

**WebSphere** IBM WebSphere Process Server for Multiplatforms  
Versión 7.0.0

*Common Event Infrastructure*





**WebSphere** IBM WebSphere Process Server for Multiplatforms  
Versión 7.0.0

*Common Event Infrastructure*



**Abril de 2010**

Esta edición se aplica a la versión 7, release 0, modificación 0 de WebSphere Process Server for Multiplatforms (número de producto 5724-L01) y a todos los releases y las modificaciones subsiguientes hasta que se indique lo contrario en nuevas ediciones.

Para enviar comentarios sobre este documento, envíe un mensaje de correo electrónico a [doc-comments@us.ibm.com](mailto:doc-comments@us.ibm.com). Esperamos sus comentarios.

Cuando se envía información a IBM, se otorga a IBM un derecho no exclusivo de utilizar o distribuir la información del modo que estime apropiado sin incurrir por ello en ninguna obligación con el remitente.

© Copyright IBM Corporation 2007, 2010.

# Contenido

## Capítulo 1. Common Event

### Infraestructure . . . . . 1

Modelo de Common Base Event. . . . . 2

## Capítulo 2. Configuración de Common Event Infraestructure . . . . . 5

Componentes de Common Event Infraestructure. . . 5

Configuración de Common Event Infraestructure mediante la consola administrativa . . . . . 7

Despliegue de la aplicación Common Event Infraestructure . . . . . 9

Despliegue de Common Event Infraestructure en un clúster . . . . . 10

Configuración de la mensajería de sucesos . . . . 12

Configuración de colas JMS adicionales . . . . 13

Configuración de la mensaje de suceso mediante un proveedor JMS externo . . . . . 13

Configuración de alias de autenticación JMS . . 14

Configuración de la base de datos de sucesos . . 15

Limitaciones de la base de datos de sucesos . . 15

Configuración de una base de datos de sucesos Derby . . . . . 17

Configuración de una base de datos de sucesos DB2 (sistemas Linux, UNIX y Windows ) . . . 18

Configuración de una base de datos DB2 en un sistema z/OS . . . . . 20

Configuración de una base de datos DB2 en un sistema iSeries . . . . . 22

Configuración de una base de datos de sucesos Informix . . . . . 23

Configuración de una base de datos de sucesos Oracle . . . . . 25

Configuración de una base de datos de sucesos de SQL Server . . . . . 27

Ejecución manual de los scripts de configuración de base de datos. . . . . 28

Actualización de la base de datos de sucesos desde una versión anterior . . . . . 35

Configuración de Common Event Infraestructure entre células para WebSphere Business Monitor . . 40

Eliminación de la configuración de Common Event Infraestructure. . . . . 40

Suprimir la base de datos de sucesos . . . . . 40

Eliminación de la aplicación Common Event Infraestructure. . . . . 41

Supresión de la mensajería de sucesos del servidor Common Event Infraestructure . . . . 42

## Capítulo 3. Administración de Common Event Infraestructure . . . . . 43

Administración del servicio de sucesos con la consola administrativa. . . . . 43

Habilitación e inhabilitación del servicio de sucesos con la consola administrativa. . . . . 43

Creación de una fábrica de emisores de sucesos mediante la consola administrativa . . . . . 44

Creación de un grupo de sucesos mediante la consola administrativa. . . . . 44

Creación de un filtro de sucesos mediante la consola administrativa. . . . . 45

Administración del servicio de sucesos mediante scripts . . . . . 46

Habilitación del servicio de sucesos mediante scripts . . . . . 46

Inhabilitación del servicio de sucesos mediante scripts . . . . . 47

Creación de una fábrica de emisores con scripts . 47

Creación de un grupo de sucesos con scripts . . 50

Creación de un filtro de sucesos con scripts . . 52

Anotación cronológica y rastreo de los componentes de Common Event Infraestructure . . . . . 53

Mantenimiento de la base de datos de sucesos. . . 54

Actualización de las estadísticas de base de datos de sucesos DB2 . . . . . 54

Reorganización de las tablas de base de datos de sucesos DB2 . . . . . 55

Depuración de sucesos de la base de datos de sucesos . . . . . 56

## Capítulo 4. Protección del acceso a las funciones de Common Event Infraestructure . . . . . 63

## Capítulo 5. Resolución de problemas de Common Event Infraestructure . . . . 67

Problemas durante el inicio . . . . . 67

El servicio de sucesos no se inicia (mensaje CEIDS0058E) . . . . . 67

Problemas al enviar sucesos . . . . . 67

Error al enviar suceso (mensaje CEIDS0060E) . . 67

Error al enviar suceso (ServiceUnavailableException) . . . . . 68

Error al enviar suceso (NameNotFoundException) . . . . . 69

Error al enviar suceso (mensaje CEIEM0025E) . . 70

Error al enviar suceso (mensaje CEIEM0034E) . . 70

El suceso no es válido (mensaje CEIEM0027E). . 70

Modalidad de sincronización no soportada (mensaje CEIEM0015E) . . . . . 71

Modalidad de transacción no soportada (mensaje CEIEM0016E). . . . . 72

Problemas al recibir o consultar sucesos . . . . . 73

Error al consultar sucesos (mensaje CEIDS0060E) 73

Los sucesos no se almacenan en el almacén de datos persistente. . . . . 74

Los clientes no reciben los sucesos (ningún mensaje de error) . . . . . 75

Los clientes no reciben los sucesos (NameNotFoundException) . . . . . 76

El grupo de sucesos con elementos de datos ampliados no contiene sucesos . . . . .	76
Error al consultar un grupo de sucesos (mensaje CEIES0048E) . . . . .	78
Problemas varios . . . . .	78
La consulta de patrón del catálogo de sucesos falla en un sistema Windows . . . . .	78

