



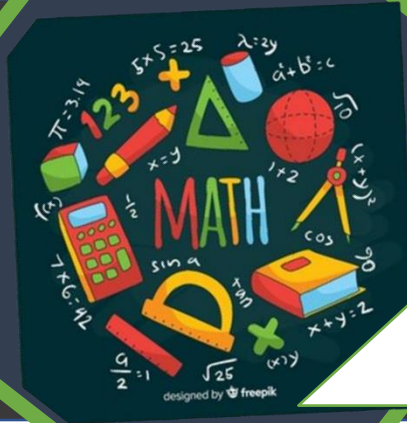
UNIVERSIDAD
DE COLIMA

09/05/2022

Computo en la nube

**CARLOS ALEJANDRO
BALTAZAR PADILLA**

6 ■



Azure App Service es un servicio basado en HTTP para hospedar aplicaciones web, API REST y back-ends móviles. Puede desarrollar en su lenguaje favorito, ya sea .NET, .NET Core, Java, Ruby, Node.js, PHP o Python. Las aplicaciones se ejecutan y escalan con facilidad en entornos basados en Windows y Linux.

App Service no solo agrega el poder de Microsoft Azure a su aplicación, como seguridad, equilibrio de carga, escalado automático y administración automatizada. También puede aprovechar sus capacidades de DevOps, como la implementación continua desde Azure DevOps, GitHub, Docker Hub y otras fuentes, administración de paquetes, entornos de prueba, dominio personalizado y certificados TLS/SSL.

Con App Service, paga por los recursos informáticos de Azure que utiliza. Los recursos informáticos que utiliza están determinados por el plan de App Service en el que ejecuta sus aplicaciones. Para obtener más información, consulte Información general sobre los planes de Azure App Service.

¿Por qué usar el servicio de aplicaciones?

Azure App Service es una oferta de plataforma como servicio (PaaS) completamente administrada para desarrolladores. Estas son algunas características clave de App Service:

Múltiples lenguajes y marcos: App Service tiene soporte de primera clase para ASP.NET, ASP.NET Core, Java, Ruby, Node.js, PHP o Python. También puede ejecutar PowerShell y otros scripts o ejecutables como servicios en segundo plano.

Entorno de producción administrado: App Service aplica parches automáticamente y mantiene el sistema operativo y los marcos de lenguaje por usted. Dedique tiempo a escribir excelentes aplicaciones y deje que Azure se preocupe por la plataforma.

Contenedorización y Docker: Dockerize su aplicación y aloje un contenedor personalizado de Windows o Linux en App Service. Ejecute aplicaciones de varios contenedores con Docker Compose. Migre sus habilidades de Docker directamente a App Service.

Optimización de DevOps: configure la integración y la implementación continuas con Azure DevOps, GitHub, BitBucket, Docker Hub o Azure Container Registry. Promueva actualizaciones a través de entornos de prueba y ensayo. Administre sus aplicaciones en App Service mediante Azure PowerShell o la interfaz de línea de comandos (CLI) multiplataforma.

Escala global con alta disponibilidad: escala hacia arriba o hacia afuera de forma manual o automática. Aloje sus aplicaciones en cualquier lugar de la infraestructura del centro de datos global de Microsoft y el SLA de App Service promete una alta disponibilidad.

Conexiones a plataformas SaaS y datos locales: elija entre más de 50 conectores para sistemas empresariales (como SAP), servicios SaaS (como Salesforce) y servicios de Internet (como Facebook). Acceda a datos locales mediante Hybrid Connections y Azure Virtual Networks.

Seguridad y cumplimiento: App Service cumple con las normas ISO, SOC y PCI. Autentique a los usuarios con Azure Active Directory, Google, Facebook, Twitter o cuenta de Microsoft. Cree restricciones de direcciones IP y administre identidades de servicios.

Plantillas de aplicaciones: elija entre una extensa lista de plantillas de aplicaciones en Azure Marketplace, como WordPress, Joomla y Drupal.

Integración de Visual Studio y Visual Studio Code: las herramientas dedicadas en Visual Studio y Visual Studio Code agilizan el trabajo de creación, implementación y depuración.

API y funciones móviles: App Service proporciona soporte CORS llave en mano para escenarios de API RESTful y simplifica los escenarios de aplicaciones móviles al habilitar la autenticación, la sincronización de datos sin conexión, las notificaciones automáticas y más.

