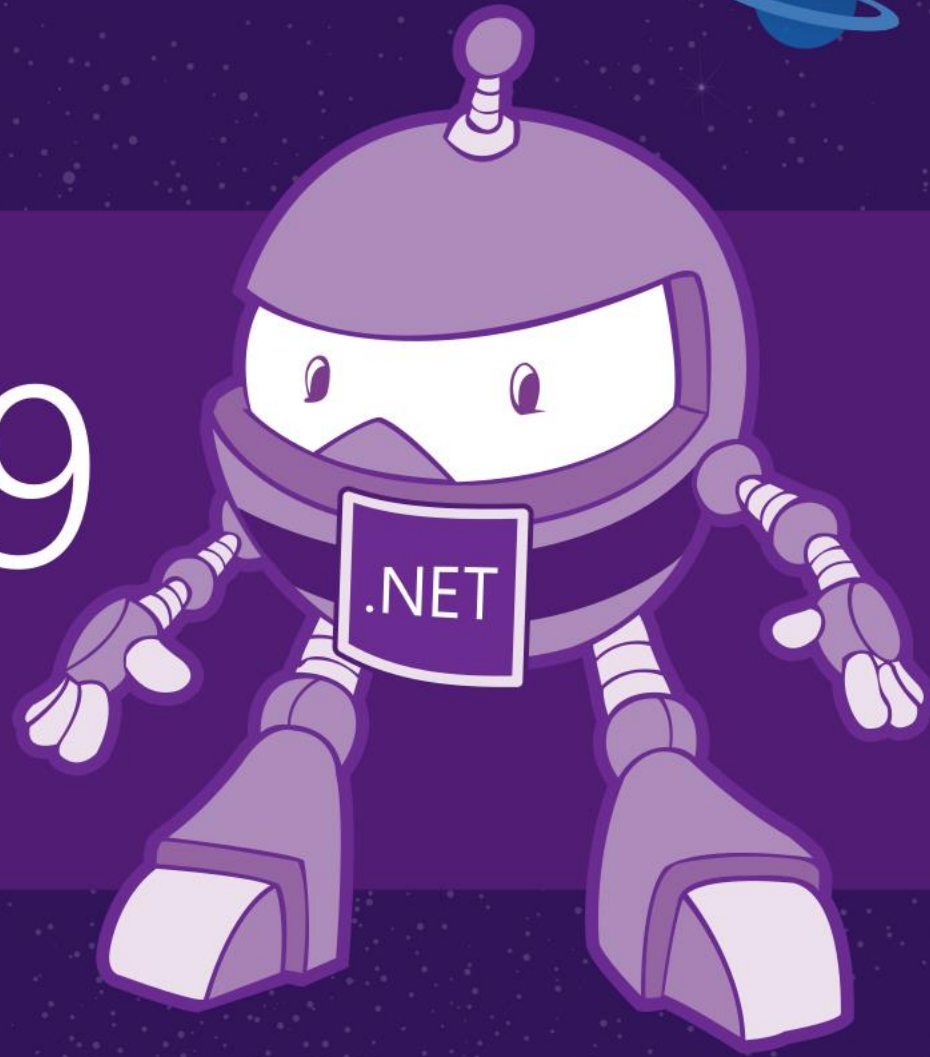


.NET Conf 2019

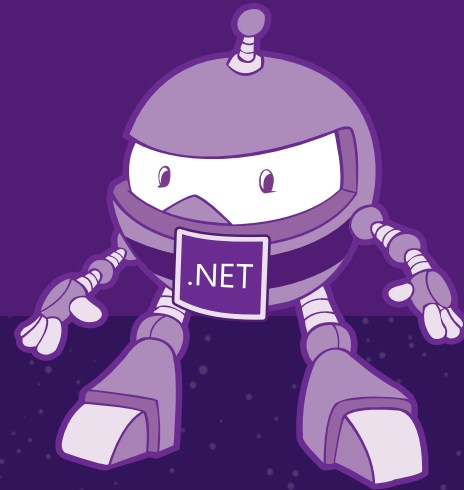
Discover the world of .NET



www.dotnetconf.net

Desarrollo nativo de nube con AKS y Cosmos DB

Jesus Angulo
Microsoft MVP



Jesús Angulo

Autor en SomosTechies.com

Microsoft Certified Trainer

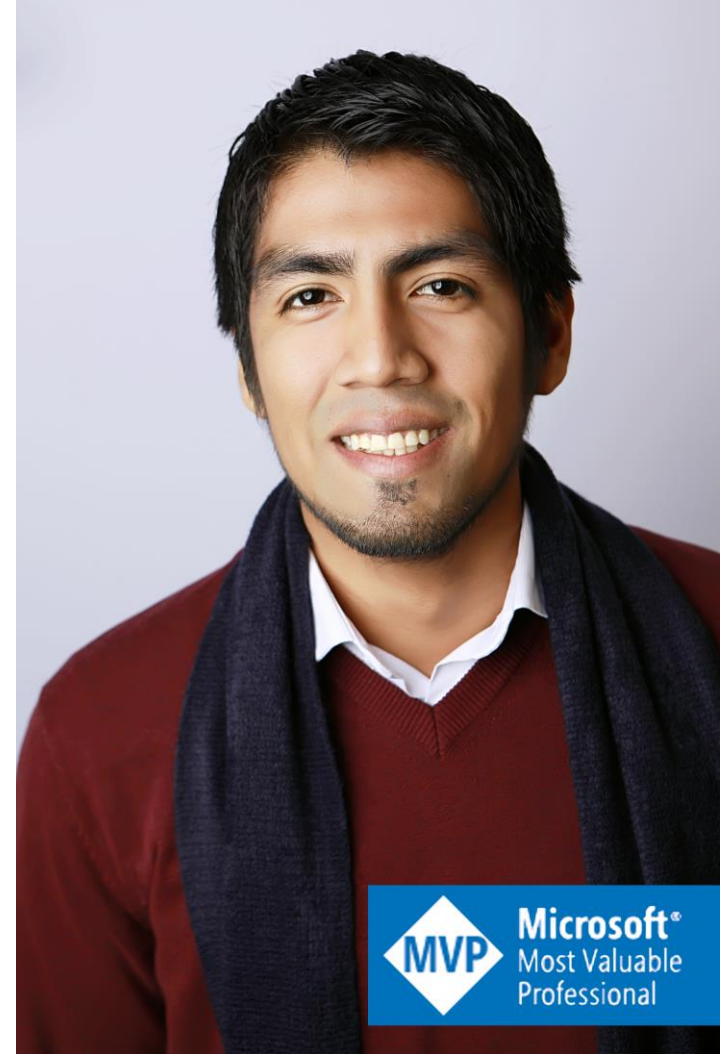
 Software Architect en Baufest Perú

 Microsoft Most Valuable Professional

 @jesulink2514

 jesus@somostechies.com

 /somostechies-com