**Capítulo 6. Mandatos de Common  
Event Infrastructure . . . . . 79**

Mandato configEventServiceDB2DB . . . . .	79
Mandato configEventServiceDB2iSeriesDB . . . . .	82
Mandato configEventServiceDB2ZOSDB . . . . .	84
Mandato configEventServiceDerbyDB . . . . .	87
Mandato configEventServiceInformixDB . . . . .	89
Mandato configEventServiceOracleDB . . . . .	91
Mandato configEventServiceSQLServerDB . . . . .	94
Mandato deployEventService . . . . .	96

Mandato deployEventServiceMdb . . . . .	98
Mandato setEventServiceJmsAuthAlias . . . . .	99
Mandato enableEventService . . . . .	100
Mandato disableEventService . . . . .	102
Mandato showEventServiceStatus . . . . .	103
Mandato removeEventService . . . . .	104
Mandato removeEventServiceMdb . . . . .	105
Mandato removeEventServiceDB2DB . . . . .	106
Mandato removeEventServiceDB2iSeriesDB . . . . .	108
Mandato removeEventServiceDB2ZOSDB . . . . .	109
Mandato removeEventServiceDerbyDB . . . . .	110
Mandato removeEventServiceInformixDB . . . . .	112
Mandato removeEventServiceOracleDB . . . . .	113
Mandato removeEventServiceSQLServerDB . . . . .	114
Programa de utilidad de línea de mandatos eventbucket . . . . .	116
Programa de utilidad de línea de mandatos eventpurge . . . . .	117

---

## Capítulo 1. Common Event Infrastructure

Common Event Infrastructure es una tecnología incorporable concebida para proporcionar servicios de gestión de sucesos básicos a las aplicaciones que requieren estos servicios.

Esta infraestructura de sucesos hace las veces de punto de integración para consolidar y conservar los sucesos en bruto procedentes de diferentes fuentes heterogéneas y distribuir dichos sucesos a los consumidores de sucesos. Los sucesos se representan con el modelo Common Base Event, un formato estándar basado en XML que define la estructura del suceso. Para obtener más información, consulte el subtema de modelo de Common Base Event.

Con esta infraestructura común, productos diferentes que no se asocian rigurosamente entre sí pueden integrar su gestión de sucesos, con lo cual se obtiene una vista completa de los recursos de la empresa y se correlacionan los sucesos a través de los límites de los dominios. Por ejemplo, los sucesos generados por una aplicación de supervisión de redes se pueden correlacionar con sucesos generados por una aplicación de seguridad. Dicha correlación puede ser difícil de lograr cuando cada producto utiliza su propio método para la gestión de sucesos.

Common Event Infrastructure proporciona recursos para generar, propagar, conservar y consumir sucesos pero no define los sucesos. En su lugar, son los desarrolladores de aplicaciones y los administradores los que definen los tipos de sucesos, los grupos de sucesos, los filtros y la correlación.

### Componentes de Common Event Infrastructure

Common Event Infrastructure consta de los siguientes componentes principales:

#### Suceso básico común

El componente Common Base Event da soporte a la creación de sucesos y al acceso a sus datos de propiedades. Los orígenes de sucesos utilizan las API de Common Base Event para crear sucesos nuevos que se ajustan al modelo Common Base Event, a continuación, los consumidores de sucesos utilizan las API para leer los datos de las propiedades de los sucesos recibidos. Asimismo, las aplicaciones pueden convertir sucesos al formato de texto XML y también convertirlos desde este formato, dando soporte al intercambio con otras herramientas. El componente Common Base Event forma parte de la plataforma TPTP (Test and Performance Tools Platform) de Eclipse.

#### Emisor

El componente de emisor da soporte al envío de sucesos. Después de que un origen de sucesos cree un suceso y lo rellene con datos, el origen de suceso somete el suceso a un emisor. El emisor realiza opcionalmente la cumplimentación automática del contenido y, a continuación, valida el suceso para asegurarse de que se ajusta a la especificación Common Base Event. Asimismo, compara el suceso con criterios de filtro configurables. Si el suceso es válido y pasa el criterio de filtro, el emisor envía el suceso al servicio de sucesos. Un emisor puede enviar sucesos al servicio de sucesos de forma síncrona (utilizando llamadas JavaBeans™) o de forma asíncrona (utilizando una cola JMS (Java™ Message Service)).

