up view of migration status across a portfolio of migration projects. This is useful for both internal tracking and customer reports.



Guide: Optimize and Grow

Leverage the Microsoft resources available in the [Optimize and Grow guide](#), for details on building customer lifetime value, executing nurture marketing efforts, optimizing and growing from feedback, refining the customer value proposition, growing partnerships, and measuring results.



Microsoft
Partner
Network

Hire & Train

Cloud Migration and Modernization

aka.ms/practiceplaybooks



Executive Summary

In the previous section, we evaluated several strategies for building or enhancing a migration or modernization practice. With some avenues of success identified, we'll move on to how to build and train your team.

A very real and pressing challenge is the skills gap in the industry. To state it simply, there are not enough subject matter experts that know Microsoft Azure and the surrounding technologies to fulfill the opportunities available.

A cloud migration practice will need to decide whether to reskill existing employees, hire new, contract out, or utilize

a combination of these options to fulfill the operational needs.

In this section, we will help define the members of a cloud migration team and the skills they should contribute. We will provide detailed job descriptions, ideas on where to look for resources, and the characteristics to look for in a candidate's skillset.

Then we will show how to build a plan for reskilling and maintaining your team's skills. This will involve understanding what skills are needed and practical ways of training your team to close the skills gap and foster team growth.

Top 5 things to do

Measure twice and cut once. Here are the top 5 things to do when planning for hiring and training.

- Define roles for the technical team
- Write job descriptions
- Develop a readiness and onboarding plan
- Identify applicable certifications
- Get trained

Building a Migration Team

The team needed for a migration varies greatly depending on the size and scope of the effort. An important consideration is that most migrations are performed in conjunction with technical and business stakeholders from the customer, so it is important to ensure that they are brought in at the right times throughout the project and that communication is clear on timelines, objectives and responsibilities.

The following examples of roles are based on interviews with partners and are made slightly generic to outline the types of professionals on a migration team as well as roles often encountered in a customer's organization.

PARTNER

Cloud Architect – the Cloud Architect is responsible for the overall vision of what the solution will consist of once it is in Azure, as well as building out the migration plan. They are typically the go-to-resource for helping the customer to understand the tradeoffs of the approach to migrating or modernizing workloads as well as setting up the target Azure environment with appropriate controls for governance.

Cloud Infrastructure Engineers – the Cloud Infrastructure Engineers are the experts that are doing the actual work of a migration project. This may include creating resources in Azure, uploading data, writing scripts, and in general doing the actual migration.

Senior Software Developer – the Senior Software Developer is responsible for designing and delivering a modernized application that takes advantage of new capabilities Azure provides.

Technical Specialist – the Technical Specialist is a solution engineer that specializes in a certain area such as databases, networking, storage, or security/identity management. They may or may not be part of the migration team depending on the workload.

Project Manager – the Project Manager is tasked with ensuring that milestones are reached on time and communication occurs between the members of the partner team and the customer. In addition to strong project management skills and experience, the PM should also have solid technical background, so they can understand the project in depth and make sound technical judgement calls.

CUSTOMER

Application/Business owners – these are the teams directly responsible for business processes that may vary by the migration project.

Database administrators – these experts will play an integral role in identifying dependencies, availability requirements, and migration SLAs for moving data as part of the migration.

Security and compliance specialists – work with the security and compliance experts to understand existing security processes and compliance criteria. Often these professionals are some of the most important to create a productive relationship with because companies rightly view questions about security as a blocker.

IT Architects – The partner's team will work with the IT architects at the customer to understand existing services and policies and what the future services and policies after the migration should look like.

Application developers – the application developers are an incredibly useful resource to work with when it comes to deciding whether an application should be migrated as-is, modernized, or split into a hybrid model.

End user representatives – end user feedback is important to validate that the migrated system is functioning and performing correctly and to validate any user experience changes.

Job Descriptions for the Migration Team

The following tables provides detailed job descriptions you can utilize to hire the key technical resources. All technical skills, non-technical skills, certifications, and technologies listed are potential items a candidate should have, but no candidate will have all the items listed.

CLOUD ARCHITECT

A Cloud Architect (CA) drives high-priority customer initiatives in collaboration with customers and your sales team. The CA is a technical, customer-facing role that is accountable for the end-to-end customer cloud deployment experience. CAs own the Azure technical customer engagement, including architectural design sessions, specific implementation projects and/or proofs of concepts. The ideal candidate will have experience in customer-facing roles and success leading deep technical architecture and application design discussions with senior customer executives to drive cloud deployment. Bachelor's degree in computer science or related field preferred.

Technical Skills

- Solid understanding of modern authentication protocols and a background in cyber security.
- Deep understanding of cloud computing technologies, business drivers, and emerging computing trends.
- Deep technical experience in enterprise mobile, identity and access control, & security solutions.
- Understanding of cloud governance technologies for cost management and control.
- Understanding of common database technologies such as SQL Database/Server, Oracle, MySQL
- Working knowledge with AGILE development, SCRUM and Application Lifecycle Management (ALM) with one or more of the following programming languages: PowerShell, Bash, .NET, C++, Java, JSON, PHP, Perl, Python, Ruby on Rails, HTML, CSS, JavaScript, Responsive Web Design.

Non-Technical Skills

- Proven track record of building deep technical relationships with senior executives and growing cloud consumption share in large or highly strategic accounts.
- Proven track record of driving decisions collaboratively, resolving conflicts & ensuring follow through.
- Presentation skills with a high degree of comfort with both large and small audiences.
- Prior work experience in a consulting/architecture position within a software & services company.
- Problem-solving mentality leveraging internal and/or external resources.
- Exceptional verbal and written communication.

Certifications

- Microsoft Certified Azure Solutions Architect Expert, CompTIA Security+, CISSP, AWS Certified Solution Architect.
- Exam priorities: [AZ-300 Microsoft Azure Architect Technologies](#), [AZ-301 Microsoft Azure Architect Design](#).

Project Experience Types/Qualities

- 5+ years of architecture, design, implementation, and/or support of highly distributed applications (i.e. having an architectural sense for ensuring availability, reliability, etc.).
- 2+ years of experience in “migrating” on premise workloads to the cloud.
- 5+ years of success in consultative/complex technical sales and deployment projects (where necessary, managing various stakeholder relationships to get consensus on solution/projects).
- Oversight experience on major transformation projects and successful transitions to implementation support teams.

Technologies

- Enterprise Mobility Suite, Intune, Azure Information Protection, Azure Active Directory, Octa, Auth0, LDAP, OAuth, SAML, Cloud App Security, Firewalls, Office 365, Windows Server Active Directory, Azure AD Connect, Active Directory Federation Services (ADFS), Mobile Iron, Airwatch, iOS, Android, Windows, Azure Virtual Machines, Virtual Networks, ExpressRoute, Operations Management Suite, Azure Site Recovery, Azure Backup, Azure App Services, Azure Storage, Azure Import/Export, Azure SQL Database, Azure Web Jobs, Azure ExpressRoute, Azure SQL Database, MySQL, Azure SQL DW, Azure DB for MySQL, Azure Cosmos DB, SQL Server, SQL Server on Azure IaaS, SharePoint on Azure, AWS EC2, S3, AWS DirectConnect



CLOUD INFRASTRUCTURE ENGINEER

The Cloud Infrastructure Engineer delivers technical solutions and support to customers allowing them to maximize their investment in cloud technology. The ideal candidate will have experience in customer facing roles and success implementing cloud-based solutions, migrating workloads to the cloud, and experience with connecting and managing hybrid cloud environments.

Building upon solid IT project experience relative to their level, consultants will work with customers in:

- The delivery of high-quality engagements around Microsoft's solution areas, technologies and products in diverse client environments.
- The design and development of integrated solutions using the latest Microsoft products and technologies.
- Understanding the relevant application development, infrastructure and operations implications of the developed solution.

Technical Skills	<ul style="list-style-type: none"> • Deep understanding of cloud computing technologies, business drivers, and emerging computing trends. • Deep technical experience in infrastructure design including private and public cloud, networking, virtualization, identity, security and storage. • Understanding of how to build resilient multi-site architectures. • Experience with Windows, Linux and OSS technologies. • Experience with configuration management and automation technologies such as PowerShell DSC, Chef, ARM Templates, and Puppet. • Experience with deploying and managing the infrastructure for databases such as SQL Server, Oracle, Maria, Cassandra • Working knowledge with AGILE development, SCRUM and Application Lifecycle Management (ALM) with one or more of the following programming languages: PowerShell, Bash, .NET, C++, Java, JSON, PHP, Perl, Python, Ruby on Rails.
Non-Technical Skills	<ul style="list-style-type: none"> • Services project management. • Building customer/partner relationships. • Proven track record of driving decisions collaboratively, resolving conflicts and ensuring follow through. • Presentation skills with a high degree of comfort with both large and small audiences. • Prior work experience in a consulting/architecture position within a software and/or services company. • Problem-solving mentality leveraging internal and/or external resources. • Exceptional verbal and written communication.
Certifications	<ul style="list-style-type: none"> • Microsoft Certified Azure Administrator Associate, AWS Certified SysOps Administrator • Exam priorities: AZ-103 Microsoft Azure Administrator.

Project Experience Types/Qualities	<ul style="list-style-type: none">• 3-5+ years senior (Tier 3) level support with cloud infrastructure as part of responsibilities.• 5+ years of architecture, design, implementation, and/or support of highly distributed applications (i.e. having an architectural sense for ensuring availability, reliability, etc.)• 2+ years of experience in “migrating” on premise workloads to the cloud.• 5+ years of success in consultative/complex technical sales and deployment projects (where necessary, managing various stakeholder relationships to get consensus on solution/projects.• Oversight experience on major transformation projects and successful transitions to implementation support teams.
Technologies	<ul style="list-style-type: none">• Azure Virtual Machines, Virtual Networks, ExpressRoute, Azure Active Directory, Operations Management Suite, Azure Site Recovery, Azure Backup, Azure App Services, Azure Storage, Azure Import/Export, Azure SQL Database, Azure Web Jobs, Azure ExpressRoute, MySQL, SQL Server, SQL Server IaaS, SharePoint on Azure, AWS EC2, S3, DirectConnect, Hyper-V, VMWare, System Center, Citrix, StorSimple, SAN, firewalls, web app proxies, PowerShell, Bash, JSON, ARM Templates, BGP, Site-to-Site VPN, Chef, Puppet, Ansible, SaltStack, Windows Server, Linux, OSS Technologies, Azure SQL Database, MySQL, Azure SQL DW, Azure DB for MySQL, Azure Cosmos DB, SQL Server, SQL Server on Azure IaaS, SharePoint on Azure, AWS EC2, S3, AWS DirectConnect



SENIOR SOFTWARE DEVELOPER

A Senior Software Developer has a history of designing, owning and shipping software, as well as excellent communication and collaboration skills. With a focus on cloud-based application development, the candidate must have demonstrable experience architecting and deploying applications to cloud platforms, the ability to effectively integrate disparate services as needed, and decide when to implement IaaS, SaaS, and PaaS components. As a mentor to junior developers, the senior software developer should have a solid understanding of the software development cycle, from architecture to testing. They should have a passion for quality and be a creative thinker. A senior developer will write secure, reliable, scalable, and maintainable code, and then effectively debug it, test it and support it live. This person should also be comfortable owning a feature and making decisions independently, and should have leadership experience with agile methodologies, such as the Scrum approach to agile software development.

A Senior Software Developer can also effectively gather customer requirements and ask clarifying questions when needed. This person must be able to translate these requirements to actionable tasks they will perform, or delegate to members of the team. The ideal candidate will have experience in customer facing roles and success leading deep technical architecture and design discussions with senior executives.

Eight plus years of experience with deep understanding of web technologies, API consumption/development, full lifecycle application development, database development (relational and/or NoSQL), and enterprise/cloud architecture. Technical BS degree in Computer Science desirable, and experience in:

- The delivery of high quality engagements around Microsoft's solution areas, technologies and products in diverse client environments.
- Stabilizing developed solutions using Microsoft methodologies in complex customer environments.
- The design and development of integrated solutions using the latest Microsoft products and technologies.
- Understanding the relevant application development, infrastructure and operations implications of the developed solution.

Technical Skills

- API development, Application architecture, application development, application lifecycle management (ALM), caching, capacity planning, cloud archival, cloud disaster recovery, cloud storage, cloud systems management, cloud systems operations, cloud transformation, compliance (PCI, HIPPA, etc.), data architecting, data migration (cross platform / upgrade), data modeling (physical and logical), data movement, data transformation, database and server virtualization, database architecture, database design, database lifecycle management, database management, dev ops, diagnostics, distributed application design, distributed application development, distributed database design, event sourcing, HADR / replication, health checks, identity and security, information architecture, information management, IoC, mission critical DB design and architecture, modern applications, monitoring, package management (npm, NuGet, etc.), performance tuning, polyglot resiliency, reporting services design and deployment, responsive design, RESTful services, resiliency (clustering, etc.), scalability (up and out, high performance), security architecture, security compliance, source code repository management (git, TFS, svn, etc.), technical migration upgrades, technology architecture, testing / TDD, unstructured data formats (e.g. JSON), structured data formats (e.g. XML), UI / UX.

Non-Technical Skills	<ul style="list-style-type: none"> • Collaboration, stakeholder management, relationship management, technical oversight, technical recommendations, problem solving, risk management, architecture design session, program management, proof of concept design, technical demonstration, excellent communication skills.
Certifications	<ul style="list-style-type: none"> • Microsoft Certified Azure Developer Associate, Certified ScrumMaster, AWS Certified Developer. • Exam priorities: 70-487: Developing Microsoft Azure and Web Services, AZ-203 Developing Solutions for Microsoft Azure.
Project Experience Types/Qualities	<ul style="list-style-type: none"> • API consumption and development, coordinate and execute pilots, prototypes or proof of concepts, provide validation on specific scenarios, document and share technical best practices, further customer investment, hybrid solutions on premises or in the cloud, industry-visible, CI / Continuous Deployment, large project relative to size of customer, lift and shift, migrations and upgrades (SQL, etc.), on-premises to cloud, production environment, projects where data is born in the cloud, cross-platform SQL Server migration, server-side/desktop development, service architecture, size of project team (complexity), significant challenges, source code repository implementation support teams.
Technologies	<ul style="list-style-type: none"> • AWS API Gateway, AWS EC2, AWS SWF, AWS, AWS RDS, AWS VM, AWS Redshift, AWS S3, Angular, Aurelia, Azure Active Directory, Azure App Service Environment, Azure Data Catalog, Azure Data Factory, Azure Data Lake, Azure Logic App, Azure Mobile App, Azure Storage, Azure Cosmos DB, Azure SQL Data Warehouse, Azure Functions, Azure Import/Export, Azure SQL Database, Azure SQL DW, Azure DB for MySQL, Azure Search, Azure Event Hubs, Azure Web App, Azure Functions, Azure Cognitive Services, BizTalk, Business Objects, Cassandra, CDN, Cortana Intelligence, CouchDB, Data warehouse, Database, DB2, Docker, Excel, IBM Bluemix, Google App Engine, HTML, IBM, IBM Teradata, IoT Solutions, Java, Media Services, MongoDB, Microsoft Dynamics CRM, Microsoft SharePoint, MySQL, MVC, MVVM, Mobile Development, Networking, Node.js, NoSQL, Oracle, Oracle Exadata, Oracle SOA, PostgreSQL, Python, REST, Security, SQL Server, SQL Server IaaS, SQL Server Integration Services, Storage, Sybase, T-SQL, UWP, Virtualization, Web Services, WCF, WPF, XML • Programming/Scripting Languages: .NET (C#, F#, VB.NET), Java, Python, JavaScript, Scala, Go, Ruby, PHP, SQL, T-SQL, PowerShell. Platforms: Linux (Red Hat, Ubuntu, Debian, etc.), Windows.

PROJECT MANAGER

The Project Manager is responsible for the overall success of the project. They are responsible for ensuring the initial vision and goals of the project are clearly defined and aligned with all relevant stakeholders and executing the project to meet those goals. This includes building project plans, tracking and managing risks, analyzing dependencies, and communication within the team, with management, and with the customer.

Throughout the project, the Project Manager will need to make scoping and prioritization decisions as issues arise. Making good decisions is only possible if the Project Manager can fully understand each issue. In addition, a good Project Manager can anticipate issues before they arise and take pre-emptive corrective action. In addition to strong project management skills, the Project Manager should also have a strong technical background.

Technical Skills

- Solid technical background in IT infrastructure and application architectures.
- Solid understanding of cloud computing technologies, business drivers, and emerging computing trends.
- Solid understanding of cloud migration approaches and supporting tools.

Non-Technical Skills

- Strong project management skills including experience of a variety of project management methodologies such as Agile, SCRUM, waterfall methodologies, etc.
- Demonstrated success in driving complex projects with multiple stakeholders and dependencies.
- Proven track record of building deep technical relationships with senior executives and growing cloud consumption share in large or highly strategic accounts.
- Proven track record of driving decisions collaboratively, resolving conflicts & ensuring follow through.
- Strong written and spoken presentation skills with a high degree of comfort with senior audiences. Able to represent the project to the customer.

Certifications

- A relevant Project Management qualification such as PMP, Certified ScrumMaster or PRINCE 2.
- Microsoft Certified Azure Fundamentals, Microsoft Certified Azure Administrator Associate.
- Exam priorities: [Microsoft Azure Fundamentals AZ-900](#), [AZ-103 Microsoft Azure Administrator](#).

Project Experience Types/Qualities

- 5+ years technical project management experience leading complex projects on business-critical IT systems.
- 2+ years of experience in "migrating" on premise workloads to the cloud.
- 3+ years in hands-on technical IT role (e.g. developer, operations engineer).

Technologies

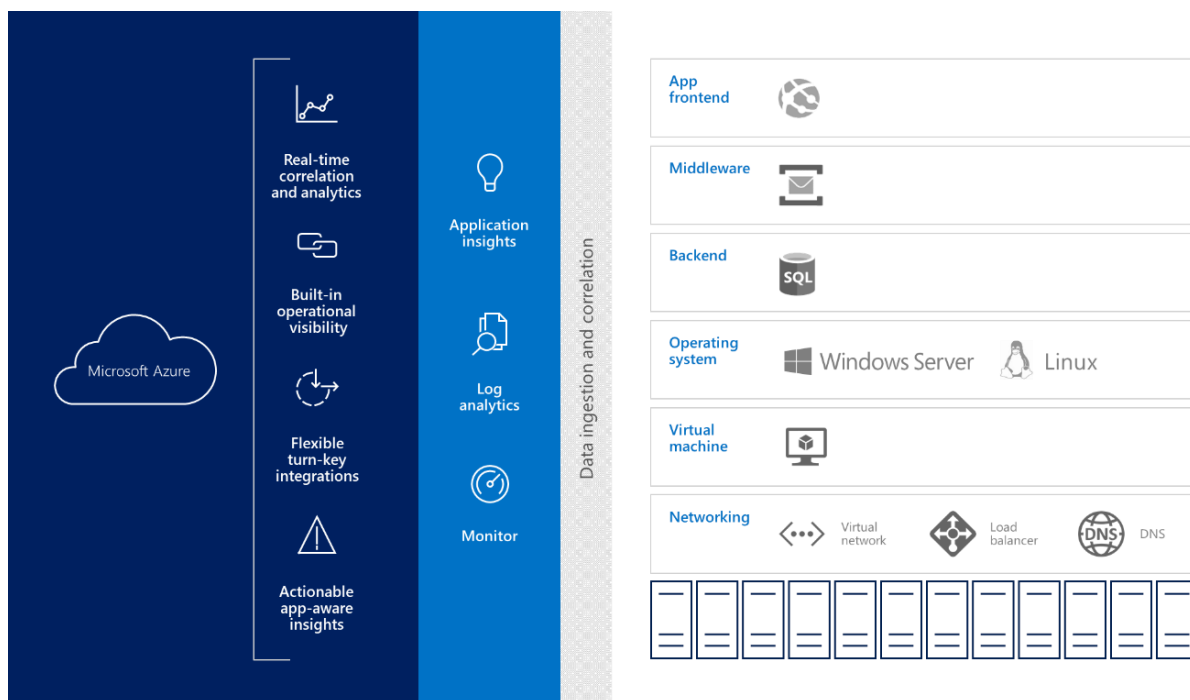
- Project Management and issue tracking tools (VSTS, MS Project, or similar).
- Infrastructure, Networking and Storage technologies (including MS SQL or other databases).
- Azure IaaS, Azure Backup, Azure Site Recovery.

Reskilling Existing Technical Resources

The cloud offers enterprises unprecedented opportunities for agility, innovation, and cost reduction relative to traditional IT. Transforming enterprise IT to take advantage of the cloud requires deep changes to existing IT processes.

DIFFERENCES IN THE CLOUD

Building a cloud practice and keeping the right resources on staff requires a fundamental shift in the way partners and customers have approached their existing environments. Many organizations have dedicated resources for managing their data centers and on-premises workloads, from application teams to the resources in the data center that are responsible for monitoring heating, ventilation, and cooling. These same resources can leverage their existing skills and with training can become cloud operators.



PLANNING CAPACITY

Planning, provisioning, and managing compute and storage capacity changes in several meaningful ways. Resources that were responsible for planning the location of racks and servers to meet data center plans no longer have to worry about where a SAN is located or if a rack is going to be close enough to an existing fiber run. Another fundamental difference in resource planning is the transition from planning for resources multiple years out to estimating consumption and costs of cloud resources and ensuring the workloads deployed are resilient and support cloud features such as horizontal scaling (or scale out).

In public cloud implementations such as Microsoft Azure, the cloud resources and services deployed will be managed to the boundaries and limits of Azure. Employees will still need to help customers understand these constraints and ensure application architects are designing systems that meet these constraints.

APPLICATION ARCHITECTURE AND DEVELOPMENT

Many applications are not designed for the cloud. As organizations transition to Microsoft Azure, there are opportunities with existing applications to refactor and transform either the entire application or specific components to take advantage of Azure. For instance, a traditional on-premises Microsoft SQL Server database requires not just a server to host the instance, but also the backend storage, networking, and configuration of functionality like clustering to meeting customer requirements. In Azure, there are opportunities to transition these workloads over to PaaS services where appropriate, allowing an operations teams more time to focus on the issues that matter.

Through managing less servers and licensing, application architects and developers can now focus on business requirements and objectives for customers. This brings quicker time to market with faster deployments, lower risk by leveraging platform services with defined SLAs, and the ability to make informed decisions based on the service capabilities.

NETWORKING

Existing network engineers and networking resources will have to make a transition to the cloud as well. While there will still be a need for resources who understand existing on-premises hardware such as firewalls and routers, there will also be a need for resources who understand software-defined networking.

The networking stack in Azure is very rich, with support for hundreds of network virtual appliances, including load balancers, next-generation firewalls, and much more. The technical team will need to understand how these virtual appliances can be integrated into customer deployments and know the Azure networking stack and how it can be configured to secure and protect customers in both cloud-only and hybrid cloud implementations.

DEPLOYMENT AND MONITORING

It is critical that technical resources understand how to manage deployments in Azure as well as monitor not just deployment health, but also platform health and service availability. They will need to learn new monitoring tools, including Application Insights and Azure Monitor logs with Log Analytics workspaces and how these tools can integrate both with each other and existing IT service management (ITSM) tools.

As the team masters the monitoring and maintenance tools in Azure, they will be able to monitor and maintain on-premises environments and/or hybrid environments to offer customers holistic management of their environment – regardless of where the servers or clients they have today reside. Being able to monitor and manage non-Azure environments with Azure services can accelerate revenue growth as customers begin consuming Azure management and monitoring services before they begin hosting applications in Azure.

This is also where the adoption of DevOps becomes critical. As deployments shift from “rack and stack” exercises to declarative and imperative provisioning and the creation of infrastructure as code, developers will need to manage your templates and scripts the same way they would manage the development of an application. This includes treating the templates as source code and storing and managing them appropriately.

HIGH AVAILABILITY, DISASTER RECOVERY, AND BACKUP

Deploying applications and workloads that are highly available is easier than ever with Microsoft Azure. Azure brings unprecedented scale to the public cloud with 50+ Azure regions and availability in 140 countries. However, access to a global infrastructure does not mean that applications can take advantage of it in their current state.

Applications may need to be refactored or rearchitected to take advantage of multiple regions and the technical team will need to understand the available services and platform functionality, including Azure Backup, Azure Site Recovery, and Azure Backup.

SECURITY AND COMPLIANCE

One of the best reasons to use Azure for customer applications and workloads is to take advantage of its wide array of security tools and capabilities. Microsoft Azure provides confidentiality, integrity, and availability of customer data.

The team will need to understand and be able to articulate to customers Microsoft's approach to Azure security as well as helping the application developers adopt the right services to make sure the applications they build and deploy and secure from the start.

This includes understanding services like Azure Security Center, Azure Monitor, Application Insights, Azure Advisor, and the networking controls that are available to secure both IaaS and PaaS services with service endpoints and service-level firewalls.

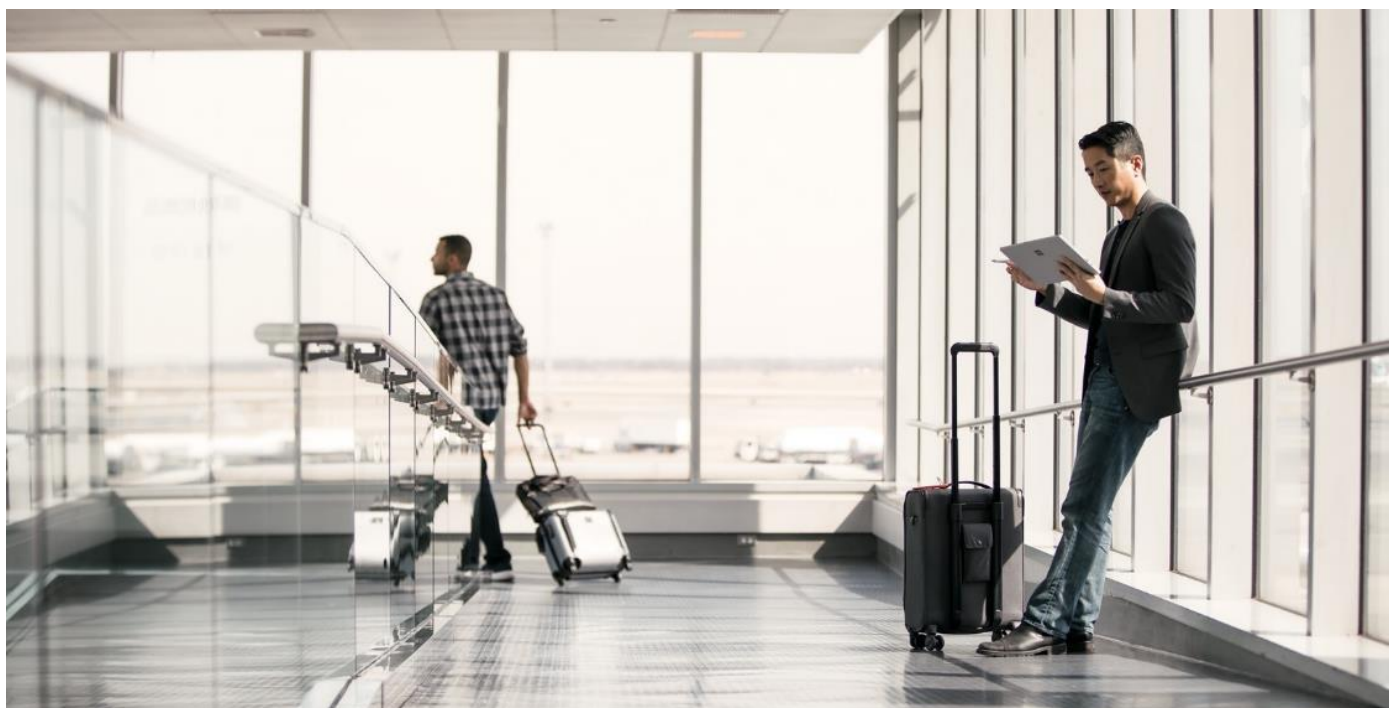
For the latest and most current information on the compliance and security of the Microsoft cloud, visit the Trust Center at <https://www.microsoft.com/trustcenter>.

BUDGETING AND COST CONTROL

Partners selling services around Microsoft Azure, your resources will need to understand the billing and cost management constructs Microsoft provides for the Azure platform. This will give your clear line of sight to not just your billing, but also downstream billing to your customers.

Up front costing can be determined with tools like the [pricing calculator](#) and the [Total Cost of Ownership \(TCO\) calculator](#). These tools can help you estimate your costs before a workload is even to deployed to Azure. Partners who participate in the CSP program have access to the [CSP Pricing Calculator](#) and the [CSPARM Pricing Calculator](#) on GitHub.

After applications have been deployed, they need to be tracked for spend. Your team will need to understand how to apply core platform governance features such as Azure policy to ensure resources are properly tagged for billing. There are additional tools to learn as well, such as [Cost Management](#) (formerly Cloudyn) which is critical to managing your spend in Azure.

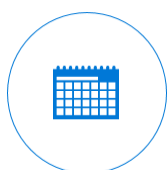


Hiring and Onboarding

As part of embracing a cloud-based world, there are steps to manage the acquisition and growth of technical expertise.



Mapping Existing Staff - the cloud requires new skills and a successful reskilling program will require identifying the skills and roles your practice will need going forward and then mapping the existing team to those roles. Once identified, an onboarding plan will be identified to help guide the new experts on their chosen path.



Interviewing/Hiring New - in many cases, the technical team will lack experience in cloud solutions and in those cases, skilled talent will need to be hired. Start with a defined role and the needed soft and technical skills for the role and then move forward with a recruitment plan to find the best fit. We will discuss options for finding new talent later in this section.



Onboarding - Every existing team member or new hire will need an onboarding plan to be successful. This includes external and internal training to learn the technology they will use on a day-to-day basis and the systems and IP that has been created to deliver solutions consistently. A good onboarding plan will also involve a method for ongoing training such as access to on-demand training, lab environments and the technical community.



Building a Technical Community - Technical communities can be an incredibly beneficial resource for increasing technical expertise. Putting subject matter experts on tools such as Yammer, Microsoft Teams, Slack (and many others) has the benefits of spreading their expertise across the organization as well as providing easy access to often asked questions and scenarios.



Rewarding Assistance - Giving the technical team the ability to communicate with each other and help others in the organization is a big step in building a learning and helping culture. Leaders should be encouraged to publicly praise or reward team members when they go out of their way to help each other.



Retaining your staff - Hiring and onboarding is only the first step among many to build a rock solid technical and business delivery team. Frequent communication about the company's goals, and about expectations is essential. Work with human resources to establish an employee retention plan. There are many low cost and low impact techniques to ensure a happy workforce.



Training & Readiness

Preparing and Training Technical Staff for the Cloud

Follow a learning curriculum to build the skills needed to stay relevant.

Suggested resources to help with onboarding your team and training are available in this section. This includes a range of on-line learning resources for self-paced learning, as well as options for instructor-led training for rapid technology adoption.

Use the following resources for training new and existing staff:

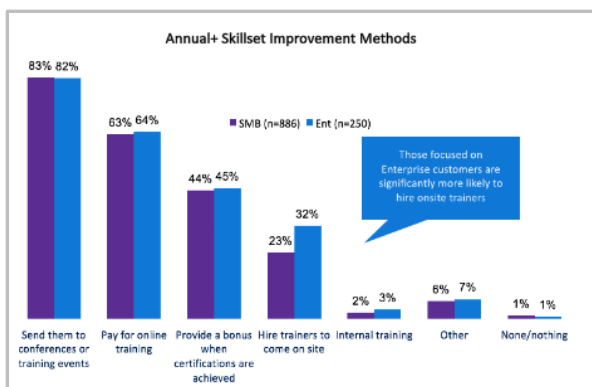
- [Microsoft Learn](#) offers training from the people who helped to build Microsoft Azure. From the basic overview to deep technical training, IT staff will learn how to leverage Microsoft Azure for their business.
- [Azure Training and Certification](#) provides free online training options including online courses, learning paths, hands-on labs, and resources to help find learning partners who can help develop skills using Microsoft Azure services.
- [Microsoft Azure Hands-on Labs](#) provides free, self-paced labs to stay current with Azure. The live environments are fully self-contained. Partners do not need their own Azure subscription to complete the labs, just login with a remote desktop (RDP) client and get started.
- [Microsoft Virtual Academy](#) offers training from the people who helped to build Microsoft Azure. From the basic overview to deep technical training, IT staff will learn how to leverage Microsoft Azure for their business.
- [Microsoft Learning](#) offers a wide variety of official curriculum on-demand, as well as [edX courses that are taught by Microsoft experts](#) for hands-on learning experiences with a broad range of Azure technologies.
- [The Microsoft Partner Network \(MPN\) Learning Portal](#) provides a centralized interface with training opportunities and certification options organized by products, competencies, certifications, and job role.

General Technical Training

Whether filling a skills gap or looking to improve the overall skill surface area, technical training is critical to partner success.



In our research, we found conferences and paid online training are the most common learning mechanisms.



Source: Microsoft Cloud Practice Development Study, MDC Research, November 2016

INTELLIGENT CLOUD SOLUTIONS

The [Intelligent Cloud Solutions](#) page provides a source of sales and technical training for partner practices and key areas of specialization. Resources include customer success stories, sales and technical training, tools, engines, and resources available to help build skills around selling, deploying, and architecting cloud infrastructure and management, cloud application development, data platform and analytics, and security and compliance solutions.

MPN LEARNING PORTAL

The [Microsoft Partner Network \(MPN\) Learning Portal](#) provides a centralized interface with training opportunities and certification options organized by products, competencies, certifications, and job role.

MICROSOFT CONFERENCE RECORDINGS

The [Microsoft Inspire Conference](#) and [Microsoft Ignite](#) provide many of their sessions as on-demand recordings — no conference pass required. Events such as Microsoft Ignite are live-streamed if your resources cannot attend in person.

PARTNER COMMUNITY EVENTS, CALLS & WEBINARS

The [Microsoft Partner Events](#) maintains a schedule of trainings available to partners. Visit often and plan a training calendar.

Additional Resources

[Microsoft Learning Partners](#) are available worldwide for Microsoft Azure via live instructor-led training. This can be scheduled as a dedicated delivery at a partner's location or virtually using remote learning technologies. Many courses are scheduled as open-enrollment courses, which do not require partners to schedule a dedicated class.

- [Opsgility](#) is a key Microsoft partner that offers Azure and Office 365 training. Find more than 200+ online classes focused on Azure with full learning paths for Azure certification. Opsgility also offers a full set of instructor-led Azure trainings that focus on architects, developers, DevOps, operations, sales, and decision makers.
- [Pluralsight](#) is a key Microsoft partner that offers Azure training. Gain the know-how and confidence the job demands through these free online courses, delivered in partnership with Pluralsight.
- [O'Reilly Safari](#) provides subscription access to more than 40,000 books, videos, and interactive tutorials from over 200 of the world's best publishers, including O'Reilly, Pearson, Harvard Business Review, and Packt. It also offers live online training courses led by instructors from O'Reilly's network of tech innovators and expert practitioners.



