

Making the most of the cloud everywhere



Information in this document, including URL and other internet website references, is subject to change without notice. Unless otherwise noted, the companies, organizations, products, domain names, email addresses, logos, people, places, and events depicted in examples herein are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2019 Microsoft Corporation. All rights reserved.
Microsoft, Azure, Active Directory, Dynamics, ExpressRoute, Hyper-V, SQL Server, SharePoint, Windows, and Windows Server are either registered trademarks or trademarks of the Microsoft group of companies.

Making the most of the cloud everywhere: Unifying development and modernizing practices in hybrid environments

01 /

Innovations unlocked
by the cloud

02 /

Benefits of developing
with the cloud

03 /

People, process,
and DevOps

04 /

New architecture
and tools

Resources

Introduction

Your company has made significant investments in information technology. For mature companies—those existing in the pre-cloud era or those operating in a highly regulated field—it's likely that many of these investments exist primarily on physical servers, in a data center, or in similarly disconnected or “non-cloud” infrastructure.

Significant money and resources go into securing these IT investments, and you're responsible for making sure they continue to run smoothly. However, the more time you spend on keeping your current systems running, the less time you can dedicate to transforming IT.

The cloud can help companies with their push toward transforming their operations. Yet the transformation takes time and resources, which for many companies leads to business operations and a development process that must span both on-premises and in-the-cloud infrastructure.

This hybrid approach is fairly common. In a recent survey, 67 percent of mid- to large-size companies are already operating, or planning to operate, in a hybrid cloud environment.¹ Many companies have migrated business functions to cloud computing in a piecemeal way, because moving an entire infrastructure and every service to the cloud in a single massive project is unrealistic. This poses a fundamental issue for IT teams: existing business assets that cannot currently transition to public cloud risk being alienated from the agility and innovation promised by a shift to cloud computing.

¹By the Numbers: Hybrid Cloud (Redmond, WA: Microsoft, 2018) Infographic. Accessed 4 Sep 2018. <https://azure.microsoft.com/resources/hybrid-cloud-by-the-numbers/>

What is hybrid cloud infrastructure?