Código sin servidor: ejecute un fragmento de código o un script a pedido sin tener que aprovisionar o administrar la infraestructura explícitamente, y pague solo por el tiempo de cómputo que realmente usa su código (consulte Funciones de Azure).

Además de App Service, Azure ofrece otros servicios que se pueden usar para alojar sitios web y aplicaciones web. Para la mayoría de los escenarios, App Service es la mejor opción. Para la arquitectura de microservicios, considere Azure Spring-Cloud Service o Service Fabric. Si necesita más control sobre las máquinas virtuales en las que se ejecuta su código, considere Azure Virtual Machines. Para obtener más información sobre cómo elegir entre estos servicios de Azure, consulte Comparación de Azure App Service, Virtual Machines, Service Fabric y Cloud Services.

Servicio de aplicaciones en Linux

App Service también puede alojar aplicaciones web de forma nativa en Linux para pilas de aplicaciones compatibles. También puede ejecutar contenedores de Linux personalizados (también conocidos como aplicación web para contenedores).

Lenguajes y marcos incorporados

App Service en Linux admite varias imágenes integradas específicas del idioma. Simplemente implemente su código. Los lenguajes admitidos incluyen: Node.js, Java (JRE 8 y JRE 11), PHP, Python, .NET Core y Ruby. Ejecute `az webapp list-runtimes --os linux` para ver los últimos idiomas y versiones compatibles. Si el tiempo de ejecución que requiere su aplicación no es compatible con las imágenes integradas, puede implementarlo con un contenedor personalizado.

Los tiempos de ejecución obsoletos se eliminan periódicamente de las hojas de creación y configuración de aplicaciones web en el portal. Estos tiempos de ejecución se ocultan del Portal cuando la organización de mantenimiento los descarta o cuando se descubre que tienen vulnerabilidades significativas. Estas opciones están ocultas para guiar a los clientes a los tiempos de ejecución más recientes donde tendrán más éxito.

Cuando un tiempo de ejecución desactualizado se oculta del Portal, cualquiera de sus sitios existentes que utilicen esa versión continuará ejecutándose. Si un tiempo de ejecución se elimina

por completo de la plataforma de App Service, los propietarios de su suscripción de Azure recibirán un aviso por correo electrónico antes de la eliminación.

Si necesita crear otra aplicación web con una versión de tiempo de ejecución obsoleta que ya no se muestra en el Portal, consulte las guías de configuración de idiomas para obtener instrucciones sobre cómo obtener la versión de tiempo de ejecución de su sitio. Puede usar la CLI de Azure para crear otro sitio con el mismo tiempo de ejecución. Como alternativa, puede usar el botón Exportar plantilla en la hoja de la aplicación web en el Portal para exportar una plantilla ARM del sitio. Puede reutilizar esta plantilla para implementar un nuevo sitio con el mismo tiempo de ejecución y configuración.

En App Service (aplicaciones web, aplicaciones de API o aplicaciones móviles), una aplicación siempre se ejecuta en un plan de App Service. Además, Azure Functions también tiene la opción de ejecutarse en un plan de App Service. Un plan de App Service define un conjunto de recursos informáticos para que se ejecute una aplicación web. Estos recursos informáticos son análogos a la granja de servidores en el alojamiento web convencional. Se pueden configurar una o más aplicaciones para que se ejecuten en los mismos recursos informáticos (o en el mismo plan de App Service).

Cuando crea un plan de App Service en una determinada región (por ejemplo, Europa Occidental), se crea un conjunto de recursos informáticos para ese plan en esa región. Las aplicaciones que coloque en este plan de App Service se ejecutan en estos recursos informáticos según lo define su plan de App Service. Cada plan de App Service define:

Sistema operativo (Windows, Linux)

Región (Oeste de EE. UU., Este de EE. UU., etc.)

Número de instancias de VM

Tamaño de las instancias de VM (Pequeño, Mediano, Grande)

Nivel de precios (Gratis, Compartido, Básico, Estándar, Premium, PremiumV2, PremiumV3, Aislado, AisladoV2)

El nivel de precios de un plan de App Service determina qué funciones de App Service obtiene y cuánto paga por el plan. Los niveles de precios disponibles para su plan de App Service dependen del sistema operativo seleccionado en el momento de la creación. Hay algunas categorías de niveles de precios:

Cómputo compartido: Gratis y Compartido, los dos niveles básicos, ejecutan una aplicación en la misma máquina virtual de Azure que otras aplicaciones de App Service, incluidas las aplicaciones

de otros clientes. Estos niveles asignan cuotas de CPU a cada aplicación que se ejecuta en los recursos compartidos y los recursos no pueden escalar horizontalmente.