La combinación de la potencia de un servicio nativo de nube (como AKS) y un servicio de base de datos administrada como Cosmos DB

Agenda

Características de las aplicaciones nativas de nube

Consideraciones de aplicaciones nativas de nube con estado

¿Y las herramientas nativas de nube?

Service Broker

¿Y SRE (con servicios externos en la mezcla)?

Escrituras multi-región

¿Y la portabilidad?

Aplicación nativa de nube (Cloud Native Application)

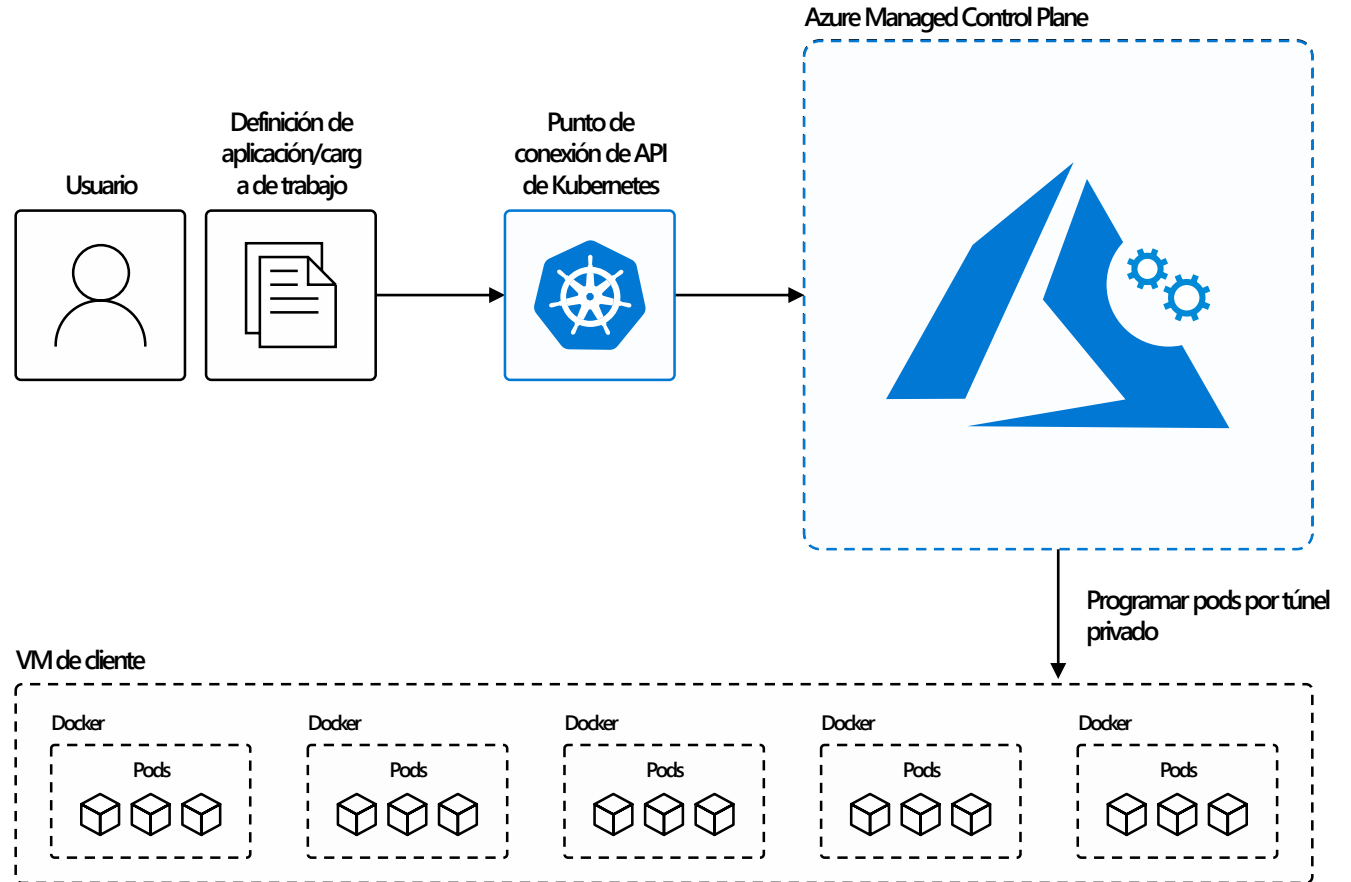
Definida como "nativa de nube"