A woman in a white blazer is standing in front of a whiteboard, holding a marker and writing. The background is a teal gradient with a faint image of the same woman. The text is overlaid on the left side of the image.

“

#1 challenge for the cloud adoption is access to talent. Building a learning culture inside the organization is the success mantra for keeping our azure rockstars up-to-date on the ever improving azure platform.

ANIL SINGH

CEO, Hanu Software

Competencies and Certifications

There are numerous assessments and certifications to consider as motivation for advancing skills, creating proof points for the practice and achieving competencies and certifications.

MPN Competencies

One of the next steps is to ensure alignment between the technical team and the MPN competency need for the practice. The competencies most applicable to the Operations and Management practice are:

- Cloud Platform Competency
- DevOps Competency
- Application Development

The following tables summarize the **skill requirements** needed to achieve either gold or silver level for the competencies relevant to a SaaS practice. Some competencies have alternative options for achieving the competency. Only needs to be chosen for any given competency.

CLOUD PLATFORM COMPETENCY	SILVER REQUIREMENTS	GOLD REQUIREMENTS
Azure Consumption Option	<p>1 individual must pass one of the following exams:</p> <p>Developer: Exam AZ-203: Developing Solutions for Microsoft Azure</p> <p>Administrator: Exam AZ-103: Microsoft Azure Administrator</p> <p>Architect: Exam AZ-300: Microsoft Azure Architect Technologies* Exam AZ-301: Microsoft Azure Architect Design*</p> <p>* Both exams together are the Azure Solutions Architect Expert.</p>	<p>2 individuals must complete one of the following exams:</p> <p>Developer: Exam AZ-203: Developing Solutions for Microsoft Azure</p> <p>Administrator: Exam AZ-103: Microsoft Azure Administrator</p> <p>Architect: Exam AZ-300: Microsoft Azure Architect Technologies* Exam AZ-301: Microsoft Azure Architect Design*</p> <p>* Both exams together are the Azure Solutions Architect Expert.</p>

DEVOPS COMPETENCY	SILVER REQUIREMENTS	GOLD REQUIREMENTS
DevOps Partner Option	Two individuals must each pass the following assessment: Azure DevOps Assessment	Four individuals must each pass the following assessment: Azure DevOps Assessment

APPLICATION DEVELOPMENT COMPETENCY	SILVER REQUIREMENTS	GOLD REQUIREMENTS
Application Builder Option	Two individuals must each pass one exam. They can choose any exam from any of the options below: Azure App Dev Focus Developer: Exam AZ-203: Developing Solutions for Microsoft Azure Exam 70-487: Developing Microsoft Azure and Web Services Administrator: Exam AZ-103: Microsoft Azure Administrator	Four individuals must each hold a current version of the following certification: MCSD: App Builder

RESOURCES

→ [Microsoft Role-based Certifications](#)



Migration Assessment

Microsoft
Partner
Network

Cloud Migration and Modernization

aka.ms/practiceplaybooks



Executive Summary

Help customers identify their infrastructure and opportunities for migrating applications with Azure.

Before migrating a workload to Azure, first understand the current infrastructure and define what the migrated workload will look like to fully understand the migration process and costs.

Therefore, the first step to migrating or modernizing a workload with Azure is to build a Migration Assessment Plan.

Creating this plan typically has three main phases:

1. A **discovery** phase, in which the current applications and infrastructure are documented, as well as the business context and overall goals for the migration.
2. A **planning** phase, in which a detailed cloud design and migration plan are constructed.
3. An **evaluation** phase, which builds the business case for the migration for review and sign-off by the final decision makers.

This section of the playbook discusses each of the above stages in detail.

Creating the Migration Assessment Plan should be an iterative process. Identify applications for migration, create the assessment plan, and migrate. This allows improvement in future assessments as well as increased velocity as partners identify areas that can be improved in their process.

The migration assessment should answer the following questions for the customer:

- What applications and infrastructure am I currently running?
- Of these applications, which should be migrated, modernized, retired, replaced, or maintained on-premises?
- What are the risks associated with a migration, and how long will it take?
- What will be the return on my investment for migration, based on my current running costs, my post-migration running costs, and the cost of migration itself?
- What additional benefits will cloud migration bring to my business?

Top 4 things to do

A migration assessment is the foundation for successful migration project. Here are the top 4 things for successful migration assessments.

- Develop a comprehensive migration assessment methodology
- Choose the migration assessment tools
- Learn how to accurately calculate return on investment
- Build migration roadmaps, prioritizing the simplest migrations

Discovery

The goal of the discovery phase is to fully understand the existing infrastructure and applications, and the business context and goals surrounding those applications and their move to the cloud. This informs the planning and evaluations phases which follow.

It is important to understand how each application contributes to the business. What does it do? Who uses it? What is the impact of an outage? How important is business continuity and business assurance to the workloads being migrated? Placing the existing applications in their business context is essential to making informed decisions regarding prioritization, design, and indeed every aspect of the migration project.

Equally important is an understanding of the end users and how they use the application. In some cases, cloud migration will be a seamless change, of which users will be unaware. In other cases, users may experience significant changes, and may need to access applications differently, or perform specific tasks in new ways. To help users embrace rather than resist this change, it's important to understand the user experience, and to keep end users informed and engaged throughout the migration process.

Existing pain points (such as reliability, performance, or issues with functionality) should be identified. Migration to cloud is often an opportunity to reduce or remove such problems. These kinds of positive changes make it easier to get buy-in from both decision-makers and end users.

Non-functional requirements, such as reliability, performance, and forecasted scale must be understood so they can be factored into the design. The cloud offers far

greater flexibility than on-premises infrastructure and is therefore able to adapt quickly to changes in demand. Even so, some requirements—such as very high availability delivered through redundancy across more than one Azure region—have design and cost implications that need to be captured up-front.

Likewise, security and compliance requirements must be captured. Azure supports an extremely wide range of compliance certifications spanning many international, national, and industry-specific standards. Delivering an application that is compliant with a specific set of standards requires that the design be reviewed against the Azure guidance for those standards.

Of course, the discovery phase must also capture the details of the existing application implementation. The hardware, network, and storage infrastructure must be documented. It is important to capture the actual usage as well as the physical specifications. Traditional infrastructure is often over-provisioned to handle expected future demand or worst-case scenarios. The agility and elastic scale of the cloud offers the opportunity to optimize significantly on this approach. Usage should be measured at both normal and peak expected load. Data should be gathered on CPU, memory, network (latency and bandwidth) and storage (capacity, IOPS and throughput).



Any dependencies between components and systems, such as between applications and databases, must be identified and mapped. Understanding these dependencies is important when grouping and sequencing migrations during the planning phase.

Capture the current version of all software, and all operating systems—in some cases, updating the software to more recent, supported versions may be required as a pre-requisite to migration. Where software is developed in-house, the availability of source code and skilled staff familiar with the code must be established. Where software is licensed, vendor support for cloud technologies must be understood. For example, does the vendor already offer a cloud-based version of the software? If the software uses Microsoft SQL Server, has the vendor certified use with Azure SQL Database?

UNDERSTAND THE TOTAL COST OF OWNERSHIP

Having mapped the existing infrastructure and applications, the total costs of delivering those applications can then be analyzed. The [Azure Total Cost of Ownership \(TCO\) calculator](#) can help estimate on-premises costs but cannot capture all costs such as 3rd-party software licenses. Building a complete business case for the cloud requires building a full view of these costs. Some costs, such as servers and software licensing, are specific to each application; other costs such as operations staff and buildings are spread across applications and therefore may need to be apportioned appropriately. Remember to include backup, disaster recovery, software licensing, power, space, operations staff, support agreements, networking equipment, warranties, and Internet access. It is also important to understand the renewal dates for any leasing, licensing, warranty or support agreements, and the refresh cycle for all hardware, since this may create hard deadlines for migration, or impact prioritization to better leverage existing assets.

DISCOVERY TECHNIQUES AND TOOLS

A variety of methods must be employed to gather all this information. First, it is important to identify key stakeholders, such as application owners, relevant executives, technical staff, and end users. Interviews with each stakeholder will be necessary to understand their perspectives and priorities, and to gather their input on the topics listed above.

Various tools are also available to assist with gathering technical data on the existing infrastructure. In many cases, these tools can also help with the subsequent migration planning, costing, and even with the migration execution. Examples of available tools, from both Microsoft and third-party vendors, include:

MICROSOFT OFFERINGS















[Azure Migrate](#): The Azure Migrate tool can be used to assess on-premises workloads for suitability, as well as offering advice on performance-based VM sizing and cost estimations. The initial release of Azure Migrate only supports assessment for VMware VMs. Support for Hyper-V assessment and VMware migration is coming soon. Azure Migrate offers the following capabilities:

- Discover and assess on-premises VMs
- Confidently plan the migration
- Easily migrate customer workloads to Azure

More resources:

- [Assess on-premises workloads for migration to Azure](#)
- [Watch a Demonstration of Azure Migrate](#)
- [Azure Migrate hands-on lab](#)

	<p>Azure App Service Migration Assistant: The Azure App Service Migration site is a dedicated site to support migrating application to Azure App Service. For Internet-facing sites, an online assessment provides an initial migration compatibility report in seconds.</p> <p>Then download the App Service Migration Assistant for an in-depth assessment of both on-premises and Internet-facing applications, and automated migration of most modern ASP.NET applications.</p>
	<p>Azure Database Migration Service: The Azure Database Migration Service is a fully managed service designed to enable seamless migrations from multiple database sources to Azure Data platforms with minimal downtime.</p>
	<p>SQL Server Data Migration Assistant (DMA): Upgrade to a modern data platform by detecting compatibility issues that can impact database functionality in a new version of SQL Server and Azure SQL Database. DMA recommends performance and reliability improvements for the target environment and moves the schema, data, and uncontained objects from the source server to the target server.</p>
	<p>Azure SQL Database DTU Calculator: A Database Transaction Unit (DTU) is a blended measure of CPU, memory, and I/O used by an Azure SQL Database. Within each SQL Database service tier, Microsoft guarantees performance in terms of DTUs. The SQL Database DTU Calculator can be used to analyze the performance of existing on-premises databases, to calculate the number of DTUs (and hence service tier) required after migration to Azure SQL Database.</p>
<h3>THIRD-PARTY OFFERINGS</h3>	
	<p>Turbonomic</p> <p>Turbonomic plans reflect what your workloads actually need to run in the cloud—no more, no less. Get to the cloud quickly and safely, while avoiding cost-overruns or performance issues.</p>
	<p>Cloudamize</p> <p>The Cloudamize cloud infrastructure analytics platform helps make data-driven decisions with ease and confidence throughout the entire cloud journey.</p> <ul style="list-style-type: none"> • Assess: Which cloud is right for me and what will it cost? • Plan: How do I prioritize my applications for migration? • Migrate: How do I ensure my migration execution is right on the first try?
	<p>Movere</p> <p>More than just a point-in-time assessment, Movere enables a depth of monitoring, analysis and optimization unseen in any other platform. Movere organically scans environments globally at a rate of up to 1,000 servers per hour and multiple instances/environments in less than one day.</p>

	<p>RISC Networks</p> <p>RISC Networks CloudScope provides IT professionals with the most relevant infrastructure performance analysis needed to properly prepare for cloud, data center, and infrastructure projects. Agentless discovery of Network Devices, routers, switches, Windows and Linux Servers and more. Review the Asset Report in the RISC Networks Portal or download an excel spreadsheet.</p>
	<p>BitTitan Azure Assessments</p> <p>Provide detailed readiness reporting using cost analysis and planning tools to convince customers to adopt Azure. Take advantage of massive opportunities to move data out of SQL servers at end of life or support. Even uncover security concerns in customer infrastructure.</p> <ul style="list-style-type: none"> • Readiness check • Cost analysis • Detailed planning
	<p>TSO Logic</p> <p>The TSO Logic Platform provides the industry's most accurate data-driven analysis of total cost of ownership and cost modelling for an ideal future state. It ingests millions of data points from the current environment, including age, generation and configuration of all hardware and software running, and each instance's historical utilization. The Platform creates a fine-grained statistical model of compute patterns for all OS instances, showing how much is being spent, where a customer is over-provisioned, and where there are opportunities to realize significant savings both now and in the future.</p>
	<p>Corent</p> <p>Corent's SurPaaS® Platform is an Azure SaaS service that automates the scan, assessment, planning and cost modeling for customers workloads. It automatically migrates them to the cloud, and then monitors, manages, optimizes, and operates those workloads in the cloud.</p>
	<p>BMC Discovery for Multi-Cloud</p> <p>BMC Discovery for Multi-Cloud automates asset discovery and application dependency mapping to build a holistic view of all data center assets, multi-cloud services, and their relationships.</p>

Planning

The purpose of the planning phase of the Migration Assessment Plan is to build a proposal of what to applications to migrate, how to migrate each application, and when each migration should take place.

Having understood the various workloads and their underlying architectures during the discovery phase, the first step in the planning phase is to logically group the infrastructure and application components, and to prioritize their migration. This prioritization will be based on several factors: simplicity of application and infrastructure, number of dependencies, application criticality, limitations of existing hardware, and hardware and license refresh cycles. It is useful to approach applications from two different vantage points:

TOP DOWN

The top-down approach is focused on reaching that goal and begins with an evaluation of the various technical and security aspects of each application:

- Categorization of data, compliance, sovereignty and security risk requirements
- Current complexity of interface, authentication, data structure, latency requirements, coupling and application life expectancy of the application architecture
- Operational requirements like SLAs, integration, maintenance windows, monitoring and insight

Once analyzed, these aspects generate an overall score that reflects the difficulty of moving that application to the cloud. The top-down assessment also involves evaluating the application's financial benefits:

- Operational efficiencies, TCO, return on investment (or similar measurements)
- Overall computer load, seasonal fluctuations in usage levels, types of users (casual vs. expert, always online vs. only occasionally), necessary levels of scalability or elasticity
- Business continuity and resiliency requirements, any dependencies in the event of a service disruption

BOTTOM UP

Simultaneous to the top-down assessment, a bottom-up assessment can also be performed. And because this is more about the technical requirements and where an application could go, much of the information can be pulled from the assessment tool of choice. Requirements typically addressed with a bottom-up approach include:

- Max. memory, number of processors, operating system disk space, data disks, Network interface cards, IPv6, Network load balancing, Clustering, OS/ DB version, Domains supporting, Third-party components/ software packages

Once the applications and infrastructure have been grouped and prioritized, a migration plan should start to emerge. Depending on the number and complexity of applications, the migration plan may range from a relatively straightforward application migration schedule, or may be a complex multi-year strategic migration roadmap, with detailed application-level planning taking place iteratively as the roadmap is executed.

A pragmatic approach is preferred, preferring simpler, non-critical workloads for the early migrations. These are less risky and likely to be quicker to migrate and to present fewer issues. Early success builds confidence and allows demonstrable return on investment. It also builds experience, which reduces the risks associated with migrating the more complex workloads which follow.

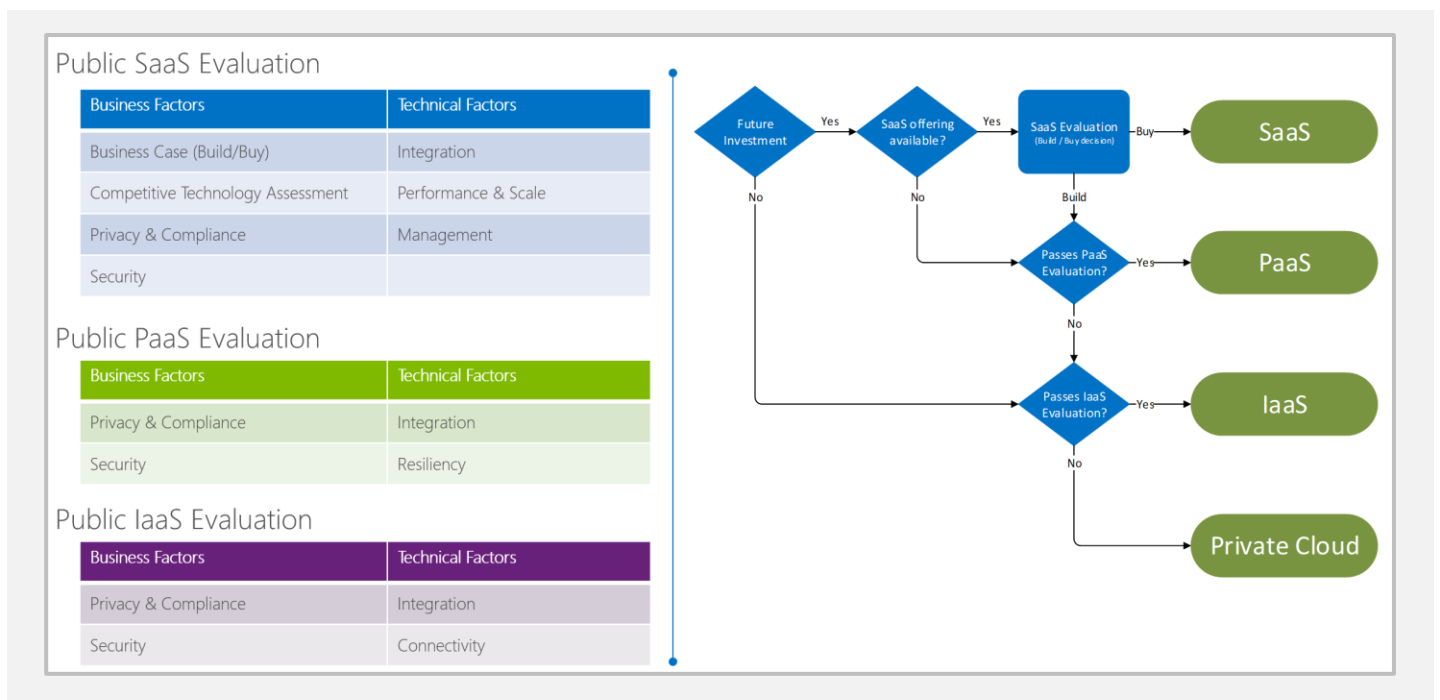
COMMON MIGRATION APPROACHES

When designing the migration for a specific workload, there are a number of choices available.

- **Retire:** Some applications may be end-of-life and can more easily be retired than migrated.
- **Replace:** Many common business workloads (such as Exchange or SharePoint) have equivalent SaaS offerings. Migrating to SaaS services offers an alternative to running application infrastructure in the cloud, typically with higher availability and lower TCO.
- **Rehost:** A 'lift and shift' approach, [in which applications are migrated to IaaS virtual machines](#), offers a fast migration path with a high level of compatibility with existing software. [SQL Managed Instances](#) is another option when rehosting SQL Servers. The compatibility benefits of virtual machines, but the benefits of PaaS.
- **Rearchitected:** [Converting applications to run as PaaS services](#) offers significant advantages over a simple rehosting in IaaS virtual machines, due to the lower on-going management complexity and costs. However, converting to PaaS may take longer and require greater technical skills, and the level of change—from minor refactoring to a complete application re-write—will depend on the existing codebase and the choice of PaaS technology adopted. As a result, while some applications may be converted to PaaS services as part of a migration project; more commonly they are first rehosted to IaaS and then evolved to take advantage of PaaS.
- **Retain on-premises:** For some applications, continuing to run on-premises may be the only realistic option, for example where regulatory requirements require data to reside within national borders and no local Azure region is available. In these cases, [Azure Stack may be a viable option](#).

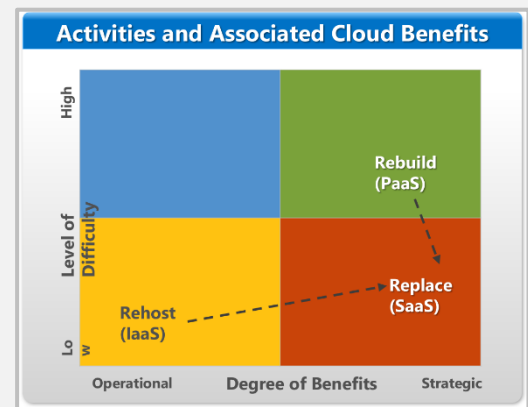
DECISION TREE FOR APPLICATION MIGRATION STRATEGY

A good practice is to build a decision tree based on the customer’s priorities and requirements to help decide the correct migration strategy. The following tree is an example of how to approach the application based on whether the workload is applicable to SaaS, IaaS or should be modernized with PaaS, or if it cannot be migrated at all should reside in a private cloud.



CONSIDER MIGRATION VELOCITY AND BENEFITS

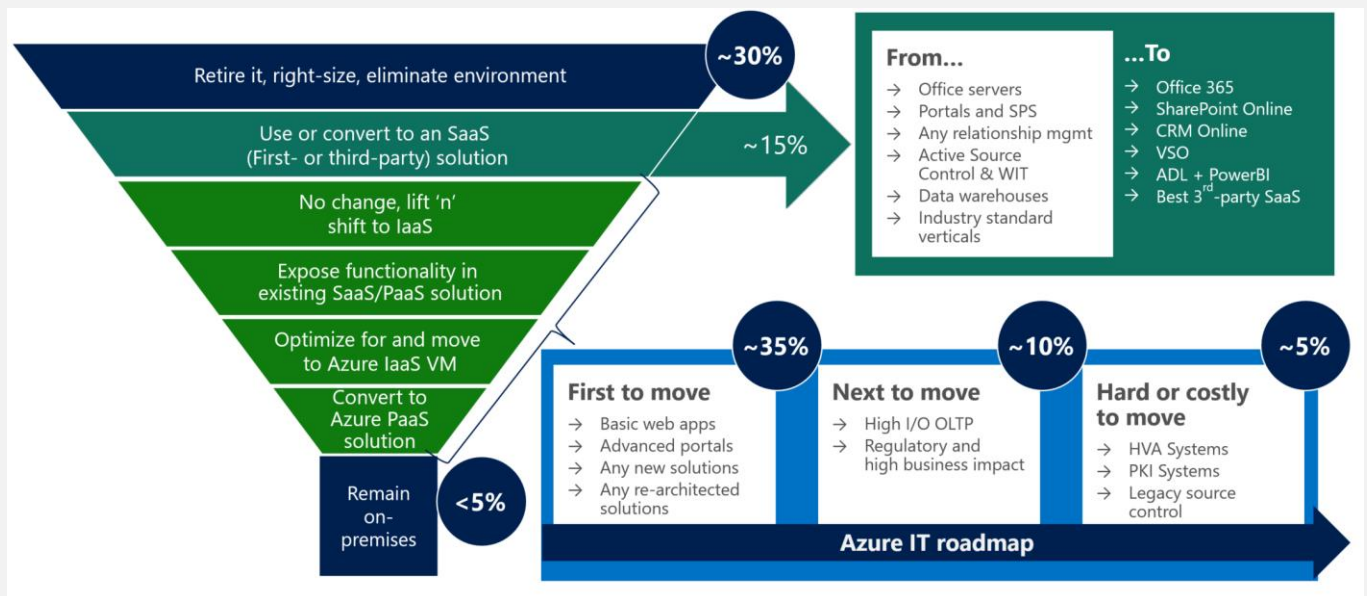
When considering the priority order of which workloads to migrate first, it's important to understand the relative difficulty level of each migration type as well as the benefit received for that option. For instance, rebuilding with PaaS will typically be the most difficult, but high on the strategic scale because it moves partners to managed services with more capabilities and developer agility. While rehosting with IaaS is most likely going to require the least amount of effort, it is also going to provide the least benefit. SaaS is the most strategic because it offloads everything but the administration and consumption of the service to the service provider.



STARTING THE MIGRATION PROCESS

After prioritizing the application portfolio, and then going through the decision-making process of the correct migration strategy for each workload, your team can then start the migration process in earnest. Migrations should be planned in sprints, so work with the customer to ensure that changes are communicated correctly, not just to stake holders but also to the product owners and any users that will be impacted by changes.

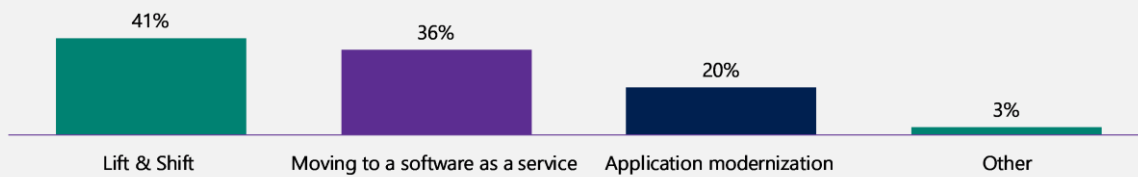
Example of an Enterprise Cloud Migration



SURVEY DATA

Our survey of Microsoft partners with an Azure migration practice measured how frequently each of the main migration approaches were used. Lift and Shift to IaaS was the most common approach, more than twice as common as application modernization.


Percentage of Migration Types Applied to Specific Workloads
(n=364)



Our survey also studied how the approach to migration varied across a range of common workloads. Most workloads employed an IaaS migration, except for database server and web servers which favored a PaaS migration, and training and service desk applications, which were more commonly retained in existing infrastructure.

Percentage of Migration Types Applied to Specific Workloads
(n=364)

	IaaS	PaaS	SaaS	None
File servers	57%	27%	26%	16%
Dev and Test environments	47%	40%	25%	24%
Active Directory / Identity management	46%	36%	32%	20%
Line of business applications from ISV	42%	34%	29%	22%
Custom Client-Server Applications	41%	32%	32%	23%
Management and monitoring applications	40%	33%	26%	27%
Database servers	45%	49%	30%	12%
Custom Web Applications	29%	36%	34%	25%
Training environments	27%	27%	22%	45%
Service desk applications	34%	21%	16%	45%

 Highest service frequency by workload

IAAS OR PAAS?

The choice between IaaS and PaaS is not all-or-nothing. A range of blended options exists, combining both IaaS and PaaS components in a single application architecture. For example, a traditional 3-tier application may see the application tier migrated to IaaS VMs, while the front-end tier is migrated to Azure App Service and the database tier is migrated to Azure SQL Database.

Where third-party software is used, licensing and support agreements must be reviewed to ensure that cloud-based deployments are fully supported, and to evaluate which cloud-based services may be used. For example, some software may support SQL Server running in IaaS, but not be validated with Azure SQL Database. In such cases it may be worthwhile to contact the software vendor to understand the cloud roadmap for the product.

While PaaS migrations may be more time-consuming and require greater initial investment, this cost is frequently outweighed by the benefits of reduced management overhead. This benefit is especially appealing to managed service providers. In addition to the efficiency gains of a PaaS approach, a modern application architecture also creates a foundation for higher-value customer offerings such as data analytics for increased business insight.

Many of the migration partners we interviewed described a 'PaaS-first' approach, preferring PaaS solutions where possible, recognizing that use of IaaS is inevitable in many cases, during a transition to PaaS or due to technical constraints such as the use of third-party software. In many cases, migration takes a phased approach, with initial migration executed as a 'lift and shift' to IaaS VMs, followed by additional phases to convert the application to make greater use of PaaS services.

OPPORTUNITY TO CLEAN UP LEGACY DEPLOYMENTS

In addition, cloud migration is often used by an organization as an opportunity to clean up legacy infrastructure and applications. It may be necessary to modernize certain infrastructure in-place before it can be migrated, for example, upgrading a legacy database or OS to a more modern version. Another common example is to consolidate on-premises databases prior to migration. Similarly, converting applications to PaaS may be an opportunity to combine several related applications into one.

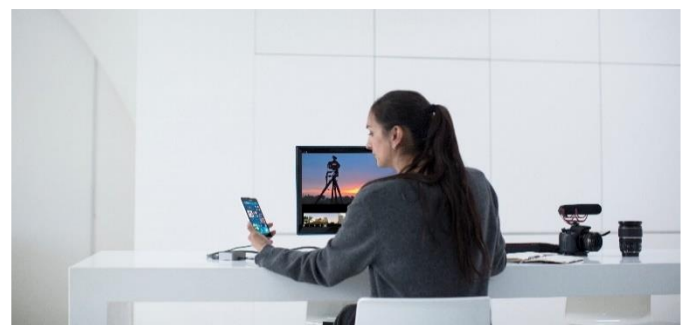
aka.ms/practiceplaybooks

WHICH AZURE SERVICES AND TOOLS

Having decided on the overall approach to migration, the design for the migrated application can be completed. This design describes in detail which Azure services are to be used, and how they will be configured. This includes:

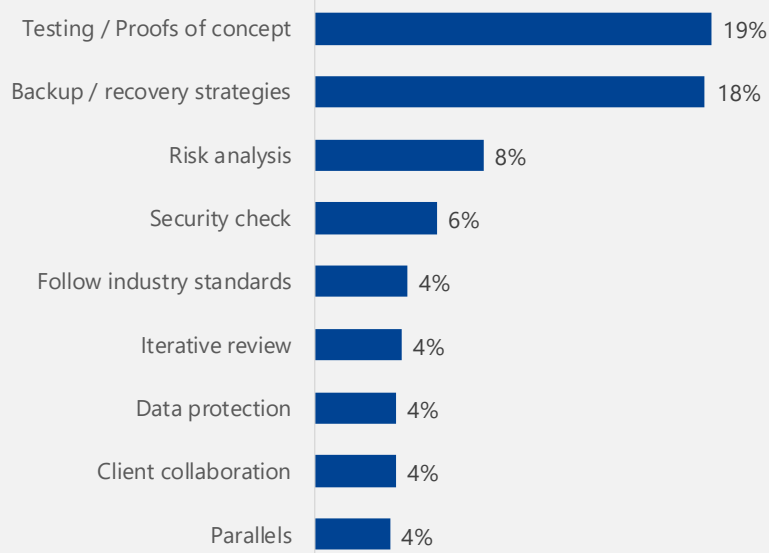
- The design for identity, such as whether Azure AD will extend an on-premises Active Directory or not, and whether to adopt hybrid identity.
- The design for [subscription and resource governance](#), including the [resource naming convention](#).
- The list of Azure services and resource types used, their deployment region, and the SKU or service tier in each case.
- The network design, including virtual networks and subnets, peering, network security groups, routes, connectivity to on-premises networks, and use of virtual appliances.
- The storage design, accounting for capacity, IOPS, and data throughput requirements.
- The design for backup, high availability and disaster recovery.
- The tools used for the initial deployment to Azure and on-going updates.
- Plans for on-going operations, including monitoring, alerting, reporting, patching, and scaling.

Remember when specifying the compute, network, and storage capacity of the migrated service not to simply copy the hardware specification of the on-premises system. Instead, scale the system based on the actual usage data gathered during the discovery phase, and design the system to scale as needed in future.



SURVEY DATA

In our survey of Microsoft partners with an Azure migration practice, we asked what safeguards and processes they used to mitigate risks for a migration project. The top responses were testing and proof of concepts, followed by backup and recovery strategies.



MINIMIZE DOUBT AND RISK WITH PROOF-OF-CONCEPTS

Technical uncertainties impacting the design can be resolved using Proof-of-Concept implementations, which can reduce the risk of unexpected discoveries impacting the migration schedule during the execution phase.

With the goal state of the migrated application in place, the design for the initial deployment and application migration can be completed. This describes in detail how each application will be migrated from the on-premises environment to Azure. This includes:

- How existing application data will be transferred.
- How traffic will be switched to new application endpoints.
- The nature and duration of any user impact during migration.
- Details of how the migration will be monitored and verified.

- The process for roll-back should the migration fail.
- Details of any supporting tools, such as Azure Migrate or Azure Site Recovery, that will be used.

PROJECT SCHEDULE

Finally, the Planning phase includes creating the schedule for executing the migration. Many of the migration partners we interviewed use a traditional milestone-based methodology, such as PMP or PRINCE2. A minority adopted agile methodologies such as Scrum.

PaaS migration projects typically follow traditional software development phases and milestones for coding, testing, user acceptance, staging, and production deployment, followed by additional milestones for data migration, verification, and endpoint cut-over.

IaaS 'lift and shift' projects may forego the coding phase, but the other phases are still typically required.

Evaluation

In the Evaluation phase, the last phase of the migration assessment plan,——help the customer understand the value proposition of going forward with a cloud migration by reviewing the benefits identified.

Earlier in this playbook, we discussed the different objectives and priorities that different organizations have when moving to Azure—from cost reduction, to increased agility, to improved service quality, to enabling new business scenarios. Understanding the customer and their motivations behind each migration is crucial to presenting the migration assessment in terms that relate to the customer and their concerns.

FORECASTING COST AND RETURN ON INVESTMENT

Cost is a universal concern. The discovery phase should already have captured an accurate picture of the existing on-premises application costs. To complete the business case, an accurate forecast of Azure costs is required. When computing these costs, consider the following points:

1. The choice of SKU or service tier is important since it can make a significant difference to pricing. A direct translation of existing on-premises hardware specifications into Azure Virtual Machine SKUs may not be optimal, since on-premises hardware may be under-utilized, and Azure hardware is typically newer, and hence faster, than on-premises hardware with a similar number of CPU cores. The [Optimize and Manage phase](#) is a key talking point in this discussion.
2. Remember to account for the hours that each service will run. It may be possible to scale back usage of Production environments at times of low usage, especially when using PaaS. Non-production environments such as Dev/Test should only be deployed when they are in active use.
3. The Azure Hybrid Benefit allows existing Windows Server software licenses (with Software Assurance) to be re-used on Azure Virtual Machines. This allows existing licensing investments to be leveraged, so the Virtual Machine is only charged at the 'bare metal' (Linux) rate. This can result in reductions in the Azure bill of 40% or even more. This benefit is also available for SQL Server licensees, allowing them to reuse their SQL Server licenses without paying the premium rate as well as significant cost savings for using Azure SQL Database. See <https://azure.microsoft.com/pricing/hybrid-benefit/> for further details.
4. Similarly, the License Mobility program enables Microsoft Windows Server application licenses (with Software Assurance) to be used in Azure, again allowing existing licensing investments to be leveraged in Azure. See <https://azure.microsoft.com/pricing/license-mobility/> for further details.
5. Some costs, such as Virtual Machine SKU, are relatively easy to predict. Other costs, such as bandwidth or storage access charges, vary according to application usage and can be difficult to predict without accurate data. Usage may need to be estimated based on transaction logs or other available usage information.
6. Where appropriate, take advantage of Azure Reserved Instances (<https://azure.microsoft.com/pricing/reserved-vm-instances/>) to further reduce Azure subscription costs.

These approaches, when used in combination, can offer very substantial cost savings. Further details on optimizing costs are given in the Azure Management Best Practices section of this Playbook.

Azure Migrate will provides cost estimates as part of a migration assessment. These estimates can be tailored based on a wide range of parameters, such as the VM family and size, use of discount plans, and hours of operation. The [Azure pricing](#)

[calculator](#) is another useful tool for predicting Azure costs based on expected usage and service tiers. Many third-party tools also offer pricing estimation.