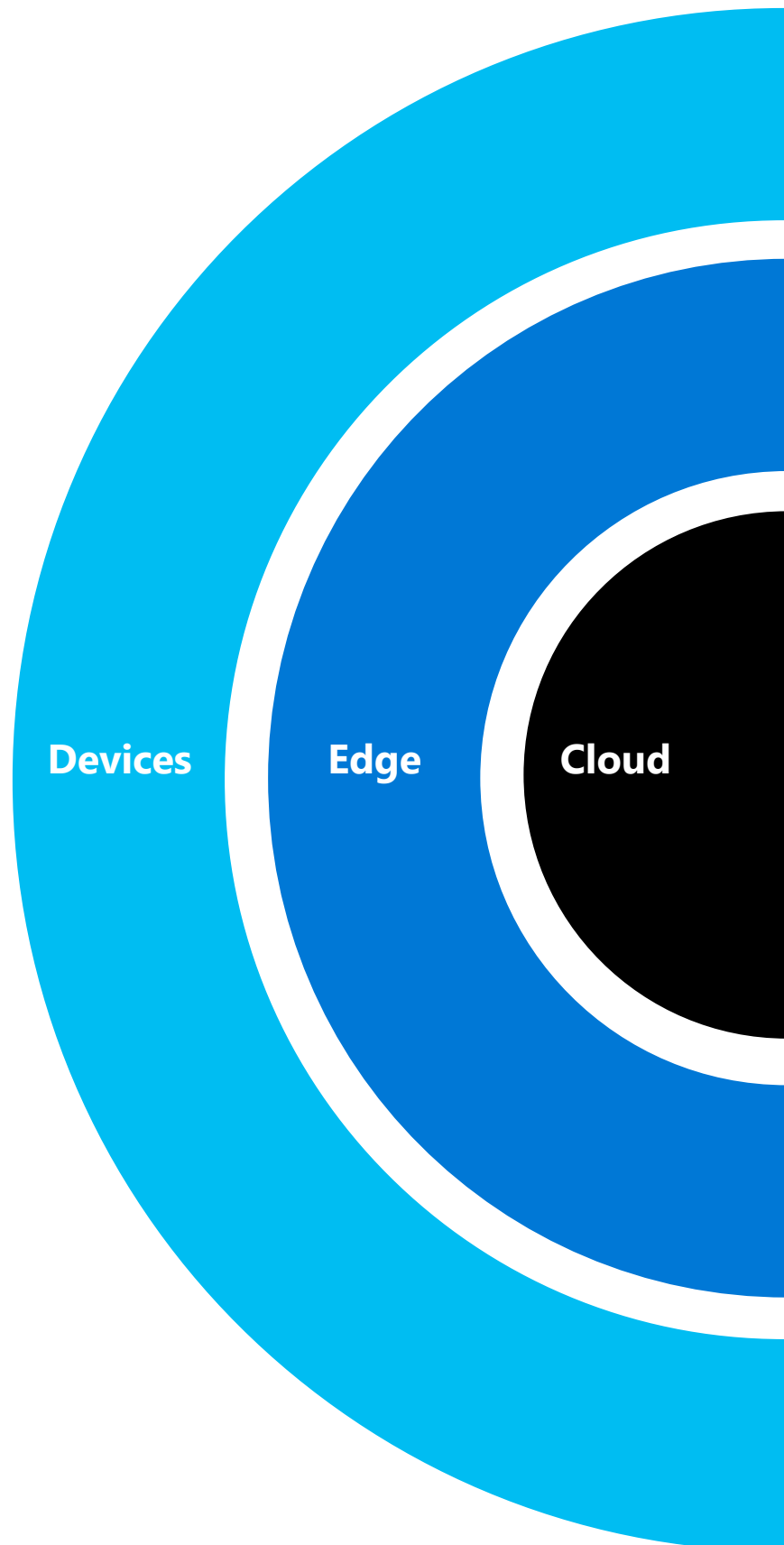
Hybrid cloud infrastructure is a natural evolution of information technology where companies transition some of their hardware and software to cloud services and technologies. As a result, their cloud computing environment combines on-premises, private cloud, and public cloud services, such as software as a service (SaaS) and platform as a service (PaaS). A key challenge is providing a truly integrated solution across these two environments for their users, developers and administrators. To help businesses deal with the blended infrastructure, certain technologies have been developed to help manage and maintain environments which span these on-premises and cloud infrastructures.

Companies with legacy IT assets—or those with complex regulatory or edge computing requirements—can struggle to keep up with the pace of innovation at companies that are able to adopt cloud computing from their inception. Yet remaining flexible and agile within a hybrid environment isn't intuitive, and finding ways to build and maintain solutions across multiple environments creates costs, requires specific skills and can tax management.

This e-book will help you take a fresh look at your IT portfolio through the lens of hybrid cloud, highlighting the key innovations needed to understand when building cloud-native services and applications and how to apply these to your overall IT estate. You'll also be able to better manage your existing assets while still improving your adoption of cloud technologies to ultimately help drive business transformation.

What is edge computing?

Edge computing encompasses systems and processes that are isolated from the internet, either because of intermittent connectivity or for security and regulatory requirements. Edge-computing devices pose a challenge for cloud-infrastructure models because the devices are usually disconnected from the network, and so cannot benefit from the regular updates and monitoring that help manage other cloud-connected devices.



01 /

Innovations
unlocked by
the cloud

A decade ago, companies either deployed their services to an online server connected to the internet or as a standalone piece of software to be installed on an endpoint system or internal server. Likewise, those companies' developers immersed themselves in either creating software for a specific on-premises platform or for a web application server. Today, this legacy, bimodal approach to development results in silos that prevent integrated and unified workflows from efficiently creating software.

With the rise of cloud computing, however, developers gain new processes and tools to integrate infrastructure with cloud-based applications and create software more flexibly and efficiently. Infrastructure is part of the application, and by adopting cloud, developers can remove the silos around development, deployment, and operations. Where there was once an application and a server, there is now a solution.

The cloud has changed the development process in three major ways: how we buy things, how we build things, and our tools and techniques. Rather than buy infrastructure with huge capital outlays, cloud computing allows infrastructure to be paid as operational expenditures, giving businesses greater flexibility. By eliminating the need to order, configure, and manage physical servers and data centers, the cloud gives companies the ability to scale infrastructure very quickly and potentially reduces costs.