- Aplicaciones escalables que funcionan en entornos como nubes públicas, privadas e híbridas
- Sistemas sin conexión directa que son resilientes, manejables y observables
- Compuestos de contenedores, mallas de servicio, microservicios, infraestructura inmutable y API declarativas
- Basados en un ecosistema de proyectos de proveedor neutral de código abierto
- Automatización robusta que permite a los ingenieros realizar frecuentes cambios de alto impacto y previsiblemente con una mínima fatiga

Fuente: <https://github.com/cncf/foundation/blob/master/charter.md>

Azure Kubernetes Service o AKS

- Actualizaciones y parches automáticos
- Alta disponibilidad
- Escalación de clúster fácil y segura
- Autorrecuperación
- Supervisión de API Server



Demostración

Aplicación de referencia — Contoso Neuro 2019



Azure Kubernetes Service

Linux Node

Web Site Containers



Admin Site



Conference
Web Site

Service Containers



Registration



Email



Content



Config

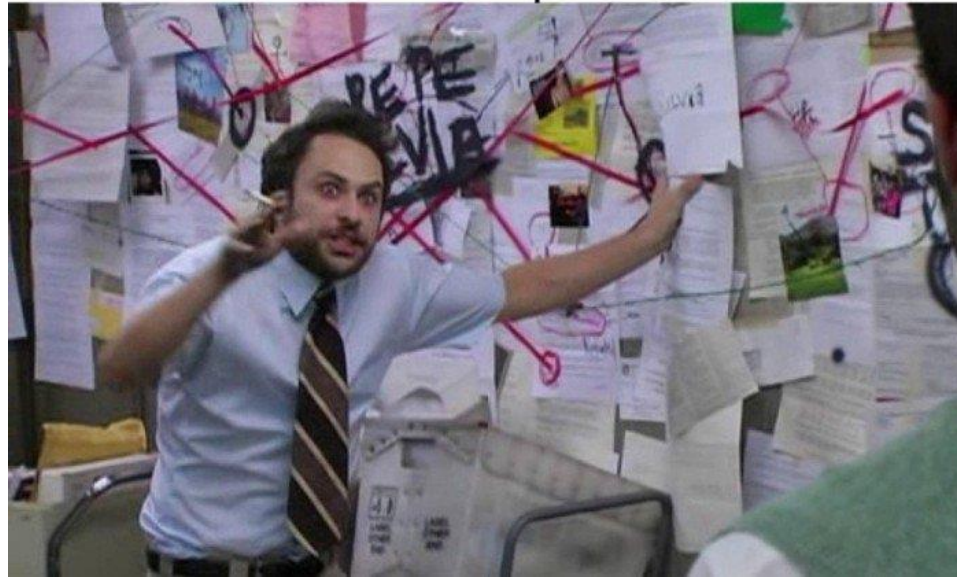


CosmosDB

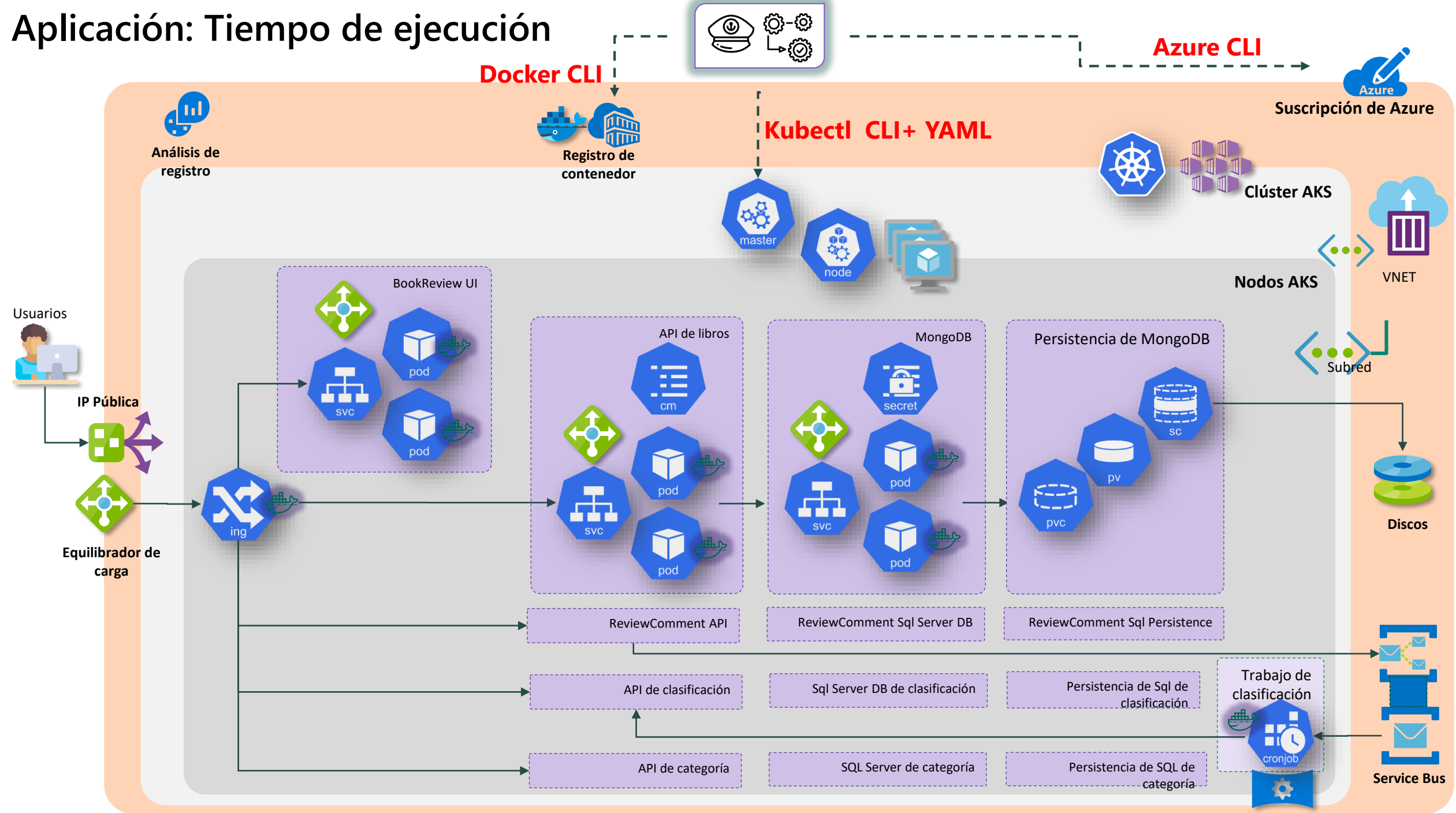