On-premises infrastructure has the advantage of well-understood up-front pricing. By comparison, cloud costs can be more difficult to predict, and may vary over time, for example as usage changes, or as the service is optimized, or as the unit cost of Azure resources changes. As a Managed Service Provider, it is important to set appropriate expectations with the customer regarding costs, and it is generally better to slightly overestimate rather than underestimate what future costs will be.

Once the costs of the migrated service have been fully understood, they can be compared with the costs of the existing service as captured during the discovery phase of the migration assessment. This shows the return on investment that the cloud migration will deliver.

COMMON CONCERNS

There are several common questions or concerns that may be raised in the context of an Azure migration. Common concerns include:

- **Concerns over regulatory compliance:** Requirements should be captured during the discovery phase and addressed in the proposed design. Be careful of making assumptions or taking requirements on face value, since in some cases regulations may have been updated or misunderstood.
- **Concerns over security:** The proposed design should explicitly address security concerns and the Azure technologies used to mitigate common threats. In some cases, and particularly when using PaaS services, the security design of the Azure-based solution may be based on a different approach than that used on premises, with which the customer is more familiar (for example, being more focused on access control and credentials and less focused on network-level protections). Addressing these concerns may require the customer to adopt new security models. Alternatively, using IaaS services at least initially may be a pragmatic way forward.
- **Concerns over service availability:** The proposed design should address the requirements for backup, availability and disaster recovery, consistently with the published Azure SLAs.
- **Concerns over functionality, compatibility and performance:** A proof of concept is a common way to address functionality and performance concerns and build confidence in the proposed solutions.
- **Concerns over cost and cost variability:** A sound understanding of fully loaded costs for the existing application implementation, together with a complete costing for the Azure implementation, should be presented. While a high-level cost approximation may provide a clear business case, providing additional detail can improve forecasting and customer confidence.

A good understanding of the customer will enable Managed Service Providers to anticipate what is likely to be asked and prepare appropriate responses.

Microsoft
Partner
Network

Lift & Shift

Cloud Migration
and Modernization
aka.ms/practiceplaybooks



Executive Summary

Now the actual work of migrating workloads studied during the assessment phase begins. In this section, we'll consider 'lift and shift' migrations to Azure infrastructure services.

Migrating applications to Azure using Infrastructure-as-a-Service is often the quickest way to move applications to Azure, as well as requiring the least work and presenting the fewest risks.

For some applications, and some customers, migrating to Azure infrastructure is the end goal, and the application will continue to be operated in that way indefinitely. In many cases however, an infrastructure migration is used as a starting point, after which a longer-term program of application modernization begins. In this case, the end goal is for the application to be operated using Platform-as-a-Service technologies, giving lower management overhead and greater agility. Application modernization is the topic of the next chapter of this playbook.

The foundation of an Azure infrastructure deployment comprises:

- **Identity** – how will users authenticate and how resources are secured
- **Compute** – configuring virtual machines for availability, scale and performance
- **Network** – designing the network for current and future connectivity requirement
- **Storage** – planning for performance, durability, scalability and archival

This section discusses how to design and build this foundation and provides several resources to help.

We then discuss how to migrate the existing application servers—physical or virtual—to Azure. We provide guidance on how to choose the right Azure virtual machine and the various tools and methodologies to execute the migration itself.

Finally, we'll present some key topics that are important for Azure infrastructure migrations:

- Migrating VMware workloads
- Configuring backup and disaster recovery for migrated workloads
- Migrating existing Azure applications to Cloud Solution Provider (CSP) subscriptions
- Using Azure Stack as a migration target for customers that cannot migrate to the public cloud
- Migrating data and databases

Top 5 things to do

Here are the top 5 things to focus on during an Azure infrastructure migration.

- ✓ Deploy the network, compute and storage infrastructure
- ✓ Implement user identity
- ✓ Understand the available tools and approaches for server migration
- ✓ Configure high availability and backup
- ✓ Understand how to migrate data

Building Out the Network

The network forms the backbone of any application, both on-premises and in the cloud. It is the doorway to the end users, the glue that enables the application to function, and the security boundary against outside attacks. Establishing the right network architecture is a critical step in designing any cloud migration.

Designing the network can be one of the most technical and challenging aspects of a cloud migration project. The network design must consider a range of requirements and security threats. Common considerations include:

- Designing network topologies with the right connectivity for application access and internal traffic
- Creating hybrid networks connecting on-premises and Azure-based resources
- Routing traffic for scale, resilience and high availability
- Securing the network against outside threats, including DDoS
- Monitoring and trouble-shooting networking issues

Microsoft Azure offers an extensive range of networking services and features, enabling almost any networking topology to be created in Azure. This enables existing network topologies to be reproduced in Azure as part of a migration. Migration to the cloud is also an opportunity to align the network design with current best practices for performance and security.

In addition to designing the Azure network, remember to consider the existing on-premises network. What changes—temporary or permanent—are needed to make the existing network compatible with and extensible to the cloud?

AZURE NETWORKING

Azure provides a comprehensive range of networking services to support a wide range of network topologies. These are built using Microsoft's software defined networking technology, which offers cloud scale, fast provisioning, and virtual isolation of network traffic in the multi-tenant Azure environment.

This guide gives a short overview of the core Azure networking services. Microsoft publishes substantial additional documentation on each of these services online. For further reading, we recommend the [Azure Networking Overview](#).

VIRTUAL NETWORKS

[Virtual networks](#) are the most fundamental resource in Azure networking. A virtual network is a dedicated, private network space (for example, 10.0.0.0/16) within the Azure cloud. Resources, such as Azure virtual machines, can be allocated private IP addresses within this space, and use those addresses to communicate with each other.

Each virtual network can be divided into subnets, and virtual networks can be connected with each other using [peering](#) connections. Configure [user-defined routes](#) and [network security groups](#) (NSGs), which are like firewall rules, to control traffic in, out, and between subnets.

Inbound Internet traffic is supported by creating a [public IP Address](#), which can be either IPv4 or IPv6 (many IPv6 networking features are currently in preview). These IP addresses can be statically or dynamically assigned. Outbound Internet traffic is supported both with and without the presence of a public IP Address (and can be blocked using an NSG if required).

LOAD BALANCING OPTIONS

There are three alternative load-balancing technologies available in Azure. It is important to understand all three, and to design the application appropriately.

- [Azure Load Balancer](#) is a Layer 4 (TCP, UDP) load balancer that distributes incoming traffic among healthy virtual machines or other service instances. It can be used for both Internet-facing and internal application endpoints.

- [Azure Application Gateway](#) is a Layer 7 load-balancer with security and routing features, such as web application firewall, SSL off-loading, URL path-based routing and cookie-based session affinity. It is an example of a network appliance; a range of 3rd-party appliances is also available via the Azure Marketplace.
- [Azure Traffic Manager](#) is a DNS-based global traffic management service. It provides a range of traffic-routing capabilities, based on end user geo-location, endpoint proximity (based on network latency), and endpoint availability. Traffic Manager can be used to direct traffic between endpoints in different Azure regions, or between Azure and non-Azure endpoints.

Note that all three load-balancing services include endpoint health probes for back-end instances, and the ability to deliver high-availability by automatically removing failed instances from service and restore them once they return to health.

DNS SERVICES

Azure supports a range of DNS services and features, for use by both Internet-facing and internal applications:

- [App Service Domains](#) allows administrators to register a domain name, through a partnership with the name registrar GoDaddy. Originally part of Azure App Service, this is now available as a standalone service.
- [Azure DNS](#) host the DNS domain (whether purchased via Azure or elsewhere) on a global network of authoritative DNS name servers for high availability, low latency and support for all common DNS record types. Azure DNS is generally available for Internet-facing domains; Intranet-facing [private domain support is available in preview](#).
- [Azure-provided DNS](#) is the name given to the recursive DNS service provided by default to all Azure virtual machines. [Override the virtual machine DNS](#) settings at either the virtual network or individual virtual machine level to specify the recursive DNS server; the most common scenarios are to specify the DNS service of the Active Directory deployment when using domain joined virtual machines, or to enable DNS lookup for on-premises servers when using hybrid networking.
- Reverse DNS lookup is used to create a mapping from an IP address to a DNS name. Using Azure, I [configure the reverse DNS name](#) assigned to the public IP addresses assigned to your virtual machines. Or [host the reverse lookup zone](#) for the IP address block using Azure DNS.

SECURITY, MONITORING AND TROUBLESHOOTING

- [DDoS Protection](#): Azure provides two levels of DDoS protection. The basic level, which is free of charge, provides always-on traffic monitoring and real-time mitigation against common attacks. The paid-for standard-level service, which is currently in Preview, provides policies tuned using machine learning and real-time telemetry.
- [Azure Firewall](#) provides a central, fully scalable firewall-as-a-service for Azure-based applications. It can be used to filter all Azure traffic (inbound, outbound, between on-premises and Azure, and between networks in Azure).
- [Network Watcher](#) provides a central hub for a range of tools to view network settings across the deployment. It also provides several very useful tools for investigating network issues, such as the ability to run packet captures, and to verify connectivity from a virtual machine to a given endpoint.

HYBRID NETWORKING

Most Azure migrations use Hybrid Networking to connect to on-premises resources.

In some migrations, for a variety of reasons including data sovereignty or industry-specific regulations, it may be necessary for some parts of an application (typically the application database) to reside on-premises, whilst the other tiers of the application are moved to Azure. In these so-called 'hybrid' networks, a secure and robust connection is required between the database in the on-premises environment and the rest of the application in Azure. We call this connectivity between Azure and on-premises networks 'hybrid networking'.

Another example is Intranet applications. Even if the entire application stack is moved to Azure, some organizations will prefer to access Intranet applications over their internal network, rather than a public IP address. Here again, a secure connection between the on-premises network and the Azure network is required.

Azure provides two approaches to implement hybrid networking: Virtual Private Networks and ExpressRoute.

Virtual Private Networks

In a Virtual Private Network (VPN), traffic flows over the public Internet through a secure, encrypted tunnel, but appears from a networking perspective to be between two private networks.

Azure supports two types of VPN:

- [Site-to-Site VPN](#): Used to join on-premises networks to Azure, for example to connect application servers with database servers. At the Azure end, a VPN Gateway is deployed into a dedicated subnet in the virtual network. The on-premises network endpoint is a VPN gateway device. These devices form a VPN tunnel over which traffic between the networks flows.
- [Point-to-Site VPN](#): Used to join individual machines to the Azure network, for example to connect remote worker laptops to an Azure application, or for Dev/Test purposes. Once again, a VPN Gateway is deployed in Azure, in this case connecting to a VPN client deployed on the remote machine.

ExpressRoute

[Microsoft Azure ExpressRoute](#) extends on-premises networks into the Microsoft cloud over a private connection facilitated by a connectivity provider. All traffic flows over this private connection, not over the public Internet. As such, ExpressRoute connections offer a higher level of performance and reliability compared to VPN connections.

Where VPN connections provide connectivity only to a single Azure virtual network in a single region, ExpressRoute connectivity supports all Azure regions in a geopolitical region, or all regions worldwide with the ExpressRoute Premium add-on.

VPN connections only provide connectivity to Azure resources. ExpressRoute can establish connections to all Microsoft cloud services, including Microsoft Azure, Office 365, and Dynamics 365. For guidance on using ExpressRoute to access Office 365 visit <http://aka.ms/ExpressRouteOffice365>.

ExpressRoute offers a choice of [connectivity models](#):

- **Co-located at a cloud exchange** – For co-located facilities with a cloud exchange, order virtual cross-connections to the Microsoft cloud through the co-location provider's Ethernet exchange. Co-location providers can offer either Layer 2 cross-connections, or managed Layer 3 cross-connections between the infrastructure in the co-location facility and the Microsoft cloud.
- **Any-to-any (IPVPN) networks** – IPVPN providers (typically MPLS VPN) offer any-to-any connectivity between the branch offices and datacenters for integrating the WAN with the Microsoft cloud. The Microsoft cloud can be interconnected to the WAN to make it look just like any other branch office. WAN providers typically offer managed Layer 3.
- **Point-to-point Ethernet connections** – Connect the on-premises datacenters/offices to the Microsoft cloud through point-to-point Ethernet links. Point-to-point Ethernet providers can offer Layer 2 connections, or managed Layer 3 connections between the site and the Microsoft cloud.

ExpressRoute capabilities and features are all identical across all the above connectivity models.

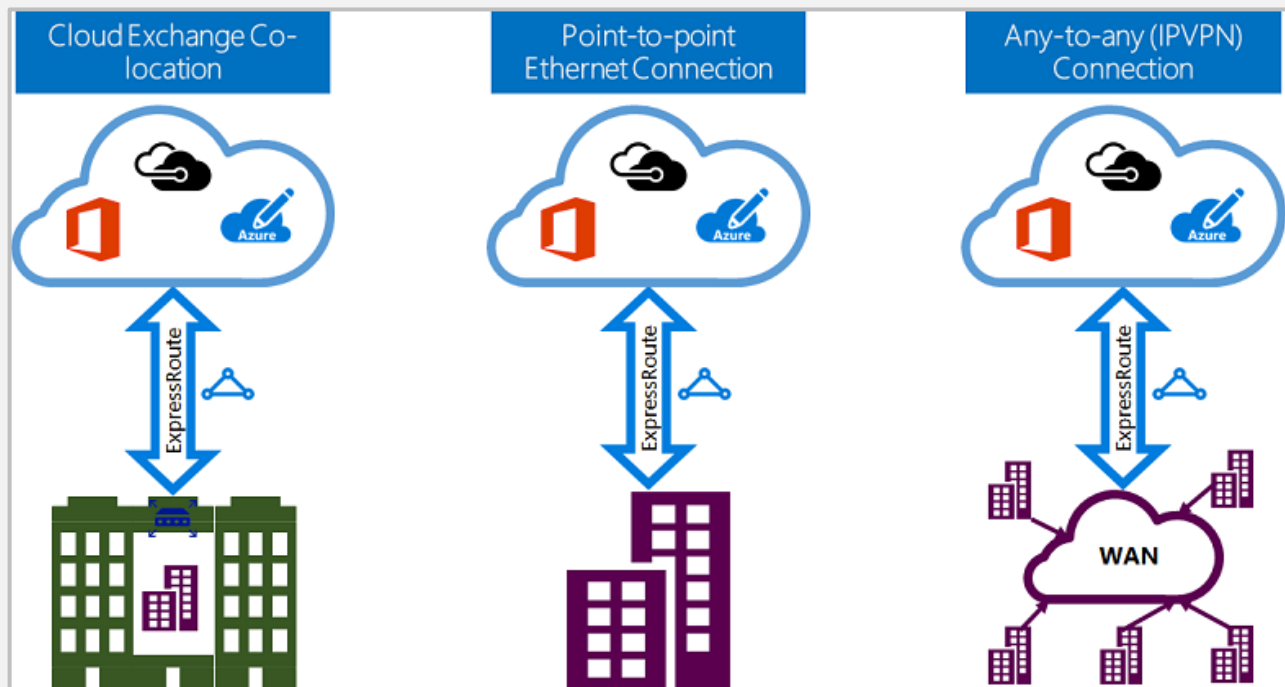
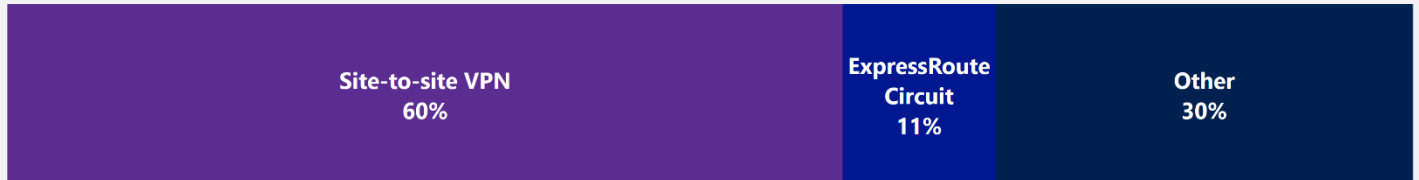
RESOURCES

➔ [Reference Architecture: Hybrid Networking](#)

SURVEY DATA

In our survey of Microsoft partners with an Azure migration practice, over 70% of migration projects included hybrid networking. Most of these (60%) used site-to-site VPNs, while 11% used ExpressRoute. These ratios changed significantly depending on the Partner’s customer focus, with enterprise-focused customers over three times more likely to use ExpressRoute than SMB-focused partners (22% vs 7%, respectively).

Mean Percentage of Migration Project Setup in the Past 12 Months
(n=364)



Network Appliances

Using Network Virtual Appliances can improve application security, help meet existing security policies, leverage existing licensing investments, and re-use existing skills.

Many on-premises applications make use of third-party network appliances, for example to provide additional security, availability, or custom routing features. Many of these appliances are available as '[network virtual appliances](#)' (NVAs) in the Azure Marketplace. Having identical or similar functionality available in Azure makes it much easier to migrate existing applications that use these devices.

Customers may have made a significant investment in these appliances, in terms of hardware and licensing, and in the configuration of the appliance to support their security policy and in training their staff. Minimizing change and maximizing re-use of existing investments can be an important way to remove obstacles from a migration project. Using a virtual appliance from the

customer's existing vendor makes it easy to re-use existing configurations and policies, as well as providing a familiar interface to existing staff. Moreover, in many cases the virtual appliances support 'bring your own license', so existing license investments can be re-used.

Microsoft provides its own network appliance, Azure Application Gateway, which we discussed earlier. Third-party network virtual appliances are available in the Azure Marketplace from a wide range of vendors, including Cisco, Barracuda, Check Point, Citrix, F5, and many more. For a full list, see the [Azure Marketplace](#).

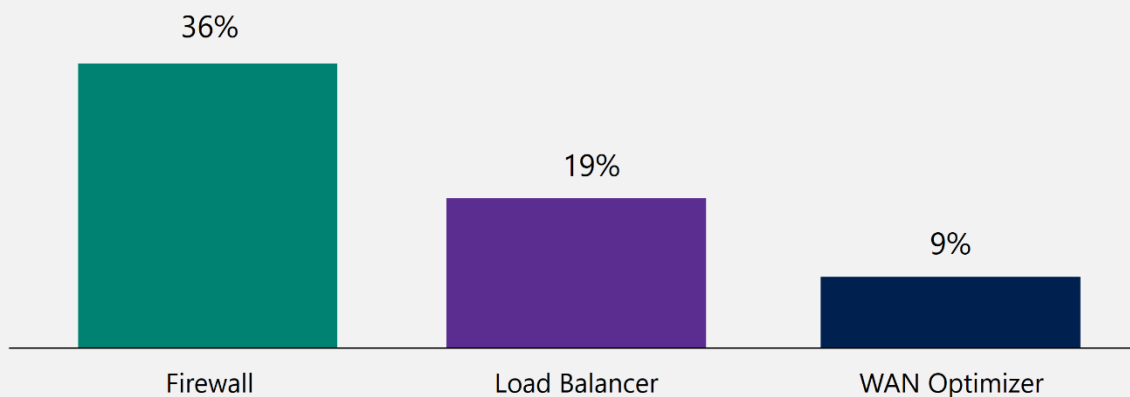
RESOURCES

→ [Reference Architecture: Networking DMZs with Network Virtual Appliances](#)

SURVEY DATA

In our survey of Microsoft partners with an Azure migration practice, network appliances from the Azure Marketplace were used in a significant number of projects. Most common were Firewall appliances (36% of projects), followed by Load Balancers (19%) and WAN Optimizers (9%).

**Mean Percentage of Migration Project Using Azure Certified Solutions
In the Past 12 Months
(n=364)**



The Virtual Data Center

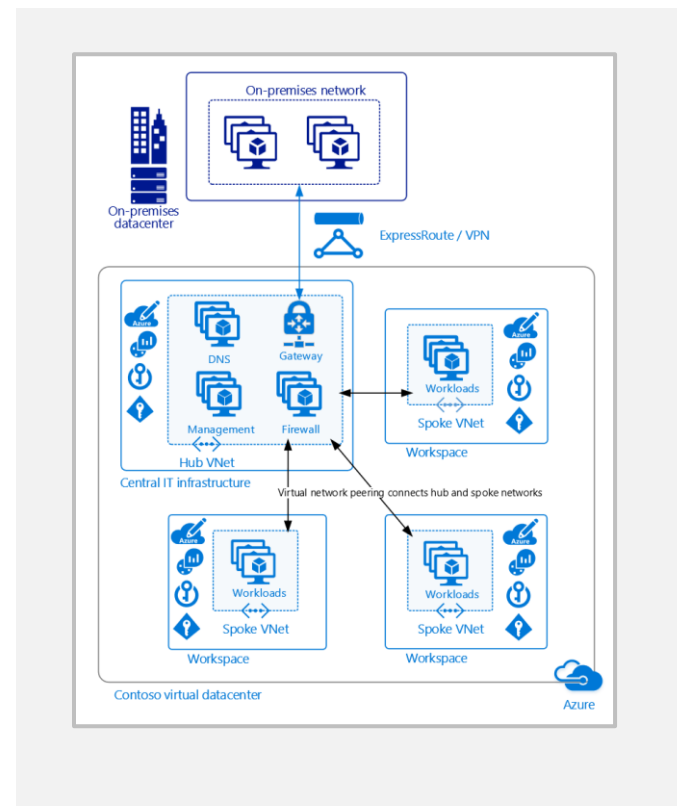
Azure Virtual Datacenter is an approach to making the most of the Azure cloud platform's capabilities while respecting existing security and networking policies. When deploying enterprise workloads to the cloud, IT organizations and business units must balance governance with developer agility. Azure Virtual Datacenter provides models to achieve this balance.

Unlike an existing on-premises datacenter environment, the Azure public cloud operates using shared physical infrastructure and a software-defined environment abstraction. The Azure Virtual Datacenter model can be used to structure isolated workloads in the Azure multi-tenant environment that meet existing governance policies.

With this approach, a set of shared services, access controls and policies are deployed and managed independently of the actual application workloads. These services include shared components such as load balancers, hybrid network connections, network security appliances, and management jump-boxes. Policies ensure that all traffic is routed through this shared infrastructure, which is responsible for implementing and enforcing governance standards.

Individual application workloads are then deployed separately into this infrastructure. Each workload uses a dedicated virtual network, integrated with the shared infrastructure using peering and routing rules. The resulting network is a 'hub and spoke' model, with a central hub of shared components, with each application workload isolated in separate spokes.

With this approach, partners can deliver a flexible and agile and yet also highly secure and compliant infrastructure, with the aim of meeting any customer security policy requirements and assuaging security concerns. By sharing network security infrastructure, partners can also demonstrate significant cost savings over siloed application deployments. These advantages will be especially important for Enterprise customers, who typically have more demanding requirements and a larger number of applications to migrate.



FURTHER READING

Deploying and configuring an Azure Virtual Datacenter requires a deep understanding of a broad range of Azure technologies. Fortunately, the Microsoft Customer Advisory Team (AzureCAT) have published extensive guidance, based on their experience of helping Microsoft's largest customers on their Azure journeys. For more information, see the [Azure Virtual Datacenter portal on the Azure Architecture Center](#), the [Azure Virtual Datacenter White Paper](#) and the [Mesh and Hub-and-Spoke Networks on Azure White Paper](#).

Enabling Hybrid Identity

As business move from using on-premises applications to cloud-based services, controlling access to those services becomes a challenge. Azure Active Directory provides the solution.

Controlling access to cloud-based services poses multiple challenges, such as knowing which services employees are using, ensuring access is authorized and authenticated And knowing to manage authentication credentials across all those services?

Azure Active Directory is a cloud-based directory and identity management service designed to address these challenges. It enables a single set of credentials to be used to authenticate access to thousands of common cloud-based services. By integrating with on-premises directories, this enables a common set of credentials to be used for both corporate and cloud applications. We call this *hybrid identity*. Enabling hybrid identity is a foundational step for most enterprise migrations.

When migrating applications to the cloud, it is important to understand how user identity will be managed. This should be designed early before any migration efforts have taken place beyond a proof-of-concept. Options include:

- Run Windows Server Active Directory (commonly referred to as AD) in the cloud using virtual machines created with Azure Virtual machines. This approach makes sense when e using Azure to extend an on-premises datacenter into the cloud.
- Use Azure Active Directory to give users single sign-on to Software as a Service (SaaS) applications. Microsoft's Office 365 uses this technology, for example, and applications running on Azure or other cloud platforms can also use it.
- Applications running in the cloud or on-premises can use Azure Active Directory Access Control to let users log in using identities from Facebook, Google, Microsoft, and other identity providers.

Organizations already using Office365 or Azure will already be using Azure Active Directory, since it is the authentication mechanism behind both services. In some cases, where different directories are in use, it is useful to [transfer ownership of their subscription to another account](#).

aka.ms/practiceplaybooks

SERVICE PRINCIPAL ACCOUNTS

As well as supporting user authentication, applications also use Azure Active directory to authorize access to the resources they need. They do this using special types of accounts, called [Service Principal accounts](#). Service Principal accounts can be created using the [Azure portal](#), [PowerShell](#), or [CLI](#).

Service Principal accounts should be configured with just enough permissions to run the tasks needed by the application, and no more. It is important that applications do not run in the context of user accounts since this breaks the principles of least privilege and individual accountability.

MANAGED IDENTITY

Managed Identity enables service principal accounts to be created automatically for Azure resources such as virtual machines. The credentials for the account are automatically provisioned into the virtual machine, using a virtual machine extension. Applications running on the virtual machine can then use those credentials to request access tokens which are used to authenticate requests to other Azure resources.

Managed Identities automate the full credential management lifecycle—from initial provisioning, through rollover, to deleting the Service Principal account when the resource is deleted. This greatly reduces the management overhead associated with credential management, whilst also increasing security.

To learn more, see [Managed Identity Overview](#).

RESOURCES

- [What is Azure AD?](#)
- [Deploying a Hybrid Identity Solution](#)
- [How to Deploy ADFS in Azure](#)
- [Azure AD Domain Services](#)
- [Azure Active Directory Proof of Concept Playbook](#)

Planning for Storage

Data storage is a critical feature of any application. Choosing the right storage technology will help create performant, cost-effective cloud deployments.

There are a wide range of data storage technologies available in Azure. Each offers different features, performance, resiliency and cost characteristics. It is important to understand the options before choosing the storage for the applications.

In this section, we will consider the storage options available to Azure Virtual Machines, considering both the disks attached to the virtual machines themselves, and shared file shares.

In addition, Azure supports a wide range of database options. These are considered later in this playbook: see the sections on [Migrating Databases](#) and [Modern Data Platform](#).

VIRTUAL MACHINE DISKS

The right storage combination can achieve [up to 2 PiB of storage per virtual machine](#), with up to 80,000 IOPS (input/output operations per second) and up to 2 GB per second disk throughput, with extremely low latencies for read operations. However, achieving this performance requires a large (and costly) deployment, and so while Azure supports extreme levels of disk performance, it is important to 'right-size' the design to avoid unnecessary cost.

The first step in planning disk storage is to identify the storage requirements—capacity, throughput, and read/write operations per second. This information should be gathered in the assessment phase, using tools such as Azure Migrate or Azure Site Recovery Deployment Planner. This information will help determine the storage architecture to use, for example the size, type and number of disks.

There are two technologies available for virtual machine disks in Azure. The original approach, which is still supported, is to store the disk image in a 'blob' in Azure storage. The current approach, called [Managed Disks](#), abstracts the disk as a first-class resource in its own right.

Managed Disks offer numerous advantages over blob storage and is the recommended approach for all new deployments.

When planning disk storage, choose an appropriate storage tier:

- [Standard HDD disks](#) are the lowest-cost storage tier. It uses traditional spinning disks, which offer higher latency than SSDs. Throughput and transaction rates are also lower than for SSDs, except for the smallest disk sizes. They are suitable for less performance-critical workloads and are ideal for Dev/Test environments.
- [Standard SSD disks](#) use solid-state drives to offer significantly lower latency rates than Standard HDD disks. Reliability is also higher, although the SLA is the same. Throughput and transaction rates are similar to Standard HDD disks, with some variation depending on disk size. Standard SSDs are suitable for Web servers, low IOPS application servers, lightly used enterprise applications, and Dev/Test workloads.
- [Premium SSD disks](#) also use solid-state drives to deliver low latency, and offer much higher transaction and throughput rates than Standard disks. They are suitable for database servers, file servers and interactive applications that require high throughput and low latency. They are also required to take advantage of the Azure SLA for single-instance virtual machines.
- [Ultra SSD disks](#) (in preview) offer the highest levels of throughput and IOPS, with consistent low-latency in the sub-millisecond range. They are designed for data-intensive workloads such as SAP HANA and critical databases.

In each case, it is important to understand the capacity, IOPS and throughput options available, and to choose a storage design that meets requirements. In some cases, to meet the required performance, it will be necessary to [stripe data across multiple disks](#).

File Shares

Cloud file shares make sharing files across cloud and on-premises servers easy.

[Azure Files](#) offers fully managed file shares in the cloud that are accessible via the industry standard [Server Message Block \(SMB\) protocol](#) (also known as Common Internet File System or CIFS). Azure File shares can be mounted concurrently by cloud or on-premises deployments of Windows, Linux, and macOS. Additionally, Azure File shares can be cached on Windows Servers with Azure File Sync for fast access near where the data is being used.

Azure File shares can be used to:

- **Replace or supplement on-premises file servers:**
Azure Files can be used to completely replace or supplement traditional on-premises file servers or NAS devices. Popular operating systems such as Windows, macOS, and Linux can directly mount Azure File shares wherever they are in the world. Azure File shares can also be replicated with Azure File Sync to Windows Servers, either on-premises or in the cloud, for performance and distributed caching of the data where it's being used.
- **"Lift and shift" applications:**
Azure Files makes it easy to "lift and shift" applications to the cloud that expect a file share to store file application or user data. Azure Files enables both the "classic" lift and shift scenario, where both the application and its data are moved to Azure, and the "hybrid" lift and shift scenario, where the application data is moved to Azure Files, and the application continues to run on-premises.
- **Simplify cloud development:**
Azure Files can also be used in numerous ways to simplify new cloud development projects, for example for shared application settings, diagnostics, or shared tools.

Azure files are a fully-managed service, requiring no patching or on-going management, and have built-in resiliency.



Choosing Virtual Machines

Helping customers choose the right virtual machines family and size, with the correct availability options, is an important value-add.

Migrating existing servers—whether physical, VMware or Hyper-V—comprises two main steps. First, an appropriate Azure virtual machine must be selected and provisioned. Secondly, the application files (executables, configuration, and local data) stored on the disks associated with the existing servers must be migrated.

In this section, we'll focus on the first step—choosing the right the virtual machines, and choosing the right high availability design for their deployment.

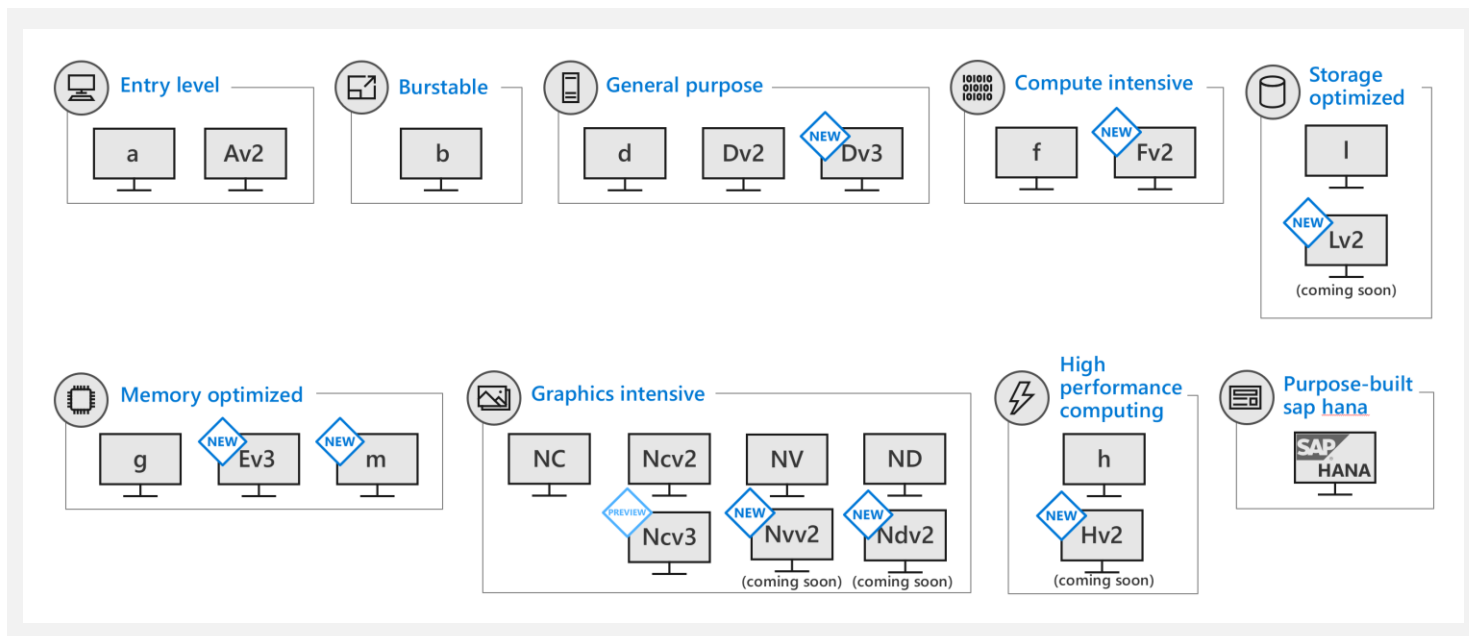
VIRTUAL MACHINE FAMILIES

Azure supports a wide range of [virtual machine families](#), with a wide range of compute and memory capabilities. The family and size of each virtual machine used should

be determined during the planning phase of the discovery process, based on the analysis of the existing application and hardware. For example, database servers and web servers may have very different memory requirements, and so use different virtual machine families.

Many of the assessment tools can help with this right sizing exercise. Even so, having intimate knowledge of the available options will undoubtedly be required.

Helping customers navigate the various VM compute series to ensure they are choosing the optimal size both for performance and for cost effectiveness for their workload is an incredible value-add and will be critical to the success of the project.



Availability Options

Beyond choosing the right family and size of virtual machine, choosing the right availability option will also be integral to the project at hand.

In addition to understanding the native availability requirements for the workload (for example: SQL Server Always On), plan and choose the right level of availability with Azure Virtual Machines, backed by a suitable [virtual machine availability SLA](#).

SINGLE INSTANCE VM

When any Azure virtual machine is backed by Azure Premium Storage (for all disks), Microsoft provides a 99.9% availability SLA.

AVAILABILITY SETS

[Azure Availability Sets](#) are a logical grouping capability in Azure to ensure that the virtual machines placed within it are isolated from each other when they are deployed within an Azure datacenter.

Using availability sets helps protect against outages caused by local failures, such as a top-of-rack network switch, or by rolling system updates such as host OS patching.

Availability sets offer a 99.95% availability SLA. This requires that at least two virtual machines are deployed.

