



UNIVERSIDAD DE COLIMA

Dirección General de Educación Superior

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

INGENIERÍA EN COMPUTACIÓN INTELIGENTE

Semestre febrero 2022 – junio 2022

10/05/2022

Nombre: Ronaldo Adán Avalos de la Mora.

Semestre: 6

Materia: Computo en la nube **Práctica:** Cloud Services

Acerca de Cloud Services (soporte extendido)

Cloud Services (soporte extendido) es un nuevo modelo de implementación basado en `Azure Resource Manager` del producto `Azure Cloud Services`, y ahora está disponible con carácter general. Cloud Services (soporte extendido) tiene la ventaja principal de proporcionar resistencia regional junto con la paridad de características con `Azure Cloud Services` implementado mediante `Azure Service Manager`. También ofrece algunas funcionalidades de ARM, como el control de acceso basado en roles (RBAC), etiquetas y directivas, y admite plantillas de implementación.

Con este cambio, el modelo de implementación basado en `Azure Service Manager` para `Cloud Services` pasará a llamarse `Cloud Services (clásico)`. Conservará la capacidad de compilar e implementar rápidamente sus aplicaciones y servicios web y en la nube. Podrá escalar su infraestructura de servicios en la nube en función de la demanda actual y garantizar que se mantiene el rendimiento de sus aplicaciones al tiempo que se reducen los costes.

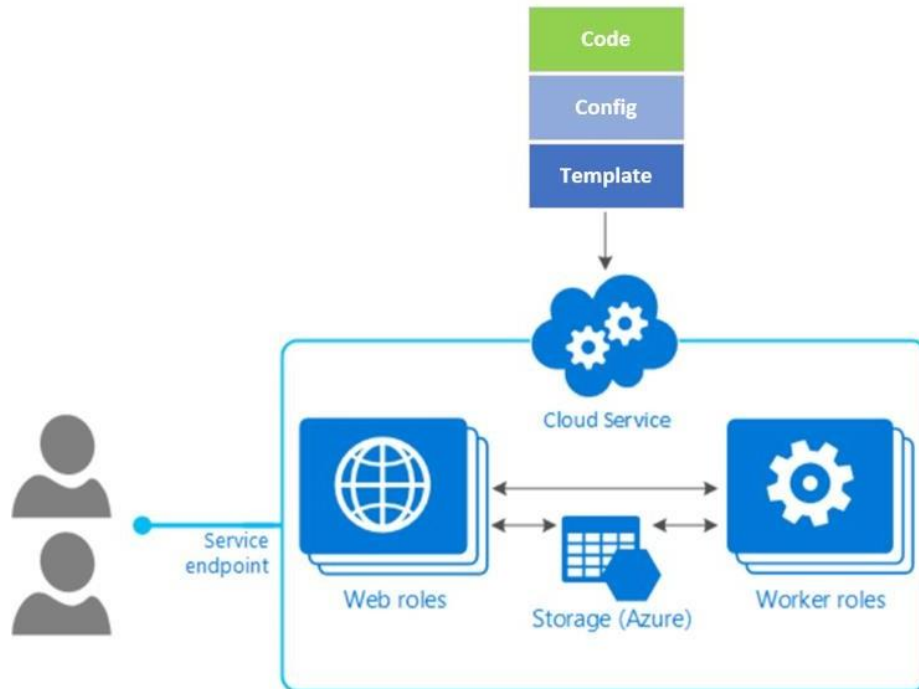
QUÉ NO CAMBIA

- El usuario crea el código, define las configuraciones y lo implementa en Azure. Azure configura el entorno de proceso, ejecuta su código y lo supervisa y mantiene automáticamente.
- Cloud Services (soporte extendido) también admite dos tipos de roles: `web` y `de trabajo`. No hay ningún cambio en el diseño, la arquitectura o los componentes de los roles `web` y `de trabajo`.