Explaining the feature
to management



Explaining the code
to other developers



Aplicación: Tiempo de ejecución



MongoDB en Azure o Kubernetes?

Consideraciones de aplicaciones nativas de nube con estado

Problemas con administración de estado

- Muchas de las recientes mejoras para aplicaciones con estado en Kubernetes
 - Conjuntos con estado, interfaz de almacenamiento de contenedores (CSI)
 - Volúmenes persistentes (PV), reclamaciones de volumen persistentes (PVC), clases de almacenamiento
 - Operadores
- Sin embargo, el soporte no está a la par con los servicios administrados
 - Clústeres difíciles de administrar con (nodos sin estado y con estado)
 - Distribución geográfica
 - Escalabilidad elástica (almacenamiento y rendimiento)
 - Garantías de rendimiento (disponibilidad, leer, escribir)



Kelsey Hightower 
@kelseyhightower

Follow



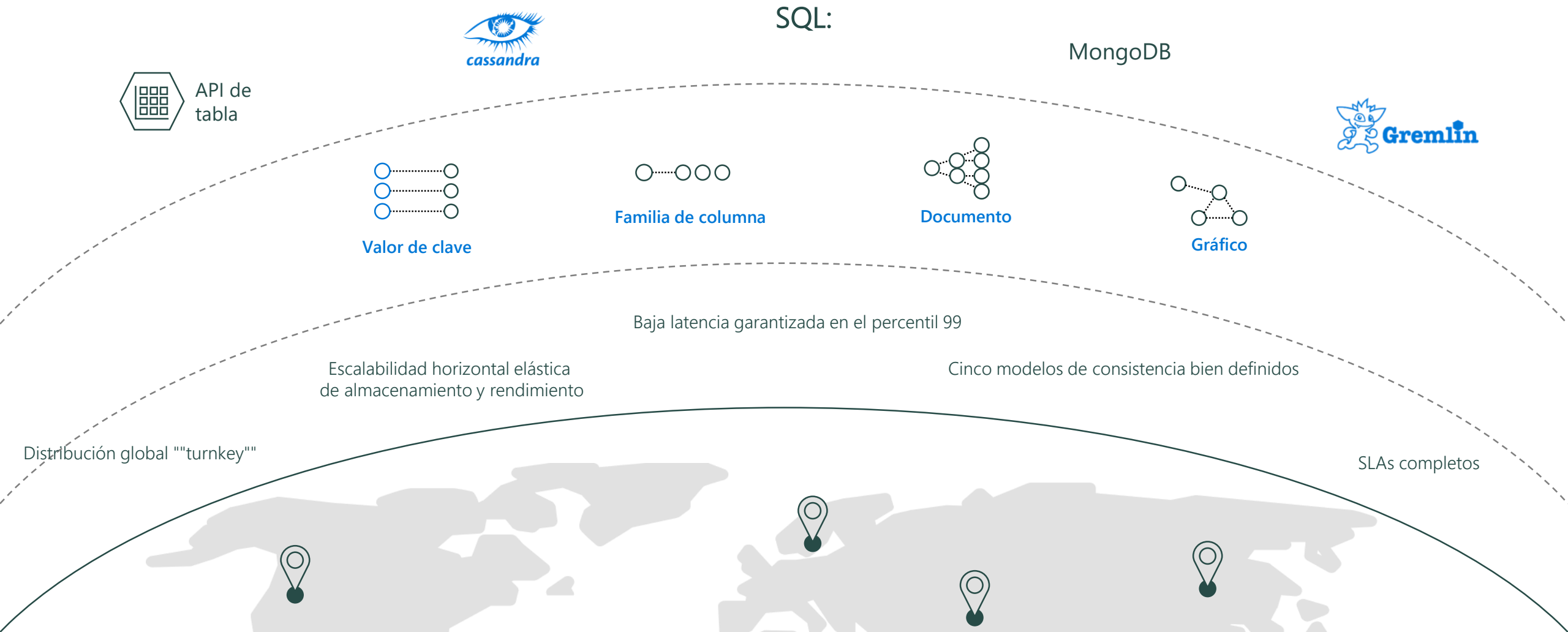
Kubernetes has made huge improvements in the ability to run stateful workloads including databases and message queues, but I still prefer not to run them on Kubernetes.