AVAILABILITY ZONES

[Azure Availability Zones](#) are fault-isolated locations within an Azure region, designed with independent power, cooling, and networking. They help protect mission-critical applications from failures of entire datacenters, caused by events such as power or cooling failures, fire, or flood.

Availability zones are designed to be sufficiently isolated to protect against coordinated failures, yet close enough for low network latency between zones, so that write operations to [zone-redundant storage](#) take place synchronously.

Availability zones also support zone-redundant networking. Zone-redundant load-balancers can be used

to distributed traffic across virtual machine instances, both within and across availability zones. Zone-redundant public IP addresses enable a single public IP address to be shared across all zones, with traffic routed away from failed zones automatically.

Availability zones were made generally available as of March 30, 2018. They offer a 99.99% availability SLA. However, they are not yet supported in all Azure regions—see [Azure Regions](#) for details.

REGION PAIRS

Deploying application to more than one Azure region helps protect against large-scale region-wide disasters (such as hurricanes) with the potential to impact all availability zones within a region. However, this comes at a significant cost—as well as the increased Azure consumption arising from the larger deployment footprint, the application design must account for data consistency between regions and traffic routing, both before and during a disaster.

Cross-region data replication can be achieved using a range of database technologies, including [Azure SQL Database](#). Cross-region traffic routing and failover is provided by [Azure Traffic Manager](#), which supports a variety of traffic-routing policies.

When deploying an application to more than one Azure region, take advantage of [Azure region pairs](#). Each Azure region has a 'paired' region, and Azure avoids deploying system updates to both regions at the same time. Spreading the load across paired Azure regions helps protect against unexpected outages caused by Azure system updates.

RESOURCES

→ [Azure resiliency](#)

Customized Virtual Machine Images

Take advantage of custom virtual machine images to accelerate deployment time.

Many customers use virtual machine images in their existing virtualization environment complete with 3rd party and custom software ready for deployment. These images can be used in Azure as well, which can accelerate deployments by removing the need to change configuration settings and deploy software after the virtual machine is created.

CUSTOM IMAGES FOR WORKING TOOLS

Many of the partners we interviewed create custom images (Windows and Linux) that contain their custom applications and third-party tools they commonly use as part of a migration or modernization project.

With this approach, they get the additional benefit of a common working environment which allows for consistent behavior and a common set of tools. New teams can get started much faster and with less confusion since all of the environments have the expected set of tools and services when they start.

CREATING IMAGES

A first step should be to browse the Azure Marketplace, to see if there is an existing virtual machine image available that meets the needs. Using an existing image saves valuable time, however, if a suitable image cannot be found in the Marketplace, create a custom image as a baseline for the virtual machines.

The easiest way to create a new image is to start by provisioning a VM from the Azure Marketplace and then customizing it by installing software and services. After the VM is configured, run sysprep.exe with the generalize and shutdown options selected. Once the VM is shutdown, use the Azure capture command to store the image for later use. For more information, see [Creating Custom VM Images](#).

The open source tool '[Packer](#)' can also be used to create custom virtual machine images. To build images, define a Packer template file specifying the build process for the image. Packer supports integration with Azure, to define Azure resources such as service principal credentials. Packer will then deploy a virtual machine to Azure, perform the necessary build steps, create the image, and then clean up the virtual machine. This image can then be used as a baseline for more virtual machines. For more information, see [How to use Packer to create Windows virtual machine images in Azure](#).

UPLOADING EXISTING IMAGES

Using the Azure Command line tools or Storage Explorer, upload existing VHD files and register them as managed images that can be used to create new virtual machines in Azure.

For details, see the Migrating Disks section in this playbook.



Migrating to Virtual Machines

A critical step in any migration to Azure infrastructure is the migration of the servers themselves. We will discuss the migration approaches to use and the tools available to help.

There are two main approaches to migrating the application files to virtual machines in Azure:

1. Start with a clean Azure virtual machine and re-install and re-configure the application software,
OR
2. Port the existing machine disks, so that the virtual machine in Azure continues where the on-premises machine left off.

A clean installation on a new virtual machine has the advantage of leaving behind any legacy issues associated with the existing machine, such as patching history and device drivers. It creates a clean, up-to-date disk image and a clean install of the application software. It is also repeatable, which helps with reliable testing.

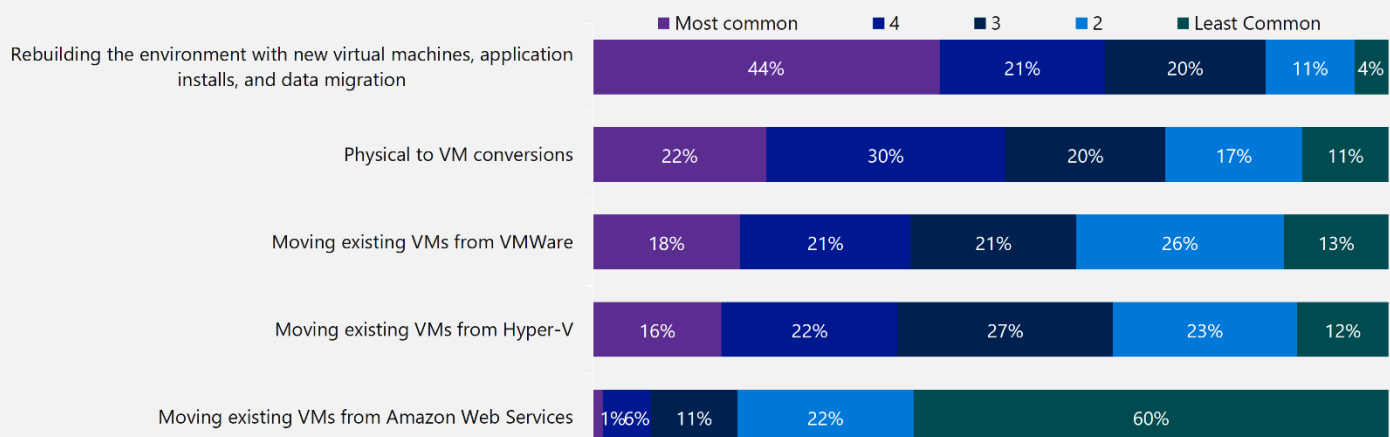
However, a clean install is only possible where the application installers and the knowledge to configure the application are available. For legacy applications, in practice, this is frequently not the case. For these applications, porting the existing disks to Azure is the fastest and most reliable way to migrate the application.

In the following sections, we'll discuss a variety of ways of implementing the second migration approach—porting existing disks to Azure. First, we'll consider the simplest possible approach of simply copying the disks. Next, we'll look at how migration can be streamlined using Azure Site Recovery. Finally, we'll discuss the features and resources available specifically to support VMware to Azure migrations.

SURVEY DATA

In our survey of Microsoft partners with an Azure migration practice, 44% reported that a clean application install was their most common approach to migration. The remaining 56% reported that their most common approach was to migrate existing servers.

Use of Lift & Shift Scenarios
(n=303)



Migrating Disks

Take care to configure the machine correctly when copying an OS disk to Azure.

The simplest way to migrate existing servers to Azure is simply to migrate their disks to Azure and then attach those disks to new Azure virtual machines. This is straightforward for data disks but requires some careful OS configuration when porting OS disks.

By using snapshots, disks can be exported from existing servers and imported into Azure while those servers are running. For stateless servers, this works well. However, for stateful servers (such as databases), any changes after the snapshot is created will be lost during the migration. In this case, to avoid loss of data, the service will need to be stopped during the migration process. Migrating disks in this way therefore incurs a longer system downtime than other migration approaches, such as using Azure Site Recovery. However, the simplicity of disk migration means it may be a good choice where this downtime is acceptable.

MIGRATING HYPER-V OR VMWARE VIRTUAL MACHINES TO AZURE

Importing Hyper-V disks to Azure only supports the VHD disk format, which must be a fixed-size disk no larger than 1,023GB. In addition, only 'Generation 1' VMs are supported. The VHDX format can be converted to VHD, and a dynamically expanding disk can be converted to a fixed-size disk. However, the VM's generation cannot change, and will therefore need to use a different migration approach with Generation 2 VMs, such as Azure Site Recovery.

Converting Hyper-V disks from VHDX to VHD format, and to fixed size, can be achieved using either Hyper-V Manager or PowerShell. For details, see [Prepare a Windows VHD or VHDX to upload to Azure](#).

The VMware VMDK disk format can also be converted to VHD using the [Microsoft VM Converter](#). For more information, see the blog article [How to Convert a VMware VMDK to Hyper-V VHD](#).

Also choose whether to 'sysprep' the VM. This is typically done to create a template to deploy several other VMs

that have a specific configuration. This is called a **generalized image**. For creating only one VM from one disk, sysprep is not required. In this situation, just create the VM from what is known as a **specialized image**.

There is a long list of additional OS changes that are required. For example, any static persistent routes must be removed, WinHTTP proxy removed, the SAN disk policy configured, the time zone set to UTC, and the power profile set to High Performance. For Windows, the full list is documented at [Prepare a Windows VHD or VHDX to upload to Azure](#). For Linux, see [Information for Non-endorsed Distributions](#) and the additional per-distro guidance pages. Once complete, the VHD file can be uploaded as a blob in Azure Storage. From there, use the VHD blob either to [create a Managed Disk](#) (in the case of a specialized image) or to [create a Managed Image](#) (in the case of a generalized image). These can then be used in Azure virtual machines.

MIGRATING AWS EC2 INSTANCES TO AZURE

Migrating AWS EC2 instances to Azure follows a similar process as used for Hyper-V or VMware, as described above. First, export the EC2 instance to a VHD file in an Amazon S3 bucket, using the instructions provided in [Exporting an Instance as a VM using VM Import/Export](#). This VHD file is then transferred to Azure and used to create a Managed Disk or Managed Image (both specialized and generalized VHDs are supported).

For further information, see [Move a Windows VM from AWS to Azure using PowerShell](#).

MIGRATING PHYSICAL SERVERS TO AZURE

Physical servers can be migrated to Azure by converting their disks to VHD files, then following a similar process as described above. This can be done using the [Disk2vhd utility](#). This uses the Windows Volume Snapshot capability to create consistent point-in-time disk snapshots, allowing the VHD to be created without system downtime.



Migrating using Azure Site Recovery

Azure Site Recovery (ASR) is an Azure service designed initially to enable customers to deliver high availability applications by enabling automatic failover to Azure infrastructure in the event of failure. By treating an application migration as a controlled failover without failback, it can also be used to migrate applications to Azure.

The requirements for failover to a backup site are similar to the requirements for a migration. Key concerns focus on data replication and integrity, re-directing the network traffic, and minimizing downtime. While initially designed for failover, [Azure Site Recovery](#) supports both failover and migration scenarios.

[Azure Site Recovery for Migration](#) supports migration from physical machines and both VMware and Hyper-V virtual machines to Azure. The [ASR Deployment Planner](#) can be used to help assess the workload to be migrated and determine details of the migration such as the network capacity needed and Azure virtual machines to use.

Once installed and configured, the migration starts by copying data (disks) from each machine to be migrated. Once completed, continuous incremental data transfers are used. This all occurs while the on-premises application is fully operational, prior to any application migration work window.

Data synchronization integrates with a wide range of common application workloads to ensure the synchronized snapshots are application consistent. Common workloads supported by Azure Site Recovery

include SQL Server, Active Directory, Exchange and Oracle Data Guard. For a full list of supported workloads, see [What workloads you can protect with Azure Site Recovery](#).

During application cut-over, virtual machines in Azure are booted using the replicated data. A final incremental data transfer is used to ensure the Azure virtual machines are fully up-to-date. Azure Site Recovery supports an RPO (Recovery Point Objective) as low as 30 seconds, enabling the work window during which applications are unavailable to be kept as small as possible.

The application traffic is then switched to the Azure-based service. For Internet-facing applications, the Azure service will have a new IP address, hence this switchover happens at the DNS level. Azure Site Recovery integrates with Azure Traffic Manager, enabling this transition to happen quickly and easily. For Intranet applications, there are two possible approaches: changes to the internal IP address require the internal DNS record to be updated; alternatively, the internal IP address can be preserved, with switchover by changing the IP subnet routing tables. This last approach is useful if application dependencies require that existing internal IP addresses must be preserved.

Migrating VMware Workloads

Many customers use VMware to run virtualized applications, either on-premises or with a hosting provider. The overall process of migrating VMware virtual machines to Azure follows a similar sequence to any other migration. However, there are some considerations that are specific to VMware.

Migrating VMware virtual machines to run in Azure is one of the most common migration scenarios. Microsoft has published a detailed [Migrating VMware to Microsoft Azure TCO Guidance](#) document to help demonstrate the value proposition that underpins these migrations.

The overall process for migrating VMware workloads to Azure follows a similar path to other workloads, using tools we have already seen such as Azure Migrate and Azure Site Recovery. To help support VMware customers on their migration journey, these services are designed and tested specifically with VMware workloads in mind. Existing VMware administrators unfamiliar with Azure may appreciate [free online training to introduce Azure VMs to existing VMware vSphere administrators](#).

There are many technical compatibility issues to consider when planning a [VMware to Azure migration](#), such as firmware version, OS version, VMware version, disk size, disk format, network capacity, etc. The Azure Migrate and [Azure Site Recovery Deployment Planner](#) tools can be used to automatically assess an existing VMware environment prior to migration, giving a report of issues that needs to be addressed.

There are a variety of approaches available for VMware workload migration, summarized below.

AZURE MIGRATE

As well as supporting VMware migration assessment, Azure Migrate also provides migration of VMware workloads. Using Azure Migrate for both assessment and migration provides an integrated experience, streamlining the overall migration process.

VMware support for migration requires registration in advance. The preview is offered with full support for production workloads.

AZURE SITE RECOVERY

Microsoft publishes [step-by-step guidance on how to prepare and configure VMware migration to Azure using Azure Site Recovery](#), including specific considerations for Cloud Solution Provider (CSP) subscriptions.

In some cases, partners may already be hosting customer applications in their own on-premises VMware infrastructure. Where this is a multi-tenant environment (supporting more than one customer), it is important to ensure tenant isolation during the migration process. This can be achieved by [ensuring the vCenter access account used by the Azure Site Recovery configuration server only has the correct access permissions](#).

PORTING VMWARE DISKS









Instead of using Azure Site Recovery, an alternative approach for VMware to Azure migration is to simply port the existing disks to Azure and assign them to a new VM. Several steps are required to properly configure the OS for Azure, and to convert the disks to the correct format. For details, see the Migrating Disks section of this playbook.



VMWARE IN AZURE

In the rare event that compatibility issues prevent an existing VMware workload being migrated to an Azure virtual machine, there is an alternative. Microsoft recently announced preview support for [VMware virtualization in Azure](#). This offering allows customers to access the full VMware stack on dedicated Azure hardware, co-located with other Azure services. While not offering the same cost-savings or agility of migrating to Azure virtual machines, this option does provide an additional option to reduce on-premises infrastructure.

Migration Tools

Following is a list of the most common migration tools and services used when migrating servers to Azure Virtual Machines. See also the tools listed under Discovery, many of which can also support migration.

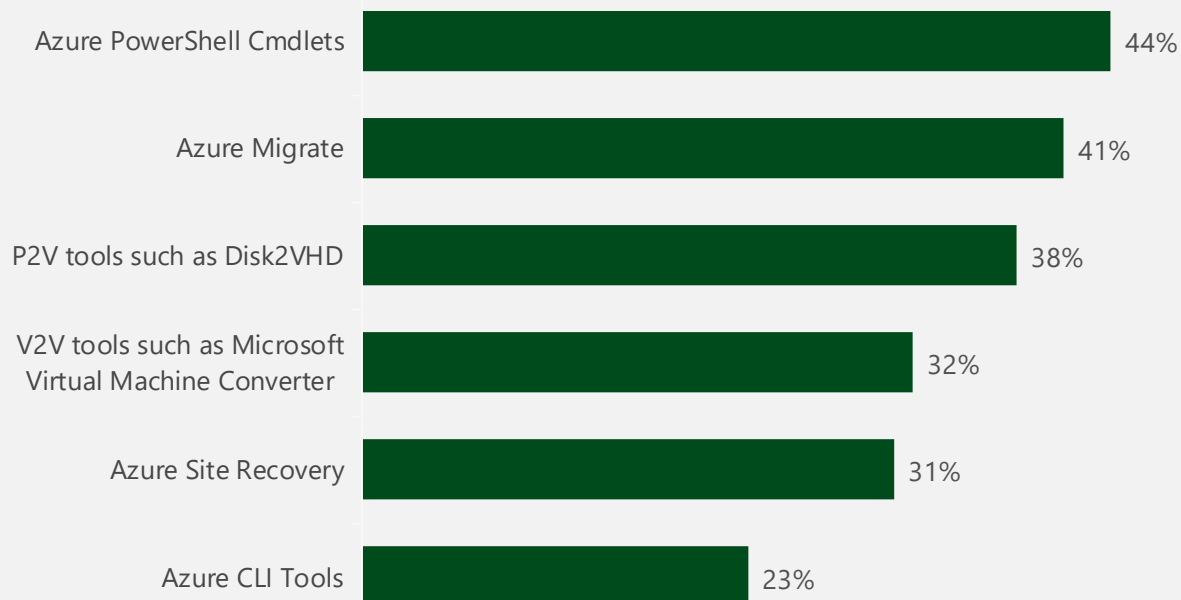
	<p>Azure PowerShell and Azure CLI</p> <p>The Azure command line tools are some of the most frequently used tools with migration. Tasks range from uploading images to Azure Storage, to automating Azure infrastructure during a failover from Azure Site Recovery with a recovery plan.</p>
	<p>Azure Migrate</p> <p>Azure Migrate support for VMware migration is in preview, and requires registration in advance. The preview is offered with full support for production workloads.</p>
	<p>Azure Site Recovery</p> <p>As well as offering a Disaster Recovery solution for failover to Azure VMs, Azure Site Recovery can also be used to replicate and migrate VMs to Azure at scale. Azure Site Recovery can be used with Azure Migrate (which helps with migration assessments).</p>
	<p>CloudEndure</p> <p>CloudEndure® provides an automated migration solution that utilizes block-level continuous replication, application stack orchestration, and automated machine conversion to ensure data integrity during migrations to, between, or within the cloud.</p>
	<p>ATADATA</p> <p>ATADATA's automates the migration of workloads and also auto-provisions multi-tiered live workloads directly into Hyper-V, Azure Classic or Azure Resource Manager environments. The agentless architecture is managed through the ATADATA Console which installs entirely behind client firewalls, or via a hosted/SaaS deployment.</p>
	<p>Dataometry</p> <p>Datometry Hyper-Q simplifies enterprise management of existing applications and develops new applications on cloud databases.</p>
	<p>Unittrends</p> <p>Automatically remodel VMs to native cloud virtual machines in low cost AWS S3 or Azure Blob storage. This free tool even handles the transformation of networking differences between VMware and AWS.</p>
	<p>Attunity</p> <p>Attunity Replicate empowers organizations to accelerate data replication across a wide range of heterogeneous databases.</p>

	<p>CloudAtlas</p> <p>CloudPilot's static code analysis ensures a deep understanding of applications and data to successfully migrate them with no code changes to Azure Containers, Azure Virtual Machines, or Azure App Service (PaaS). Migrate in minutes, not months. CloudPilot checks applications for security, governance, and regulatory compliance, and with coding best practices to reduce migration risks and costs.</p>
	<p>ZeroDown Software</p> <p>ZeroDown enables companies to move applications from their data center or other hosting platform to Azure with no interruption of service. The tool automatically synchronizes the source and target environments and is transparent to the migrated application's users with no cutover problems or loss of in-flight transactions.</p>

SURVEY DATA

In our survey of Microsoft partners with an Azure migration practice, PowerShell was the most-used tool during workload migration. Use of Azure Migrate was in second place and was first place among practices with an enterprise focus.

Azure Tools Used During Workload Migration (n=364)



Business Continuity

Providing guaranteed continuity of service even in the event of failures is a key requirement of any business application. Services in Azure are no exception.

Business continuity requires that services keep running, or recover quickly, even when disaster strikes. That could be a large-scale IT failure, or a data loss event.

This presents both a challenge, and an opportunity. Use Azure services to business continuity for customers. In designing a business continuity service, be prepared to answer common customer questions, such as:

- Will you help me restore my data when it is corrupt or lost? Will you take care of my data's long-term retention compliance requirements?
- Will you protect my mission critical applications? Will you make DR and recovery plans and run DR drills?
- Will you ensure business continuity in case of any interruption? What kind of SLAs will you provide?

To protect against IT failures, Azure Site Recovery enables critical workloads to replicate their running state to Azure, and to rapidly fail over to an Azure-based infrastructure when needed. Whilst this creates data redundancy, it does not protect against data corruption, accidental data deletion, or ransomware, and thus additional data backups, as provided by Azure Backup, are also essential.

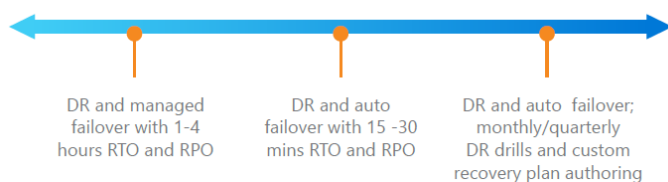
These services can be used to protect both cloud workloads and on-premises deployments. As a Managed Service Provider, providing additional resilience to on-premises applications by integrating these Azure services

can be a 'quick win', helping to build customer trust prior to a full Azure migration.

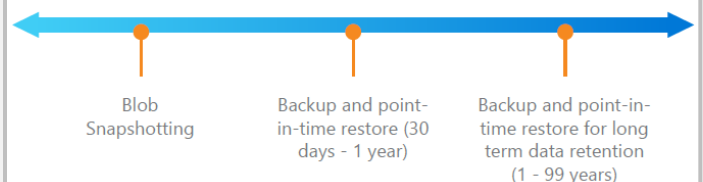
Microsoft Azure offers a rich set of services to backup workloads and fail over critical workloads, in addition to hybrid storage support.

- [Azure Backup](#) supports file and folder-based workloads, virtual machine backup, as well as workload specific support such as Hyper-V, VMware, SharePoint, and Active Directory straight to Azure, removing the need for physical backups like tape. Azure Backup also provides backup for SQL Server running in Azure VMs (without requiring any backup infrastructure) and for Azure Files shares.
- [Azure StorSimple](#) is an appliance (physical and virtual) that provides hybrid tiered storage to automatically offload data to the cloud.
- [Azure Site Recovery](#) protects important applications by coordinating the replication and recovery of physical or virtual machines. Replicate to the datacenter, to a hosting service provider, or even to Azure to avoid the expense and complexity of building and managing a secondary location. Azure Site Recovery continuously monitors service health and helps automate the orderly recovery of services in the event of a site outage at the primary datacenter.

VALUE ADD SPECTRUM - DISASTER RECOVERY



VALUE ADD SPECTRUM - BACKUP



Migrating Existing VMs to CSP

In some cases, a customer may have an existing Azure-based application, which they wish to transfer to the Managed Service Provider for on-going maintenance and monitoring.

Migrating an existing application to a Managed Service Provider may require that the Azure resources that implement that application be migrated to a different Azure subscription. This will typically be a CSP subscription managed by the Managed Service Provider on behalf of the customer.

Azure supports moving resources between subscriptions. However, there are a small number of limitations to be aware of.

First, Azure only supports resource move where both source and destination subscriptions reside under the same Microsoft tenant (directory). The existing subscription may use a different Azure AD tenant or may use a Microsoft account. In either case, it must be transferred to the tenant created when the customer account is created in the CSP portal.

Second, not all Azure resources are available in CSP subscriptions. The Azure CSP Migration Assessment Tool can be used to identify any existing resources that are not currently supported in CSP subscriptions, or which do not support resource move. In some cases, the tool will specify modifications which are required prior to migration; in other cases, the tool may identify existing resources which cannot currently be migrated. In this case, changes to the application may be required to remove this component.

Third, CSP subscriptions only support the Azure Resource Manager deployment model, and not the Azure Service Management ('classic') model. Resources using the 'classic' model will be identified by the Azure CSP Migration Assessment Tool and will need to be migrated in-place to the Azure Resource Manager deployment model before continuing.

Finally, the resources can now be migrated to the new subscription. Cross-subscription migrations require resources and all their dependencies to be moved at once. For example, if VM resources are in one resource group, and that VM's disks are stored in a storage account in another resource group, first move all dependent resources to a single resource group. Then migrate the entire resource group at the same time.

The links given in the Resources section below provide detailed step-by-step instructions for each step of the migration process.



RESOURCES

- [Migrating resources from an EA subscription to CSP](#)
- [Migrating resources from a PAYG subscription to CSP](#)
- [Azure CSP Migration Assessment Tool](#)
- [Upgrade resources from Classic to Resource Manager](#)
- [Moving resources between Azure subscriptions](#)
- [ASR Capacity Planning guide for migration to CSP](#)

Migrating Databases

Migrating databases is a crucial step in any migration project. Fears over application downtime or data loss are common, so have a well thought-through and proven approach.

Alongside migrating servers to virtual machines, migrating the application database is perhaps the most critical step in any Azure workload migration. Maintaining integrity and availability of data is critical, and the time taken to synchronize data between old and new systems may determine the duration of any service disruption during migration.

With data migrations, partners are responsible for:

- Understanding compatibility issues between existing databases and new databases in Azure
- Assessing and optimizing databases
- Executing data migration safely.

There are two main options for running a database in Azure: either as a database installed on an Azure VM or using the Azure SQL Database service. In both cases, a range of database technologies is supported, including Microsoft SQL Server, Oracle, PostgreSQL, NoSQL, and more.

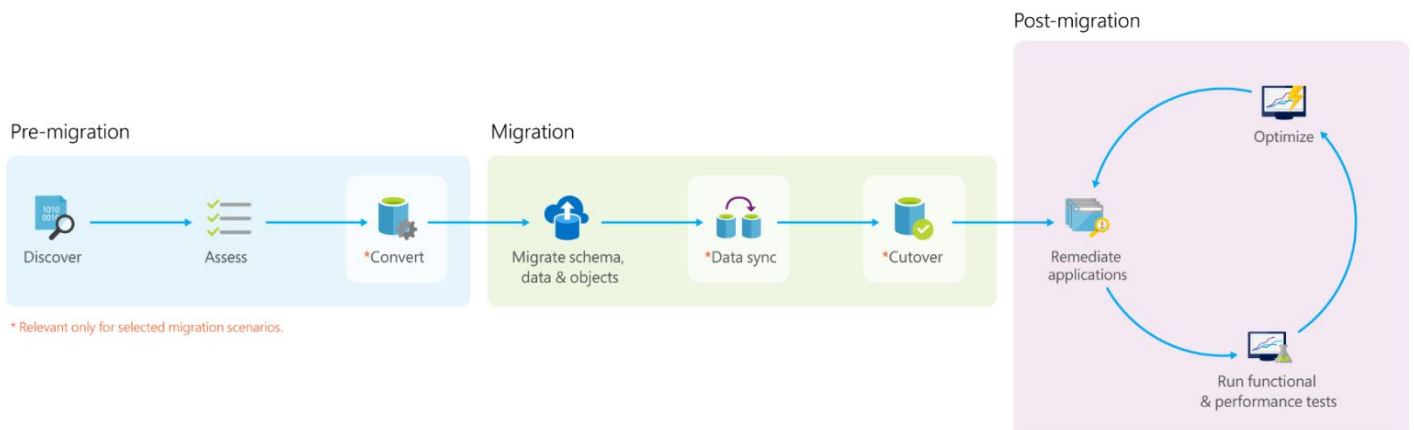
In many cases, even where an infrastructure-only migration strategy is being followed, the database will be migrated directly to Azure SQL Database, rather than SQL-on-iaaS. The motivation is to take advantage of the PaaS benefits of Azure SQL Database such as lower management overhead for underlying infrastructure, quick provisioning and service scaling, and integration with other PaaS services. Azure SQL Database also offers

SQL Database Managed Instances for near 100% compatibility with on-premises SQL Server. This provides all the advantages of a PaaS deployment along with the ease of migration inherent in a SQL Server on IaaS deployment. This makes SQL Database Managed Instances an ideal target for database migrations.

A key goal of migration is to avoid any loss of data. In some cases, a real-time synchronization between old and new databases is possible, enabling migration without service impact. In many cases however, the most pragmatic approach is to accept that the application may be unavailable (or available as read-only) during the migration work window during which the data will be replicated. The duration of this window can be kept to a minimum by performing an initial data transfer prior to migration, followed by an incremental sync containing only subsequent changes during the migration work window. Migrations should be approached with the same rigor and processes as a full software or hardware project – a solid methodology is required for success. When planning a database migration project, we recommend considering the process shown in the following graphic:

KEY SERVICES FOR THIS OFFERING

- [Azure Database Migration Service](#): The Azure Database Migration Service is a fully managed service



designed to enable seamless migrations from multiple database sources to Azure Data platforms with minimal downtime. The service uses the Data Migration Assistant to generate assessment reports that provide recommendations on any changes required prior to performing a migration, and performs all of the associated steps, taking advantage of best practices as determined by Microsoft.

- [Data Migration Assistant](#): This tool can be used to assess a Microsoft SQL Server database in preparation for migration, identify compatibility issues, and to

execute the migration, either to SQL-on-IaaS or to Azure SQL Database.

- **Third-party tools:** A range of third-party tools is available to assist with data migration. See the [Database Migration Guide](#) for a list of tools

RESOURCES

- ➔ [Azure Database Migration Hub](#)
- ➔ [Data Migration Blog](#)
- ➔ [Azure Blog post: Migrating to Azure SQL Database with zero downtime for read-only workloads](#)
- ➔ [Azure SQL Database Managed Instance](#)



Microsoft
Partner
Network

Modernizing Apps

Cloud Migration
and Modernization
aka.ms/practiceplaybooks



Executive Summary

Modernizing applications to use Azure platform services maximizes the value of migrating to the cloud. Which applications can be modernized, and how?

In the previous chapter, we looked in depth at how to migrate workloads to the cloud using a 'lift and shift' approach (rehosting) to Azure's infrastructure services. In this chapter, we will study how to migrate applications to Azure's platform services (rearchitecting).

While a 'lift and shift' migration to Azure infrastructure services can offer significant benefits, it does not take advantage of everything the cloud has to offer. Virtual machines still need to be patched, services cannot scale automatically based on usage, software updates have to be managed, and many of the higher-level services offered by the cloud are not available.

To take full advantage of the cloud requires a migration to Azure platform services. Here, the full benefits of the cloud can be realized: patching and many other maintenance tasks are handled automatically by the platform, streamlined workflows support rapid software updates, services can auto-scale based on usage, and the full power of all Azure services is available.

Earlier in this playbook, we discussed the various motivations behind a cloud migration, and the different ways that migration can deliver value for a business. To recap, the four main cloud benefits are: cost saving,

agility, service quality, and the new scenarios that cloud-based technologies can enable. Only a platform-based architecture can take full advantage of these benefits.

In this chapter, we will discuss migrating applications to Azure platform services. We will start by taking a closer look at the platform services available in Azure, the benefits of a platform-based approach, and we will provide some best practices and design considerations for 'cloud-native' application architectures.

We will then go on to discuss the most commonly used approaches, focusing firstly on Azure App Service, and then on microservices and containers. We will also look at the various technologies available for storing and processing data, including 'big data' technologies used to extract insight and value from large data volumes.

Finally, we will discuss higher-level platform services for machine learning and artificial intelligence. These cloud-only services are increasingly used to light up new application scenarios, delivering competitive advantage beyond cost savings and agility.

Top 4 things to do

Cloud-native designs offer the greatest cloud benefits. Here are the top 4 things to get started.

- ✓ Understand the benefits of platform vs infrastructure approaches to the cloud
- ✓ Build technical expertise in cloud-native design
- ✓ Choose which platform services your practice will use
- ✓ Build experience at migrating existing applications to platform services

Modernizing Applications with Azure

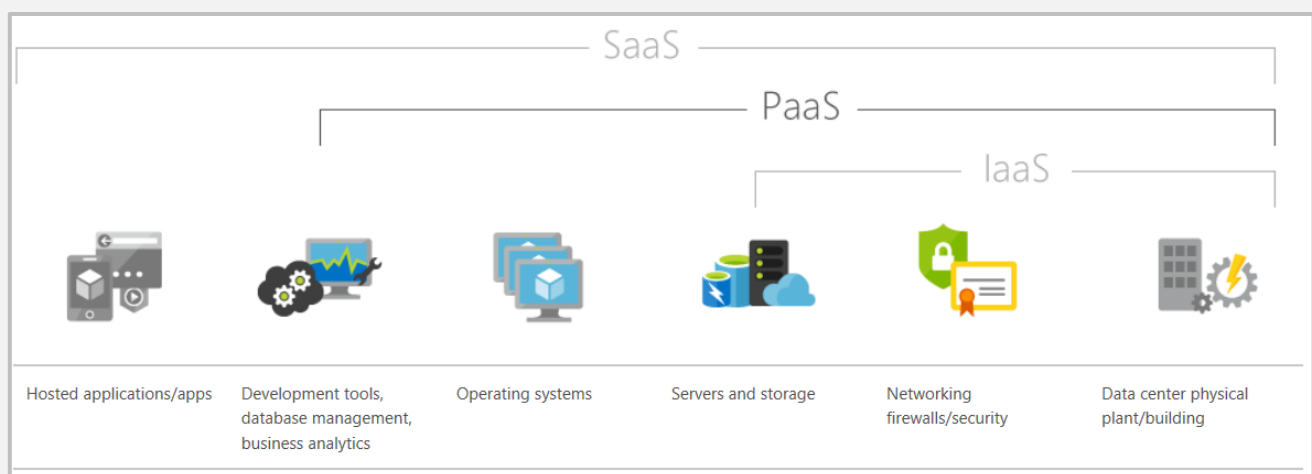
One of the most effective ways to realize the benefits of migrating applications and system architectures to the cloud is to utilize PaaS (Platform as a Service) services.

Platform as a Service (PaaS) is a complete development and deployment environment in the cloud, with resources that enable the delivery of everything from simple cloud-based apps to sophisticated, cloud-enabled enterprise applications. Like IaaS (Infrastructure as a Service), PaaS includes infrastructure (servers, storage, and networking) but also middleware, development tools, business intelligence (BI) services, database management systems, and more. PaaS is designed to support the complete application lifecycle.