Computación dedicada: los niveles Básico, Estándar, Premium, PremiumV2 y PremiumV3 ejecutan aplicaciones en máquinas virtuales de Azure dedicadas. Solo las aplicaciones del mismo plan de App Service comparten los mismos recursos informáticos. Cuanto más alto sea el nivel, más instancias de VM estarán disponibles para escalar horizontalmente.

Aislado: estos niveles Aislado y Aislado V2 ejecutan máquinas virtuales de Azure dedicadas en redes virtuales de Azure dedicadas. Proporciona aislamiento de red además del aislamiento informático para sus aplicaciones. Proporciona las máximas capacidades de escalamiento horizontal.

The new **PremiumV3** pricing tier guarantees machines with faster processors (minimum 195 **ACU** per virtual CPU), SSD storage, and quadruple memory-to-core ratio compared to **Standard** tier. **PremiumV3** also supports higher scale via increased instance count while still providing all the advanced capabilities found in **Standard** tier. All features available in the existing **PremiumV2** tier are included in **PremiumV3**.

Similar to other dedicated tiers, three VM sizes are available for this tier:

- Small (2 CPU core, 8 GiB of memory)
- Medium (4 CPU cores, 16 GiB of memory)
- Large (8 CPU cores, 32 GiB of memory)

For **PremiumV3** pricing information, see [**App Service Pricing**](#).

To get started with the new **PremiumV3** pricing tier, see [**Configure PremiumV3 tier for App Service**](#).

Un entorno de servicio de aplicaciones puede alojar su:

Aplicaciones web de Windows

Aplicaciones web de Linux

Contenedores Docker (Windows y Linux)

Funciones

Aplicaciones lógicas (estándar)

Los entornos de App Service son apropiados para cargas de trabajo de aplicaciones que requieren:

Escala alta.

Aislamiento y acceso seguro a la red.

Alta utilización de la memoria.

Altas solicitudes por segundo (RPS). Puede crear varios entornos de App Service en una sola región de Azure o en varias regiones de Azure. Esta flexibilidad hace que un entorno de servicio de aplicaciones sea ideal para escalar horizontalmente aplicaciones sin estado con un alto requisito de RPS.

Un entorno de servicio de aplicaciones puede alojar aplicaciones de un solo cliente y lo hace en una de sus redes virtuales. Los clientes tienen un control detallado sobre el tráfico de red de aplicaciones entrante y saliente. Las aplicaciones pueden establecer conexiones seguras de alta velocidad a través de VPN a recursos corporativos locales.

Escenarios de uso

Los entornos de App Service tienen muchos casos de uso, incluidos:

Aplicaciones internas de línea de negocio.

Aplicaciones que necesitan más de 30 instancias del plan App Service.

Sistemas de un solo inquilino para satisfacer los requisitos internos de cumplimiento o seguridad.

Alojamiento de aplicaciones aisladas en red.

Aplicaciones de varios niveles.

Hay muchas funciones de red que permiten que las aplicaciones en un servicio de aplicaciones de múltiples inquilinos lleguen a recursos aislados de la red o se aíslen de la red. Estas funciones están habilitadas en el nivel de la aplicación. Con un entorno de servicio de aplicaciones, no se requiere ninguna configuración adicional para que las aplicaciones estén en una red virtual. Las aplicaciones se implementan en un entorno aislado de la red que ya se encuentra en una red virtual. Si realmente necesita una historia de aislamiento completa, también puede implementar su entorno de servicio de aplicaciones en hardware dedicado.

Entorno dedicado

Un entorno de App Service es una implementación de un solo inquilino de Azure App Service que se ejecuta en su red virtual.

Las aplicaciones se alojan en planes de App Service, que se crean en un entorno de App Service. Un plan de App Service es esencialmente un perfil de aprovisionamiento para un host de aplicaciones. A medida que amplía su plan de App Service, crea más hosts de aplicaciones con todas las aplicaciones de ese plan de App Service en cada host. Un solo App Service Environment v3 puede tener hasta 200 instancias totales del plan de App Service en todos los planes de App Service combinados. Un solo plan App Service Isolated v2 (Iv2) puede tener hasta 100 instancias por sí mismo.