6:04 AM - 13 Feb 2018

**Traslado el estado hacia un servicio externo
(Cosmos DB)**

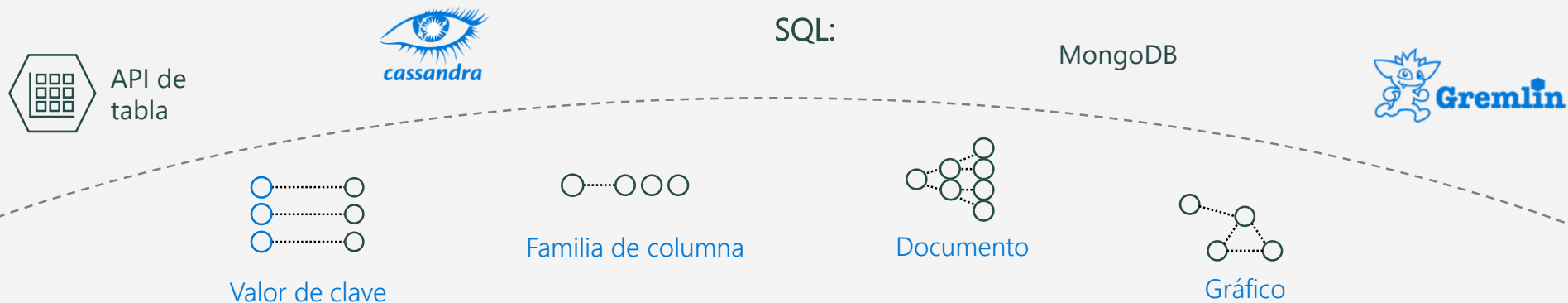
Azure Cosmos DB

Un servicio de base de datos globalmente distribuido, escalable masivamente y multi-modelo



Múltiples modelos de datos y API

- Cosmos DB ofrece una multitud de API para el acceso y la consulta de datos incluyendo, SQL, diversas APIs OSS populares y soporte nativo para cargas de trabajo de NoSQL
- Azure Cosmos etcd API