BENEFITS OF USING PAAS

Here are the biggest benefits of using PaaS services to modernize and host applications in the cloud:

- **Managed Virtual Machines (VMs):** PaaS hosting is provided by Managed VMs that do not require the traditional maintenance and updates required by traditional IaaS VMs. This provides an abstraction that automates Operating System updates and patches, so the development team only needs to be concerned with the application, data, and deployment rather than infrastructure maintenance.
- **Cut coding time:** PaaS development tools can cut the time it takes to code new apps with pre-coded application components built into the platform, such as workflow, directory services, security features, search, and so on.
- **Add development capabilities without adding staff:** PaaS components can give the development team new capabilities without needing to add staff.
- **Develop for multiple platforms—including mobile—more easily:** Some service providers offer development options for multiple platforms, such as computers, mobile devices, and browsers making cross-platform apps quicker and easier to develop.
- **Use sophisticated tools affordably:** A pay-as-you-go model makes it possible for individuals or organizations to use sophisticated development software and business intelligence and analytics tools that they could not afford to purchase outright.
- **Support geographically distributed development teams:** Because the development environment is accessed over the Internet, development teams can work together on projects even when team members are in remote locations.
- **Efficiently manage the application lifecycle:** PaaS provides all the capabilities needed to support the complete web application lifecycle: building, testing, deploying, managing, and updating within the same integrated environment.



MIGRATING APPLICATIONS TO PAAS

Using Azure platform features is not all-or-nothing. Partners can combine infrastructure and platform services in a single deployment. For example, it is common for an infrastructure migration to run application servers using infrastructure virtual machines, but to use the Azure SQL Database platform service as the data tier, rather than running SQL Server on virtual machines. Many other combinations as possible.

While it is most convenient to build “cloud-native” applications from the start, that is often not possible with many enterprise applications. There is often a significant investment in “legacy” enterprise applications that make it cost prohibitive to start over from a blank slate to benefit from the cloud. For this reason, it’s most feasible with many enterprise systems to refactor the applications to include cloud-native design considerations and leverage PaaS technologies.

To support this, Azure provides a range of platform services, each offering a range of benefits and each requiring a different degree of application modernization. For example, an application might be migrated to Azure App Service, or converted to run in containers, with relatively few changes. At the other extreme, a fully serverless architecture utilizing Azure Functions and higher-level Azure services may require the application to be re-written. Choosing the right approach to application modernization requires an understanding of the suitability, cost and migration complexity of each of the platform approaches available.

KEY CHALLENGES

- Focus on building applications and business logic, instead of managing infrastructure, reliability, scalability, and latency.
- Quickly build powerful web, mobile, and API apps using multiple languages / frameworks (.NET, .NET Core, Java, Ruby, Node.js, etc).
- Reliably deploy, update and scale applications effortlessly across many virtual machines, using either Windows or Linux.
- Globally deliver content to any device with low latency.

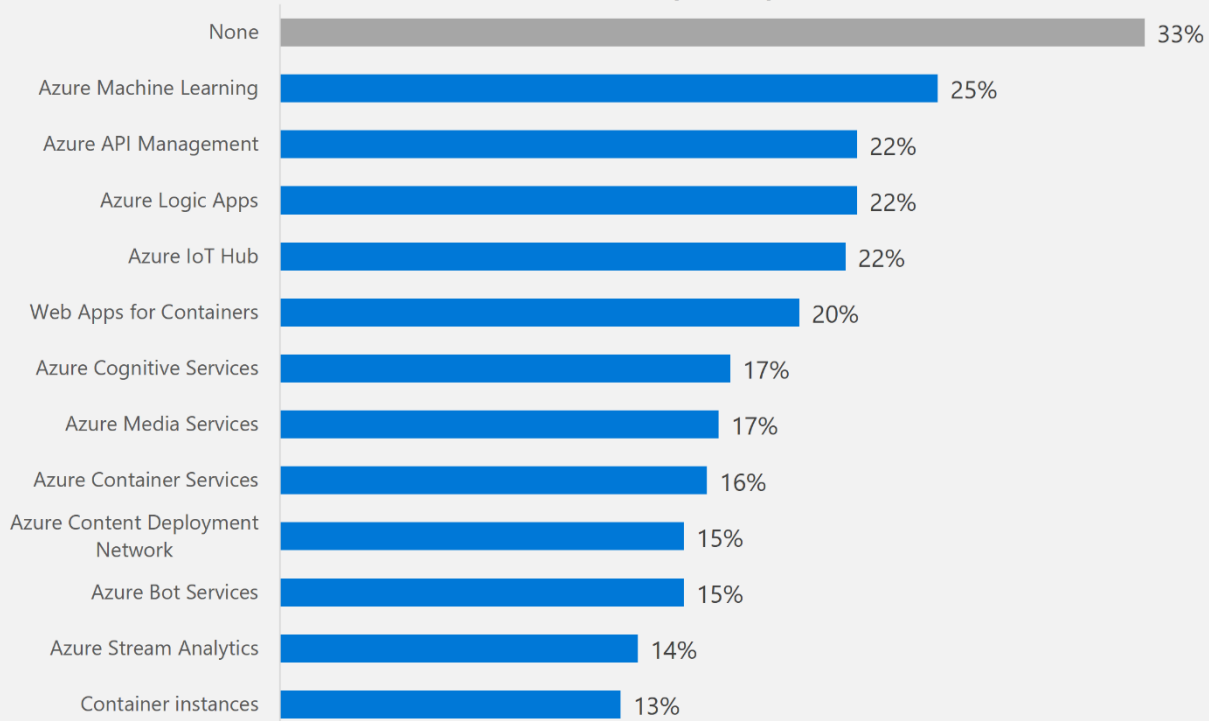
KEY SERVICES FOR THIS OFFERING

- **Virtual Machine Scale Sets:** Apply autoscaling to virtual machines for high availability. Create thousands of identical virtual machines in minutes.
- **Azure App Service:** Quickly create powerful cloud apps using a fully-managed platform.
- **Azure Container Services (AKS):** Simplify the deployment, management, and operations of Kubernetes
- **Service Fabric:** Build and operate always-on, scalable, distributed apps. Simplify microservices development and application lifecycle management.
- **Azure Functions:** Build apps faster with a serverless architecture. Accelerate development with an event-driven, serverless compute experience.

SURVEY DATA

In our survey of Microsoft partners with an Azure migration practice, two-thirds reported using one or more Azure services for application modernization.

Azure Services Used in the Past 12 Months for Modernization (n=227)





Cloud-Native Architecture and Design

Most of the traditional application designs and architectures that are commonplace in on-premises datacenters can run in the cloud without change. However, the cloud brings with it many new capabilities and features. Applications that make use of cloud capabilities are often referred to as “cloud-native” applications.

On the surface, designing applications for the cloud is not very different than designing for on-premises. All the same development tools, language, and frameworks can be used in the cloud. This enables all the familiar tools and existing skillsets of the development team to be used.

However, the cloud also offers a range of additional capabilities, and taking advantage of these requires some design changes. In addition, there are a wide range of cloud services and features available, and a variety of design approaches available. Migration partners are responsible for:

- Choosing the right cloud-native application architecture for the application.
- Incorporating proven best practices into the cloud designs.
- Optimizing implementation by leveraging existing deployment templates for common architectures.

Microsoft has published extensive guidance on designing applications for the cloud. This guidance can be found in the [Azure Architecture Center](#), and provides a wealth of resources and proven cloud architecture best practices,

based on real-world experiences gained from working directly with the largest Azure customers. Using this guidance can accelerate the design process, and ensure that the designs follow proven best practices. Amongst other resources, the Azure Architecture Center includes:

- The [Azure Application Architecture Guide](#), which presents a number of common architecture styles, technology choices, and design principles for Azure applications.
- [Azure reference architectures](#), which demonstrate recommended practices and include deployable solutions which can be used as the basis of deployments.
- [Azure architecture best practices](#) for a wide range of common topics, including API design and implementation, autoscaling, use of background jobs, monitoring, fault handling, and more.
- [Design review checklists](#) for [Availability](#), [Resiliency](#) and [Scalability](#), which can be used to validate and improve the designs, and catch potential problems early to avoid expensive re-work later.



Cloud Design Considerations

Cloud-native applications are designed differently from conventional applications.

There are a few design considerations necessary when building software to be “cloud-native” and utilize PaaS services. These design considerations address many of the key differences to resource allocations and availability with PaaS services and IaaS resources. Here are the key design considerations for building “cloud-native” applications, or migrating existing applications, using PaaS services:

- **Scale Out over Scale Up:** One of the traditional methods of scaling to meet increased load in on-premises environments is to Scale Up; simply add more CPU power, Memory capacity, or storage capacity to the server. This is a common practice on-premises with both Single Server and Multi-Server hosting solutions. In the cloud, server resources are more of a readily available commodity. In the cloud, a new server VM can be provisioned when needed. For this reason, it is much easier to add more servers when extra capacity is needed. An additional benefit of Scaling Out (adding more servers / VMs) is an increase in resiliency to the application deployment. If a 1 server out of 10 goes down, it affects the overall performance of the application and SLA to the company’s clients and users much less than if 1 server out of 2 goes down. Scaling Out and spreading load across server instances has many benefits, including higher availability, higher resilience against failure, as well as the ability to tweak server resources to optimize cost when comparing pricing tiers.
- **Graceful Fault Handling:** Every software system has failures. When distributing load across multiple server instances, or integrating a distributed / microservices architecture, an application needs to be coded to gracefully handle all error when possible. These errors can even include transient failures, such as not being able to connect to a database or service, isolated network outage on a single VM, or any other sort of error that occurs at random times for a short period of time and then goes away. When a call to an external service fails, the application can be coded with Retry Logic to re-initiate making the service call 1 or more times before throwing an exception. Another possible implementation is to cache a user’s request using some type of message queue, so the user can continue with their work while the system handles the error and performs the requested work in the background.

- **Multi-Zone and Multi-Region Deployment:** One of the aspects to hosting application in on-premises environments that can easily become cost prohibitive is the ability to deploy to multiple datacenters for redundancy and increased resiliency. In the cloud, Microsoft Azure is made up of many Azure Regions spread across continents and geographies that can be utilized for any cloud application deployment. There are also multiple Availability Zones (currently in Preview) within each Azure Region that can be utilized to enable application instances to be deployed to specific datacenters within the Region. Application deployments that utilize Zones and Regions have the capacity to be globally resilient against Regional or Datacenter failures. This extra resiliency can be combined with Scaling Out to host applications with a much greater level of resiliency and availability than is possible in most on-premises datacenter environments.
- **Do Not Write to Local File System:** When using PaaS services like Azure App Service to host an application, it's important to not write anything persistent to the local file system. The reason for this is that multiple instances of the application running in the service do not share the same local file system as they are hosted on different VMs. Instead of writing to the local file system, any files and/or data that needs to be accessed across application instances should be written to a shared storage service like Azure Blob Storage, Azure Redis Cache, or another database service.
- **Stateless Design:** When designing applications to Scale Out, instead of Scale Up, the design to share state across instances can prove to be difficult depending on the overall application architecture. Modern application architecture is to design applications to remain Stateless. This has become the modern standard in web application and Rest API design. It's common to use Cookies with an HTTP-based application to remain Stateless, but another option is to use a cache service like Azure Redis Cache to share state across instances for applications that still require state.
- **Serverless:** Serverless computing, such as Azure Functions, takes the benefits of PaaS even further by adding an additional abstraction layer that enables for an individual function of code to be deployed and hosted without requiring a full application to be maintained. This larger abstraction of Azure Functions is integrated with Input and Output Bindings that provide built-in capabilities to integrate with other Azure services, as well as third-party services and applications. Using Serverless compute decreases the overall development and maintenance costs that enable faster time to delivery and easier production support.
- **PaaS Services:** Where a specific service is available for a task, using that service is usually a better choice than building a custom service using lower-level platform components. These services have already been designed to deliver the availability, performance, security and scale required by Microsoft's most demanding customers—saving significant work and providing a service level that would be difficult and expensive to match. Using services built specifically for the cloud enhances the development team's capacity to deliver value, in addition to providing an abstraction that removes many of the monotonous and time-consuming aspects of Server and Virtual Machine maintenance.

Cloud Design Patterns

Use proven design patterns to accelerate and improve cloud-native application designs.

Among the many application design challenges developers will face, many have already been solved and documented as cloud 'design patterns'. These patterns provide proven out-of-the-box solutions to challenges involved in designing applications to make best use of the cloud. Using proven, established patterns streamlines development by saving the development team from having to design solutions for these common scenarios on their own.

Here are some of the most common cloud design patterns used to build "cloud-native" applications:

- **Retry Pattern:** Enable the application to handle transient failures when it tries to connect to a service or network resource, by transparently retrying a failed operation. This can improve the stability of the application.
- **Circuit Breaker Pattern:** Handle faults that might take a variable amount of time to recover from, when connecting to a remote service or resource. This can improve the stability and resiliency of an application.
- **Competing Consumers Pattern:** Enable multiple concurrent consumers to process messages received on the same messaging channel. This enables a system to process multiple messages concurrently to optimize throughput, to improve scalability and availability, and to balance the workload.
- **Priority Queue Pattern:** Prioritize requests sent to services so that requests with a higher priority are received and processed more quickly than those with a lower priority. This pattern is useful in applications that offer different service level guarantees to individual clients.
- **Queue-Based Load Leveling Pattern:** Using a Queue that acts as a buffer between a task and a service it invokes in order to smooth intermittent heavy loads that can cause the service to fail or the

task to time out. This can help to minimize the impact of peaks in demand on availability and responsiveness for both the task and the service.

- **Cache-Aside Pattern:** Load data on demand into a cache from a data store. This can improve performance and help to maintain consistency between data held in the cache and data in the underlying data store.
- **Throttling Pattern:** Control the consumption of resources used by an instance of an application, an individual tenant, or an entire service. This can allow the system to continue to function and meet service level agreements, even when an increase in demand places an extreme load on resources.
- **Command and Query Responsibility Segregation (CQRS):** This design pattern segregates operations that read data from operations that update data by using separate interfaces. This can help maximize performance, scalability, and security. It also supports the evolution of the software system over time through higher flexibility and prevents update commands from causing merge conflicts at the domain level.

The Microsoft Architecture Center includes an additional list of [cloud design patterns](#), as well as a list of [cloud "anti-patterns"](#)—common mistakes that can negatively impact the performance of cloud-native applications.



Azure DevTest Labs

DevTest is probably the only workload that doesn't have corporate issues slowing down its migration. Aside from the security and backup of a company's source code, there's no reason for IT managers to balk nor are there regulatory or compliance rules that are stopping the march of DevTest to the cloud.

Microsoft [Azure DevTest Labs](#) provides an end-to-end experience to address the testing environment needs of modern development teams. Before looking at the benefits of DevTest labs, let's first review the challenges of building and maintaining DevTest environments.

DevTest Challenges

Skills and Effort Required to Build Out DevTest Environments

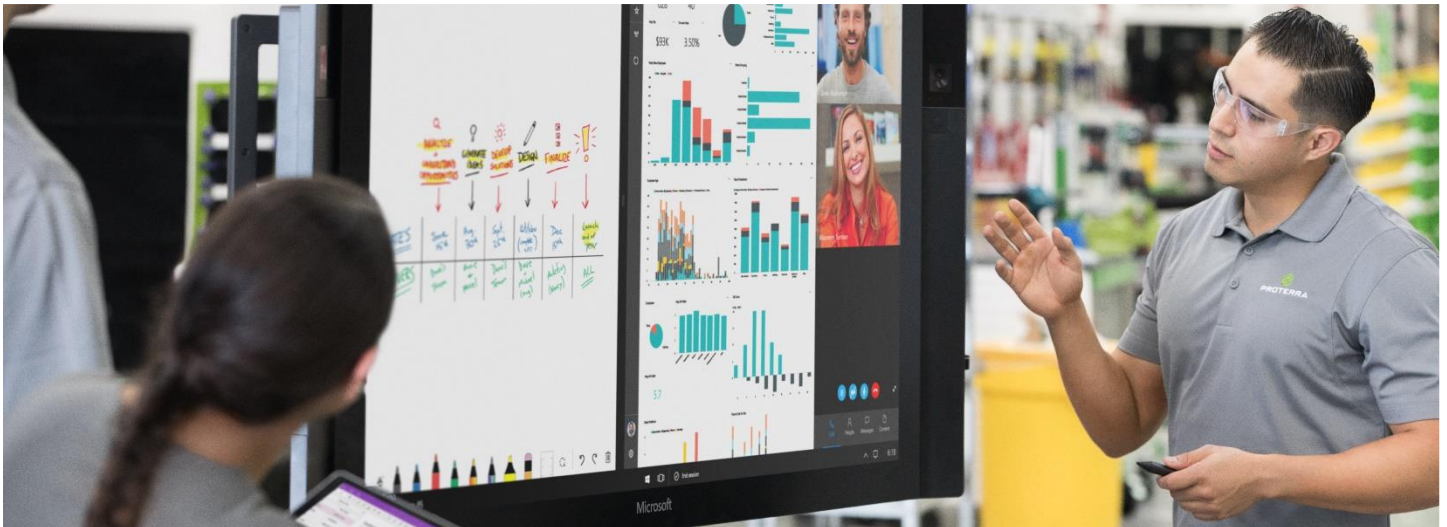
Many development shops build an infrastructure from scratch as their companies haven't moved any workloads to the cloud yet, and that can be overwhelming for developers. Even though they are leveraging cloud tools, solving mundane issues such as designing Virtual Networks, IP address spaces, subnetting, DNS for name resolution, Virtual Machines templates, file shares, storage connections to DevOps package repositories will all need to be designed and implemented. This requires skills in all areas of IT, many of which are outside of a developer's expertise.

Each environment that needs to be built will require a design to be created and implemented. In addition, providing for both authentication and connectivity for in-house developers and external contractors must be addressed. DevTest in today's world is all about speed, and without the right skills or environment isolation, things can quickly grind to a crawl.

Augmenting Previously Deployed Continuous Integration and DevOps Tools

Many development and testing teams have invested in continuous integration and DevOps tools that are heavily utilized for their operations. When migrating DevTest environments from on premise to the cloud, these investments must be leveraged and augmented to ensure their investment is extended to the cloud.





Customizing the Environment to Meet the Needs of Developers and Testers

Developers and testers need their environments to be exactly right to ensure efficiency and to minimize bugs. Development teams have become very detail-oriented to ensure that each environment being developed can be replicated quickly and can be the same every time, no matter what. Often, this means building Virtual Machine (VM) templates and scripts to ensure configurations with DevOps tools. Cloud solutions must allow for this type of very precise customization and ease of use.

Providing a “Ready to Test” Solution

“Ready to test” is all about efficiency. Testing teams need to focus on their role in the development cycle, which is ensuring the highest quality of the application. Oftentimes, testers must build their environment by hand, which takes time away from testing the application and can also introduce human error. When testers show up for work, they should login and start their test, nothing else. The bits they are testing and all their tools should already be installed on the VMs they are working with to ensure they catch every issue, instead of installing software over and over.

Ensuring Access Control and Isolation from Production Environments

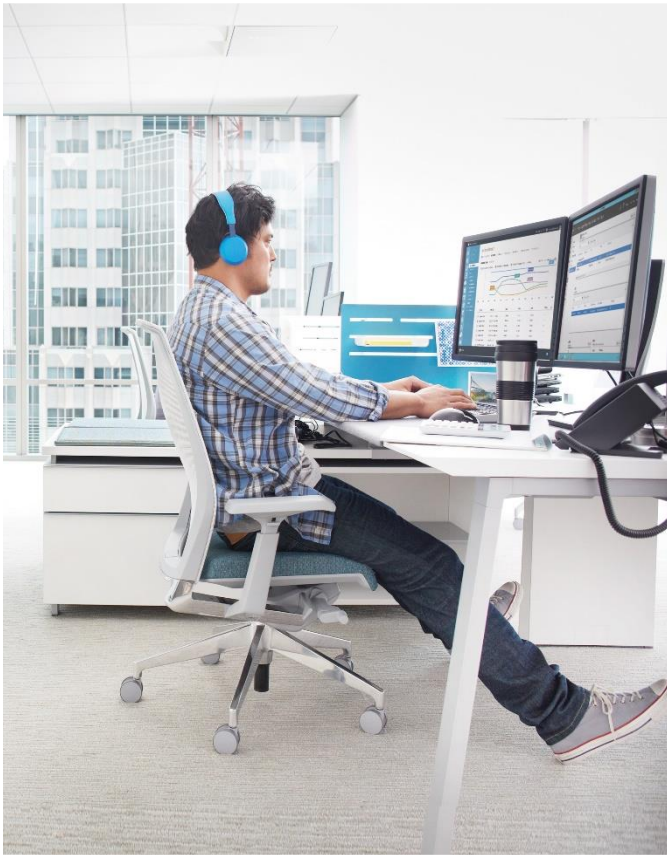
A development and testing environment is meant just for that – development and testing – and those who use these environments should only have access to leverage the cloud for such activities. It’s important to provide them

with the access they need to be successful in their role without risking the safety of production systems.

Accidental shutdowns of systems should never happen, and sufficient isolation between DevTest and production should be in place to ensure proper change control practices are followed. Moving to the cloud doesn’t mean pioneering to the Wild West. Governance must be in place to ensure that all teams are playing their roles properly while utilizing the resource.

Leveraging the Cost Efficiencies of the Cloud via “Pay-As-You-Go” While Capping Costs and Staying on Budget

One of the most interesting aspects of moving to the cloud for DevTest is the idea that costs are only incurred while developing and testing. Some companies can dip as low as 0% utilization of on premise DevTest environments at times during the year. Leveraging the flexibility of the pay-as-you-go model is one of the most compelling reasons to move to the cloud. If the users of the resources aren’t cognizant of how many resources they are using or when those resources are being used, the scalability of the public cloud can turn into some very large bills. Just think of an electric bill if everything in a home was left on 24 hours a day, seven days a week; the bill would be astronomical. IT projects are known for always being over budget, so there needs to be a way to ensure that the resources required don’t put a project over budget.



Azure DevTest Labs

To solve these problems and let developers and testers do what they do best, Microsoft has developed a turnkey solution for DevTest in the cloud: **Azure DevTest Labs**. This service addresses the above problem by means of the following features.

Quickly be “Ready to Test”

DevTest Labs provides multiple options to help developers and testers to get their environments ready quickly. It offers three different types of VM bases that developers and testers can use to create the Dev/Test environments:

- **Marketplace images:** VM images directly from Azure Marketplace.
- **Custom images:** leveraging a VHD file built by the customer.
- **Formulas:** a reusable base where VM creation settings (such as VM image, VM sizes, virtual network, etc.) are pre-defined, so that environments can be created without requiring any more input.

Reusable artifacts in the DevTest Labs allow users to run VM extensions and install tools, deploy applications or execute custom actions on demand once a lab VM is created.

Worry-free self-service

DevTest Labs enables a team self-service model where developers and testers go to the Azure portal directly to create environments, instead of going through a complicated “request” process. In this case the DevTest Lab is leveraging Azure Role-Based Control (RBAC), which empowers teams to use the portal to build VMs as needed on their own.

Lab policies makes it easier to control costs by allowing lab owners to set boundaries and standard operating procedures around what is spend on the project. By using policies Development and Testing teams can ensure they won't go over and have an unexpected bill at the end.

Using Lab policies and Azure Role-Based Access Control (RBAC), DevTest Labs enables a sandbox environment for developers and testers to provision their own environments without unexpected accidents that can introduce a big bill.

Create once, use everywhere

DevTest Labs resources (labs, custom images, formulas, artifacts, etc.) are reusable across labs, so they do not need to be re-created again from scratch.

DevTest Labs fully supports Azure Resource Manager (ARM) and follows the best practice of using resource groups. Create multiple labs with the same settings/policies by deploying the same ARM template.

ARM templates are fully supported to deploy labs and resources in a lab. Reusable custom images and formulas can be created from an existing VM, and artifacts loaded from VSTS Git or GitHub repositories can be used cross different labs.

Integrates with existing toolchain

Azure DevTest Labs provides pre-made plug-ins, command-line tool and APIs for integrating Dev/Test environments from labs to the release pipeline.

In addition to APIs and command line tools, [Azure DevTest Labs Tasks](#) are available in Visual Studio Marketplace to better support the release pipeline.

Migrating Applications to Azure App Service

Azure App Service is a powerful and flexible platform for hosting web applications in Azure. Migrating applications to Azure App Service is often the simplest and quickest way to take advantages of the benefits of PaaS.

[Azure App Service](#) is a fully-managed platform to run and scale both Internet-facing and Intranet web applications and services, on both Windows and Linux. It supports a wide range of development languages and offers a high level of developer productivity with features such as CI/CD, easy and safe application updates, and integration with Azure Repos, BitBucket, Docker Hub and GitHub. Azure App Service also supports auto-scaling of the infrastructure supporting the app, enabling significant cost savings.

Review the following considerations before migrating applications to Azure App Service:

- **Port Bindings:** Azure App Service support port 80 for http and port 443 for HTTPS traffic. If there are sites using any other port after migration to Azure App Service, do remember that these are the only ports that will be used.
- **Usage of assemblies in the GAC (Global Assembly Cache):** This is not supported. Consider bin placing the assemblies in the local bin.
- **IIS5 Compatibility Mode:** IIS5 Compatibility Mode is not supported. In Azure App Service each Web App and all the applications under it run in the same worker process with a specific set of application pool settings.
- **IIS7+ Schema Compliance:** One or more elements and/or attributes are being used which are not defined in Azure App Service IIS schema. Consider using XDT transforms.
- **Single Application Pool Per Site:** In Azure App Service each Web App and all the applications under it run in the same application pool. In cases where there are applications in different application pools in IIS, consider establishing a single application pool with common settings or creating a separate Web App for each application.
- **COM and COM+ components:** Azure App Service does not allow the registration of COM components on the platform. If the site(s) or application(s) make use of any COM components, these would need to be rewritten in managed code and deployed with the site or application.
- **ISAPI Extensions:** Azure App Service can support the use of ISAPI Extensions; however, the DLL(s) need to be deployed with the site and registered via the web.config.

Once the above limitations have been taken into consideration, migrate the applications. The easiest way to migrate is by using the [Azure App Service Migration Assistant](#), which can assist with both migration assessment and migration execution.

The App Service Migration Assistant can provide initial migration assessment for Internet-facing applications simply by entering the site URL into the tool. The tool will then check various details, such as the language and hosting platform, to see if they are compatible with the Azure App Service.

For a deeper assessment, the App Service Migration Assistant tool is downloaded and run on the application web server. This provides an in-depth assessment of the application configuration. For modern applications



without migration blockers, the tool can also be used to migrate the application itself to the Azure Web Apps service.

The App Service Migration Assistant tool can be used with both public and Intranet sites. A current limitation is that only ASP.NET applications are supported. The App Service team intends to support more platforms in future, so check the web site for the current support matrix.

With the decision made to migrate, the following areas need to be considered for migrating applications to Azure App Service. Also review the considerations for migrating databases to Azure listed earlier in this playbook.

- **On-premises integration:** When applications are communicating with other applications that will not be migrated to Azure, consider how the communication will happen when the application moves to cloud. One solution is to enable the other application to communicate over the internet using REST. This may require changes in both the applications, not to mention the additional risk of exposing the server onto internet. Another approach would be to establish a secure connection to the on-premises server from Azure App Service, where the application is hosted. This can be done in any of the following ways depending on the requirement: Deploying apps in an [App Service Environment](#) using an Isolated App Service Plan; enabling [virtual network integration](#) with an Azure VNet; establishing a Site to Site VPN between this Azure virtual network and on-premises, and then enable routes between your App Service and the on-premises server; and establishing hybrid connections.
- **Authentication:** When on-premises, using no authentication or Windows authentication may be acceptable as there was mutual trust with AD. When migrating to Azure, enable authentication with Azure Active Directory. This means [modifying some of the configuration to be able to authenticate users via Azure AD](#).
- **Session State:** In an ideal case, make your application stateless in order to scale/switch at will. In case it is not possible, have the session state configured to be persisted in [Azure Redis Cache](#).
- **File Persistence:** Usually, websites might require uploading files that need to be persisted. On Azure App Service, it is recommended to persist any files outside of the App Service into something like a blob store. Modify the application to now use either the Azure Storage SDK or the REST APIs for saving and accessing files.
- **App Settings and Connection Strings:** There will be App Settings and Connection Strings that will change based on environment, whereas some will stay same. For the ones that change based on environment, also define them on the portal or deployment template so that they can be overridden for different deployment slots.
- **Logging:** If the logging framework is logging to files saved locally, update them to either log in to Azure Diagnostics or to a centralized blob store. Also include Azure App Insights to get deeper insights into how the application is performing.
- **Certificates:** Certificates are not migrated directly. They must be explicitly uploaded to be able to work on Azure, as detailed [in this Bind SSL Certificate documentation](#). Purchase certificates directly from Azure, as detailed [in this buy SSL cert documentation](#).
- **Custom Domains:** Custom domains can be associated to Azure Web Apps via a CNAME record change. Update App Service to validate the DNS. [Details are available in this map custom domain documentation](#).
- **Email:** Sending Emails requires an SMTP server. App Service does not provide the same and there is no way to configure it within App Service. While an SMTP server can be set up to send emails on Azure IaaS VMs, we do put in restrictions. We recommend using relay services to send email, such as [Office 365](#).
- **LDAP Queries:** If you are building internal applications that are querying the LDAP store, such as AD, those may not work on Azure App Service. Specifically, in the case of Active Directory, move AD to Azure AD and then use the graph APIs to make the necessary queries to Azure. For this, register the application with Azure AD to permit querying Directory Objects. A [complete list of graph APIs is here](#).



Microservices and Containers

The microservices architecture has become an important part of building distributed mission-critical software systems.

Traditional application architectures build applications in large, monolithic components. These large components are deployed as a single unit, making it hard to maintain strict separation between internal components. This results in long integration, test and release cycles, which slow development, reduce agility and increase costs.

The primary design principle of a microservices architecture is to design an entire software system to be built using smaller software components, called microservices. Each microservice performs a single function of the overall system that can be developed, deployed, and scaled independently. This independence enables accelerated development by enabling individual component teams to work more independently, thereby avoiding long integration, test and release cycles. For more information, see the article [Why a microservices approach to building applications?](#)

Containerization is an approach to deployment and application management that combines an application with its dependencies and configurations (via manifest files) into a container image. Due to their highly efficient resource consumption, containers are an ideal platform for the development of microservices. Each microservice is built into a dedicated container image, and can then be more easily deployed, scaled, and managed as a single package. Using microservices and containers reduces the

effort required to manage the deployment and scalability of an application.

Azure supports several services to support microservices and containers:

- [Azure Container Services \(AKS\)](#): Kubernetes is the leading platform for orchestrating container deployments. The Azure Container Service for Kubernetes (AKS) simplify the deployment, management, and operation of Kubernetes.
- [Azure Container Instances \(ACI\)](#) provides a fully-managed service to run containers, without any need to deploy or manage the underlying infrastructure. Easily run containers on Azure with a single command, and with per-second billing.
- [Azure Container Registry](#) is a fully managed Docker Registry service. Container registries can be used to store and manage container images across all types of deployments.
- [Azure Service Fabric](#) is a platform for deploying and operating always-on, scalable, distributed, microservice-based applications. Service Fabric simplifies microservices development and application lifecycle management.
- [Web App for Containers](#) makes it easy to deploy and run containerized web apps that scale and provides a fully managed platform for infrastructure maintenance.

What is Docker?

Docker has become the de-facto standard for containers.

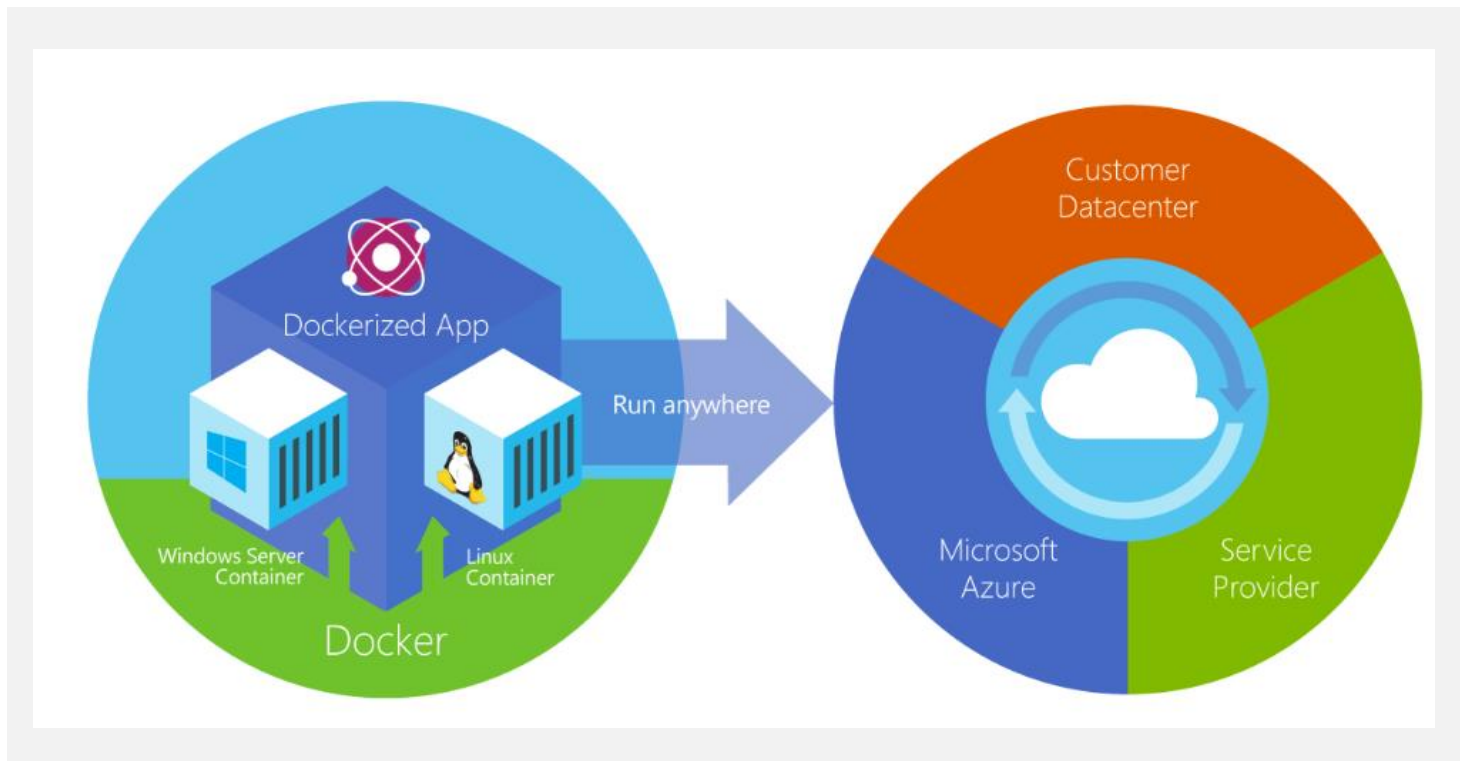
Docker is an open-source project for automating the deployment of applications as portable, self-sufficient containers that can run on the cloud or on-premises. Docker is also a company that promotes and evolves this technology. Docker works in collaboration with cloud, Linux, and Windows vendors, including Microsoft.

Docker image containers run natively on Linux and Windows. Windows images run only on Windows hosts and Linux images run only on Linux hosts. The host is a server or a virtual machine.