Cuando está implementando en hardware dedicado (hosts), está limitado a la hora de escalar en todos los planes de App Service a la cantidad de núcleos en este tipo de entorno. Un entorno de App Service que se implementa en hosts dedicados tiene 132 núcleos virtuales disponibles. I1v2 usa dos núcleos virtuales, I2v2 usa cuatro núcleos virtuales e I3v2 usa ocho núcleos virtuales por instancia.

Soporte de red virtual

La función App Service Environment es una implementación de Azure App Service en una subred única en una red virtual. Cuando implementa una aplicación en un entorno de servicio de aplicaciones, la aplicación se expone en la dirección de entrada asignada al entorno de servicio de aplicaciones. Si su entorno de servicio de aplicaciones se implementa con una dirección IP virtual interna (VIP), la dirección de entrada para todas las aplicaciones será una dirección en la subred del entorno de servicio de aplicaciones. Si su entorno de servicio de aplicaciones se implementa con una dirección VIP externa, la dirección de entrada será una dirección de Internet y sus aplicaciones estarán en un sistema de nombres de dominio público.

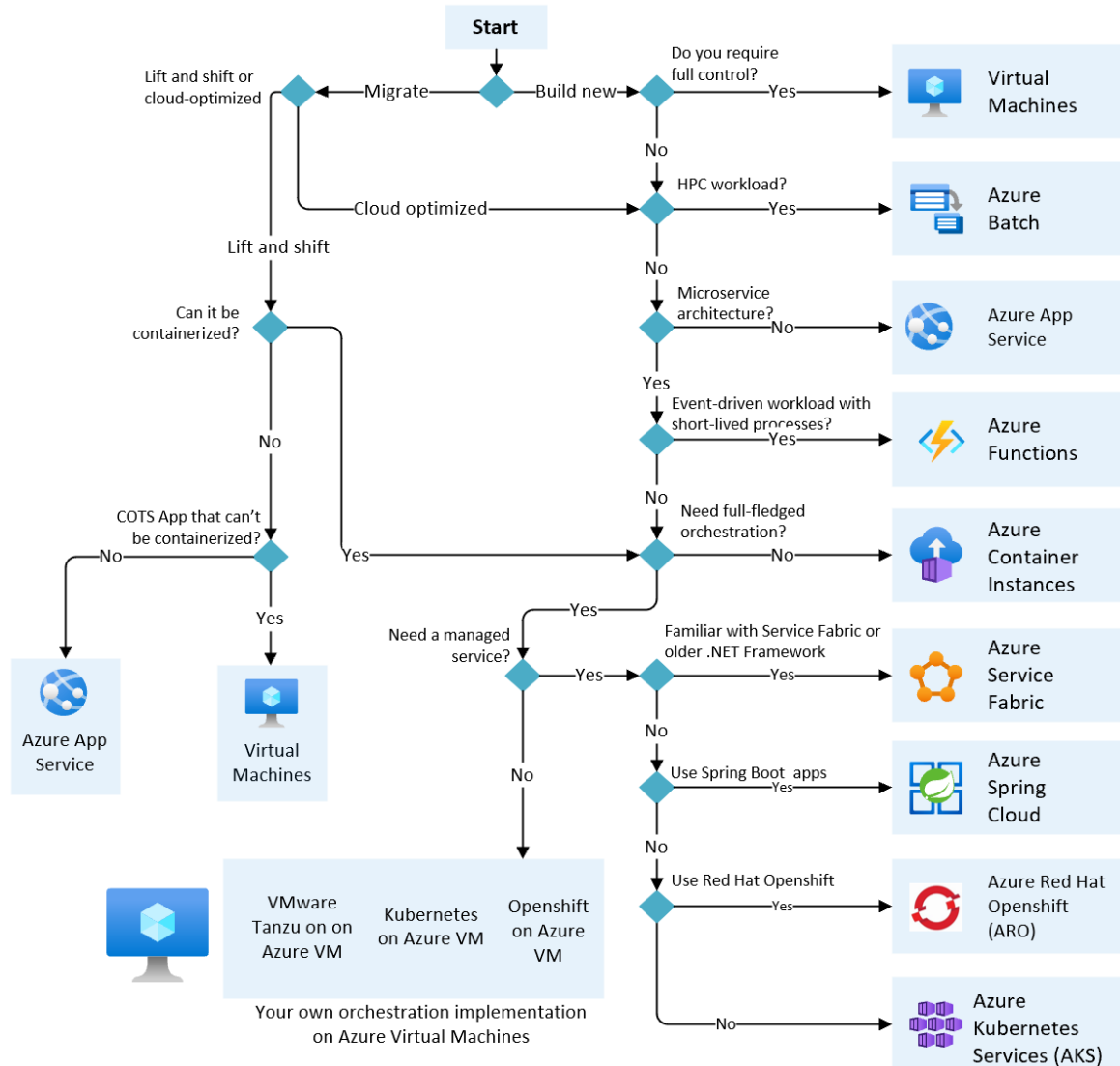
La cantidad de direcciones que utiliza un entorno de servicio de aplicaciones v3 en su subred variará según la cantidad de instancias y la cantidad de tráfico. Algunas funciones de infraestructura se escalan automáticamente, según la cantidad de planes de App Service y la carga. El tamaño recomendado para la subred de App Service Environment v3 es un bloque de enrutamiento entre dominios sin clase (CIDR) /24 con 256 direcciones, porque ese tamaño puede alojar un App Service Environment v3 que se escala hasta su límite.