Compatibilidad con el protocolo de transferencia

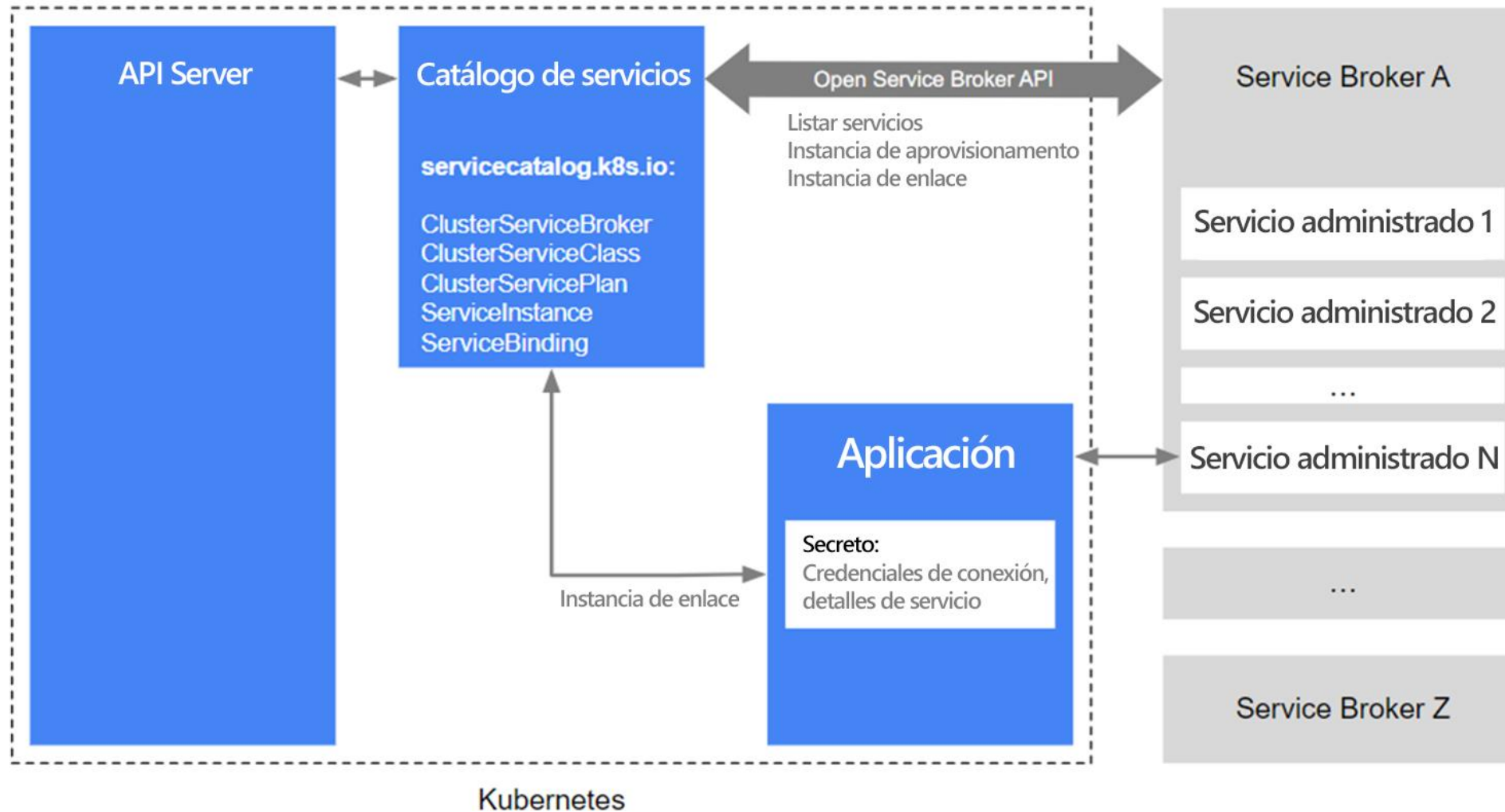


¿Y las herramientas nativas de nube?

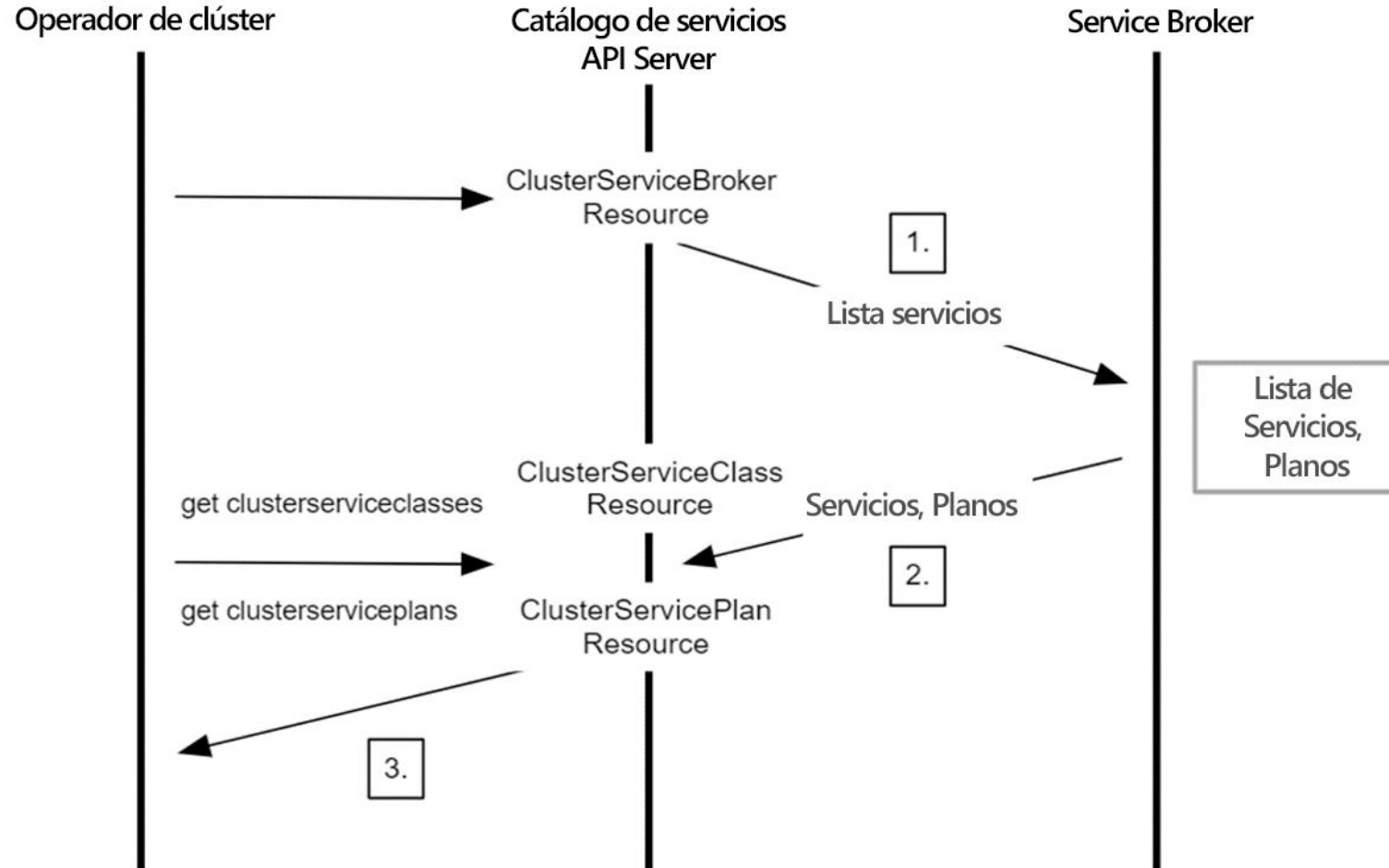
Open Service Broker

- Catálogo de servicios — API que habilita servicios que se ejecutan en Kubernetes para utilizar servicios administrados externos
- Proporciona una forma para enlistar, aprovisionar y enlazar con agentes de servicio externo (sin un conocimiento detallado del servicio)
- API de Open Service Broker — punto de conexión para los servicios ofrecidos por un tercero