Develop on Windows, Linux, or macOS. The development computer runs a Docker host where Docker images are deployed, including the app and its dependencies. On Linux or macOS, use a Docker host that is Linux-based and can create images only for Linux containers (on macOS you can edit code or run the Docker CLI, but as of this writing, containers do not run directly on macOS). On Windows, create images for either Linux or Windows containers.

For further reading, see:

- [Introduction to Containers and Docker](#)
- [.NET Microservices: Architecture for Containerized .NET Applications](#)



Modern Data Platform

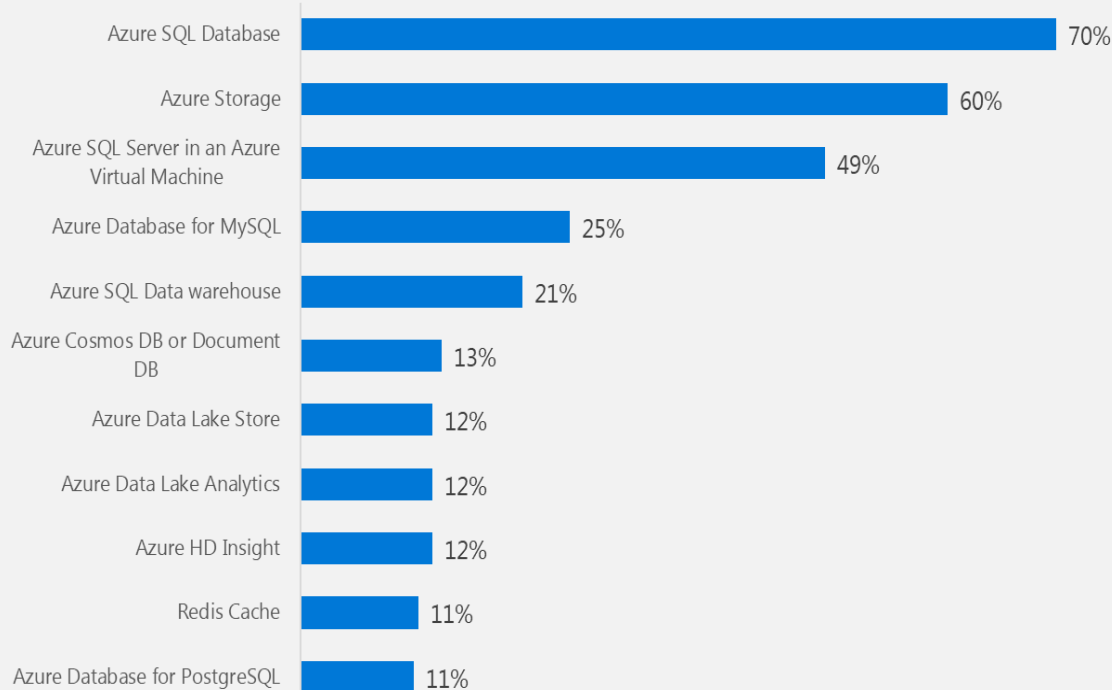
The cloud has driven rapid changes in how applications handle data. Whatever the data needs, Azure offers a service to suit.

Modern data platforms are designed to ingest and process petabytes of data for a variety of purposes. These systems are capable of ingesting and storing data in nearly any format and at any scale. Data may be structured like a relational database or unstructured such as a web log. These modern data platforms enable a variety of applications types such as large scale cognitive and AI applications and high throughput IoT data ingestion.

There are several options for running a modern data platform in Azure. There are big data stores such as Azure Data Lake Store and Azure Storage and there are compute technologies such as HDInsight and Data Lake Analytics. There are also more focused database options such as Azure SQL Data Warehouse and Cosmos DB that can also operate on large amounts of data.

SURVEY DATA

In our survey of Microsoft partners with an Azure migration practice, those using data-related services reported Azure SQL Database as the most-used service (70%), followed by Azure Storage (60%) and Microsoft SQL Server running in a virtual machine (49%).



Microsoft partners add value by helping customers choose and implement their modern data platform solution. This requires an understanding of the data platform services available and the ability to choose the most suitable services for the given application. Once chosen, further work will be needed to choose the correct configuration, for example assessing and optimizing the cluster and storage size.

Transferring large data volumes to the cloud can be another challenge. Online transfers, either via the Internet or an ExpressRoute connection, are the default approach, enabled by tools such as AzCopy. For very large volumes, use disks or the Azure Data Box appliance to ship data to Microsoft data centers.

A common scenario is data analytics, perhaps from incoming telemetry or using existing data stores. Azure offers several services to help, such as Azure Analysis Services for data modelling and analytics to Azure Event Hubs and Stream Analytics for real-time processing of incoming data streams.

In many on-premises environments, it is common to configure a single monolith Hadoop cluster to handle all types of data ingestion and processing. The architecture we use in the cloud allows us to store all of our data in the same location such as an Azure Data Lake Store and then spin up any number of compute clusters to operate on that data. This separation of compute and storage buys us two things:

- **Decoupled compute and storage scale.** We can scale compute and storage independently of each other. If we need more processing power, we can simply increase the size of our HDInsight cluster.
- **Optimized spending.** We can now have multiple, purpose-built compute clusters allowing us to optimize cluster size and runtime based on a single workload rather than all workloads in aggregate.

A summary of the Modern Data Platform services offered by Microsoft is given below:

- [Azure HDInsight](#): Azure HDInsight is a fully-managed cloud service that makes it easy, fast, and cost-effective to process massive amounts of data. Use popular open-source frameworks such as Hadoop, Spark, Hive, LLAP, Kafka, Storm, R & more. Azure

HDInsight enables a broad range of scenarios such as ETL, Data Warehousing, Machine Learning, IoT and more

- [Azure Data Lake Analytics](#): Develop and run massively parallel data transformation and processing programs in U-SQL, R, Python, and .NET over petabytes of data with zero infrastructure.
- [Azure Data Lake Store](#): Store unstructured, semi-structured and structured data with no limits on size or throughput. Secure, massively scalable, and built to the open HDFS standard to run massively-parallel analytics.
- [Azure Cosmos DB](#): A low latency, horizontally scalable and globally distributed multi-model database. Support for many APIs such as SQL, JavaScript, Gremlin (Graph), MongoDB, Cassandra and Azure Table storage. Cosmos DB was formerly known as DocumentDB.
- [Azure SQL Database](#): A fully-managed database service for structured, relational data.
- [Azure SQL Data Warehouse](#): A massively parallel processing server with independent compute and storage scalability to integrate with big data stores, and create a hub for data marts and cubes—to drive highly tailored, enterprise-grade performance, while leveraging existing SQL and BI skills.
- [Azure Data Factory](#): Fully managed ETL service in the cloud. Connect all data sources and orchestrate the data workflows wherever the data lives.
- [Azure Storage](#): Offering fast and scalable blob, table and queue storage, and shared file storage for Azure virtual machines
- [Azure Storage Import/Export Service](#) and [Azure Data Box](#): Use physical disks or a custom-built appliance to ship large data volumes to Azure
- [Azure Analysis Services](#): A fully-managed service for combining data from multiple sources into a single semantic model, enabling reporting through client tools such as Power BI and Excel.
- [Azure Stream Analytics](#): A managed event-processing engine for real-time analysis on streaming data.
- [Azure Event Hubs](#): A hyper-scale telemetry ingestion service supporting real-time and batch processing.

For more information on the Microsoft Data Platform, see the Microsoft partner [Data Platform & Analytics Playbook](#).

Cognitive Services and AI

Microsoft's AI and cognitive services can process data in new and exciting ways, extracting patterns and meaning that would previously have required human input. These services enable a wealth of new scenarios.

One of the business benefits of cloud migration is the ability to enable new scenarios that would be impossible (or at least, prohibitively expensive) previously. This concept is at the heart of the digital transformation paradigm shift, and Microsoft's cognitive and AI services are at the heart of this capability.

Partners can differentiate their offerings by helping customers migrate to the cloud and maximize the new business opportunities offered by digital transformation. Use the unique data generated by operating their services to enhance their business in new and unexpected ways, for example by interpreting business and customer data in real-time and large scale, including text, docs, images, video, and voice.

The [Microsoft AI Platform](#) provides a comprehensive set of flexible AI services and enterprise-grade AI infrastructure that runs AI workloads anywhere at scale. The Microsoft AI Platform accelerates the development of AI solutions with high-level services. Modern AI tools designed for developers and data scientists help create AI solutions with maximum productivity.

Machine Learning enables computers to learn from data and experiences, and apply that learning to sense, process and act on information in future. There are many applications for this technology. A common use case is predictive analytics—using historical data to predict future behavior, for example using telemetry data from machines in a factory to predict forthcoming hardware

failure and enable pro-active maintenance, thereby reducing downtime.

Microsoft's suite of Cognitive Services enable insight from human interactions, from facial expressions, to speech, and natural language processing. For example, extracting the positive or negative sentiment from comments on a review forum or twitter feed can enable a business to respond more rapidly to a service quality issue, thereby improving their overall customer experience.

Enhancing existing applications to take advantage of these services is one of the ways a Managed Services Provider can provide additional value to customers.

Core services of Microsoft's AI platform include:

- [Microsoft Cognitive Services](#): Use AI to solve business problems. Infuse apps, websites, and bots with intelligent algorithms to see, hear, speak, and understand natural methods of communication.
- [Azure Machine Learning](#): Model AI algorithms and experiment with ease. Customize based on the requirements.
- [Azure Bot Service](#): Accelerate development for conversational AI. Integrate seamlessly with Cortana, Office 365, Slack, Facebook Messenger, and more.

For further information on how to build a partner practice taking advantage of AI, see the [AI Practice Development Playbook](#).



Microsoft
Partner
Network

Optimize & Manage

Cloud Migration and Modernization

aka.ms/practiceplaybooks



Executive Summary

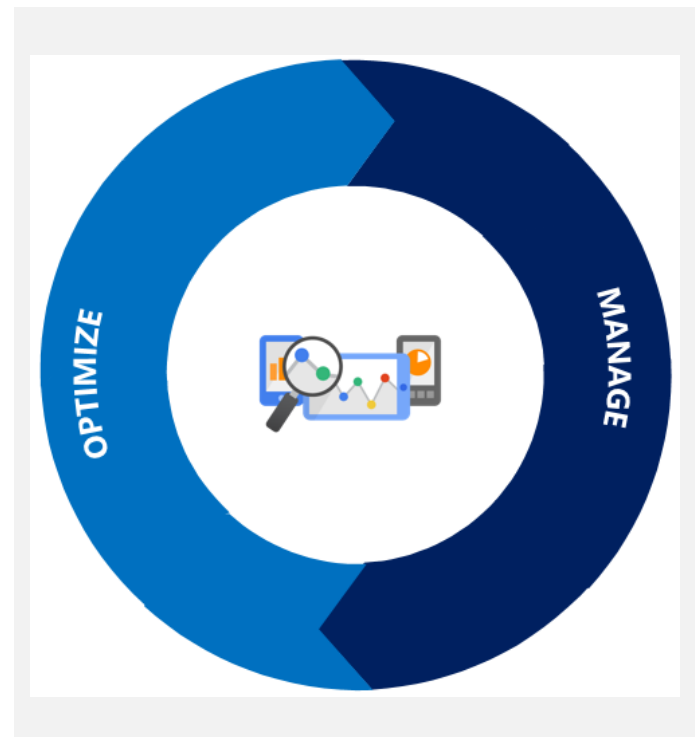
After a successful migration, the workload will be handed off to the managed services team. Here, a whole new set of services such as monitoring, patching, cost optimization and support offer additional revenue and differentiation opportunities.

The [Azure Managed Services Playbook for CSP Partners](#) offers detailed guidance on the value-added services which MSPs can provide, including infrastructure management, backup and disaster recovery, identity management, monitoring and security.

In this section, we will focus first on cost optimization. Azure offers a wide range of possibilities for reducing running costs and creating cost savings, which we'll discuss in depth. Taking advantage of these is key to maintaining a competitive edge. Azure consumption optimization is commonly used as a selling point in negotiating a migration project as a way of demonstrating how to increase the return on investment from the cloud. It can also be a great way of identifying additional opportunity for services.

Next, we'll review some best practices for managing services in Azure. We'll consider subscription management, resource organization, and how to protect Azure resources against unauthorized or accidental changes. We'll also look at the security and compliance tools and guidance that are available from Microsoft.

Last, we'll discuss how to automate the deployment and management of Azure services using Infrastructure as



Code. Consider both the advantages of this approach, and the variety of technologies available in Azure to deliver, including how to develop custom tools.

Top 4 things to do

Here are 4 top tips to increase efficiency and differentiate your practice.

- ✓ Study the Managed Services Playbook
- ✓ Learn how to understand, forecast and optimize costs
- ✓ Use best practices for Azure resource management
- ✓ Automate using Infrastructure as Code

Cost Optimization

Cost optimization is the key to a successful managed services business.

With the pay-as-you-go nature of the cloud there are many ways to overspend, even when being careful. A common fear that customers have when it comes to adopting the cloud is the fear of runaway spending. This is where partners play a critical role, both prior to and after a migration to Azure—by helping customers understand, manage, and optimize their spend.

It is always a good idea to design and deploy Azure based systems that are designed for optimization from the start. Then, by analyzing application performance and cloud spend, and by taking advantage of new Azure features, to help customers optimize spend over time.

Additionally, Microsoft partners can help clients make sense of their Azure bill and attribute the spend to different projects, departments, teams, applications and cost centers. If done correctly, this function can add a lot of value to both a partners' practice and the relationship with clients. Spending money on an Azure service that is not needed will cause customers to rethink their strategy and can put the account at risk over time. There are critical areas of expertise both from a reporting and Azure

feature set that customers will expect the partner to provide or include as a part of their migration to Azure.

KEY CUSTOMER CHALLENGES

- Understanding current cloud spend and forecasting future spend
- Gaining the most value from cloud spend
- Guarding against unexpected costs
- Optimizing deployments for cloud efficiency

KEY SERVICES FOR THIS OFFERING

- **Azure Cost Management** : Microsoft's tool to help customers and partners gain full visibility and control over cloud spend, optimize cloud efficiency, and maximize the potential of the cloud
- **Pre-Purchasing Azure**: Helping cloud customers save by using upfront commitments to Azure
- **Auto Shut Down**: Building a strategy for running VMs only when it services are needed
- **Optimized Architecture**: Deploying the best architecture to Azure to ensure performance and cost optimization



Azure Cost Management

Effective cost management starts with understanding the costs, at a granular level.

As enterprises accelerate cloud adoption, they are finding that it's getting more difficult to manage cloud spend across the organization. In a recent survey, managing cloud spend was listed as a challenge by 76% of respondents, second only to security (source: *RightScale 2018 State of the Cloud Report*).

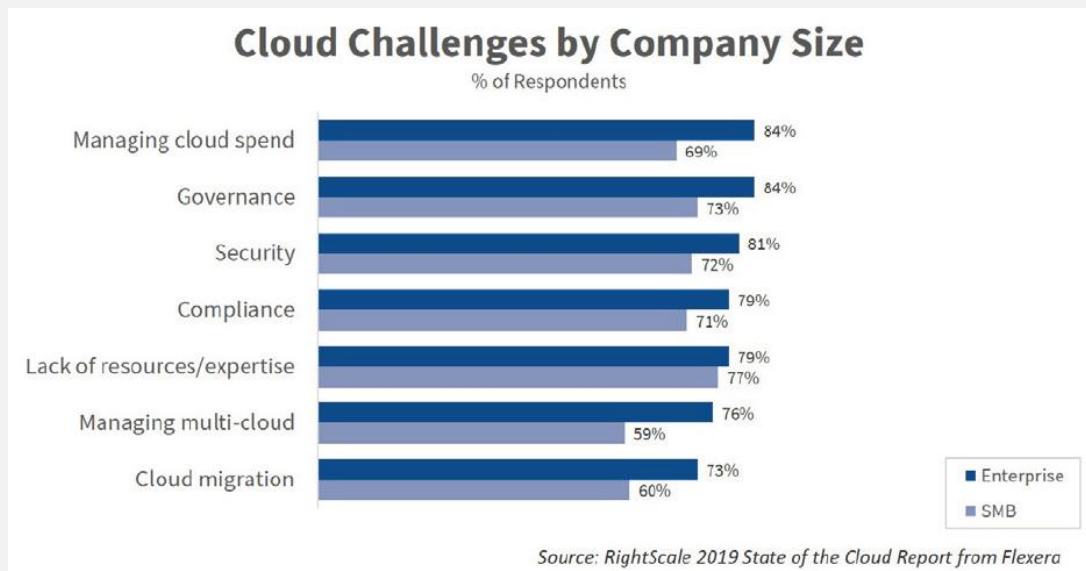
To effectively control and optimize costs on behalf of customers, partners need visibility into the customer's IT environment and the ability to optimize their deployment to reduce costs. And when dealing with customers who have multiple applications, cost transparency is needed to break down costs across each of those applications.

That's why there are Microsoft and third party tools for visualizing and optimizing a customer's IT environment costs—on premises, on Azure, or in a hybrid deployment.

Azure Cost Management is an Azure service designed to help understand and manage cloud spend. It supports cost analysis, forecasting, budgets, and alerting. It also provides cost optimization recommendations.

Cost Management in Azure was originally provided via the standalone Cloudfy service, which Microsoft acquired in 2017. This service is gradually being replaced by the Azure Cost Management service.

At this time, Azure Cost Management does not support CSP subscriptions, so CSP Partners should continue to use Cloudfy for their cost management scenarios. New Cloudfy registrations are limited to Microsoft CSP Partner administrators only. Existing Cloudfy customers can continue using Cloudfy for a limited period.



VISIBILITY

Azure Cost Management and Cloudfy provide a view into the costs of the Azure cloud environment. These reports can be analyzed and the subscription or resource group

level. They can also be aggregated across subscriptions in a team or department. For Azure Cost Management, the Azure Management Group hierarchy is used; for Cloudfy a custom hierarchy can be defined. Cost can also be

filtered using resource tags to create a wide range of views.

Tracking usage and costs trends is provided by the Cost Analysis area of the tool, which provides a report of costs against time. When first used, the report will have no groups or filters applied, so this shows the all-up cost for the entire Azure environment. The report can be filtered by the various Azure services consumed by this subscription or by groups that can be added. Some examples of groups are departments or applications that have been identified using Azure Tags. The use of the filters allows a partner to identify and monitor spending on behalf of each customer.

Historical data can help manage costs by analyzing usage and costs over time to identify trends. Trends are then used to forecast future spending. Cloudyn also includes useful projected cost reports.

Access control helps manage costs by ensuring that users and teams access only the cost management data that they need. For Azure Cost Management, this is integrated into the Azure role-based access controls. For Cloudyn, use an entity structure, user management, and scheduled reports with recipient lists to assign access.

MONITORING AND ALERTING

Monitoring usage and spending is critically important for cloud infrastructures because organizations pay for the resources they consume over time. When usage exceeds agreement thresholds, unexpected cost overages can quickly occur.

There are a few important factors which can make ad-hoc monitoring difficult. First, projecting costs based on average usage assumes that consumption remains consistent over a given billing period. Second, when costs are near or exceed the budget, it's important that both the provider and customer get notifications proactively as to allow for planning or to adjust Azure spending.

Azure Cost Management and Cloudyn can alert stakeholders automatically to spending anomalies and overspending risks. Various reports support alerts based on budget and cost thresholds.

An alert can be created for any spending using any Cost report. For this to work the report must first be configured using the filters and then scheduled to run at intervals that partners determine with the customer. The threshold of

aka.ms/practiceplaybooks

spending will also be configured. The trigger for the alert will be the difference between the value the report returns as the Active Spend vs. the Threshold.

OPTIMIZATION (RIGHT-SIZING)

An important aspect to any cloud deployment is ensuring that the right resources are being used for the job at hand. Azure Cost Management enables partners to drive optimization into their clients' environments.

Resource use optimization or "rightsizing" on behalf of customers is a valuable service offering for an MSP practice. In a hybrid or public cloud deployment, consolidate workflows running on multiple, under-utilized resources, which has a direct impact on cost. Conversely, spin up additional resources when one is no longer enough to handle your customer's workloads.

There are many aspects to cost optimization that add varying degrees of value to both an MSP practice and the customers served. While the goal with any cost optimization offering should be increased efficiency and lowered TCO, an MSP will need to choose to offer will differ and can range from "rightsizing" to harnessing the power of analytics to offer actionable intelligence with demand forecasting.

There is also the means to receive recommendations, for example on switching from on-demand to pre-purchase VMs, or to reveal underutilized VMs which can be moved to a lower cost VM family or size. For Azure Cost Management, these recommendations are surfaced via Azure Advisor. Cloudyn also provides recommendations, including on how to reallocate workloads or where to switch from standard to low priority VMs for cost effectiveness. Some recommendations provide information that would be very difficult to figure out otherwise—such as showing unattached block-blob accounts that were left behind from a deleted VM—helping to minimize waste.

The Optimizer reports in Cloudyn improve efficiency, optimize usage, and identify ways to save money spent on cloud resources. They are especially helpful with cost-effective sizing recommendations intended to help reduce idle or expensive VMs.

The Cost-Effective Sizing Recommendations report identifies potential annual savings by comparing VM

instance type capacity to their historical CPU and memory usage data.

TRANSPARENCY AND ACCOUNTABILITY

Prior to moving to the cloud, most IT departments don't have a true idea of what their services cost. Often the IT department is seen as overhead to a business or even a black hole that is difficult to understand. With the cloud, this model is turned up-side down, as IT can now see the exact cost of each and every resource. However, this again makes it very difficult to understand how the charges should be allocated across their business.

With Azure Cost Management and Cloudyn, MSP can help their customers get an enterprise-wide cloud accountability by enabling accurate cost allocation and chargeback across the company. These costs can be rolled up and reported on to different entities that can be defined including subscriptions, accounts, departments and cost centers.

This could be as simple as tagging Azure resources for simplified cost allocation. More complex models can also be implemented with different types of cost allocation such as blended/average/normalized rates, compute pre-purchase rates, or any other policy.

Cost allocation manages costs by analyzing costs based on tagging policy. Use tags on custom accounts, resources, and entities to refine cost allocation. Category Manager organizes tags to help provide additional governance. Use cost allocation for show back/chargeback to show resource utilization and associated costs to influence consumption behaviors or charge tenant customers.

Alerting helps manage costs with automatic notifications when unusual spending or overspending occurs. Alerts

can also notify other stakeholders automatically for spending anomalies and overspending risks.

MSP & CSP COST MANAGEMENT

Partners providing managed services for multiple customers face cost management challenges. Customers are looking to partners, as the experts and providers of these services, to help ensure they are getting the most out of their commitment to Azure.






MSPs or CSPs can use [Cloudyn for CSPs](#) to manage and monitor customers' cloud deployments for optimal efficiency and growth. Azure Cost Management manages and optimizes multi-platform clouds by enabling full visibility and accountability, packaged with continuous optimization across all clouds. Supported platforms include Azure, AWS, Google Cloud, and cloud containers.

Azure Cost Management for CSPs supports additional capabilities designed specifically for CSPs:

- Manage and monitor end-customers' consumption, cost and profitability
- Get visibility into enterprise-grade n-tier hierarchy per end-customer
- Support any value-chain business model (n-tiers, direct, indirect, partner, distributor)
- Guarantee full data segregation through Azure Cost Management's multi-tenant application
- Implement control policies and consumption limits through reports and alerts
- Customize margins and discounts per end-customer
- Apply preferred cost allocation methods within end-customer
- Manage customers' subscriptions and billing via custom-built portal

THIRD-PARTY COST MANAGEMENT

These are just some of the third-party tools offering cost management functionality for Azure.

	<p>APPTIO Cost Transparency</p> <p>Apptio Cost Transparency provides visibility to all public cloud costs across providers like AWS and Azure. It can monitor public cloud spend and trends by service type, such as compute, storage, network, and understand which departments are consuming cloud services to help proactively manage spend.</p>
	<p>CloudCheckr</p> <p>CloudCheckr is a multi-cloud management platform enabling enterprises and service providers to save money, reduce risk, and ensure governance at scale.</p>
	<p>Corent SurPaaS®</p> <p>SurPaaS® is a platform for migrating and operating applications. It has monitoring and metering features that collect usage data for the Azure infrastructure, and to examine the actual application transactional activity on a tenant basis in order to create customized tenant cost allocations and billing.</p>
	<p>SCALR</p> <p>Cost is affected by all other aspects of cloud usage, and Cost Control doesn't stop with visibility. Gain insights that turn into financial guardrails that encourage smart behavior.</p>
	<p>HANU Insight</p> <p>Track, optimize, budget, chargeback & invoice Microsoft Azure spend</p> <p>Hanu Insight is a Continuous Financial Governance product that provides End-to-end visibility of Azure spend and optimizes consumption to maximize Azure investments.</p>

PRE-PURCHASING

Helping cloud customers save by using up-front commitments to Azure is one of the most important roles that a partner can play during and after a migration. There are many program and strategies that are available for partners to help their customers manage their budgets.

A move to the cloud is a shift in mindset for customers with respect to purchasing their technology. Traditionally, they have purchased hardware and software upfront using capital expenditures with no ongoing commitment to the platforms they have chosen beyond support. With a migration to the cloud this relationship changes, due to the pay-as-you-go nature of the cloud. This means there are new strategies that cloud partners must bring to the table both in terms of licensing and implementation. This guidance and optimal deployment of Azure features can make or break the success of a migration.

Customers are going to be seeking information about how the transition will impact their budget and their existing investments. In addition, they will require input into strategies for saving on a long-term commitment to Azure as their cloud platform.

Pre-purchasing Azure is a great way to manage those costs and make the most out of every dollar that is spent on the platform. With the right strategies, customers will often accelerate their transition which will and help them gain business velocity from their decision to migrate to the cloud.

In this section, we'll look at five pre-purchase or credit schemes to significantly reduce Azure spend.

ENTERPRISE AGREEMENT (EA) WITH AZURE MONETARY COMMITMENT

The [Enterprise Agreement](#) offers enterprise customers the distinct benefit of having one contract to purchase all of their Microsoft products and services. Any Enterprise Agreement customer can add Azure to their agreement by making an upfront monetary commitment to Azure. This is essentially a dollar amount that the customer expects to spend on Azure over the course of each year during the EA. EA's do allow a customer to negotiate their rates based on their commitment to Microsoft which is one attractive reason for entering into this type of contract.

When an EA is signed, the customer will make an upfront payment to cover the cost of Azure for the first year. Each subsequent year another payment is made to Microsoft. That commitment is consumed throughout the year by using any combination of the wide variety of cloud services Azure offers from its global datacenters. Microsoft will also extend the same rates that were negotiated for services consumed beyond the commitment. This allows customers to use Azure even with this unplanned growth, so they can meet their organization's needs. Enterprise Agreement customers can pay Microsoft at the end of the year for unplanned growth, as long as use is within certain thresholds. If the customer doesn't use all of the pre-purchased budget for the year it is forfeited and can't be rolled over to the next year.

Azure can be added to an Enterprise Agreement at any time, although the anniversary or expiration of an existing Enterprise Agreement commitment is a great time to evaluate usage and future plans for the platform.

Azure, via the Enterprise Agreement, also gives customers access to the Enterprise Portal, a great resource for customers managing multiple accounts or subscriptions—see the section on [Azure Subscriptions](#) for further information.

SERVER AND CLOUD ENROLLMENT

[Server and Cloud Enrollment](#) is an enrollment under the Microsoft Enterprise Agreement that enables customers to commit to one or more key server and cloud technologies from Microsoft. In exchange, they get the best pricing and terms, plus other benefits such as cloud-optimized licensing options and simplified license management.

The following are some of the benefits of Server and Cloud Enrollment:

Best value

- Get the best pricing, discounts, and added benefits designed to support server and cloud technologies

- Best pricing and terms for server and cloud products, including discounts on new licenses and Software Assurance
- Full Software Assurance benefits for all deployed licenses, including new version rights
- Unlimited Problem Resolution Support for qualifying customers

Flexible

Move to the cloud as needed and grow organically without losing the value of existing investments.

- Application license mobility to the cloud through Software Assurance
- New subscription-based licensing offers more flexibility to retire workloads, consolidate, or migrate to the cloud

Manageable

Adopt the latest technologies while simplifying deployment and license management.

- Simplified licensing management streamlines overall deployment and management
- Standardized terms, conditions, and discounts
- A standardized management platform across on-premises and Microsoft Azure comes with commitment to the Core Infrastructure Suite (CIS) in Server and Cloud Enrollment

To enroll, customers must make an installed-base commitment to one or more of the following four Server Cloud Enrollment components:

- **Core Infrastructure:** Windows Server and System Center
- **Application platform:** SQL Server, BizTalk Server, and SharePoint Server
- **Developer platform:** Visual Studio Enterprise
- **Microsoft Azure:** All Azure cloud services

This means committing to full Software Assurance coverage across the installed base of a Server and Cloud Enrollment component. For the Core Infrastructure component, however, customers can commit to full System Center coverage on the Windows Server installed base through the Core Infrastructure Suites (CIS). Microsoft Azure is automatically available when enrolling in any of the other three components, and it can also be licensed standalone.

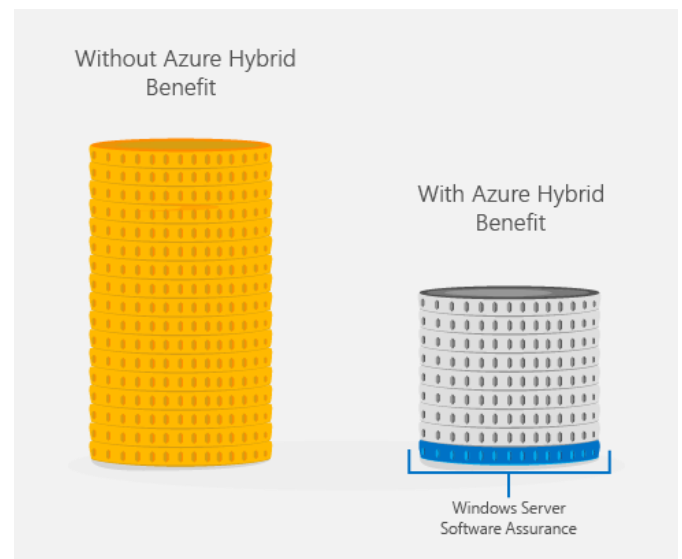
HYBRID BENEFIT

When creating a Windows Server virtual machine in Azure there are two components that make up the cost per minute:

- **Base compute cost:** cost for the hardware of the VM, comprising the CPU, memory, and temp disk (OS and data disks are managed separately)
- **Windows OS license:** cost of the Windows operating system

If a customer already has Windows OS licenses with Software Assurance, then Microsoft extends a benefit which can help them save up to 40 percent on Windows Server VMs. This is known as the [Azure Hybrid Benefit for Windows Server](#). This allows customers to use their on-premises Windows Server licenses with Software Assurance or Windows Server Subscriptions to save when migrating to Azure. With this benefit, for each license Microsoft will cover the cost of the OS, while the customer is only charged for the base compute costs (which are the same as the Linux VM costs).

The Azure Hybrid Benefit helps customers get more value from their Windows Server licenses for machines that are migrated to the cloud since the investment in that software isn't lost.



Hybrid Benefit is also available for running SQL Server workloads in Azure. Existing SQL Server licenses with Software Assurance can be used to offset the license costs of either SQL Server running in Azure VMs, or the Azure SQL Database service.

Hybrid Benefit is a great benefit for customers, but it is very important to understand the details of the program. Partners can help them plan and be successful with this program, as this can provide huge cost savings and could even be the reason a deal closes. Understanding some of these basics will go a long way toward ensuring their customer's success and optimizing their spend on Azure.

Hybrid Benefit for Windows Server VMs

Customers can use the benefit with Windows Server Datacenter and Standard edition licenses covered with Software Assurance that they own. Depending on the edition, customers can convert or re-use their licenses to run Windows Server virtual machines in Azure and pay a lower base compute rate (Linux virtual machine rates).

Each 2-processor license or each set of 16-core licenses are entitled to two instances of up to 8 cores, or one instance of up to 16 cores. Customers can also use multiple licenses to cover licensing for a large VM in Azure. For example, a virtual machine with more than 16 cores can be covered with HUB by "stacking licenses". With two 2-processor licenses or two 16-core licenses, the customer would have the HUB to cover the OS license for a VM up to 32 cores. There is a 90-day assignment rule for Windows Server, which requires licenses to stay assigned to the same hardware for a minimum of 90 days and this rule does apply to Azure virtual machines.

For every 2-processor Windows Server license or Windows Server license with 16-cores, Software Assurance covers either of the following:

- Up to two virtual machines with up to 8 cores, or
- One virtual machine with up to 16 cores

There are a few differences on how the HUB can be used based on the type of license the customer owns:

- **Standard Edition Licenses** can only be used once either on-premises or in Azure. Once assigned, to Azure, the Azure Hybrid Benefit cannot use the Standard Edition license on-premises again.
- **Datacenter Edition Licenses** allow simultaneous, indefinite usage both on-premises and in Azure.

Hybrid Benefit for SQL Workloads

Azure Hybrid Benefit for SQL Server can be used with both SQL Server Enterprise Core and SQL Server Standard Core licenses. In both cases, Software Assurance is required.

For SQL Server running in Azure VMs, these licenses transfer directly to Azure, core-for-core. That is each Enterprise Core license qualifies for one core of SQL Server Enterprise Core in Azure VMs, and similarly for Standard Core licenses. In both cases, a minimum of four core licenses must be allocated per VM.

With Azure SQL Database, the licensing is a little more complicated. The SQL Database can use any of the Managed Instance, Elastic Pool or Single Database deployment models. However, the database must use the vCore pricing model, not the DTU-based pricing model.

Each SQL Server license provides the following Azure SQL Database license rights:

- **SQL Server Enterprise Edition:** 1 core on-premises license provides either 1 core at the Business Critical tier, or 4 cores at the General Purpose tier.
- **SQL Server Standard Edition:** 1 core on-premises license provides 1 core at the General Purpose tier only.

Each license can only be used either on-premises or in the cloud. However, there is a 180-day grace period to use the same license both on-premises and in the cloud, to facilitate migration.

Hybrid Benefit with CSP

If customers will be purchasing Azure services through a Microsoft partner acting as a Cloud Solution Provider, they are still eligible for the Azure Hybrid Benefit. The same rules apply in that the software must have active Software Assurance or Windows Server Subscription coverage. There is nothing different about Hybrid Benefit use on Azure subscriptions purchased from via CSP.

Deployment Options

For Windows Server, there are several ways to deploy VMs to take advantage of Hybrid Benefit pricing:

- Partners can deploy Windows Server virtual machines for their clients pre-configured with the Azure Hybrid Benefit straight from the Azure Portal, using ARM Templates, PowerShell, or the Azure CLI. Windows Server Datacenter Azure Platform images are made available via in the Azure Marketplace. There are no Windows Server Standard images in Azure, so Microsoft allows Windows Standard licenses to be used with the Windows Datacenter Azure Platform images at no additional cost.