Las aplicaciones en un entorno de servicio de aplicaciones no necesitan ninguna característica habilitada para acceder a los recursos en la misma red virtual en la que se encuentra el entorno de servicio de aplicaciones. Si la red virtual del entorno de servicio de aplicaciones está conectada a otra red, las aplicaciones en el entorno de servicio de aplicaciones puede acceder a los recursos en esas redes extendidas. El tráfico puede ser bloqueado por la configuración del usuario en la red.

La versión multiinquilino de Azure App Service contiene numerosas funciones para permitir que sus aplicaciones se conecten a sus diversas redes. Con esas funciones de red, sus aplicaciones pueden actuar como si estuvieran implementadas en una red virtual. Las aplicaciones en un App Service Environment v3 no necesitan ninguna configuración adicional para estar en la red virtual.

Una ventaja de usar un entorno de App Service en lugar de un servicio de múltiples inquilinos es que los controles de acceso a la red para las aplicaciones alojadas en App Service Environment son externos a la configuración de la aplicación. Con las aplicaciones en el servicio multiusuario, debe habilitar las funciones aplicación por aplicación y usar el control de acceso basado en roles o una política para evitar cualquier cambio de configuración.

Choose a candidate service



Understand the hosting models

Cloud services, including Azure services, generally fall into three categories: IaaS, PaaS, or FaaS. (There is also SaaS, software-as-a-service, which is out of scope for this article.) It's useful to understand the differences.

Infrastructure-as-a-Service (IaaS) lets you provision individual VMs along with the associated networking and storage components. Then you deploy whatever software and applications you want onto those VMs. This model is the closest to a traditional on-premises environment, except that Microsoft manages the infrastructure. You still manage the individual VMs.

Platform-as-a-Service (PaaS) provides a managed hosting environment, where you can deploy your application without needing to manage VMs or networking resources. Azure App Service is a PaaS service.

Functions-as-a-Service (FaaS) goes even further in removing the need to worry about the hosting environment. In a FaaS model, you simply deploy your code and the service automatically runs it. Azure Functions is a FaaS service.

Note

Azure Functions is an Azure serverless compute offering. You may read [Choose the right integration and automation services in Azure](#) to know how this service compares with other Azure serverless offerings, such as Logic Apps which provides serverless workflows.

There is a spectrum from IaaS to pure PaaS. For example, Azure VMs can autoscale by using virtual machine scale sets. This automatic scaling capability isn't strictly PaaS, but it's the type of management feature found in PaaS services.

In general, there is a tradeoff between control and ease of management. IaaS gives the most control, flexibility, and portability, but you have to provision, configure and manage the VMs and network components you create. FaaS services automatically manage nearly all aspects of running an application. PaaS services fall somewhere in between.