### Servicio de sucesos

El servicio de sucesos es el conducto entre los orígenes de sucesos y los consumidores de sucesos. El servicio de sucesos recibe los sucesos que envían los orígenes de sucesos a los emisores. Los almacena en un almacén de datos persistente y, a continuación, los distribuye de forma asíncrona a los consumidores de sucesos suscritos. Asimismo, el servicio de sucesos da soporte a las consultas de sucesos históricos del almacén persistente.

### Catálogo de sucesos

El catálogo de sucesos es un depósito de metadatos de sucesos. Las aplicaciones utilizan el catálogo de sucesos para recuperar la información sobre las clases de sucesos y el contenido permitido.

Asimismo, una aplicación o solución que utilice Common Event Infrastructure también puede incluir los componentes siguientes (que no forman parte de la propia infraestructura):

### Origen de suceso

Un origen de suceso es cualquier aplicación que utilice un emisor para enviar sucesos a un servicio de sucesos.

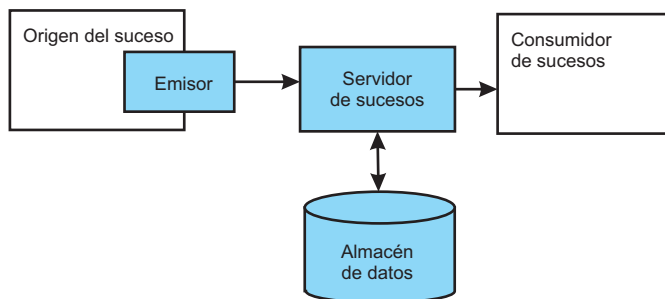
### Consumidor de sucesos

Un consumidor de sucesos es cualquier aplicación que recibe sucesos del servicio de sucesos.

### Aplicación del catálogo de sucesos

Una aplicación del catálogo de sucesos es cualquier aplicación que almacene o recupere metadatos de sucesos en el catálogo de sucesos. Puede ser una herramienta de gestión o de desarrollo, o bien un origen de suceso o consumidor de sucesos.

En el siguiente diagrama se muestra el flujo general de sucesos desde el origen de sucesos al consumidor de sucesos utilizando Common Event Infrastructure.



---

## Modelo de Common Base Event

El modelo de Common Base Event es un estándar que define una representación común de sucesos ideada para su uso en aplicaciones de empresa y de gestión de empresas. Este estándar, desarrollado por IBM® Autonomic Computing Architecture Board, da soporte a la codificación de sucesos de anotaciones cronológicas, rastreos, gestión y empresa utilizando un formato común basado en XML, lo que hace posible correlacionar diferentes tipos de sucesos que se originan en diferentes aplicaciones. El modelo Common Base Event forma parte de IBM Autonomic Computing Toolkit; si desea información, visite la dirección <http://www.ibm.com/autonomic>.



Common Event Infrastructure actualmente da soporte a la versión 1.0.1 de la especificación.

El concepto básico detrás del modelo Common Base Event es la *situación*. Una situación puede ser cualquier cosa que ocurre en la infraestructura de sistemas como, por ejemplo, si concluye el servidor, un error de unidad de disco o un inicio de usuario erróneo. El modelo de Common Base Event define un conjunto de tipos de situaciones estándar que cubre la mayor parte de las situaciones que pueden surgir (por ejemplo, StartSituation y CreateSituation).

Un *suceso* es una notificación estructurada que genera información relacionada con una situación. Un suceso notifica tres tipos de información:

- La situación propiamente dicha (lo que ha pasado)
- La identidad del componente afectado (por ejemplo, el servidor que ha concluido)
- La identidad del componente que notifica la situación (que puede ser el mismo que el componente afectado)

La especificación Common Base Event define un suceso como un elemento XML que contiene propiedades que proporcionan los tres tipos de información. Estas propiedades están codificadas como atributos y subelementos del elemento raíz, CommonBaseEvent.

El formato de Common Base Event se puede extender. Además de las propiedades de suceso estándares, un suceso también puede contener elementos de datos ampliados, que son elementos específicos de la aplicación que pueden contener cualquier tipo de información asociada a la situación. El atributo *extensionName* etiqueta un suceso con un nombre de clasificación opcional (una clase de suceso), que indica a las aplicaciones la clase de elementos de datos ampliados que se han de esperar. El catálogo de sucesos almacena las definiciones de suceso que describen estas clases de suceso y el contenido permitido.

Para obtener información detallada sobre el formato Common Base Event, consulte el documento de especificación y el esquema XSD que se incluye en IBM Autonomic Computing Toolkit.



---

## Capítulo 2. Configuración de Common Event Infrastructure

Puede configurar los recursos de Common Event Infrastructure o cambiar recursos existentes, utilizando el objeto AdminTask del servidor.

### Acerca de esta tarea

Common Event Infrastructure (CEI) se puede instalar con una configuración por omisión completamente funcional en una configuración de servidor autónomo. Esta configuración sólo se debe llevar a cabo para crear un perfil de servidor autónomo mediante la herramienta de gestión de perfiles. En todos los otros casos, debe utilizar la consola de administración para configurar CEI, por ejemplo, cuando está instalando en un entorno de despliegue de red o en un clúster, para asegurarse de que la configuración es adecuada para su sistema.