- Another option is to upload OS images to the customer's subscription and use these as the basis for deployments. These images should be generalized to ensure that they don't have issues as duplicates in the environment. Windows VMs should have the sysprep tool run on them using the Out of Box Experience. The image should then be uploaded to Azure Storage and then made into a Managed Image. For more information on how to use PowerShell, view this article on [uploading on-premises VMs](#). All VMs created using this image will be charged at the base compute rate and the OS licensing will be covered under the Hybrid Benefit scheme.
- Another option is to upload a current Windows Server virtual machine disk (VHD) to Azure and create an Azure VM using this disk. Upload the VHD using PowerShell cmdlets, but only after the VM has been prepped for Azure. The process to prepare a VM to upload to Azure can be reviewed in this article on [preparing a Windows VHD or VHDX to upload to Azure](#). Once this VMs is up and running in Azure it will be charged at the base compute rate and the OS licensing will be covered under the Hybrid Benefit scheme.
- Another option is to [migrate an existing workload to Azure using Azure Site Recovery](#). Once migrated, the virtual machine in Azure will be running as a custom image and thus will be charged at the base compute rate and the OS licensing will be covered under the Hybrid Benefit scheme

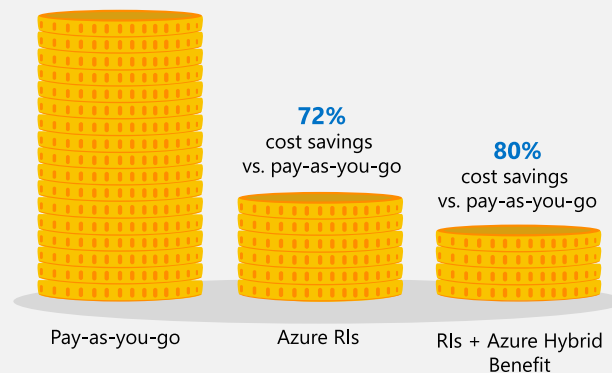
AZURE RESERVATIONS

Another way to save on Azure costs is through pre-purchasing compute capacity using Azure Reservations. Azure Reservations are available for Azure Virtual Machines, Azure SQL Database, Azure Cosmos DB and Azure SQL Data Warehouse.

Azure VM Reserved Instances

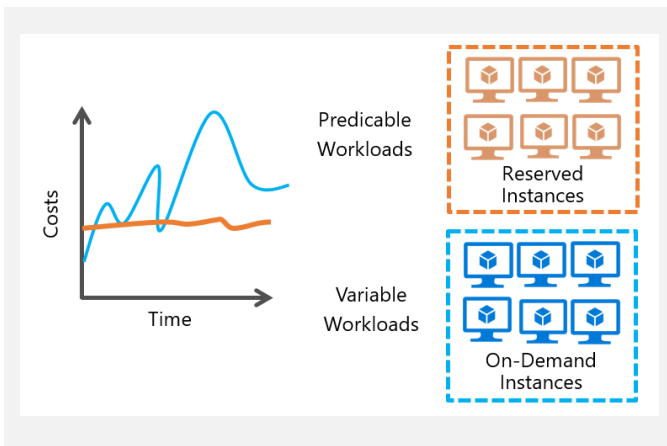
Reservations for Azure VMs are called [Azure Virtual Machine Reserved Instances](#). Customers can reserve virtual machines in advance and enjoy cost savings of up to 72% on pay-as-you-go prices. When combining the cost savings gained from Azure Reserved Instances with the added value of the Azure Hybrid Benefit, customers can save up to 80 percent on their Windows workloads.

Save up to **80%** with RIs and Azure Hybrid Benefit



Reserved VM Instances provide price predictability, as well as the flexibility to exchange or cancel to get pro-rated refund, should their needs change. Discounts for Reserved Instances are determined based on the customer's commitment to a one-year or three-year term on Windows and Linux virtual machines (VMs). Reserved Instances also help customers improve their budgeting and forecasting with a single upfront payment, making it easy to understand their investment and avoiding unexpected overages.

Partners can lower their customer's total cost of ownership by combining Reserved Instances with On-Demand instances to manage costs across predictable and variable workloads. In customer environments, there will be different types of systems, some of which have known requirements for their VMs. An example of a system with known requirements would be an SAP implementation where the exact specifications for the VMs are locked and are not expected to change during the next three years. This is a great example of how Reserved Instances could be used to maximize the savings. In this same environment, a public e-commerce web application that has variable needs could leverage the ability to scale up and down based on the number of visitors on the web site. This workload would be best suited to the on-demand pricing leveraging standard the pay-as-you-go model.



Reserved Instance Purchasing and Billing

Reserved Instances are purchased in one-year or three-year terms with a single upfront payment. Purchasing can be completed very easily using the Azure portal in three steps:

- Specify the Azure region
- Select the Virtual Machine type
- Chose a term (one year or three years)

For Enterprise Agreement (EA) customers, Azure Monetary Commitment can be used to purchase Azure Reserved VM Instances. In scenarios where EA customers have used up their monetary commitment, Reserved Instances can still be purchased, and those purchases will be invoiced on their next overage bill. Reserved Instances can be assigned at the enrollment or subscription level to manage Reserved Instance usage at an organizational or individual department level. Assignments are easy to change post-purchase.

For customers purchasing via Azure.com, at the time of purchase, the credit card on file will be charged for the full



upfront payment of the Azure Reserved Instances. To see

Reserved Instance purchase details, use the Azure portal and select the Reservations menu on the left side of the Azure Portal to view all Reserved Instances associated with the account. All Reserved Instances will be displayed on the right.

The Reserved Instance will apply immediately to any existing running VMs that match the one-year or three-year terms of the Reserved Instance, depending if the Reserved Instance is scoped to a specific subscription or applied at the enrollment level. In both cases, the period of the Reserved Instance starts immediately after purchasing.

The Reserved Instance purchase can be assigned to a subscription or enrollment and can be changed as necessary. Assignment helps partners decide whether the reservation is applied at the Azure account/enrollment or at the subscription level. This provides flexibility for how to leverage the savings. For example, to simply buy reservations to save money for the customer's organization, assign all reservations to the account level. If the customer wants to apply the Reserved Instance savings to a particular business unit, such as finance, provision a subscription for that department and then assign the Reserved Instance to the subscription. Then only they would be able to take advantage of the Reserved Instance commitment savings.

Azure Reserved Instances provide a single price for each VM size in a region. There is no requirement to choose Windows or Linux VMs for the Reserved Instance purchase. If Windows Server VM is selected, there is an option to use the Azure Hybrid Benefit or pay the Windows Server rate.

There are two options for adding Windows Server licenses to an Azure Reserved VM Instance. The first option is to use the Azure Hybrid Benefit. With Windows Server with Software Assurance on-premises, assign these licenses to the Azure Reserved Instance. The second option is to add Windows Server using the Windows Server hourly meter. If the Azure Hybrid Benefit cannot be applied, Windows Server will be charged when the VM is active, based on the number of cores the VM is using.

Availability of Reserved Instances

Azure Reserved Instances are available for all VM families other than A-series, A_v2 series, or G-series, or any VM series in Preview. Azure Reserved Instances are not available in Azure Government, Germany, and China. EA customers can purchase reservations in all countries available in Azure today. For customers who have signed up through azure.com (Pay-As-You-Go customers), Azure Reserved Instance offer is not

available in India, Brazil, Taiwan, Russia, Korea, Argentina, Hong Kong, Indonesia, Liechtenstein, Malaysia, Mexico, Saudi Arabia, South Africa and Turkey. Because of the flexibility Microsoft does offer to exchange Reserved Instances to a different region or VM family, but there is no guaranteed availability of capacity in a given region or VM family.

Reserved Capacity for Database Workloads

Azure reservations are also available for Azure SQL Database, Azure Cosmos DB, and Azure SQL Data Warehouse. This is known as Reserved Capacity.

As with Reserved Instances for Azure VMs, Reserved Capacity is based on a one-year or three-year pre-purchase. The reservation is purchased separately from the service, and service billing is discounted against matching reservations.

For each service, the reservation is purchased in the following units:

- For Azure SQL Database, each reservation specifies the number of vCores (minimum 8 vCores)
- For Cosmos DB, each reservation specifies the capacity as a number of Request Units per Second (minimum 20,000 RU/S)
- For Azure SQL Data Warehouse, each reservation specifies the number of Data Warehouse Units (cDWU), in multiples of 100.

As with Azure VM Reserved Instances, each reservation must specify the Azure region and whether it is scoped to a single subscription or shared across subscriptions in an EA or account.

Exchanges and Cancellations

Azure reservations for both VMs and databases require making upfront commitments on compute capacity, but Microsoft allows for flexibility should the customer's business needs change. An exchange allows the customer to receive a prorated refund based on the unused amount which applies fully to the new purchase price. A cancellation terminates the contract and Microsoft will provide a prorated refund based on unused amount minus an early termination fee of 12 percent. Customers can cancel a reservation at any time (up to \$50,000 per year).

DEV/TEST PRICING

Microsoft provides discounted rates on Azure for customers' ongoing development and testing needs. This

includes no Microsoft software charges on Azure Virtual Machines and [special Dev/Test pricing](#) on other services.

Discounted rates on Azure to support ongoing development and testing includes:

- No Microsoft software charges on Virtual Machines
- Significant dev/test pricing discounts on a variety of other Azure services
- Exclusive access to Windows 10 Virtual Machines

There are also Dev/Test pricing discounts on the services that these teams consume in Azure. Including significant discounts on VMs, SQL VMs, Azure App Service, HD Insight amongst others.

There are two methods for customers to receive these benefits and discounts, and it based on the type of client and their Dev/Test workload in Azure.

Small Development Teams

Small teams are best suited to leverage Visual Studio subscriptions which include benefits for Azure. Each subscription includes a monthly Azure credit which is dependent upon the level of the subscription and ranges from \$50 to \$150 (full details at [Azure credits for Visual Studio subscribers](#)).

The monthly Azure credit for Visual Studio subscribers is ideal for experimenting with and learning about Azure services. When this benefit activates, it creates a separate Azure subscription with a monthly credit balance that renews each month while the subscriber remains an active. Any overage above the credit would be billed to a credit card that the Developer would put in file with Microsoft. If no credit card is on file, then Azure services are suspended if the credit balance is exceeded.

Large Development Teams and Enterprises

Development and Test subscriptions for large teams are purchased via an EA and require that each user be an active Visual Studio Online subscriber. Unlike the small team these do not include a credit each month, but instead are eligible for the discounted rates on Azure services. This is a significant benefit due to the nature of how these teams work on their projects. For example, if a group of developers were writing code that requires SQL Server Enterprise Edition, they won't be charged for that software running on their development VMs. Only the production SQL Server would have this charge—which is a very large savings over time.



Automatic Shutdown of VMs

One of the most attractive attributes of the cloud is the pay-as-you-go nature of the services. Take full advantage by only using resources when needed.

In many environments, there are times of day when certain services aren't needed and thus there is no reason for them to be running. This is particularly true of Azure Virtual Machines, which can be shut down and later re-started with no loss of data.

Non-production Cloud servers need to be online only when employees are actively working on them. In some cases, non-production environments can be turned off, or de-allocated, over 70 percent of the time, which translates into a direct 70 percent cost reduction.

As a partner that is helping a customer move to the cloud, it is important to understand the requirements placed on each virtual machine that will be deployed. Understanding which virtual machines only provide useful service during certain hours of a workday or even days of the week is critical to maximizing the potential savings. This could also change over time as the customer's business evolves or as new workloads are onboarded to the cloud.

It is important to understand that there are two ways to stop a virtual machine:

- Shutting down the virtual machine from within the virtual machine OS puts the virtual machine into the **'Stopped (allocated)'** state. In this state, the underlying infrastructure (CPU, memory) is still reserved for the virtual machine, and hence the virtual machine is still billed
- Stopping the virtual machine from the Azure portal or other Azure tools (PowerShell, CLI, etc.) puts the virtual machine into the **'Stopped (deallocated)'** state. The virtual machine is no longer billed, resulting in significant potential savings (although other resources such as VM disks associated with the VM may still be charged)

A stopped virtual machine can be re-started at any time and will continue where it left off since its disks have not been affected. Note that if the virtual machine is associated with a Public IP Address, then stopping and re-starting the virtual machine may result in a new Public IP

Address being allocated. To preserve the same Public IP Address, a static Public IP Address should be used.

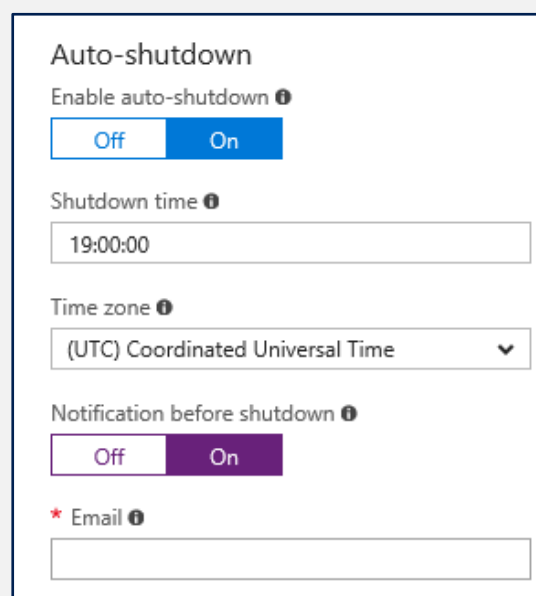
Azure supports several ways to implement automatic shut-down of virtual machines. We'll now look at each in turn.

AUTO SHUT DOWN OF VMS

[Automatic shut-down of virtual machines](#) is one of the ways that Microsoft helps customer to optimize their costs, by automatically shutting down a virtual machine at a given time.

This feature was originally introduced as a feature of DevTest Labs and was widely adopted used by customers. Because of this success, auto shut down was added to all Azure Resource Manager virtual machines.

Customers can schedule a time each day (local time is supported) when a VM will be stopped. At this time, the virtual machine is stopped and deallocated, and Microsoft stops billing the subscription for the compute time until it is re-started.



The screenshot shows the 'Auto-shutdown' configuration panel in the Azure portal. It includes the following settings:

- Enable auto-shutdown:** A toggle switch set to 'On'.
- Shutdown time:** A text input field containing '19:00:00'.
- Time zone:** A dropdown menu set to '(UTC) Coordinated Universal Time'.
- Notification before shutdown:** A toggle switch set to 'On'.
- * Email:** An empty text input field.

Auto shut-down is available via the [Azure portal](#). This method is simple to use for individual virtual machines and can be configured while the virtual machine is being provisioned or after it has been created. The settings configured include the local time zone, the time to shut down and the option to notify the shut-down 15 minutes in advance, via email.

It is also possible to set the auto shut-down settings when deploying virtual machines by other methods, such as Azure Resource Manager templates. To do so, create a separate resource of type Microsoft.DevTestLabs/schedules, specifying the target virtual machine and auto shut-down settings. This configuration will have the same result as configuring the Auto Shutdown in the portal during the provisioning of a virtual machine and is how the portal configures them behind the scenes. The configurations can later be changed using the portal just as if they were provisioned there.

It is important to note that the auto shut-down feature has no corresponding auto start-up, so the virtual machines will need to be started manually. As an alternative, consider using one of the other approaches to auto shut-down described below.

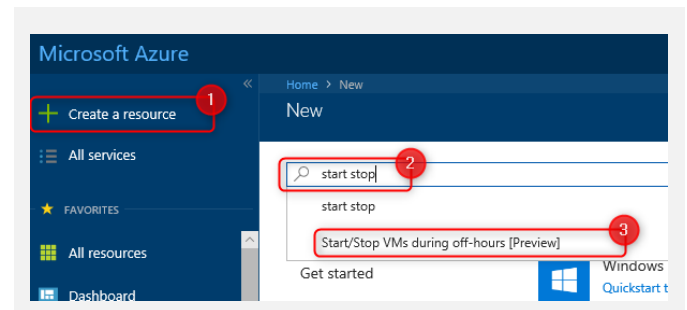
POWERSHELL VIA AZURE AUTOMATION RUNBOOKS

It is also possible to automatically shut down and restart Azure virtual machines using a 'runbook' executed using [Azure Automation](#). This is different than the Auto shut down feature using the Microsoft.DevTestLabs resource provider. This is fully dependent upon the partner or

customer to configure and leverages the Azure PowerShell cmdlets. For a packaged solution, see the next option, 'Start/stop VMs during off hours'.

START/STOP VMS DURING OFF-HOURS

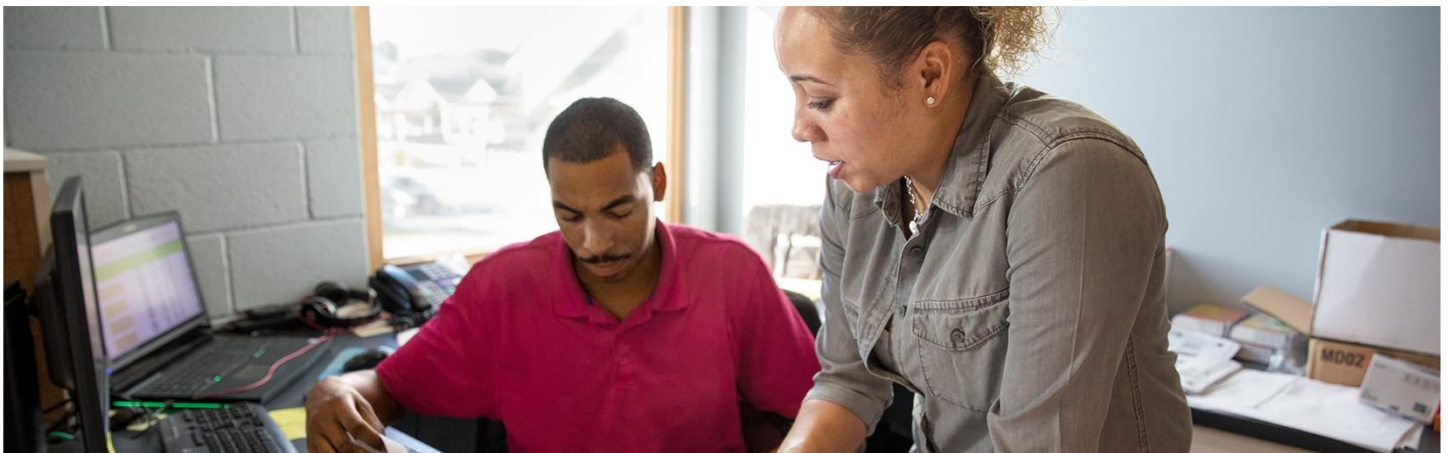
There is an Azure Marketplace solution known as '[Start/stop VMs during off-hours](#)'. Find this solution by clicking '+ Create a resource' in the Azure portal, then entering 'start stop vms' in the search field.



This solution can start and stop Azure Virtual Machines automatically, using a schedule or based on utilization. The solution relies on two Azure services and a SendGrid service:

- **Automation:** starts and stops virtual machines.
- **Log Analytics:** visualizes the successful start and stop of the machines.
- **SendGrid:** sends email notifications of stop and start activities.

This solution is more complex, but more robust in that it is a full strategy for achieving auto start and stop for an entire subscription rather than targeting individual virtual machines. For further information, see the [start/stop VMs during off hours documentation page](#).



Optimized Architecture

The choice of application architecture can have a significant impact on running costs. Incorporating cost as a design goal can result in significant savings.

A common problem that affects organizations when they initially move resources into the cloud is their virtualization strategy. They often use an approach like the one used when creating virtual machines for the on-premises virtualization environment. And, they assume that costs are reduced by moving their on-premises VMs to the cloud, without any thought to changing how it is deployed and the resources (CPU & RAM), that are assigned. However, this approach is not likely to reduce costs.

The problem with this approach is that the existing on-premises infrastructure was already paid for. Users could create and keep large VMs running if they liked—idle or not and with little consequence. Moving large or idle VMs to the cloud is likely to increase costs. Cost allocation for resources is important when customers enter into agreements with cloud service providers.

Using the features of Azure with a new mindset is required to gain efficiencies from the cloud, and not end up with additional costs by moving. There are features in Azure which allow for autoscaling of compute infrastructures and thus, when coupled with the pay-as-you-go model, can provide for the savings customers desire. These strategies often have the added benefit of high-availability given that they scale the compute out and in instead of up

and down. This means more nodes provide the service rather than one larger node providing the service.

Moving to a PaaS platform for services is also a common strategy to change the architecture of a service and gain cost savings. Web applications that were traditionally running on VMs in a customer's datacenter can now be moved to the Azure App Service and will run just as today, but with much less overhead in terms of cost and management responsibility. This is especially advantageous to MSPs that are responsible for patching and securing these VMs. The move to PaaS means that their responsibility day to day will focus only on the application itself rather than the underlying OS.

In this section, we'll discuss the following approaches to optimizing the application architecture to reduce costs:

AZURE PAAS SERVICES OVER IAAS

PaaS over IaaS is the default stance that partners should take as a best first approach.

IaaS deployments require almost the same amount of effort as deploying and managing VMs in a customer datacenter without the hardware and facilities cost and complexity. The cloud computing service provider manages the infrastructure, while customers must purchase, install, configure, and manage their own



software, operating systems, middleware, and applications. IaaS does provide the advantages of no upfront capital commitment and the customer needs to rent a VM for as long as it is needed.

PaaS is a complete development and deployment environment in the cloud, with resources that enable customers to deliver everything from simple cloud-based apps to sophisticated, cloud-enabled enterprise applications.

Like IaaS, PaaS includes infrastructure—servers, storage, and networking, but also middleware, development tools, business intelligence (BI) services, database management systems, and more. PaaS is designed to support the complete web application lifecycle: building, testing, deploying, managing, and updating.

PaaS allows customers to avoid the expense and complexity of buying and managing software licenses, the underlying application infrastructure and middleware or the development tools and other resources. Customers and MSPs will operate and manage the applications and services, while Azure manages everything else.

When making recommendations and determining how to migrate the customer to Azure, the partner should take a stance that PaaS is the first answer to how something should be migrated to Azure. Starting from PaaS and only backing away to an IaaS deployment should be only due to specific reason(s) why the deployment can't leverage a PaaS service.

There are times when a complete PaaS offering might not make sense, but partners should use a cloud mindset



when evaluating these circumstances. For example, if a web server must remain on a VM in Azure IaaS the question should be asked: "Can the images and data files that are downloaded to the client be offloaded from the VM to Azure Storage?" Azure Storage is a HTTP web server at scale that can easily host all of the applications images, PDFs or other binary files that will be sent to the clients while the app is in use. A simple change to the HTML code of the app to reference a different URL for these files and the application is now leveraging a PaaS service. In the customer's datacenter those downloads would be left to the VM, but with Azure Storage all the files could be placed on that service which would free up the burden of loading these files from the VM to the client. This could mean the difference in the size of the VM and thus lessen the customer's bill each month for that application.

AUTO SCALE VM SCALE SETS & AZURE WEB APPS

Auto-scale is a built-in feature of many Azure services to meet the changing needs of applications. Typically, these are in the form of Websites with fluctuating compute requirements based on network traffic. When autoscaling Azure will add more nodes to service the application and then remove nodes when the traffic spike diminishes. This is known as scaling-out and scaling-in to meet the performance demands. Of course, performance means different things for different applications. Some apps are CPU-bound, others memory-bound. For example, a web app that handles millions of requests during the day and none at night. Auto-scale can scale a service by any of these or via a custom metric defined by the customer or partner.

Partners should be proactive with clients and build autoscaling into their migrations. It's not wise to wait for a traffic spike to take down the app or site and scale the VM size to meet the demand. With scheduled auto-scale, partners can respond before anything ever happens. If an MSP customer has a retail shopping site and Black Friday is coming, simply use auto-scale to add 10x more virtual machines in advance to handle the load. Another example would be for a website that traffic always skyrockets at 9am Monday through Friday. This can be handled with an autoscaling rule that triggers a scale out during those hours, but contracts during the offer hours. By using this simple approach to better understand the application,

partners are ensuring their successful implementation in an optimized manor.

[Azure App Service](#) is a great example of using a PaaS service that can auto-scale. This service is used for hosting web applications, REST APIs, and mobile back ends. It supports by Windows and Linux along with many programming languages including .NET, .NET Core, Java, Ruby, Node.js, PHP, or Python.

Web Apps not only adds the power of Microsoft Azure to the application, such as security, load balancing, autoscaling, and automated management. It also has DevOps capabilities without the need to provision and build other VMs or services. These capabilities include continuous deployment from VSTS, GitHub, Docker Hub, and other sources, package management, staging environments, custom domain, and SSL certificates.

With App Service, the customer pays for the Azure compute resources they use. The compute resources used is determined by the App Service plan that is implemented during the migration. Keep in mind that providers need to give guidance on the size of the App Service plan just as they would for VM sizes. Each size and type of App Service plan provides different capabilities from the number of cores and RAM to storage space and ability to host custom domain names for their site.

For deployments that require an IaaS deployment, but also need the ability to auto-scale partners will need to leverage [Virtual Machine Scale Sets \(VMSS\)](#). VMSS is an Azure compute resource that can be used to deploy and manage a set of identical VMs. With all VMs configured the same, scale sets are designed to support true auto-scale, and no pre-provisioning of VMs is required. So, it's easier to build large-scale services that target big compute, big data, and containerized workloads.

VMSS should be used for applications that need to scale compute resources out and in, and are highly available given that they are deployed behind the Azure load balancer and balanced across fault and update domains.

The VMs that are part of the VMSS can be configured using the VM Custom Script extension, via Azure Automation Desired State Configuration (DSC) or other VM extensions such as CHEF or Puppet. By using automation to configure the application on the VMs that are created by the VMSS there is no need for the administrator to take any actions during a scaling event.



The required VMs will be created by the autoscaling rule and then we then boot for the first time they will be configured with the application.

REGION PRICING DIFFERENCES

With 42 announced regions (more than any other cloud provider), Azure makes it easy to choose the datacenter and regions that is right for partners and their customers. Within each of these regions there are services that are made available and each have a cost associated with them.

With that in mind, it is important to understand that there are two variables to these services:

- **Availability:** not all services are available in every region
- **Cost:** service costs vary by Azure Region

Availability of services is based entirely on when Microsoft provides the service to a region. For example, there are many different Virtual Machine families in Azure which are named using letters from the alphabet such as A, B, D, M amongst others. These VM types aren't always available in every region when they are released by Microsoft. Some VM types are available in many regions while others are only available in a few. The DV2 series of VMs is an example of almost full coverage across all of Azure. This contrasts with the M series VMs which are only available in two US regions and one in Europe.

There is also an availability gap between the Azure Commercial Cloud and the Azure Government or Country Clouds (Germany and China). The Azure Government and Country Clouds are entirely separate Azure instances and

not connected to the Azure Commercial Cloud. The services that are available to the Government and Country Clouds vary dramatically with large gaps, so it is very important to understand the clients' needs and where they should deploy.

Cost is of course top of mind as the partner that is working with a client to them move to Azure. Each of these regions has their own price list and these prices do fluctuate over time. In some cases, the cost difference between a service in one region as compared to another may be relatively insignificant. In other cases, it can become a critical concern.

The [Azure Calculator](#) can show the standard pricing by region for a VM (not all regions are supported in the calculator). For example, the price of a DS3_V2 in East US may be \$409.92 a month, while in East US2 it may only be \$359.41.

Tax consequences should also be considered when pricing solutions. Depending on their location and registrations with local governments, different taxes will apply. In Ireland and the EU there are rules that dictate the collection and payment of VAT. In the United States and Canada there are rules with respect to tax-exemptions status. It is best to work directly with a Microsoft reseller or distribution partner to determine the exact nature of tax issues for clients. This is a very important topic as proper billing and contract procedures must be followed or, in some cases, Microsoft may collect taxes which will make the customer's Azure bill much higher than anticipated.

More information on Pricing in Azure can be found in the [Azure Price FAQ](#).

AZURE SQL DATABASE ELASTIC POOLS

[Azure SQL Database elastic pools](#) help customer manage and scale multiple Azure SQL databases. SQL Database elastic pools are a simple, cost-effective solution for managing and scaling multiple databases that have varying and unpredictable usage demands. The databases in an elastic pool are on a single Azure SQL Database

server and share a set number of resources (elastic Database Transaction Units (eDTUs) at a set price. Elastic pools in Azure SQL Database enable SaaS developers to optimize the price performance for a group of databases within a prescribed budget while delivering performance elasticity for each database.

Pools are well suited for many databases with specific utilization patterns. For a given database, this pattern is characterized by low average utilization with relatively infrequent utilization spikes.

The more databases added to a pool, the greater the savings. Depending on the application utilization pattern, it is possible to see savings with as few as two S3 tier databases.

Elastic Pools automatically scale performance and storage capacity for elastic databases on the fly. Partners can control the performance assigned to a pool, add or remove elastic databases on demand, and define performance of elastic databases without affecting the overall cost of the pool.

There are built-in sizing recommendations that proactively identify databases which would benefit from pools and allow "what-if" analysis for quick optimization to meet the performance goals. Rich performance monitoring and troubleshooting dashboards help visualize historical pool utilization.

There are three different performance and pricing tiers: Basic, Standard, and Premium. Each of these pools offer a broad spectrum of performance, storage, and pricing options. Pools can contain up to 400 elastic databases. Elastic databases can auto-scale up to 1,000 elastic database transaction units (eDTUs).

AZURE BLOB STORAGE (HOT, COLD & ARCHIVE)

Azure Blob Storage is a Microsoft-managed cloud service providing storage that is highly available, secure, durable, scalable, and redundant. Microsoft takes care of maintenance and handles critical problems that may arise with the service.



Azure Blob Storage is a service for storing large amounts of unstructured object data, such as text or binary data, that can be accessed from anywhere in the world via HTTP or HTTPS. Use Blob storage to expose data publicly to the world, or to store application data privately.

Common uses for Azure Blob Storage include:

- Serving images or documents directly to a browser
- Storing files for distributed access
- Streaming video and audio
- Storing data for backup and restore, disaster recovery, and archiving
- Storing data for analysis by an on-premises or Azure-hosted service

[Azure storage offers three Access Tiers](#) for Blob object storage. This enables customers to choose the most cost-effective storage option, depending on their usage patterns.

Access Tiers for Blob object storage:

- **Hot:** optimized for storing data that is accessed frequently
- **Cold:** optimized for storing data that is infrequently accessed and stored for at least 30 days
- **Archive:** optimized for storing data that is rarely accessed and stored for at least 180 days with flexible latency requirements (on the order of hours)

Today, data stored in the cloud is growing at an exponential pace. To manage costs for customers expanding storage needs, it's helpful to organize the data based on attributes like frequency-of-access and planned retention period to optimize costs. Data stored in the cloud can be different in terms of how it is generated, processed, and accessed over its lifetime. Some data is actively accessed and modified throughout its lifetime. Some data is accessed frequently early in its lifetime, with access dropping drastically as the data ages. Some data remains idle in the cloud and is rarely, if ever, accessed once stored.

Each of these data access scenarios benefits from a different storage tier that is optimized for an access pattern. With hot, cool, and archive storage tiers, Azure Blob storage addresses this need for differentiated storage tiers with separate pricing models.

Hot

Hot storage has higher storage costs than cool and archive storage, but the lowest access costs.

Example usage scenarios for the hot storage tier include:

- Data that is in active use or expected to be accessed (read from and written to) frequently
- Data that is staged for processing and eventual migration to the cool storage tier

Cool

Cool storage tier has lower storage costs and higher access costs compared to hot storage. This tier is intended for data that will remain in the cool tier for at least 30 days.

Example usage scenarios for the cool storage tier include:

- Short-term backup and disaster recovery datasets
- Older media content not viewed frequently anymore but is expected to be available immediately when accessed
- Large data sets that need to be stored cost effectively while more data is being gathered for future processing. (*For example*, long-term storage of scientific data, raw telemetry data from a manufacturing facility)

Archive

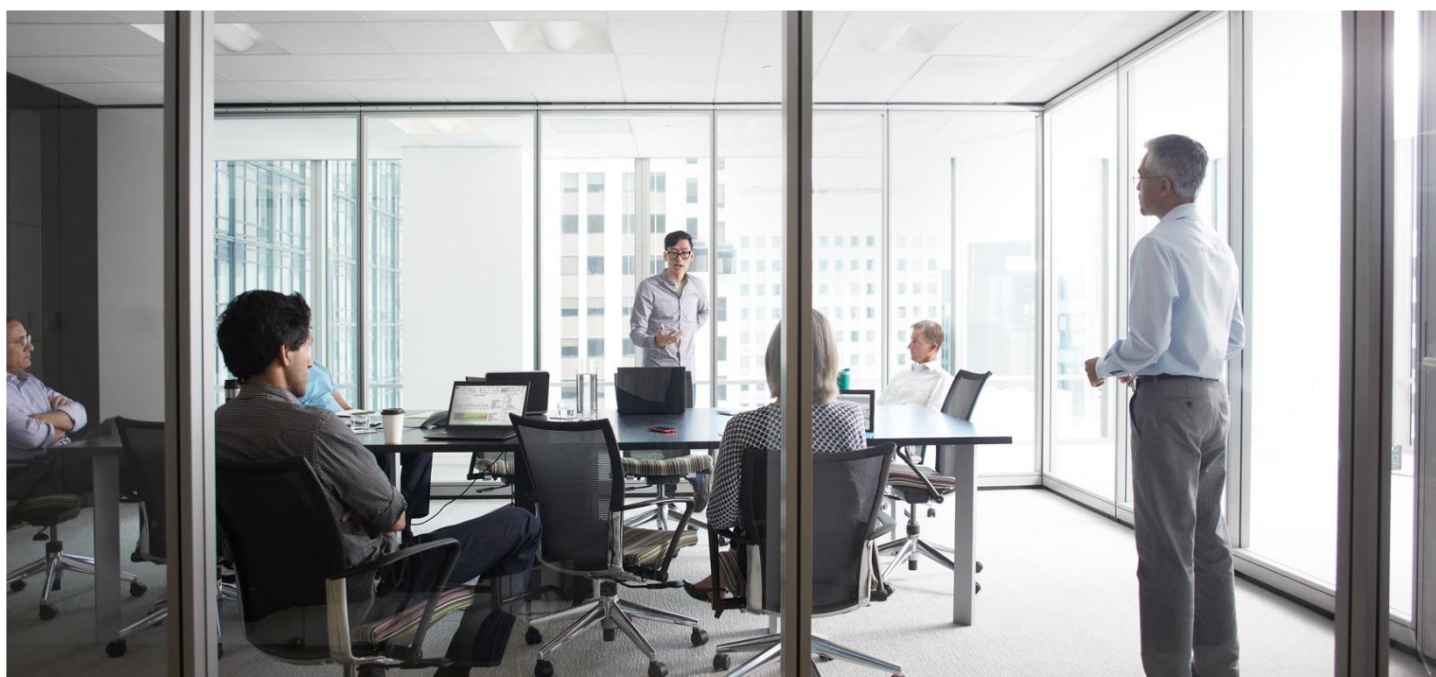
Archive storage has the lowest storage cost and higher data retrieval costs compared to hot and cool storage. This tier is intended for data that can tolerate several hours of retrieval latency and will remain in the archive tier for at least 180 days.