- Los tres componentes de un servicio en la nube —la definición del servicio (.csdef), la configuración del servicio (.cscfg) y el paquete del servicio (.cspkg)— se llevan a cabo y no hay ningún cambio en los formatos.
- No se requieren cambios en el código del entorno de ejecución, ya que el plano de datos es el mismo y el plano de control solo cambia.
- Las versiones de Azure GuestOS y las actualizaciones asociadas están alineadas con Cloud Services (clásico).
- El proceso de actualización subyacente con respecto a los dominios de actualización, cómo se aplica la actualización, la reversión y los cambios de servicio permitidos durante una actualización no cambian.

CAMBIOS EN EL MODELO DE IMPLEMENTACIÓN

Se requieren cambios mínimos en los archivos de configuración de servicios (.cscfg) y de definición de servicios (.csdef) para implementar Cloud Services (soporte extendido). No se requieren cambios en el código del entorno de ejecución. Sin embargo, los scripts de implementación deberán actualizarse para llamar a las nuevas API basadas en Azure Resource Manager.



Las principales diferencias entre Cloud Services (clásico) y Cloud Services (soporte extendido) con respecto a la implementación son las siguientes:

- Las implementaciones de Azure Resource Manager usan plantillas de ARM, que son un archivo JSON (notación de objetos JavaScript) que define tanto la infraestructura como la configuración de un proyecto. La plantilla usa sintaxis declarativa, lo que permite establecer lo que pretende implementar sin tener que escribir la secuencia de comandos de programación para crearla. El archivo de configuración del servicio y de definición del servicio deben ser coherentes con la plantilla de ARM al implementar Cloud Services (soporte extendido). Para ello, se puede crear manualmente la plantilla de ARM o utilizar PowerShell, Portal y Visual Studio.
- Los clientes deben usar Azure Key Vault para administrar certificados en Cloud Services (soporte extendido). Azure Key Vault le permite almacenar y administrar de forma segura las credenciales de la aplicación, como secretos, claves y certificados, en un repositorio central y seguro en la nube. Asimismo, las aplicaciones pueden autenticarse en Key Vault en tiempo de ejecución para recuperar las credenciales.
- Todos los recursos implementados a través de Azure Resource Manager deben estar dentro de una red virtual. Las redes virtuales y las subredes se crean en Azure Resource Manager con las API de Azure Resource Manager existentes y se debe hacer referencia a ellas en la sección NetworkConfiguration de .cscfg al implementar Cloud Services (soporte extendido).

- Cada servicio de Cloud Services (soporte extendido) es una única implementación independiente. Cloud Services (soporte extendido) no admite varias ranuras dentro de un único servicio en la nube.
 - La capacidad de intercambio de VIP se puede utilizar para intercambiar entre dos servicios de Cloud Services (soporte extendido). Para probar y ensayar una nueva versión de un servicio en la nube, implemente un servicio de Cloud Services (soporte extendido) y etiquételo como "con intercambio de VIP" con otro servicio de Cloud Services (soporte extendido)
- La etiqueta del servicio de nombres de dominio (DNS) es opcional para un servicio de Cloud Services (soporte extendido). En Azure Resource Manager, la etiqueta DNS es una propiedad del recurso de dirección IP pública asociada al servicio en la nube.

MIGRACIÓN A AZURE RESOURCE MANAGER

Cloud Services (soporte extendido) proporciona dos rutas de acceso para migrar desde Azure Service Manager a Azure Resource Manager.

1. Los clientes implementan servicios en la nube directamente en Azure Resource Manager y, a continuación, eliminan el antiguo servicio en la nube en Azure Service Manager.
2. La migración en contexto permite migrar Cloud Services (clásico) con un tiempo de inactividad mínimo o nulo a Cloud Services (soporte extendido).

OPCIONES DE MIGRACIÓN ADICIONALES

Al evaluar los planes de migración de Cloud Services (clásico) a Cloud Services (soporte extendido), es posible que desee investigar otros servicios de Azure, como: Virtual Machine Scale Sets, App Service, Azure Kubernetes Service y Azure Service Fabric. Estos servicios seguirán incorporando funcionalidades adicionales, mientras que Cloud Services (soporte extendido) mantendrá principalmente la paridad de características con Cloud Services (clásico).