También puede utilizar el mandato wsadmin para configurar CEI o puede utilizar el mandato para alterar una configuración CEI existente. En cualquiera de los casos, debe cambiar la configuración de CEI utilizando el objeto AdminTask del servidor para ejecutar mandatos administrativos.

Después de cambiar la configuración CEI, debe reiniciar el servidor o el clúster.

---

## Componentes de Common Event Infrastructure

Los componentes de Common Event Infrastructure se instalan como parte de un conjunto de aplicaciones, servicios y recursos en el servidor.

Cuando configura Common Event Infrastructure, se crean varios componentes que se despliegan en el servidor.

### Servicio Common Event Infrastructure

Un servicio que se instala en el servidor y permite que las aplicaciones y clientes utilicen Common Event Infrastructure. Puede ver la configuración del servicio Common Event Infrastructure en la consola administrativa, como se indica a continuación:

- Para un servidor, seleccione **Servidores > Servidores de aplicaciones > nombre\_servidor > Business Integration > Common Event Infrastructure > Servicio Common Event Infrastructure**.
- Para un clúster, seleccione **Servidores > Clústeres > nombre\_clúster > Business Integration > Common Event Infrastructure > Servicio Common Event Infrastructure**.

Si el recuadro de selección "Habilitar el servidor de la infraestructura de sucesos" está seleccionado, el servicio se instala y ejecuta o se inicia después de que reinicie el servidor o el clúster. Si no está seleccionado, entonces no se instala el servicio o se desinstala después de reiniciar el servidor o el clúster.

### Valores del servicio de sucesos

Un conjunto de propiedades que utiliza el servicio de sucesos para habilitar la distribución de sucesos y la persistencia mediante el almacén de datos. Por lo general, no es necesario realizar ninguna actividad de configuración para este recurso, pero es posible que deba crear valores del servicio de sucesos adicionales si desea configurar varios servicios de

sucesos en la misma célula. Para ver los valores del servicio de sucesos, pulse **Integración de servicios > Servicio de suceso > Valores del servicio de sucesos**.

### **Configuración de la mensajería de sucesos**

Los recursos que dan soporte a la transmisión de sucesos asíncrona para el servicio de sucesos utilizando el servicio JMS (Java Messaging Service). La configuración de la mensajería de sucesos utiliza la mensajería incorporada del servidor. De forma opcional puede configurar un proveedor JMS externo para la mensajería de sucesos.

### **Base de datos de sucesos**

La base de datos de sucesos se utiliza para almacenar de forma permanente los sucesos que recibe el servicio de sucesos. La base de datos Derby se incluye como parte del servidor pero no se recomienda utilizarla en entornos de producción. En su lugar, puede configurar una base de datos de sucesos externa en los productos siguientes: DB2, Oracle, SQLServer e Informix.

### **Plug-in de filtro de sucesos**

Un plug-in de filtro se utiliza para filtrar sucesos en el origen utilizando selectores de sucesos XPath. Para configurar las propiedades del filtro, pulse **Integración de servicios > Common Event Infrastructure > Fábricas de emisores de sucesos > Valores del filtro de sucesos**.

### **Fábrica de emisores**

Una fábrica de emisores es un objeto que utilizan los orígenes de sucesos para crear emisores; un emisor se utiliza para enviar sucesos al servicio de sucesos. Las propiedades de una fábrica de emisores afectan al comportamiento de cualquier emisor creado utilizando esa fábrica de emisores. Para ver las fábricas de emisores disponibles, pulse **Integración de servicios > Common Event Infrastructure > Fábricas de emisores de sucesos**.

### **Transmisión del servicio de sucesos**

Una transmisión del servicio de sucesos es un objeto que define propiedades que determinan cómo acceden los emisores al servicio de sucesos de forma asíncrona utilizando llamadas EJB; estas propiedades las utilizan las fábricas de emisores al crear nuevos emisores. Puede ver o cambiar las transmisiones del servicio de sucesos disponibles en los valores de la fábrica de emisores.

### **Transmisión JMS**

Una transmisión JMS es un objeto que define propiedades que determinan cómo acceden los emisores al servicio de sucesos de forma asíncrona utilizando una cola JMS; estas propiedades las utilizan las fábricas de emisores al crear nuevos emisores. Puede ver o cambiar las transmisiones JMS disponibles en los valores de la fábrica de emisores.

### **Grupo de sucesos**

Un grupo de sucesos es una colección lógica de sucesos que se utilizan para categorizar sucesos en función de su contenido. Al consultar sucesos en el servicio de sucesos o al suscribirse a una distribución de sucesos, un consumidor de sucesos puede especificar un grupo de sucesos de modo que sólo recupere los sucesos de dicho grupo. Los grupos de sucesos también se pueden utilizar para especificar qué sucesos se deben almacenar en el almacén de datos persistente. Para ver los grupos de sucesos disponibles en la consola administrativa, pulse **Integración de**

---

## Configuración de Common Event Infrastructure mediante la consola administrativa

Configure Common Event Infrastructure mediante la consola administrativa del servidor.