Criteria	Virtual Machines	App Service	Azure Spring Cloud	Service Fabric	Azure Functions
Application composition	Agnostic	Applications, containers	Applications, microservices	Services, guest executables, containers	Functions
Density	Agnostic	Multiple apps per instance via app service plans	Multiple apps per service instance	Multiple services per VM	Serverless
Minimum number of nodes	1 ²	1	2	5 ³	Serverless
State management	Stateless or Stateful	Stateless	Stateless	Stateless or stateful	Stateless
Web hosting	Agnostic	Built in	Built in	Agnostic	Not applicable

Criteria	Virtual Machines	App Service	Azure Spring Cloud	Service Fabric	Azure Functions
Can be deployed to dedicated VNet?	Supported	Supported ⁵	Supported	Supported	Supported
Hybrid connectivity	Supported	Supported ⁶	Supported	Supported	Supported

Notes

1. If using Consumption plan. If using App Service plan, functions run on the VMs allocated for your App Service plan. See [Choose the correct service plan for Azure Functions](#).
2. Higher SLA with two or more instances.
3. Recommended for production environments.
4. Can scale down to zero after job completes.
5. Requires App Service Environment (ASE).
6. Use [Azure App Service Hybrid Connections](#).
7. Requires App Service plan or [Azure Functions Premium plan](#).

DevOps

Criteria	Virtual Machines	App Service	Azure Spring Cloud	Service Fabric	Azure Functions
Local debugging	Agnostic	IIS Express, others ¹	Visual Studio Code, IntelliJ, Eclipse	Local node cluster	Visual Studio or Azure Function CLI
Programming model	Agnostic	Web and API applications, WebJobs for background tasks	Spring Boot, Steeltoe	Guest executable, Service model, Actor model, Containers	Functions with triggers
Application update	No built-in support	Deployment slots	Rolling upgrade, Blue-green deployment	Rolling upgrade (per service)	Deployment slots

Notes

1. Options include IIS Express for ASP.NET or node.js (iisnode); PHP web server; Azure Toolkit for IntelliJ, Azure Toolkit for Eclipse. App Service also supports remote debugging of deployed web app.
2. See [Resource Manager providers, regions, API versions and schemas](#).

Scalability

Criteria	Virtual Machines	App Service	Azure Spring Cloud	Service Fabric	Azure Functions	Azure Kubernetes Service
Autoscaling	Virtual machine scale sets	Built-in service	Built-in service	Virtual machine scale sets	Built-in service	Pod auto-scaling ²
Load balancer	Azure Load Balancer	Integrated	Integrated	Azure Load Balancer	Integrated	Azure Load Balancer or Application Gateway
Scale limit ³	Platform image: 1000 nodes per scale set, Custom image: 600 nodes per scale set	30 instances, 100 with App Service Environment	500 app instances in Standard	100 nodes per scale set	200 instances per Function app	100 nodes per cluster (default limit)

Notes

1. See [Autoscale pods](#).
2. See [Automatically scale a cluster to meet application demands on Azure Kubernetes Service \(AKS\)](#).
3. See [Azure subscription and service limits, quotas, and constraints](#).

Availability

Criteria	Virtual Machines	App Service	Azure Spring Cloud	Service Fabric	Azure Functions	Azure Kubernetes Service
SLA	SLA for Virtual Machines	SLA for App Service	SLA for Azure Spring Cloud	SLA for Service Fabric	SLA for Functions	SLA for AKS
Multi region failover	Traffic manager	Traffic manager	Traffic manager, Multi-Region Cluster	Azure Front Door	Traffic manager	Not supported

For guided learning on Service Guarantees, review [Core Cloud Services - Azure architecture and service guarantees](#).

Security

Review and understand the available security controls and visibility for each service

- [App Service](#)
- [App Spring Cloud](#)
- [Azure Kubernetes Service](#)
- [Batch](#)
- [Container Instances](#)
- [Functions](#)
- [Service Fabric](#)
- [Virtual machine - Windows](#)
- [Virtual machine - LINUX](#)

Other criteria

Criteria	Virtual Machines	App Service	App Spring Cloud	Service Fabric	Azure Functions	Azure Kubernetes Service
SSL	Configured in VM	Supported	Supported	Supported	Supported	Supported
Cost	Windows, Linux	App Service pricing	Azure Spring Cloud pricing	Service Fabric pricing	Azure Functions pricing	AKS pricing
Suitable architecture styles	N-Tier, Big compute (HPC)	Web-Queue-Worker, N-Tier	Spring Boot, Microservices	Microservices, Event-driven architecture	Microservices, Event-driven architecture	Microservices, Event-driven architecture

The output from this flowchart is a **starting point** for consideration. Next, perform a more detailed evaluation of the service to see if it meets your needs.