En función de la aplicación, Cloud Services (soporte extendido) puede requerir menos esfuerzo para pasar a Azure Resource Manager en comparación con otras opciones. Si la aplicación no evoluciona, Cloud Services (soporte extendido) es una opción viable para tener en cuenta, ya que proporciona una ruta de migración rápida. Por el contrario, si la aplicación evoluciona continuamente y necesita un conjunto de características más moderno, explore otros servicios de Azure para abordar mejor sus requisitos actuales y futuros.

Requisitos previos para la implementación

ACTUALIZACIONES NECESARIAS DEL ARCHIVO DE CONFIGURACIÓN DE SERVICIO (.CSCFG)

1) Red virtual

Las implementaciones del servicio en la nube (soporte extendido) deben estar en una red virtual. La red virtual se puede crear por medio de [Azure portal](#), [flowerShell](#), la CLI de Azure o una [plantilla de ARM](#). Además, debe haber una referencia a la red virtual y las subredes en el archivo de configuración de servicio (.cscfg), en la sección `NetworkConfiguration`.

En el caso de las redes virtuales que pertenecen al mismo grupo de recursos que el servicio en la nube, basta con una referencia al nombre de la red virtual en el archivo de configuración de servicio (.cscfg). Si la red virtual y el servicio en la nube se encuentran en dos grupos de recursos diferentes, es necesario especificar el id. de Azure Resource Manager completo de la red virtual en el archivo de configuración de servicio (.cscfg).

2) Eliminación de complementos antiguos

- Quite la configuración antigua de Escritorio remoto del archivo de configuración de servicio (.cscfg).
- Quite la configuración de diagnóstico antigua de cada rol del archivo de configuración del servicio (.cscfg).

ACTUALIZACIONES NECESARIAS DEL ARCHIVO DE DEFINICIÓN DE SERVICIO (.CSDEF)

1) Tamaños de máquina virtual

Los tamaños siguientes están en desuso en Azure Resource Manager. Pero si quiere seguir usándolos, actualice el nombre `vmSize` con la convención de nomenclatura de Azure Resource Manager asociada.

Nombre de tamaño anterior	Nombre de tamaño actualizado
ExtraSmall	Standard_A1_v2
Pequeña	Standard_A1_v2
Media	Standard_A2_v2
grande	Standard_A4_v2
ExtraLarge	Standard_A8_v2
A5	Standard_A2m_v2
A6	Standard_A4m_v2
A7	Standard_A8m_v2
A8	Obsoleto
A9	Obsoleto
A10	Obsoleto
A11	Obsoleto
MSODSG5	Obsoleto

Por ejemplo, `<WorkerRole name="WorkerRole1" vmsize="Medium"` se convertiría en `<WorkerRole name="WorkerRole1" vmsize="Standard_A2"`.

2) Eliminación de los complementos antiguos de Escritorio remoto

- Las implementaciones que usan los complementos antiguos de Escritorio remoto deben eliminar los módulos del archivo de definición de servicio (.csdef) y cualquier certificado asociado.
- Las implementaciones que usaron los complementos de diagnóstico antiguos necesitan que se quite la configuración de cada rol del archivo de definición de servicio (.csdef).

CONTROL DE ACCESO

La suscripción que contiene recursos de red debe tener acceso de `colaborador de red` o superior para Cloud Services (soporte extendido). Para más información, consulte el artículo sobre los `roles integrados` de RBAC.

CREACIÓN DE KEY VAULT

Key Vault se usa para almacenar certificados asociados a Cloud Services (soporte extendido). Agregue los certificados a Key Vault y haga referencia a las huellas digitales del certificado en el archivo de configuración de servicio. También debe habilitar las "Directivas de acceso" (en el portal) en "Azure Virtual Machines para la implementación" en Key Vault, de modo que el recurso de Cloud Services (soporte extendido) pueda recuperar el certificado almacenado como secretos de Key Vault. Puede crear un almacén de claves en [Azure portal](#) o con [PowerShell](#). El almacén de claves debe crearse en la misma región y suscripción que el servicio en la nube. Para más información, consulte [Uso de certificados con Azure Cloud Services \(soporte extendido\)](#).

Configuración y empaquetado de archivos

Un servicio en la nube se crea a partir de tres componentes: la definición de servicio (.csdef), la configuración de servicio (.cscfg) y un paquete de servicio (.cspkg).