### Acerca de esta tarea

Abra el panel Common Event Infrastructure Server de la consola administrativa:

Si va a configurar un servidor, seleccione **Servidores > Tipos de servidores > Servidores de aplicaciones WebSphere > nombre\_servidor > Business Integration > Common Event Infrastructure > Servidor de Common Event Infrastructure**.

Si va a configurar un clúster, pulse **Servidores > Clústeres > Clústeres de servidores de aplicaciones WebSphere > nombre\_clúster > Business Integration > Common Event Infrastructure > Servidor de Common Event Infrastructure**.

### Procedimiento

1. Habilite el despliegue de la aplicación de empresa Common Event Infrastructure seleccionando el recuadro de selección **Habilitar el servidor de la infraestructura de sucesos**. Si el servidor ya se ha configurado, puede habilitarlo o inhabilitarlo si selecciona o deselecciona el recuadro de selección. Si el recuadro de selección está deseleccionado, entonces Common Event Infrastructure no se ha configurado o su configuración anterior se ha inhabilitado pero no se ha reiniciado el servidor. Un mensaje de información muestra si este destino de despliegue se ha configurado Common Event Infrastructure. Si el servidor ya se ha configurado, puede cambiar los valores del origen de datos para la base de datos de sucesos o el almacén de mensajes o ambos.

**Nota:** Si selecciona el recuadro de selección para habilitar el servidor Common Event Infrastructure y el servidor todavía no se ha configurado, se utilizan los parámetros mostrados para configurarlo, a menos que los modifique.

- Si es la primera vez que realiza la configuración, las tablas de los orígenes de datos de sucesos se crean en la base de datos común. Si ya existe una configuración de servidor Common Event Infrastructure, deberá crear una base de datos.
- El servicio de mensajería se crea bajo un esquema exclusivo bajo la base de datos común.

Cuando se reinicia el servidor/clúster en el que se ha configurado Common Event Infrastructure, los cambios nuevos entran en vigor.

2. Configure (o cambie los valores actuales de una configuración existente) la base de datos de sucesos utilizando uno de los siguientes métodos para rellenar los campos con los valores adecuados.
  - Pulse **Editar** para un panel de configuración de base de datos con una lista más extensa de opciones que las que figuran en el panel.
  - Utilice los campos del panel para especificar información:
    - a. **Nombre de base de datos:** nombre de la base de datos que utiliza para almacenar sucesos.

- b. **Crear tablas:** seleccione este recuadro de selección si desea crear las tablas de base de datos en la base de datos de sucesos.

**Nota:** Si está configurando Common Event Infrastructure para que utilice una base de datos en otro servidor, no podrá crear las tablas utilizando este control. En su lugar, tendrá que utilizar los scripts de base de datos que se generarán después de que haya completado el resto de este proceso de configuración. En este caso, pulse **Editar** para mostrar el panel de detalles de orígenes de datos, que le indicará la ubicación de los scripts de creación de base de datos.

- c. **Nombre de usuario y Contraseña:** para la autenticación en la base de datos de sucesos.
- d. **Servidor:** el nombre del servidor en el que se encuentra la base de datos de sucesos.
- e. **Proveedor:** seleccione en el menú un proveedor para la base de datos.

**Nota:** El campo **Esquema** sólo se activa si la base de datos se crea mediante DB2 en una plataforma iSeries o z/OS. En todos los otros casos, el campo de esquema está inhabilitado.

**Importante:** Si existen las tablas en la base de datos de destino, es posible que la configuración falle.

- 3. Seleccione si el bus de Common Event Infrastructure es **Local** en el servidor o **Remoto** y reside en otro servidor. Si selecciona remoto, entonces selecciona la ubicación remota del menú o pulse **Nuevo** para crear un nuevo bus remoto.
- 4. Configure el soporte de Common Event Infrastructure para mensajería.
  - Pulse Editar para un panel de configuración de base de datos con una lista más extensa de opciones que las que figuran en el panel.
  - Utilice los campos del panel para especificar información:
    - a. **Nombre de base de datos:** especifique el nombre de la base de datos que utiliza para almacenar mensajes.
    - b. **Esquema:** escriba un nombre para el esquema o acepte el nombre por omisión asignado.
    - c. **Nombre de usuario y Contraseña:** para la autenticación en la base de datos de mensajería.
    - d. **Servidor:** el nombre del servidor en el que se encuentra la base de datos de mensajería.
    - e. **Proveedor:** seleccione en el menú un proveedor para la base de datos.
- 5. Cree un alias de autenticación para el bus Common Event Infrastructure.
  - a. Seleccione **Propiedades adicionales > Alias de autenticación JMS**.
  - b. Escriba el ID de usuario y la contraseña que utiliza para las comunicaciones seguras en el bus de integración del sistema. Puede aceptar los valores por omisión configurados de "CEI" para el ID de usuario y la contraseña, si la seguridad está inhabilitada. Si la seguridad está habilitada, especifique el ID de usuario y la contraseña que se utilizarán para la autenticación del bus. En un entorno de producción, seleccione sus propios ID de usuario y contraseña para proteger el sistema.
  - c. Pulse **Aceptar**.
- 6. Pulse **Aceptar** o **Aplicar**.
- 7. Reinicie el servidor o el clúster.