Next, rather than creating silos between developers and operations, cloud computing allows software to be created, tested, deployed, and maintained in an integrated way. Integrating cloud into the development process is an important step toward unifying the business, development, and operations teams. This integration makes the cloud more than just a part of the solution or infrastructure, but an integral part of the application development process as well.

Finally, the tools and techniques used to develop and deploy software and services are more unified, allowing developers to take control of more aspects of their workloads and applications.

Challenges

To gain the most benefit from your mixed environment, you should take stock of your current infrastructure and prioritize the assets and services that need to move to the cloud. For many migrations, the requirements of being flexible while simplifying development and unifying various pipelines poses challenges. Expect additional challenges if you:

- Have significant on-premises investments and legacy applications but want or need to innovate just like a born-in-the-cloud company
- Are not prepared for failure—the top reasons for project failure are a change in the organization's priorities, a change in the project objectives, or inaccurate requirements gathering
- Have not mapped the relevance of IT to the business and identified where it can enable growth and evolution
- Expect all assets to be moved to the cloud—while you don't want to be shepherding things that are constantly in danger of being outdated, be strategic about what you lift and shift versus applications you rewrite or leave as-is

Development and deployment of software and services will become **faster and more flexible**. Plus, standardizing on shared tools will **unite your development and IT operations teams** to make the largest impact.

Innovations

When you successfully migrate your operations to the cloud, you should see a number of advantages. With a cloud-focused approach that augments on-premises technology, you should see that:

- Operations will have greater control over applications, because of cloud-native technologies such as containerization and software-defined networking
- Teams will take ownership of each product, from development to operations, especially if they are educated on the entire pipeline and trained as cross-functional groups to have a better understanding of their part in software delivery
- Development and deployment of software and services will become faster and more flexible, and standardizing on shared tools will bring your developers and IT operations teams together for the biggest impact
- Costs will shrink if the business is focused on managing its use of cloud services

Resources

Learn about Hybrid cloud solutions on Azure:

<https://azure.com/hybrid>

Modernize your customer's legacy applications: <https://blogs.technet.microsoft.com/msuspartner/2018/02/13/modernize-your-customers-legacy-applications/>

Microsoft Azure Stack:

<https://azure.microsoft.com/overview/azure-stack/>

Azure Strategy and Implementation Guide: https://azure.microsoft.com/mediahandler/files/resourcefiles/azure-strategy-and-implementation-guide/Azure_Strategic_Implementation_Guide_for_IT_Organizations_New_to_Azure.pdf

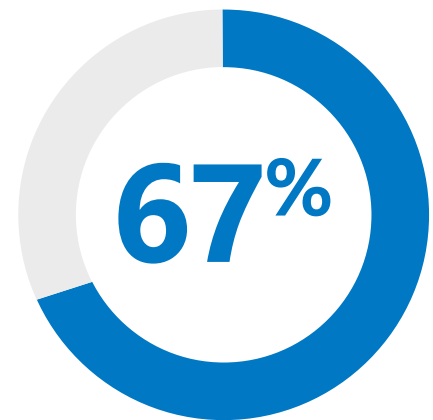
02 /

Benefits of developing with the cloud

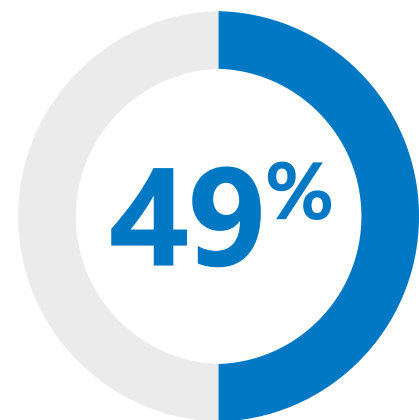
Companies that create collaborative partnerships between their IT and business groups have a strategic advantage over their competition. Using cloud infrastructure helps deliver reliability, accessibility, and future innovation. Unifying development between on-premises programmers and cloud infrastructure is essential to creating agile software pipelines that bring together business decision makers, developers, and IT operations.