Los archivos `ServiceDefinition.csdef` y `ServiceConfiguration.cscfg` se basan ambos en XML y describen la estructura del servicio en la nube y cómo se configura; lo que se conoce en conjunto como modelo. `ServicePackage.cspkg` es un archivo ZIP que se genera a partir de `ServiceDefinition.csdef` y, entre otros, contiene todas las dependencias necesarias basadas en archivos binarios. Azure crea un servicio en la nube a partir de `ServicePackage.cspkg` y `ServiceConfiguration.cscfg`.

Una vez que se ejecuta el servicio en la nube en Azure, puede volver a configurarlo mediante el archivo `ServiceConfiguration.cscfg`, pero no puede alterar la definición.

SERVICEDEFINITION.CSDEF

El archivo `ServiceDefinition.csdef` especifica los valores que usa Azure para configurar un servicio en la nube. El esquema de definición de servicio de Azure (`serviceDefinition.schema.xml`) proporciona el formato permitido para un archivo de definición de servicio. En el ejemplo siguiente se muestra la configuración que se puede definir para los roles web y de trabajo:


```

1 <?xml version="1.0" encoding="utf-8"?>
2 <ServiceDefinition name="MyServiceName"
  xmlns="http://schemas.microsoft.com/ServiceHosting/2008/10/ServiceDefinition">
3   <WebRole name="WebRole1" vmSize="Standard_D1_v2">
4     <Sites>
5       <Site name="Web">
6         <Bindings>
7           <Binding name="HttpIn" endpointName="HttpIn" />
8         </Bindings>
9       </Site>
10    </Sites>
11    <Endpoints>
12      <InputEndpoint name="HttpIn" protocol="http" port="80" />
13      <InternalEndpoint name="InternalHttpIn" protocol="http" />
14    </Endpoints>
15    <Certificates>
16      <Certificate name="Certificate1" storeLocation="LocalMachine" storeName="My" />
17    </Certificates>
18    <Imports>
19      <Import moduleName="Connect" />
20      <Import moduleName="Diagnostics" />
21      <Import moduleName="RemoteAccess" />
22      <Import moduleName="RemoteForwarder" />
23    </Imports>
24    <LocalResources>
25      <LocalStorage name="localStoreOne" sizeInMB="10" />
26      <LocalStorage name="localStoreTwo" sizeInMB="10" cleanOnRoleRecycle="false" />
27    </LocalResources>
28    <Startup>
29      <Task commandLine="Startup.cmd" executionContext="limited" taskType="simple" />
30    </Startup>
31  </WebRole>
32
33  <WorkerRole name="WorkerRole1">
34    <ConfigurationSettings>
35      <Setting name="DiagnosticsConnectionString" />
36    </ConfigurationSettings>
37    <Imports>
38      <Import moduleName="RemoteAccess" />
39      <Import moduleName="RemoteForwarder" />
40    </Imports>
41    <Endpoints>
42      <InputEndpoint name="Endpoint1" protocol="tcp" port="10000" />
43      <InternalEndpoint name="Endpoint2" protocol="tcp" />
44    </Endpoints>
45  </WorkerRole>
46 </ServiceDefinition>

```

Puede ver el esquema de definición de servicio para entender mejor el esquema XML que se usa aquí, aunque a continuación se ofrece una explicación rápida de algunos de los elementos:

Sites

contiene las definiciones de sitios web o aplicaciones web que se hospedan en IIS7.

InputEndpoints

contiene las definiciones de los extremos que se usan para ponerse en contacto con el servicio en la nube.

InternalEndpoints

contiene las definiciones de los extremos que se usan en las instancias de rol para comunicarse entre sí.

ConfigurationSettings

contiene las definiciones de configuración de las características de un rol concreto.

Certificados

contiene las definiciones de los certificados que son necesarios para un rol. En el ejemplo de código anterior se muestra un certificado que se usa para la configuración de Azure Connect.

LocalResources

contiene las definiciones de los recursos de almacenamiento local. Un recurso de almacenamiento local es un directorio reservado en el sistema de archivos de la máquina virtual en la que se ejecuta una instancia de un rol.