## Resultados

Se han configurado todas las partes principales de Common Event Infrastructure, que se están ejecutando en el servidor o clúster. La configuración incluye el almacén de datos de sucesos, el motor de mensajería y la aplicación de sucesos. Este panel de individual se puede utilizar en lugar de los numerosos mandatos y pasos que, de otro modo, debe utilizar para configurar Common Event Infrastructure.

## Qué hacer a continuación

Después de reiniciar el servidor o el clúster, podrá almacenar los sucesos de los componentes de servicio que se emiten desde las aplicaciones. Ahora puede cambiar las propiedades de tiempo de ejecución del servidor Common Event Infrastructure seleccionando el panel **Destino de Common Event Infrastructure**. Puede seleccionar si inicia el servidor Common Event Infrastructure durante el arranque y especificar el nombre JNDI de la fábrica de emisores a la que se envían los sucesos.

---

## Despliegue de la aplicación Common Event Infrastructure

Antes de utilizar Common Event Infrastructure, debe desplegar en primer lugar el servicio de sucesos y los recursos asociados en el entorno de tiempo de ejecución del servidor.

### Acerca de esta tarea

La aplicación empresarial Common Event Infrastructure incluye los componentes del entorno de ejecución del servicio de sucesos y la configuración de mensajería por omisión utilizada para el envío de sucesos asíncrono.

Para desplegar el servicio de sucesos:

### Procedimiento

En la herramienta wsadmin, ejecute el mandato administrativo **deployEventService** en modalidad de proceso por lotes o interactiva. Los parámetros del mandato administrativo **deployEventService** son los siguientes:

#### **nodeName**

Nombre del nodo en el que se debe desplegar el servicio de sucesos. Este parámetro es opcional; si no especifica un nombre de nodo, el valor por omisión es el nodo actual. Si especifica un nombre de nodo, también debe especificar el nombre de servidor mediante el parámetro **serverName**. Este parámetro no es válido si está desplegando el servicio de sucesos en un clúster.

#### **serverName**

Nombre del servidor en el que se debe desplegar el servicio de sucesos. Este parámetro sólo es necesario si especifica un nodo; no es válido si está desplegando el servicio de sucesos en un clúster.

#### **clusterName**

Nombre del clúster en el que se debe desplegar el servicio de sucesos. Este parámetro es opcional y no se debe especificar si está desplegando en el ámbito de servidor o nodo.

### **enable**

Indica si el servicio de sucesos se debe iniciar automáticamente al iniciarse el servidor. El valor por omisión es true.

## **Resultados**

Una vez finalizado el mandato administrativo, el servicio de sucesos de Common Event Infrastructure y la configuración de mensajería por omisión se despliegan en el ámbito especificado.

## **Qué hacer a continuación**

Si la seguridad de WebSphere está habilitada, también debe configurar el alias y la contraseña de autenticación JMS con el mandato administrativo **setEventServiceJmsAuthAlias**.

Si está desplegando el servicio de suceso en un clúster, también debe configurar manualmente la base de datos de sucesos.

## **Despliegue de Common Event Infrastructure en un clúster**

Hay varias formas de desplegar los recursos de Common Event Infrastructure en un entorno en clúster.

### **Despliegue de Common Event Infrastructure en un clúster existente**

Puede desplegar la aplicación de servicio de sucesos en un clúster existente.

### **Acerca de esta tarea**

El despliegue de la aplicación del servicio de sucesos en un clúster es básicamente lo mismo que el despliegue de la aplicación en un servidor autónomo. Sin embargo, en un entorno de clúster, no hay ninguna base de datos de sucesos por omisión configurada.

Para desplegar y configurar Common Event Infrastructure en un entorno en clúster:

### **Procedimiento**

1. Ejecute el mandato administrativo **deployEventService** como lo haría para un servidor autónomo, pero especifique el nombre del clúster. Utilice el parámetro `clusterName` para especificar el clúster.
2. En el sistema del gestor de despliegue, ejecute el mandato de administración de configuración de base de datos. Especifique el nombre de usuario utilizando el parámetro `clusterName`. Este mandato genera el script de configuración de base de datos.
3. Copie el script de configuración de base de datos generado en el sistema de base de datos.
4. Ejecute el script de configuración de base de datos en el sistema de base de datos para crear la base de datos de sucesos.
5. En el sistema del gestor de despliegue ejecute el mandato **enableEventService** para habilitar el servicio de sucesos. Utilice el parámetro `clusterName` para especificar el nombre del clúster.



## Creación de un clúster convirtiendo un servidor existente de Common Event Infrastructure

Puede crear un clúster tras convertir un servidor autónomo existente que ya está configurado con Common Event Infrastructure.