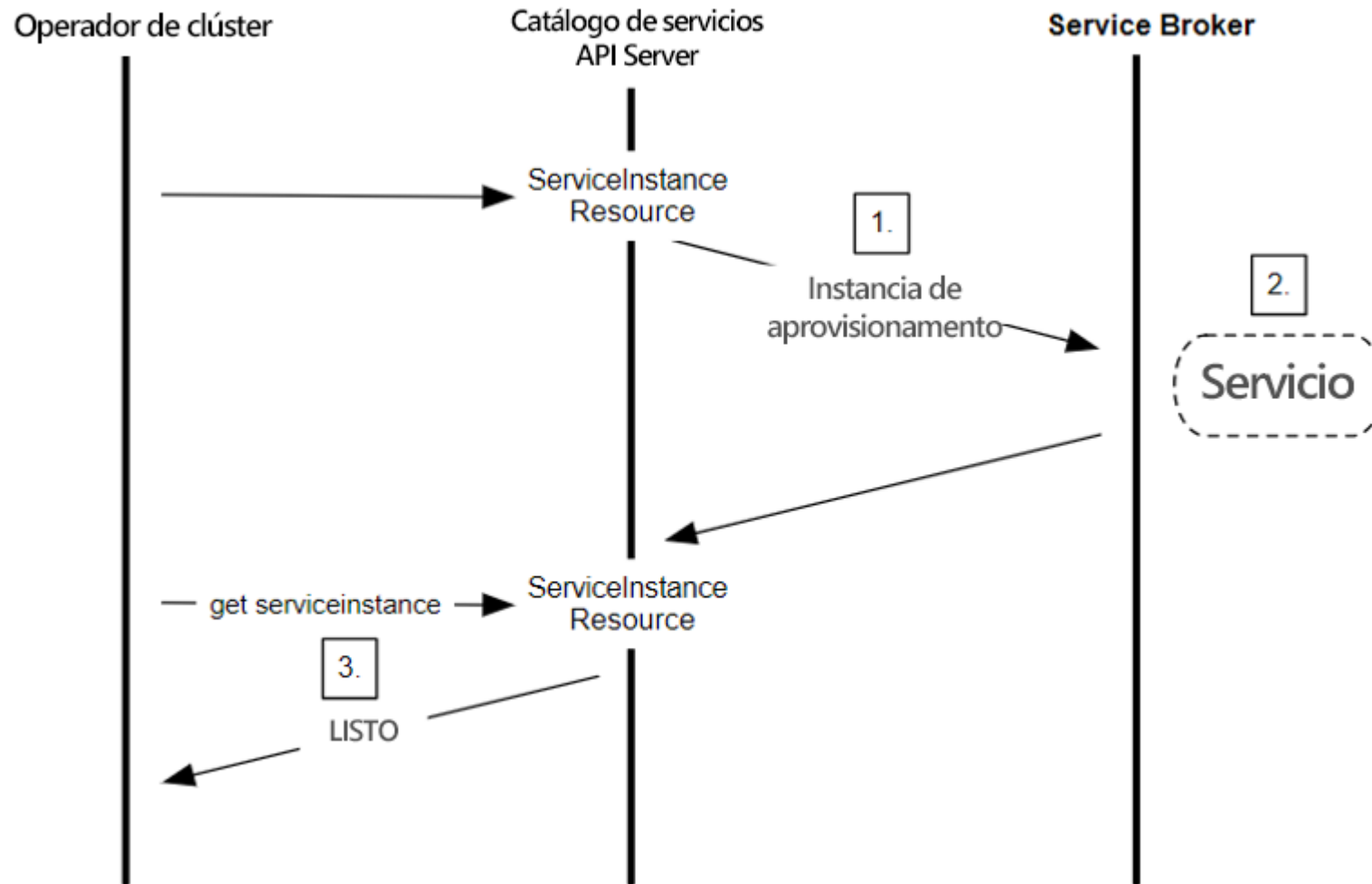
Catálogo de servicio de Kubernetes



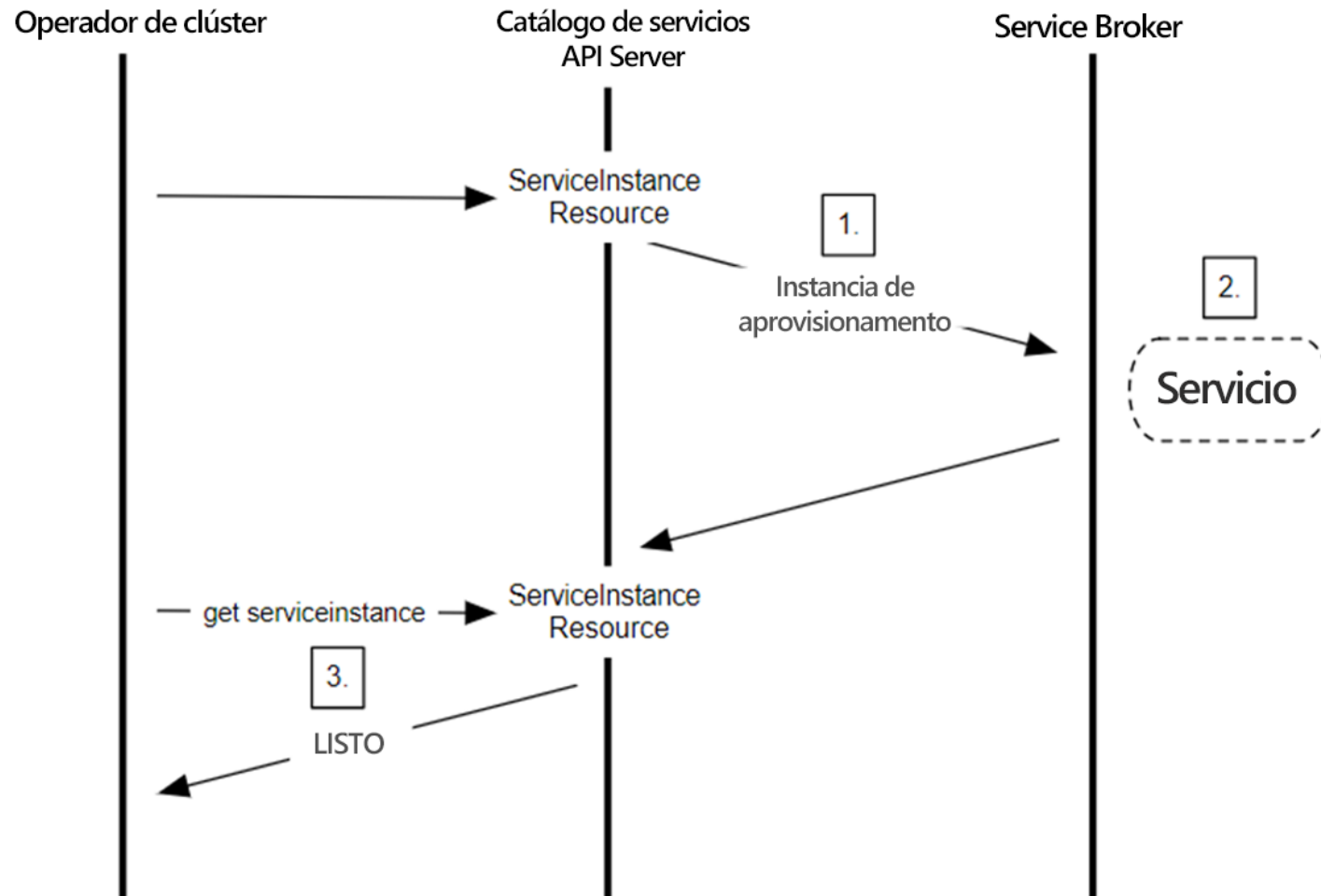
Listado de planes de servicios administrados



Aprovisionamiento de una nueva instancia



Enlace hacia un servicio administrado




Demostración

Service Broker for Cosmos DB

¿Y SRE (ingeniería de confiabilidad de sitio)?

Replicación de datos en todo el mundo

 **andrl-global - Replicate data globally**
Azure Cosmos DB account

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Quick start

Data Explorer


SETTINGS


Replicate data globally


Default consistency


Firewall

Keys

 Save


 Discard

 Manual Failover

 Automatic Failover

Click on a location to add or remove regions from your Azure Cosmos DB account.

* Each region is billable based on the throughput and storage for the account. [Learn more](#)



Agregar soporte multi-master

- Crear nueva cuenta Cosmos DB
- Habilitar multi-master
- Wins de última escritura (basados en propiedad de marca de tiempo definida por sistema)

The screenshot displays the 'Create CosmosDB Account' interface in the Azure portal. The 'Basics' tab is active, showing various configuration options. The 'DETAILS' section includes dropdowns for 'Subscription' and 'Resource Group', with a 'Create new' link for the latter. The 'INSTANCE' section contains fields for 'Account Name' (with a placeholder 'Enter account name'), 'API' (with a placeholder 'Please choose an API'), and 'Location' (set to 'Brazil South'). Below these are toggle switches for 