Importaciones

contiene las definiciones de los módulos importados. El ejemplo de código anterior muestra los módulos de Conexión a Escritorio remoto y Azure Connect.

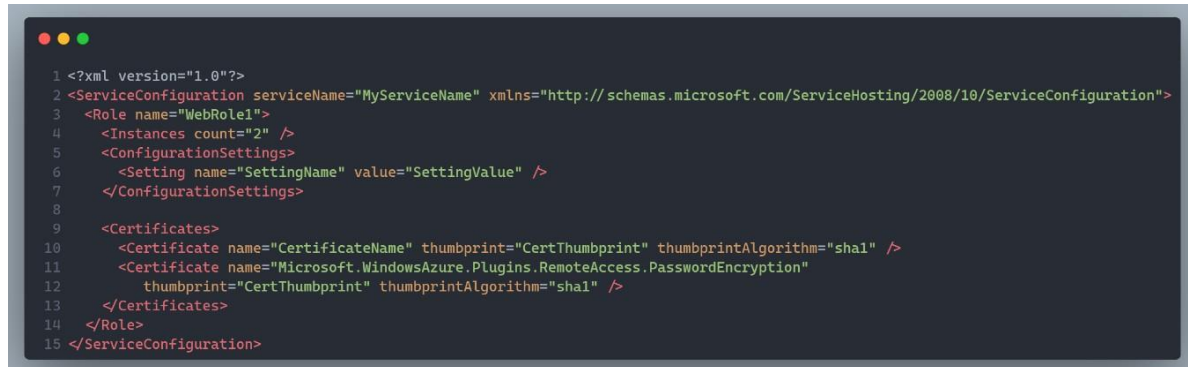
Startup

contiene las tareas que se ejecutan cuando se inicia el rol. Las tareas se definen en un archivo ejecutable o .cmd.

SERVICECONFIGURATION.CSCFG

La configuración de los valores del servicio en la nube viene determinada por los valores del archivo `ServiceConfiguration.cscfg`. Especifique el número de instancias que desea implementar para cada rol en este archivo. Los valores de configuración que ha definido en el archivo de definición de servicio se agregan al archivo de configuración de servicio. Las huellas digitales de los certificados de administración que están asociados con el servicio en la nube también se agregan al archivo. El `esquema de configuración de servicio de Azure` (archivo `.cscfg`) proporciona el formato permitido para un archivo de configuración de servicio.

El archivo de configuración de servicio no se empaqueta con la aplicación, sino que se carga en Azure como un archivo independiente y se usa para configurar el servicio en la nube. Puede cargar un nuevo archivo de configuración de servicio sin volver a implementar el servicio en la nube. Los valores de configuración del servicio en la nube pueden cambiarse mientras el servicio en la nube se está ejecutando. En el ejemplo siguiente se muestran los valores de configuración que se pueden definir para los roles web y de trabajo:

A screenshot of a code editor showing an XML configuration file. The XML is a ServiceConfiguration file for a service named 'MyServiceName'. It defines a role named 'WebRole1' with 2 instances. It also defines configuration settings, including a setting named 'SettingName' with a value of 'SettingValue'. Finally, it defines certificates, including a certificate named 'CertificateName' with a thumbprint of 'CertThumbprint' and a thumbprint algorithm of 'sha1'. The XML is well-formed and uses the 'http://schemas.microsoft.com/ServiceHosting/2008/10/ServiceConfiguration' namespace.

```
1 <?xml version="1.0"?>
2 <ServiceConfiguration serviceName="MyServiceName" xmlns="http://schemas.microsoft.com/ServiceHosting/2008/10/ServiceConfiguration">
3   <Role name="WebRole1">
4     <Instances count="2" />
5     <ConfigurationSettings>
6       <Setting name="SettingName" value="SettingValue" />
7     </ConfigurationSettings>
8   </Role>
9   <Certificates>
10    <Certificate name="CertificateName" thumbprint="CertThumbprint" thumbprintAlgorithm="sha1" />
11    <Certificate name="Microsoft.WindowsAzure.Plugins.RemoteAccess.PasswordEncryption"
12      thumbprint="CertThumbprint" thumbprintAlgorithm="sha1" />
13  </Certificates>
14 </ServiceConfiguration>
```