It's an idea that has caught on. Two-thirds of companies have adopted a hybrid approach, with 49 percent of companies doing their first cloud deployment in the past year.² allowing access to resources globally, more flexible procurement, and the rapid development of new technology and software. By eliminating the need to order, configure, and manage physical servers and data centers, cloud infrastructure gives companies the ability to scale resources and services very quickly, potentially reducing costs.

Incorporating cloud components enables you to take advantage of more agile and consistent development practices—empowering your organization to move from a position of playing catch up with your apps to fueling growth and leveraging significant IT investments and applications as a competitive advantage.



**of companies have adopted
a hybrid approach**



**of companies did their
first cloud deployment
in the past year**

²By the Numbers: Hybrid Cloud (Redmond, WA: Microsoft, 2018) Infographic. Accessed 4 Sep 2018. <https://azure.microsoft.com/resources/hybrid-cloud-by-the-numbers/>

Capital expenses versus operational expenses

One of the fundamental changes that cloud computing brings to business is the transformation of physical assets requiring upfront expenditures, known as capital expenses, into virtual assets that can be quickly created as needed and paid for through subscription as an operational expense.

Challenges

In order to reap the benefits, however, your company needs to transform its existing business infrastructure and processes to one or more cloud platforms, while strategically enabling the power of cloud computing and next generation practices across its corporate assets. Unlocking that agility requires application modernization—allowing software to run regardless of where the infrastructure resides, while avoiding a lack of functionality and stagnation from solutions that are not cloud-first.

A number of factors can hinder your company from fully benefitting from cloud-based development efforts:

- Security considerations and regulatory compliance can prevent your infrastructure and products from being deployed to the cloud
- Most companies don't know all the services and methods used by their developers; this "shadow IT" used by teams for new application development may present security issues and a sprawl of assets that makes operations expensive, fragmented, and risky
- Learning cloud-based development methodologies, software, and toolsets requires an investment of both time and resources

Considerations

Modern development focuses on loosely coupled applications and microservices, providing the ability to deploy software to consistent hardware or virtual platforms, and standardizing processes and configurations. Core applications should be developed for cloud and moved into a cloud-native environment, while legacy workloads should be moved when they can be easily transitioned.

Other considerations to fully benefit from moving applications to the cloud:

- Move applicable software into cloud-based containers or virtual machines to ease management and benefit from the instrumentation provided by such cloud-based infrastructure
- Often, an application needs to be deployed into different use cases with different requirements—for example, internal users and external partners may need to use the same application but access it in different ways, or various countries may have different requirements around how certain data is handled
- Moving from the silos of development and operations to unified development requires a comprehensive approach to the software and cloud-service lifecycle

Solutions

The one-to-many app deployment approach with Azure ensures you can deploy the same application in different environments with little to no code changes:

- Azure Security Center helps companies monitor their applications and detect anomalous behavior
- Software can be configured for different environments using the Azure Service Fabric toolset—configurations, parameters, and clusters can all be managed using the service
- Patch deployment helps companies stay in compliance by ensuring that every instance of an application is up to date
- Azure meets a variety of international and industry-specific compliance standards, such as ISO 27001, HIPAA, FedRAMP, SOC 1, and SOC 2

03 /

People, process, and DevOps

As you explore how the cloud improves the way software is built, it's important to devote some attention to the practice of DevOps, and how it can help improve innovation. DevOps integrates the development, deployment, and management of applications by leveraging the cloud, orchestration tools, and agile-development methodology to create a more flexible and fast-moving software lifecycle.

Agile development consists of quick iterations of planning, building, deploying, and monitoring applications. Any of these stages could be partially or wholly cloud-based, even when the target is a disconnected system in a high-security facility or sensitive business environment.

DevOps makes IT operations a fully collaborative partner of the software development process. They are equally represented, with no alienation between groups but instead as a partnership that's lasting, durable, but also flexible enough to accommodate change.

Yet, to do DevOps right, your team needs to build and test its software in the same environment where the code is going to run. Once your company commits to a unified development environment, DevOps can help you speed up your testing and deployment of software. With shared responsibility and toolsets, developers and operations can work as a tight team to deliver portable software more quickly.

What is DevOps?

DevOps brings together developers and IT operations groups into an integrated software development, deployment, and management lifecycle. By getting rid of the traditional silos and emphasizing collaboration and shared responsibility, DevOps focuses on delivering user requirements with fast iterative development and the ability to quickly update the product.