### Antes de empezar

Para poder convertir el servidor existente, asegúrese de que está configurado completamente para Common Event Infrastructure. La configuración incluye desplegar la aplicación de servicio de sucesos y configurar la base de datos de sucesos.

### Acerca de esta tarea

Para crear el clúster:

### Procedimiento

1. Siga el proceso habitual de WebSphere para convertir un servidor autónomo en el primer miembro de un nuevo clúster. Cuando el servidor se ha convertido, tienen lugar los siguientes pasos:
  - Los recursos de Common Event Infrastructure disponibles en el ámbito del servidor se trasladan al nuevo ámbito del clúster.  
  
**Base de datos por omisión:** Si el servidor existente está configurado con la base de datos Derby por omisión, los recursos de base de datos no se trasladan al ámbito del clúster. En su lugar, los recursos se eliminan. La configuración de base de datos por omisión no tiene soporte en un clúster. En esta situación, por omisión el servicio de sucesos del clúster está inhabilitado.
  - La lista de destinos de la aplicación del servicio de sucesos que se ha desplegado se ha modificado para eliminar el servidor convertido y añadir el nuevo clúster.
2. Opcional: Si el servidor convertido se ha configurado con la base de datos Derby por omisión, debe configurar una nueva base de datos de sucesos para el clúster y luego habilitar el servicio de sucesos:
  - a. En el sistema del gestor de despliegue, ejecute el mandato de administración de configuración de base de datos. Especifique el nombre de usuario utilizando el parámetro `clusterName`. Este mandato genera el script de configuración de base de datos.
  - b. Copie el script de configuración de base de datos generado en el sistema de base de datos.
  - c. Ejecute el script de configuración de base de datos en el sistema de base de datos para crear la base de datos de sucesos.
  - d. En el sistema del gestor de despliegue ejecute el mandato `enableEventService` para habilitar el servicio de sucesos. Utilice el parámetro `clusterName` para especificar el nombre del clúster.

## Creación de un clúster utilizando el servidor de Common Event Infrastructure como plantilla

Puede crear un clúster si especifica un servidor de Common Event Infrastructure existente como plantilla.

## Antes de empezar

Para poder crear un clúster utilizando este método, debe tener un servidor existente configurado completamente para Common Event Infrastructure. La configuración incluye desplegar la aplicación de servicio de sucesos y configurar la base de datos de sucesos.

## Acerca de esta tarea

Para crear el clúster:

### Procedimiento

1. Siga el proceso habitual de WebSphere para crear un nuevo clúster, utilizando el servidor de Common Event Infrastructure como plantilla para el primer miembro del clúster. Cuando se crea el primer miembro, tienen lugar los siguientes pasos:
  - Los recursos de Common Event Infrastructure disponibles en el ámbito del servidor existente se copian al nuevo ámbito del clúster.  
  
**Base de datos por omisión:** Si el servidor existente está configurado con la base de datos Derby por omisión, los recursos de base de datos no se copian en el ámbito del clúster. La configuración de base de datos por omisión no tiene soporte en un clúster. En esta situación, por omisión el servicio de sucesos del clúster está inhabilitado.
  - La lista de destinos de la aplicación del servicio de sucesos que se ha desplegado se ha modificado de modo que incluya el nuevo clúster.
2. Opcional: Si el servidor existente se ha configurado con la base de datos Derby por omisión, debe configurar una nueva base de datos de sucesos para el clúster y luego habilitar el servicio de sucesos:
  - a. En el sistema del gestor de despliegue, ejecute el mandato de administración de configuración de base de datos. Especifique el nombre de usuario utilizando el parámetro `clusterName`. Este mandato genera el script de configuración de base de datos.
  - b. Copie el script de configuración de base de datos generado en el sistema de base de datos.
  - c. Ejecute el script de configuración de base de datos en el sistema de base de datos para crear la base de datos de sucesos.
  - d. En el sistema del gestor de despliegue ejecute el mandato `enableEventService` para habilitar el servicio de sucesos. Utilice el parámetro `clusterName` para especificar el nombre del clúster.

---

## Configuración de la mensajería de sucesos

Puede modificar la configuración de mensajería que se utiliza para la transmisión de sucesos JMS al servicio de sucesos.

## Acerca de esta tarea

Crearé la infraestructura de mensajería para Common Event Infrastructure cuando utilice el panel de la consola administrativa para configurar Common Event Infrastructure en un servidor. Generalmente, la configuración de mensajería utilizará el proveedor de mensajería por omisión y creará una cola JMS individual para la transmisión asíncrona de los sucesos al servicio de sucesos. Si es necesario, puede modificar esta configuración de mensajería.

## Configuración de colas JMS adicionales

Si utiliza la configuración de mensajería de sucesos por omisión, puede añadir colas JMS adicionales para la transmisión de sucesos al servicio de sucesos.