'Geo-Redundancy' (set to 'Enable'), 'Multi Master' (set to 'Enable'), and 'Virtual Network' (set to 'Filter Virtual Networks'). A 'Subnet' dropdown is also present, currently showing '0 selected'. At the bottom, there are navigation buttons: 'Review + create' (highlighted in blue), 'Previous', 'Next: Summary', and a link to 'Download a template for automation'.

Configurar regiones

- Agregar regiones
- Seleccionar regiones o escribir regiones
- No hay necesidad de configurar la conmutación por error

Save Discard Manual Failover Automatic Failover

Click on a location to add or remove regions from your Azure Cosmos DB account.

* Each region is billable based on the throughput and storage for the account. [Learn more](#)



Configure regions

Configure the regions available for reads and writes. [+ Add region](#)

REGIONS	READS ENABLED	WRITES ENABLED	
North Central US	✓	✓	🗑️
North Europe	✓	✓	🗑️
Southeast Asia	✓	✓	🗑️

En su aplicación

- Política de conexión
 - UseMultipleWriteLocations = true
- Añadir regiones en orden preferido

```
//Define the connection policy
ConnectionPolicy policy = new ConnectionPolicy
{
    ConnectionMode = ConnectionMode.Direct,
    ConnectionProtocol = Protocol.Tcp,
    UseMultipleWriteLocations = true    //specify multiple write locations
};

//Add locations in preferred order
foreach (string region in regions)
{
    policy.PreferredLocations.Add(region);
}

//Create the client
DocumentClient client = new DocumentClient(new Uri(endpoint), authKey, policy);
```

¿Y la portabilidad?

¡Gracias!