Challenges

Bringing together developers and IT operations in a hybrid cloud environment requires the right tools and good process design. Code needs to be automatically tested before deployment. In addition, your team will have to learn new skills. Application security teams need to focus on building secure development environments and automate the testing of code.

Adopting DevOps in your software development lifecycle can pose challenges, especially if you have significant on-premises investments and legacy applications:

- Failures will happen and can derail any hybrid application development initiative unless both business and development teams have committed to adopting DevOps. Moving from the silos of development and operations to unified development requires a comprehensive approach to the software and cloud-service lifecycle.
- While hybrid application development will simplify overall IT in the future, the challenges lie in getting the two sides of DevOps (developers and IT ops) to collaborate and execute in the most efficient manner.
- Adopting DevOps requires development and operations teams to be invested in the process and have a desire to learn new things. The path to effective DevOps adoption is not always straightforward; it is often slow, and individual projects can run into unforeseen hurdles.

Considerations

Automating the testing, deployment, and monitoring of software is a significant force-multiplier of DevOps that helps companies quickly change, build, and deploy code. Companies need to make the investment of time, money, and commitment to bringing every software project into the DevOps pipeline.

- Iterative education and development are key to building DevOps maturity across the software lifecycle. Companies should commit to continuous learning and investment in their developer and IT operations teams.
- Commit time and money to creating DevOps practice; invest in learning, configuration, and security requirements.
- Empower developers to deploy their software with production-like environments. An effective DevOps pipeline relies on simplifying the number of tools and processes. As much as possible, build a common software pipeline with common tools to maximize collaboration between developers and operations.

Solutions

Microsoft has focused on making Azure a natural foundation for any DevOps initiative. Azure gives companies the ability to automate continuous integration and deployment using any DevOps tool—such as Jenkins, TravisCI, and Docker—while targeting any Azure service. Azure Resource Manager gives companies the ability to monitor and control the security of development and test services.

- Visual Studio Team Services has a continuous integration and continuous deployment pipeline for DevOps organizations
- DevOps Projects allows even novice development organizations to get started by selecting a language, a runtime, and an Azure service
- Tools, such as the change tracking service in Azure Automation, help manage the DevOps process

Resources

Azure DevOps Integrations:

<https://azure.microsoft.com/try/devops/>

DevOps Resource Center:

<https://docs.microsoft.com/azure/devops/learn/>

DevOps Projects:

<https://azure.microsoft.com/features/devops-projects/>

DevOps Checklist: <https://docs.microsoft.com/azure/architecture/checklist/dev-ops>

Azure DevTest Labs:

<https://azure.microsoft.com/services/devtest-lab/>

04 /

New architecture and tools

When you move your software pipeline to cloud platforms, your developers gain the ability to embrace new architectural patterns and use a plethora of tools to build your applications—essentially reducing the time it takes to deliver innovation to your customers.

The degree to which the development and operations pipeline can be implemented and maintained depends on the degree to which the infrastructure and DevOps pipelines can be implemented as infrastructure-as-code (IaC). This approach enables application developers to describe the infrastructure needed for their applications deterministically through static definition files. This allows for a consistent deployment of infrastructure that reduces cost, increases execution speed and mitigates risks and errors.

Cloud-based assets using configurations expressed in code result in easy-to-maintain and quick-to-change software and services. In addition, infrastructure-as-code makes developers part of the deployment team—when they change the configuration of the applications, the changes are carried along with the software to deployment.

In addition to architectural flexibility, IaC will also facilitate the automation of the end-to-end delivery pipeline of the software. During development, the code, both infrastructure and application, can be automatically tested and scanned for vulnerabilities, giving developers critical feedback on the state of their code. After deployment, new instances of specific components or resources can be created to scale up the application to improve availability or add necessary redundancy.

A unified approach to hybrid infrastructure also requires a consistent approach to cloud services—the services that your company depends on in one cloud platform need to be available on-premises, if necessary. Complying with regulations and the need to deploy software to isolated sites—so-called “edge computing”—requires that companies not be reliant only on cloud-based applications and services. Hybrid cloud solutions which deliver such consistency—such as Azure Stack—are an important part of delivering on that vision.