Puede hacer referencia al [esquema de configuración de servicio](#) para comprender mejor el esquema XML que se usa aquí; sin embargo, a continuación se da una explicación rápida de los elementos:

Instancias

configura el número de instancias en ejecución para el rol. Para evitar la posibilidad de que el servicio en la nube deje de estar disponible durante las actualizaciones, es recomendable que implemente más de una instancia de los roles accesibles a través de web. Al hacerlo, estará siguiendo las instrucciones del [Acuerdo de Nivel de Servicio de Azure Compute](#), que garantiza la conectividad externa del 99,95 % para los roles accesibles a través de Internet cuando se implementan dos o más instancias de rol para un servicio.

ConfigurationSettings

configura los valores de las instancias en ejecución de un rol. El nombre de los elementos `<Setting>` debe coincidir con las definiciones de configuración del archivo de definición de servicio.

Certificados

configura los certificados usados por el servicio. En el ejemplo de código anterior se muestra cómo definir el certificado para el módulo RemoteAccess. El valor del atributo thumbprint debe establecerse en la huella digital del certificado que se va a usar.

DEFINICIÓN DE LOS PUERTOS DE LAS INSTANCIAS DE ROL

Azure permite un solo punto de entrada a un rol web. Esto significa que todo el tráfico se produce a través de una dirección IP. Puede configurar sus sitios web

para que compartan un puerto configurando el encabezado host para dirigir la solicitud a la ubicación correcta. También puede configurar las aplicaciones para que escuchen en puertos conocidos de la dirección IP.

En el ejemplo siguiente se muestra la configuración de un rol web con un sitio web y una aplicación web. El sitio web se configura como la ubicación de entrada predeterminada en el puerto 80 y las aplicaciones web se configuran para recibir solicitudes de un encabezado host alternativo que se llama "mail.mysite.cloudapp.net".

```
1 <WebRole>
2   <ConfigurationSettings>
3     <Setting name="DiagnosticsConnectionString" />
4   </ConfigurationSettings>
5   <Endpoints>
6     <InputEndpoint name="HttpIn" protocol="http" port="80" />
7     <InputEndpoint name="Https" protocol="https" port="443" certificate="SSL" />
8     <InputEndpoint name="NetTcp" protocol="tcp" port="808" certificate="SSL" />
9   </Endpoints>
10  <LocalResources>
11    <LocalStorage name="Sites" cleanOnRoleRecycle="true" sizeInMB="100" />
12  </LocalResources>
13  <Site name="Mysite" packageDir="Sites\Mysite">
14    <Bindings>
15      <Binding name="http" endpointName="HttpIn" />
16      <Binding name="https" endpointName="Https" />
17      <Binding name="tcp" endpointName="NetTcp" />
18    </Bindings>
19  </Site>
20  <Site name="MailSite" packageDir="MailSite">
21    <Bindings>
22      <Binding name="mail" endpointName="HttpIn" hostHeader="mail.mysite.cloudapp.net" />
23    </Bindings>
24    <VirtualDirectory name="artifacts" />
25    <VirtualApplication name="storageproxy">
26      <VirtualDirectory name="packages" packageDir="Sites\storageProxy\packages" />
27    </VirtualApplication>
28  </Site>
29 </WebRole>
```

CAMBIO DE LA CONFIGURACIÓN DE UN ROL

Puede actualizar la configuración de su servicio en la nube mientras se ejecuta en Azure, sin necesidad de desconectarlo. Para cambiar la información de configuración, puede cargar un nuevo archivo de configuración, o editar el archivo de configuración existente y aplicarlo al servicio en ejecución. Pueden realizarse los siguientes cambios en la configuración de un servicio:

- **Cambiar los valores de configuración:** Cuando los valores de una configuración cambian, una instancia de rol puede elegir aplicar el cambio mientras la instancia está en línea, o bien reciclar la instancia correctamente y aplicar el cambio mientras la instancia está sin conexión.