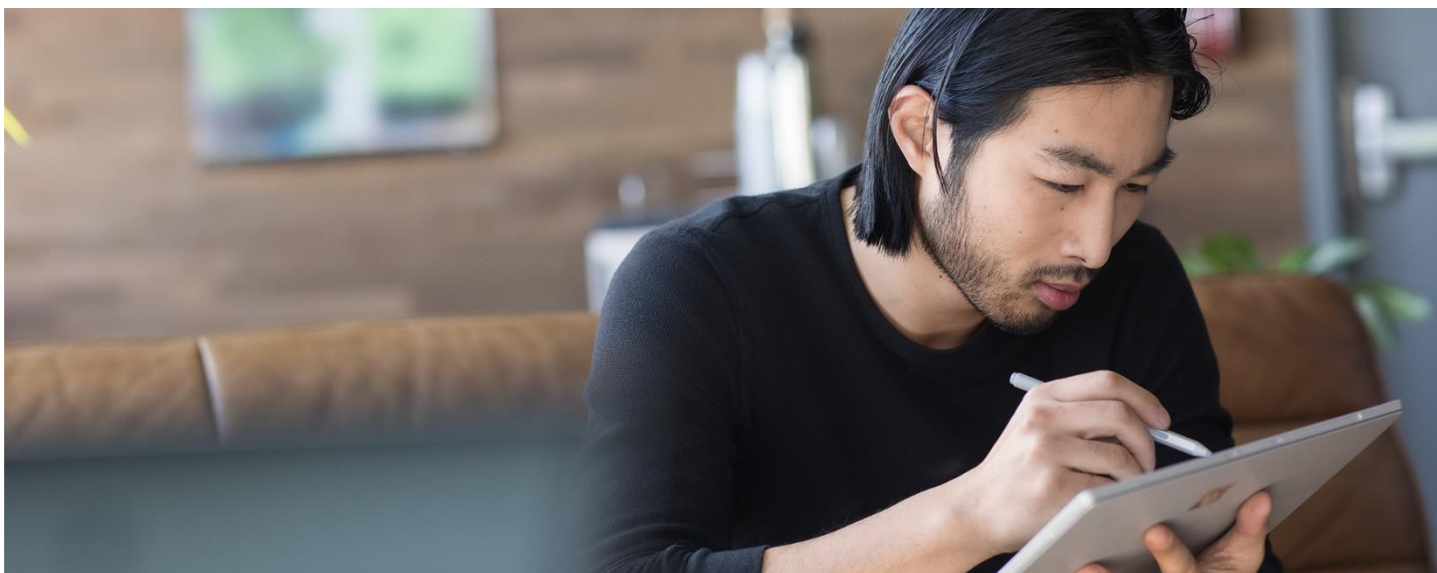
While a blob is in archive storage, it is offline and cannot be read (except the metadata, which is online and available), copied, overwritten, or modified. Nor can snapshots be taken of a blob in archive storage. However, existing operations can be used to delete, list, get blob properties/metadata, or change the tier of the blob.

Example usage scenarios for the archive storage tier include:

- Long-term backup, archival, and disaster recovery datasets
- Original (raw) data that must be preserved, even after it has been processed into final usable form. (*For example*, Raw media files after transcoding into other formats)
- Compliance and archival data that needs to be stored for a long time and is hardly ever accessed. (*For example*, Security camera footage, old X-Rays/MRIs for healthcare organizations, audio recordings, and transcripts of customer calls for financial services)

To read data in archive storage, first change the tier of the blob to hot or cool. This process is known as rehydration and can take up to 15 hours to complete. Large blob sizes are strongly recommended for optimal performance.





Azure Management Best Practices

There are several best practices that should be adhered to when migrating or operating any workload in Azure. Following these established patterns helps avoid common pitfalls and makes it easier to maintain deployments in the long term.

Often, new Azure deployments start off as an experiment from a learning exercise or a proof of concept that was converted into a production workload. When these ad-hoc type deployments occur, it is easy to get started on the wrong foot and implement anti-patterns accidentally.

In this section we'll review several key strategies that can kick migration projects off on the right foot and avoid future problems.

First, we'll take a detailed look at Azure subscriptions. Avoiding subscription bloat while enforcing clear management and security boundaries requires a subscription management strategy. We'll review several different models for subscription management.

Next, we'll consider how resources are organized and named within each subscription. Resources are organized using resource groups, and the correct resource grouping

as a large impact on how the resources are deployed and managed. Resource naming, and the use of naming conventions, is also important to promote consistency and reduce human error during resource management operations.

Organizing resources at deployment is just the start. It's also important to consider how resources will be managed throughout their lifecycle. Accidentally deleting a production resource can create a catastrophic application outage. It's also important to control resource creation to control costs. We'll review the technologies in Azure for controlling the resource lifecycle.

Finally, we'll consider Security and Compliance, and provide links to valuable resources provided by Microsoft that can help ensure that deployments are secure, and meet any local or industry regulations.

Azure Subscriptions

An Azure subscription is the most fundamental entity used to organize Azure resources.



The Azure subscription is the boundary for several independent Azure concepts:

- **Security boundary** - Each subscription forms an administrative security boundary. A subscription administrator has full control over resources within that subscription and controls the subsequent configuration of role-based access control and policies within the subscription. Where controls must be applied to more than one subscription, they must be applied to each subscription independently—there is no control mechanism that spans subscriptions.
- **Service limits** - The subscription is the boundary at which many of the Azure Service Limits are applied. Resource utilization forecasts need to be made at the subscription level to ensure they stay within these limits. For further details on Azure subscription limits, see <https://docs.microsoft.com/azure/azure-subscription-service-limits>
- **Billing** - the subscription forms the billing boundary. All Azure resources within a subscription are billed at the subscription scope. Different billing schemes (such as pay-as-you-go, CSP, or Enterprise Agreement) are applied at the subscription level.

With these points in mind, the Managed Service Providers need to choose how to organize resources across subscriptions. A common model is to use a separate subscription for each application, and to use separate aka.ms/practiceplaybooks

subscriptions for production vs non-production deployments. A single subscription can contain more than one non-production deployment, in separate resource groups. Keep the subscription model simple, whilst retaining essential flexibility.

MANAGEMENT GROUPS

Organizations typically manage multiple subscriptions. For example, different applications are typically deployed to separate subscriptions. Similarly, different environments, such as development, staging, and production, are often separated by using separate subscriptions. In addition, developers often have individual subscriptions for experimentation and learning.

Each subscription requires appropriate governance. This governance includes role-based access controls for team members, control over how each subscription is used (for example, the size and type of VM that can be created), and cost management.

A large organization may have hundreds of subscriptions. Applying and managing governance controls consistently and at scale across multiple subscriptions can require significant effort

Management Groups are an Azure feature designed to help address this challenge. Management Groups can group subscriptions into a hierarchy. Governance controls can be applied to nodes in the hierarchy in a single

operation, and flow down from there into every subscription. The root of the Management Group hierarchy is the tenant root Management Group, of which there is exactly one for each Azure Active Directory tenant. All Azure subscriptions within this tenant must sit within the Management Group hierarchy under this root.

This provides a simple and powerful way to efficiently apply and update governance across Azure subscriptions at scale. As well as being used to apply governance, several other Azure features now support management groups. This includes Azure Cost Management, Azure Advisor and Azure Security Center for visibility into costs, recommendations, and security reports spanning multiple subscriptions.

SUBSCRIPTIONS WITH AN ENTERPRISE AGREEMENT

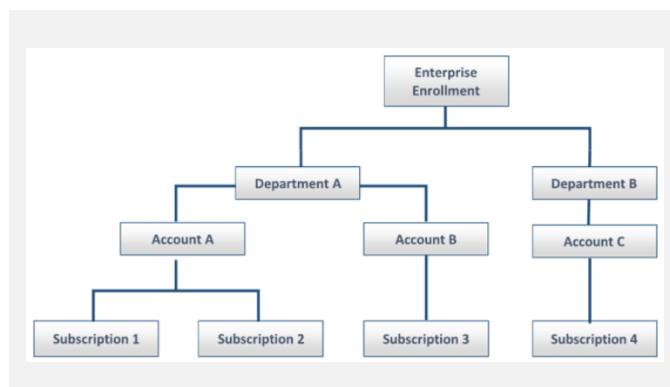
The [Azure Enterprise Agreement portal](#) allows large enterprise customers of Azure to manage Azure subscriptions and associated licensing information from a central portal. Enterprise Agreement (EA) customers can add Azure to their EA by making an upfront monetary commitment to Azure. That commitment is consumed throughout the year by using any combination of the wide variety of cloud services Azure offers from its global datacenters. Within a given enterprise enrollment, Microsoft Azure has several roles that individuals play.

The Enterprise Administrator has the ability to add or associate accounts and departments to the enrollment, can view usage data across all accounts and departments, and is able to see the monetary commitment balance associated to the enrollment. There is no limit to the number of Enterprise Administrators on an enrollment.

Department Administrators can manage department properties, manage accounts under the department they administer, download usage details, and view monthly

usage and charges associated to their department if the Enterprise Administrator has granted permission to do so.

The Account Owner can add subscriptions for their account, update the Service Administrator and Co-Administrator for an individual subscription, view usage data for their account, and view account charges if the Enterprise Administrator has provided access. Account Owners will not have visibility of the monetary commitment balance unless they also have Enterprise Administrator rights.



SUBSCRIPTIONS WITHIN A CSP

The CSP model does not have the concept of departments and accounts. Instead, each CSP subscription is created within a customer. The customer entity contains all the services that have been sold, such as Office 365, Dynamics CRM, and Azure subscriptions.

Within each CSP customer, a Management Group hierarchy can be used enabling consistent governance across subscriptions. However, there is no such facility that spans multiple customers, so each customer will need to be managed separately.

Azure CSP subscriptions are also subject to a small number of technical limitations. See [Migrating existing VMs to CSP](#) for more information.

COMMON SUBSCRIPTION MANAGEMENT STRATEGIES

Common considerations when creating a subscription creation strategy are

- **Subscription Service Limits** – will the migration be impacted by service limits such as number of virtual networks, or ExpressRoute connectivity? What about future growth?
- **Virtual Network Connectivity** – how will resources in each subscription connect to each other? Site-to-Site, ExpressRoute, Virtual Network Peering?
- **Security** – role-based access control is setup per subscription. How will this impact the subscription creation strategy?
- **Chargeback** – how to report and group Azure consumption costs?

The following are some common strategies for creating subscriptions. There is not a single model, each approach comes with its own caveats that the provider should understand and plan for.

Subscription per Department (EA Only)

In this model, each department contains different types of environments (e.g. prod, non-prod) and all Azure resources are created in the same subscription.



Pros:

- Low ExpressRoute circuit costs
- Lower number of overall subscriptions to manage

Cons:

- Granular role-based access control model required to allow permissions for different resources
- Higher risk of subscription limit issues since many potential services could be deployed into a single subscription
- Mistake in management will affect all environments in the department

Subscription per Environment

In this model, each environment contains the different types of applications and workloads.



Pros:

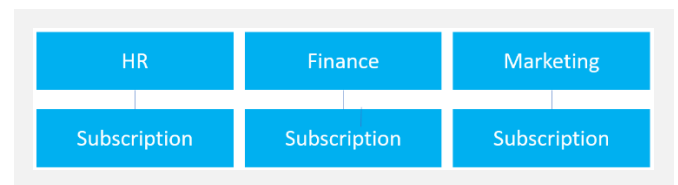
- Minimized risk of running into subscription limit issues
- Virtual network address spaces can be tailored per application
- Minimized risk of impacting one environment by changing the other

Cons:

- New ExpressRoute circuit required per 10th application unless ExpressRoute Premium is used, which has larger limits depending on circuit bandwidth
- More subscriptions to manage (RBAC, Policies, Tagging, Chargeback)

Subscription per Application

Each application uses a different subscription.



Pros:

- Minimal subscription limit issues since each application is in a subscription
- Per application RBAC model

Cons:

- Higher risk of running into cross subscription connectivity limits unless applications are isolated from each other
- More subscriptions to manage (RBAC, Policies, Tagging, Chargeback)

RESOURCES

- ➔ [Azure Onboarding Guide](#)
- ➔ [Azure Subscription Service Limits](#)

Resource Organization

Consistent naming and tagging schemes help organize Azure resources, making management easier and reducing mistakes.

Even a simple application can involve a substantial number of Azure resources, such as virtual machines, network interfaces, IP addresses, storage accounts, virtual networks, subnets, network security groups, etc. Without disciplined and intuitive resource organization, administrative mistakes are inevitable.

NAMING CONVENTIONS

Adopting a systematic and descriptive naming convention for each Azure resource is a useful way of making the purpose of each resource and the relationships between resources clearer. This increases manageability and reduces the likelihood of administrative mistakes.

Well-designed naming conventions can help identify resources in the portal, on a bill, and within scripts. Most likely, the customer will already have naming conventions for their on-premises infrastructure. When migrating to Azure, extend those naming standards to the Azure resources where possible. Naming conventions facilitate more efficient management of the environment at all levels.

Designing a naming convention is not entirely straightforward. Different Azure resource types have different limitations on what names are permitted (alphabet, case sensitivity, and length). Microsoft has [documented these rules and published a baseline recommendation for a naming convention](#) which can be used as a starting point.

RESOURCE GROUPS

A resource group is a container that holds related resources for an application. The resource group could include all the resources for an application, or only those resources that are logically grouped together. The service designer decides how to allocate resources to resource groups based on what makes the most sense for the organization. Since resource groups are commonly used

as a security boundary for role-based access permissions, a good general principle is to group resources that share the same management lifecycle.

With Resource Manager, application designers can create a simple template (in JSON format) that defines deployment and configuration of entire application. This template is known as a Resource Manager template and provides a declarative way to define deployment. By using a template, partners can repeatedly deploy the application throughout the app lifecycle and have confidence that resources are deployed in a consistent state. We'll cover more about writing [templates](#) later. Since each template must be deployed to a single resource group, the resource groups structure also impacts the deployment template design. Deployment to multiple resource groups can be achieved at the expense of the additional complication of breaking the deployment into separate, nested templates.

TAGGING RESOURCES

In addition to a naming convention, [resource tags](#) can also be used to organize Azure resources. Tags are key-value pairs, with a maximum of 15 tags per resource or resource group. Tags can be used for multiple purposes—common scenarios include:

- **Azure billing** - supports filtering and grouping using Tags, enabling tags to be used to implement internal charge-backs or billing codes ('billTo=IT-Chargeback-1234')
- **Associating resources** - for a particular application across resource groups ('application=payroll')
- **Resource owner** - Identifying the owner of a particular resource ('managedby=joe@contoso.com')
- **Environment** - Identifying the environment of a particular deployment ('environment=staging')

Controlling access to Azure Resources

With great power, comes great responsibility. Just as the cloud allows services to be provisioned in minutes, critical services can just as easily be changed or deleted. Fortunately, Azure provides comprehensive features to control access without losing agility.

A typical application will only use a small subset of the wide range of resource types available in Azure. Allowing unrestricted creation of resources allows consumption of resources far beyond what was planned. This can cause unexpectedly high bill or even affect the correct operation of the application (for example, by causing subscription limits to be breached).

Similarly, incorrectly modifying or accidentally deleting a resource can cause an application outage. It is therefore important to control who has access to Azure resources, especially those supporting production applications, and to control what operations are permitted by those who do have access.

To summarize, the main governance challenges faced by an organization running production services in Azure are:

- Prevent unexpected costs by controlling which resources are deployed
- Control who has access to Azure resources, and what changes they are permitted to make
- Separate the control of production from non-production environments
- Prevent accidental resource changes or deletions that are potentially service impacting

Azure provides three complementary mechanisms to control what resource management operations are permitted in a subscription, who has permission to perform those operations, and to prevent accidental mistakes. These are known as policies, role-based access control, and resource locks.

- [Azure Resource Manager Policy](#) helps define rules or conventions that must be applied to resources within a subscription. For example, a policy may block the use of certain types of resource or limit the SKUs or service tier of another resource type, or enforce or prevent the use of a particular Azure region. Policies can also be used to enforce rules, such as the inclusion of a tag to enable billing cross-charging or enforcing

a naming convention. Policies can be applied to the entire subscription or scoped to a resource group or resource.

- [Role-based access control \(RBAC\)](#) allows control over the actions of a user or group of users. Each role defines what actions are permitted, or not permitted, on what types of resource. Once a role has been defined, it can be applied to specific users or groups of users (as defined in Azure AD), and applied at the subscription, resource group or individual resource scope.
- [Resource Locks](#) allow certain potentially damaging management operations to be blocked. There are two types of lock: 'DoNotDelete', which permits all actions except deletion, and 'ReadOnly', which permits read operations but blocks any updates or deletes. For example, a 'DoNotDelete' lock may be applied to a storage account containing critical data to prevent accidental deletion, regardless of the role-based access permissions granted to the user. The operation can only proceed if the lock is first removed, which provides a useful checkpoint to verify that the operation really is intended. Adding and removing locks themselves are controlled by separate RBAC permissions, thereby providing role separation and enabling operator activities to be reviewed. Whilst a single lock can be applied at the subscription or resource group scope, they are typically applied to individual resources, allowing more fine-grained control.

Azure Policy, RBAC and locks are complementary. Policies focus on the resources in a subscription, regardless of the user initiating each request. RBAC works at the user level, controlling which operations are available to each user. Resource locks prevent accidental modification or deletion of individual resources. These three mechanisms work together to provide a powerful toolkit to control resource access whilst permitting necessary management operations under suitable controls.

Security and Compliance

Security is a critical requirement for cloud applications. Applications cannot be migrated to Azure unless there is a high degree of confidence in the security of the migrated application and all application data. A strong security stance and deep security skills are therefore essential for any successful Managed Service Provider offering Azure services.

Security in the cloud is a shared responsibility between Microsoft and the Azure customer developing and hosting applications in Azure. Microsoft makes substantial investments in the physical, logical and operational security of its infrastructure, networks, and software. Azure provides a wide range of features and tools which can be used to secure applications hosted in Azure. It is the responsibility of the Managed Service Provider to understand and make appropriate use of Azure so that the applications they manage are properly protected.

Security is a specialist topic, requiring both broad and deep technical skills. Technical teams require appropriate experience and training to properly design, implement, and manage secure Azure solutions. Both the threat landscape and the security features available in Azure are continually evolving. On-going investment should be made to review and update the security protections of existing applications, and to ensure staff skills stay up-to-date.

Microsoft Azure supports an extensive range of features and services which can be used to secure Azure-based applications, and to provide secure connectivity to on-premises environments where required. In addition, Microsoft publishes extensive guidance on security best practices.

Use the following resources to deliver secure and compliant customer solutions:

- The [Microsoft Trust Center](#) is a hub for security, compliance and privacy guidance from Microsoft. This includes all aspects of Microsoft's software portfolio (both on-premises and cloud), as well as guidance for third parties on how develop their own applications securely. It also has a section dedicated to [Azure Security](#).
- The [Azure Trust Center](#) provides links to additional resources on security, compliance and privacy in Azure.
- The [Azure Security documentation hub](#) contains both an overview of the full suite of security features available in Azure, as well as comprehensive guidance on specific areas such as networking, storage, compute, identity, and operations. Best practice guidance is also included, as well as links to additional resources.
- The [Security best practices for IaaS workloads in Azure](#) documentation page contains further useful guidance on secure Azure deployments.

The [Microsoft Cloud Architecture resources](#) includes a series of posters providing a technical overview of core Azure features areas aimed at Enterprise Architects. This includes guidance on networking, hybrid cloud, storage, identity, mobility, and security.

The [Security Practice Development Playbook](#) provides further guidance for Microsoft Partners seeking to build a practice with a specialist focus on security.

Infrastructure as Code

Automate infrastructure deployment and management through declarative or imperative scripts.

Infrastructure as Code (IaC) is the process of writing scripts to automate the deployment and configurations management of infrastructure. Using automation to manage their infrastructure enables partners to:

- Increase accuracy and reliability of resource deployment and configuration.
- Automate the process of replicating environment configurations across dev, test, and production environments.
- Add version control to infrastructure configuration management.

Automation is also a key component to implementing DevOps practices, and Infrastructure as Code fits in well with the other aspects of DevOps such as Continuous Integration (CI) and Continuous Deployment (CD).

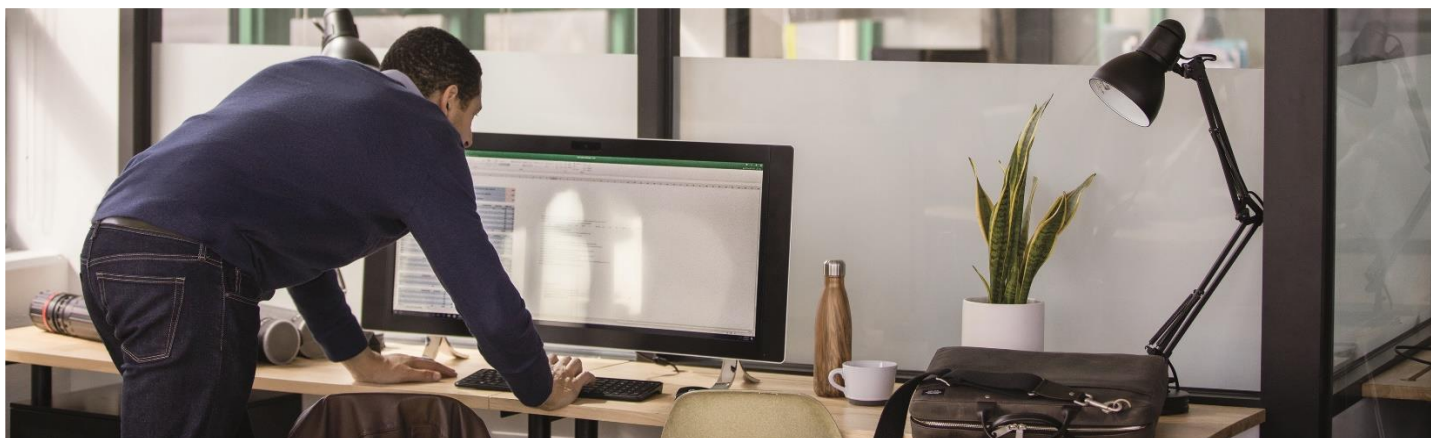
There are two methods to writing scripts for implementing Infrastructure as Code: **Imperative** and **Declarative**. The imperative approach utilizes traditional command-line scripts to define the step-by-step process to modify current state to the desired end-state. The declarative approach utilizes a definition file that declares what the desired end-state is, and the tooling automatically figures out how to modify the current state to reach the desired end-state.

With the infrastructure deployment and configuration automated with scripting, these scripts can then be

checked into Source Control, such as Git or Visual Studio Team Services. This integration with Source Control adds the ability to affectively track infrastructure version changes over time in the same manner that all other source code is tracked. It also enables Infrastructure as Code (IaC) to be integrated into the release pipeline through Continuous Integration (CI) and Continuous Deployment (CD); alongside the source code for the enterprise applications that may run on the infrastructure being automated.

In this section, we'll review three approaches to automating infrastructure provisioning and management:

- [Azure Resource Manager Templates](#): Declarative templates formatted in JSON to define the configuration of an Azure deployment.
- [Blueprints](#): Declarative management of entire environments, from initial deployment to ongoing updates. Each blueprint combines Resource Manager templates, role-based access control assignments and Azure policy assignments into a single declarative object.
- Automated scripts: Create imperative scripts using [Azure PowerShell](#) and [Azure CLI](#) for managing Azure resources.
- [Azure SDKs and REST APIs](#): Build management tools by leveraging the Azure SDKs and REST APIs to automate Azure resource management.



Azure Resource Manager Templates

Implement declarative Infrastructure as Code using Azure Resource Manager

Azure Resource Manager Templates offer a **declarative** method of implementing Infrastructure as Code (IaC) for deploying and maintaining infrastructure and application deployments. Put simply, a template consists of a text file, in JSON format, specifying the resources in a given deployment.

Templates can be parameterized, allowing a small number of inputs to control the type, number and size of resources deployed. This allows the same template to be used for multiple deployments in different environments, such as Test, Pre-Production and Production.

Consistency across the Azure ecosystem allows for Resource Manager Templates to be written that can deploy applications to both the Azure public cloud and Azure Stack without requiring any changes to the template.

Templates can be authored using a plain-text editor, or a variety of IDEs (Integrated Development Environments), such as Visual Studio 2017 and Visual Studio Code. These editors enhance the authoring experience with syntax highlighting, code completion, and other common IDE features.

Templates can be easily deployed, either directly from Visual Studio, using the Azure portal, PowerShell or CLI, or even integrated into a Continuous Integration / Continuous Delivery (CI/CD) pipeline using Visual Studio Team Services, Jenkins, or some other automated build and deployment tool.

Templates support two deployment modes, incremental and complete. In both modes, all resources specified in the template are deployed. The differences are in what happens to pre-existing resources that are *not* specified in the template—in incremental mode, they are unchanged, whereas in complete mode, they are deleted. This enables templates to be used both for clean deployments and to update existing deployments.

Developing Azure Resource Manager Templates is a skill. The template language supports a wide range of features, some of them quite advanced such as conditionals and nested templates. Learning to use the full power of Templates is a highly worthwhile investment for any team making significant use of Azure. Several guides and samples are available to help—see the links in the Resources section below.

TEMPLATE AUTHORING TOOLS

- [Azure Resource Manager Tools for Visual Studio Code](#). Visual Studio Code (VS Code) is a free code IDE from Microsoft. The Azure Resource Manager Tools is an Extension to VS Code that adds syntax highlighting, autocomplete, and other features to the IDE for authoring ARM Templates with ease.
- [Visual Studio Azure Resource Group project](#) The Azure Resource Group project (deployed as part of the [Azure SDK](#)) template adds full IDE support to Visual Studio 2015 and 2017 for authoring and deploying ARM Templates directly within Visual Studio.
- [Azure portal](#) Templates can also be loaded into the Azure portal for easy deployment. The portal also supports editing templates online and downloading templates for editing offline. In fact, every deployment made using the Portal uses a template, and partners can download these templates from the portal as a starting point for creating their own.

RESOURCES

- [Azure Resource Manager Overview](#)
- [Authoring Azure Resource Manager Templates](#)
- [Azure Quickstart Templates](#)
- [Sample templates from the Azure Resource Manager team](#)
- [Create and deploy your first Azure Resource Manager template](#)

Blueprints

Automated deployment entire environments, with full lifecycle version management.

As we have just seen, Azure Resource Manager Templates provide a powerful means of deploying applications to Azure. By using a declarative model of all the required Azure resources, they enable the engineer to focus on the goal state of the deployment, letting Azure take care of the deployment process. Integrating these templates into source control enables automated deployment as part of a Continuous Integration / Continuous Delivery pipeline.

However, templates also have their limitations. First, the template itself is created and managed outside Azure. There is no central repository where all templates must be stored. Second, there is no tracking of deployments, no way to answer the question 'where has this template been deployed?'. Third, there is no version control, no way to ask 'which version of this template was deployed here?' Fourth, there is no control over the resources deployed by the template once the deployment completes. For example, a template might define the best-practice configuration for a VPN gateway, but nothing stops the gateway being modified and diverging from this best practice.

BLUEPRINTS

Azure Blueprints are designed to address all these concerns. Like templates, blueprints are a declarative model for defining an Azure deployment. However, there are some important differences. Whereas templates are designed to deploy an application, blueprints are designed to deploy an entire environment. This environment can comprise multiple resource groups,

resources (specified using templates), role-based access control assignments and Azure policy assignments. All these elements are included in the Blueprint.

TRACKING AND VERSIONING

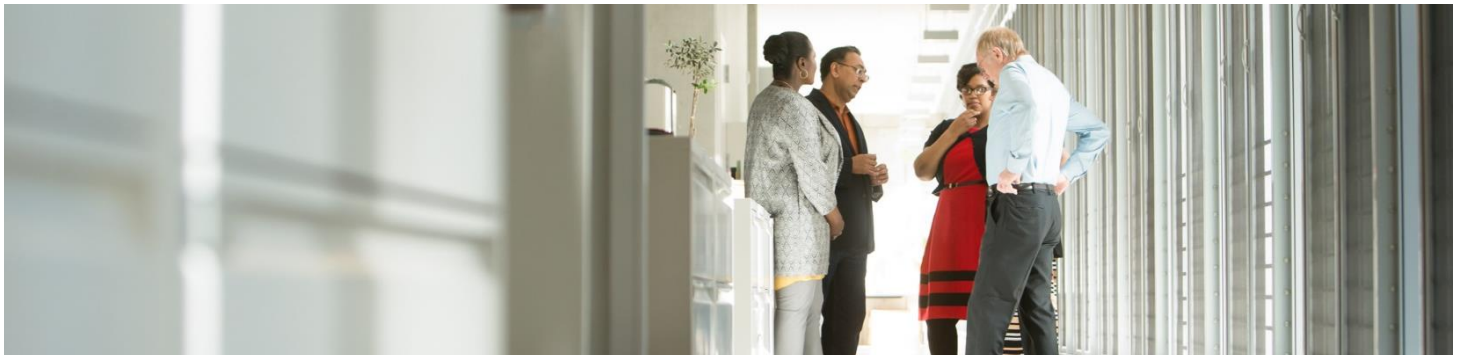
Blueprints are stored in Azure, with a full version lifecycle. Each deployment (or *assignment*) of each blueprint is tracked, to easily identify every current blueprint deployment. This tracking includes the blueprint version that is currently deployed. Upgrades are managed by creating new blueprint versions and then updating a blueprint assignment to use that new blueprint version.

BLUEPRINT LOCKS

Blueprints also support resource locking. As with Resource Manager resource locks, there are two lock types: DoNotDelete and ReadOnly. However, the locking implementation with blueprint locks is quite different. Resource Manager locks protect against accidental changes, requiring the administrator to first remove the lock before making a change. Blueprint locks are stricter—the lock **cannot** be overridden, even by the subscription owner. The only way to modify the resources is by updating or removing the blueprint assignment. This makes blueprint locks suitable for publishing best practice deployments safe in the knowledge that the best-practice configuration will always be preserved.

RESOURCES

→ [Azure Blueprints Overview](#)



Automated Scripts

Implement Imperative Infrastructure as Code (IaC) using Command-Line scripts

[Azure PowerShell](#) and [Azure CLI](#) (cross-platform command-line tool) offer an **Imperative** method of implementing Infrastructure as Code (IaC) for deploying and maintaining environment and infrastructure deployments with the Microsoft Azure ecosystem. These tools are designed for managing and administering Azure resources from the command-line, and for building automation scripts that work using Azure Resource Manager.

Automation scripts written using either tool can easily be checked into Source Control, such as Git or Visual Studio Team Services, for added version control. The scripts can even be integrated into a Continuous Integration / Continuous Delivery (CI/CD) pipeline using Visual Studio Team Services, Jenkins, or some other automated build and deployment tool.

Azure PowerShell and Azure CLI can both be downloaded and installed from the [Azure downloads page](#). They are also available via the [Microsoft Web Platform Installer](#). The tools are frequently updated, and it's worthwhile to make have an up-to-date copy to access the latest Azure features.

In addition to executing commands at the command-line of the local machine, the Azure Cloud Shell within the Azure Portal can be used to execute Azure PowerShell and Azure CLI scripts directly within a web browser from any machine. The Cloud Shell is accessed by clicking the Cloud Shell icon in the toolbar at the top of the portal, or as a standalone page at <https://shell.azure.com>.



Azure SDKs and REST APIs

Custom code can be built to automate Azure resources, too.

The Azure SDKs (Software Development Kits), in addition to templates and command-line tools, can be used for both working with Azure Resources, as well as implementing the automation of deployment, configuration, and management of Azure Resources. These tools help add additional automation capabilities to grant the capacity to work with and automate resources in Microsoft Azure.

The Azure Development SDKs are a set of reusable libraries built by Microsoft that work with Azure Resource Manager to manage Azure resources. These SDKs enable any custom automation scenario to be built out, in addition to the integration of Azure services with custom code. This allows partners to use or build exactly the right tool to solve the problem at hand.

These libraries support a wide array of programming languages and platforms, including:

- .NET Framework
- .NET Core
- Java
- Node.js
- PHP
- Python
- Ruby
- Android
- iOS

Partners can also manage Azure resources by calling the Azure Resource Manager REST APIs directly, using their own code instead of the SDKs. These APIs are the foundation of all resource management in Azure—indeed, the Azure portal, PowerShell, CLI and SDKs all use these Azure REST APIs. Using the REST APIs gives you direct access to all Azure resource management features.

RESOURCES

- ➔ [Azure SDK Downloads](#)
- ➔ [Azure REST API Reference](#)



Playbook Summary

We hope this playbook has provided new insight into the business opportunity that application migration and modernization offers, and valuable guidance on how to execute a migration or modernization project and build a cloud migration practice.

We created this playbook to help our MSP partners adapt their businesses to embrace the opportunity to migrate customer applications to run in Azure.

In the first section, **Define Your Strategy**, we detailed the unique business opportunity, and some of the ways to differentiate their business, and leverage Microsoft incentive programs to boost revenue.

In the next section, **Hire & Train**, we explained the skills needed and provided guidance on how to evaluate new hires or up-skill existing staff to meet those needs. We also provided several training resources to use, both online and instructor led.

We then moved to the technical content, which forms the bulk of this playbook, starting with a chapter on **Building a Migration Assessment**. This included detailed guidance on the three stages of the assessment process—discovery, planning and evaluation-- , and the expected contents of the resulting migration assessment plan.

Next, we discussed migration execution, starting with **Migrating to Azure Infrastructure Services**. Here we provided guidance on the various Azure services to use to build a cloud-based infrastructure. We also discussed the tools available to assist with the migration process, and some of the technical challenges and option in specific cases, such as migrating databases.

We then moved on to discuss **Modernizing Applications**, use Azure's platform services. We started with the business benefits of embracing a platform approach, before drilling deeper into the design principles behind 'cloud-native' applications and some of the technologies available in Azure to implement them.

In the **Optimize & Manage** chapter, we focused on cost optimization, discussing the various programs and techniques available to reduce Azure spend. Reducing Azure spend is one of the most effective ways that a Managed Service Provider can demonstrate clear value to their customers.

In **Best Practices**, we discussed a variety of proven techniques to make Azure usage easier to manage. By organizing subscriptions and resources, and using the range of techniques provided to control access, partners can greatly improve the long-term manageability of their deployments and reduce the risks associated with service management mistakes.

Finally, in **Leveraging Reusable IP**, we discussed the importance of developing repeatable processes and tools to help drive efficiency, quality, and time to completion in any migration project.

FEEDBACK

Please share feedback on how we can improve this and other playbooks by emailing playbookfeedback@microsoft.com.

For more information please contact Strategic Communications by visiting www.yourstrategic.com/contact-us for more information!