By breaking up larger applications into smaller components and using static files for configuration and security, each deployed in on a server on-premise or in a container in the cloud, for example, developers can also improve the security of the final application by reducing the attack surface area, network ACLs and enforcing principles of least privilege. Dividing large applications into smaller subcomponents that use a regular and well-understood method of communication can also help developers manage and control the source code.

Challenges

With new tools and architectures come new challenges. Developers have to change their mindset from a vertical application stack to peer-to-peer architecture of a containerized application. One of the key barriers of entry—and point of stickiness—for any solution is the third-party management tools and skills the organization has invested in for their staff. Having a consistent model for management and monitoring across cloud infrastructures is core to maintaining a healthy hybrid environment.

While a variety of cloud technologies—such as containers, virtual machines and infrastructure-as-code—give developers a standard environment to which they can deploy applications, they do add new requirements, such as managing and testing the software components

Infrastructure-as-code adds simplicity, but requires a fully automated pipeline to consistently test software components before deployment

Creating an application infrastructure often requires new technologies, such as microservice APIs, making the education of developers and the testing of the service both very important to insure a secure and reliable applications

Considerations

To maximize the benefits for cloud-based development, your development and operations teams should learn about the new features and architectures available to them. You do not have to refactor an application to benefit from containerization. Porting an application into a container delivers immediate benefits.

Some considerations as you move forward:

- Hybrid application development strategies focus on writing software once and deploying to multiple endpoints
- Automating testing and deployment through policy orchestration is extremely important, and technologies such as infrastructure-as-code and containerization can help companies deploy a variety of applications in a unified way to different targeted environments
- Software requirements and configuration are also part of the pipeline, and developers need to be able to change them as needed—these requirements should be expressed as code to increase deployment flexibility and consistency
- Instrument your software and pipelines to make the system observable from end to end, allowing easier and more in-depth management of the software and infrastructure and enabling the automation of alerts and operational tasks

Hybrid application development strategies focus on **writing software once and deploying to multiple endpoints.**

Solutions

Microsoft has focused on making Azure a seamless way to integrate development and deployment efforts across cloud and on-premises infrastructures. Microsoft takes this to heart in its own businesses and runs its core services on top of Azure, including Skype, Bing, Xbox, and Office 365.

- Application Insights and other tools can give you a view into what is going on in the application
- Azure's support for container instances can deliver additional instances of application components as needed as well as support for Kubernetes
- Service Fabric Explorer gives you access to the infrastructure on which your applications run
- Azure Resource Manager, DevTest Labs, and other cloud tools help developers unify their workflow irrespective of the target environment
- Software developed on Azure can be deployed to on-premises cloud installations through Azure Stack
- Automating deployment and updates is important to keeping the software pipeline efficient and producing high-quality code, and Visual Studio Team Services helps developers and operations teams quickly update their services to build knowledge into their processes
- No matter where your code resides, the software needs access to the same services: Azure Resource Manager gives developers access to the same APIs, whether you deploy to the global Azure, a sovereign Azure cloud or Azure Stack

Resources

How to parameterize configuration files in Service Fabric: <https://docs.microsoft.com/azure/service-fabric/service-fabric-how-to-parameterize-configuration-files>

Azure Stack: An Extension of Azure: <https://azure.microsoft.com/resources/azure-stack-an-extension-of-azure/>

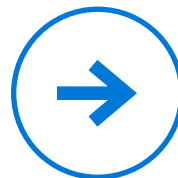
Azure IOT Edge: <https://azure.microsoft.com/services/iot-edge/>

Try Azure Stack with the Azure Stack Development Kit: <https://azure.microsoft.com/overview/azure-stack/development-kit/>

Azure Stack Development Kit Documentation: <https://docs.microsoft.com/azure-stack/asdk/>

Learn about Azure Stack HCI (hyperconverged infrastructure): <https://azure.microsoft.com/blog/announcing-azure-stack-hci-a-new-member-of-the-azure-stack-family/>

Azure Stack Pluralsight training: <https://www.pluralsight.com/courses/microsoft-azure-stack-big-picture>



Contact us: <https://azure.microsoft.com/overview/sales-number/>

Create your Azure free account: <https://azure.microsoft.com/free/hybrid-cloud/>