- **Cambiar la topología de servicio de las instancias de rol:** Los cambios en la topología no afectan a las instancias en ejecución, excepto donde se vaya a eliminar una instancia. Por lo general, el resto de instancias no es necesario reciclarlas; sin embargo, puede decidir hacerlo en respuesta a un cambio en la topología.
- **Cambiar la huella digital del certificado:** Solo puede actualizar un certificado cuando una instancia de rol está sin conexión. Si un certificado se agrega, elimina o cambia mientras la instancia de rol está en línea, Azure dejará la instancia sin conexión para actualizar el certificado y la volverá a poner en línea una vez completado el cambio.

CONTROL DE LOS CAMBIOS DE CONFIGURACIÓN CON EVENTOS DE TIEMPO DE EJECUCIÓN DE SERVICIO

La biblioteca en tiempo de ejecución de Azure incluye el espacio de nombres `Microsoft.WindowsAzure.ServiceRuntime`, que proporciona clases para interactuar con el entorno de Azure desde un rol. La clase `RoleEnvironment` define los siguientes eventos que se producen antes y después de un cambio de configuración:

- **Cambiar evento**
Esto se produce antes de que el cambio de configuración se aplique a una instancia especificada de un rol, lo que le ofrece la oportunidad de dar de baja las instancias de rol, en caso necesario.
- **Evento cambiado**
Se produce después de que el cambio de configuración se aplica a una instancia especificada de un rol.

`ServicePackage.cspkg`

Para implementar una aplicación como un servicio en la nube de Azure, primero debe empaquetar la aplicación en el formato adecuado. Puede usar la herramienta de línea de comandos `CSflack` (que se instala con el `SDK de Azure`) para crear el archivo de paquete como una alternativa a Visual Studio.

`CSflack` usa el contenido del archivo de configuración de servicio y del archivo de definición de servicio para definir el contenido del paquete. `CSflack` genera un archivo de paquete de aplicación (`.cspkg`) que puede cargar en Azure mediante el `portal` de Azure. De forma predeterminada, el paquete se denomina `[ServiceDefinitionFileName].cspkg`, pero puede especificar un nombre diferente mediante la opción `/out` de `CSflack`.

COMANDO DE EJEMPLO PARA EMPAQUETAR UN SERVICIO EN LA NUBE

En el ejemplo siguiente se crea un paquete de aplicación que contiene la información de un rol web. El comando especifica el archivo de definición de

servicio que se usará, el directorio donde se pueden encontrar los archivos binarios y el nombre del archivo de paquete.

```
cmd Copiar

cspack [DirectoryName]\[ServiceDefinition]
      /role:[RoleName];[RoleBinariesDirectory]
      /sites:[RoleName];[VirtualPath];[PhysicalPath]
      /out:[OutputFileName]
```

Si la aplicación contiene un rol web y un rol de trabajo, se usa el siguiente comando:

```
cmd Copiar

cspack [DirectoryName]\[ServiceDefinition]
      /out:[OutputFileName]
      /role:[RoleName];[RoleBinariesDirectory]
      /sites:[RoleName];[VirtualPath];[PhysicalPath]
      /role:[RoleName];[RoleBinariesDirectory];[RoleAssemblyName]
```

Donde las variables se definen como de la manera siguiente:

Variable	Value
[DirectoryName]	El subdirectorio bajo el directorio raíz del proyecto que contiene el archivo .csdef del proyecto de Azure.
[ServiceDefinition]	El nombre del archivo de definición de servicio. De forma predeterminada, este archivo se denomina ServiceDefinition.csdef.
[OutputFileName]	El nombre del archivo de paquete generado. Normalmente, se establece en el nombre de la aplicación. Si no se especifica ningún nombre de archivo, el paquete de aplicación se crea como [ApplicationName].cspkg.
[RoleName]	El nombre del rol, tal y como se define en el archivo de definición de servicio.
[RoleBinariesDirectory]	La ubicación de los archivos binarios para el rol.
[VirtualPath]	Los directorios físicos de cada ruta de acceso virtual definida en la sección Sites de la definición de servicio.
[PhysicalPath]	Los directorios físicos del contenido de cada ruta de acceso virtual definida en el nodo de sitio de la definición de servicio.
[RoleAssemblyName]	El nombre del archivo binario del rol.