### Acerca de esta tarea

Para configurar colas JMS adicionales mediante la configuración de mensajería por omisión, puede configurar varias colas JMS que se dirigen al destino de cola del bus de integración de servicios. El destino de cola del bus de integración de servicios de Common Event Infrastructure depende del ámbito en el que se despliega el servicio de sucesos:

Ámbito	Destino de cola del bus de integración de servicios
Servidor	<i>nodo.servidor.CommonEventInfrastructureQueueDestination</i>
Clúster	<i>clúster.CommonEventInfrastructureQueueDestination</i>

Para obtener más información acerca de la configuración del bus de integración de servicios, consulte la documentación.

## Configuración de la mensaje de suceso mediante un proveedor JMS externo

Si no desea utilizar la configuración de mensajería incorporada por omisión para la transmisión de sucesos, puede configurar el transporte de mensajes asíncrono para que utilice un proveedor JMS (Java Messaging Service) externo.

### Antes de empezar

Para poder configurar la mensajería de sucesos utilizando un proveedor JMS externo, primero debe crear una cola JMS y una fábrica de conexiones que utilice las interfaces adecuadas para su proveedor JMS. También debe crear un puerto de escucha o una especificación de activación.

### Acerca de esta tarea

Para configurar la mensajería de sucesos con un proveedor JMS externo:

### Procedimiento

En la herramienta wsadmin, ejecute el mandato administrativo **deployEventServiceMdb** en modalidad de proceso por lotes o interactiva. Los parámetros del mandato **deployEventServiceMdb** son los siguientes:

#### **applicationName**

Nombre de aplicación del bean controlado por mensajes del servicio de sucesos que se va a desplegar. Este parámetro es obligatorio.

#### **nodeName**

Nombre del nodo en el que se va a desplegar el bean controlado por mensajes del servicio de sucesos. Si especifica un nombre de nodo, también debe especificar un nombre de servidor. El nombre del nodo es un parámetro opcional; el valor por omisión es el nodo actual. No especifique este parámetro si está desplegando la aplicación en un clúster.

#### **serverName**

Nombre del servidor en el que se va a desplegar el bean controlado por

mensajes del servicio de sucesos. Este parámetro es necesario si está desplegando la aplicación en el ámbito del servidor; de lo contrario, es opcional. No especifique un nombre de servidor si está desplegando la aplicación en un clúster.

**clusterName**

Nombre del clúster en el que se va a desplegar el bean controlado por mensajes del servicio de sucesos. Especifique este parámetro sólo si está desplegando la aplicación en un clúster.

**listenerPort**

Nombre del puerto receptor que utiliza el bean controlado por mensajes del servicio de sucesos para publicar sucesos. El puerto receptor especificado debe existir. Debe especificar un puerto receptor o una especificación de activación, pero no ambos.

**activationSpec**

Nombre JNDI de la especificación de activación que utiliza el bean controlado por mensajes del servicio de sucesos para publicar sucesos. La especificación de activación especificada debe existir con antelación. Debe especificar un puerto receptor o una especificación de activación, pero no ambos.

**qcfJndiName**

Nombre JNDI de la fábrica de conexiones de cola JMS que debe utilizar el bean controlado por mensajes del servicio de sucesos. Este parámetro es necesario si especifica una especificación de activación; de lo contrario, es opcional. Si especifica una fábrica de conexiones de cola y un puerto receptor, la fábrica de conexiones de cola debe coincidir con la configurada para el puerto receptor.

**Resultados**

El mandato administrativo **deployEventServiceMdb** despliega el bean controlado por mensajes para el servicio de sucesos, configurado para el puerto receptor especificado o la especificación de activación. También crea una fábrica de emisores y la transmisión JMS utilizando la configuración JMS externa. Las aplicaciones pueden utilizar la fábrica de emisores por omisión (que está configurada para utilizar la configuración de mensajería por omisión) o la nueva fábrica de emisores (que utiliza el proveedor JMS externo).

**Qué hacer a continuación**

Si desea configurar más de una cola JMS en el servicio de sucesos, puede ejecutar este mandato varias veces especificando diferentes nombres de aplicación empresarial y colas JMS. Cada vez que se ejecuta el script, despliega un bean controlado por mensajes adicional y configura nuevos recursos para que utilicen la cola JMS especificada.

**Configuración de alias de autenticación JMS**

Si la seguridad WebSphere está habilitada y desea utilizar la mensajería JMS asíncrona para enviar sucesos al servicio de sucesos, debe configurar el alias de autenticación JMS.

**Acercas de esta tarea**

Para configurar el alias de autenticación JMS:

## Procedimiento

En la herramienta wsadmin, ejecute el mandato administrativo **setEventServiceJmsAuthAlias** en modalidad de proceso por lotes o interactiva. Los parámetros del mandato **setEventServiceJmsAuthAlias** son los siguientes:

### **nombre\_usuario**

Nombre del usuario que se debe utilizar para el alias de autenticación JMS. Este parámetro es obligatorio.

### **password**

Contraseña del usuario que se debe utilizar para el alias de autenticación JMS. Este parámetro es obligatorio.

### **nodeName**

Nombre del nodo en el que desea actualizar o crear el alias de autenticación JMS. Si especifica un nombre de nodo, también debe especificar un nombre de servidor. No especifique un nombre de nodo si está configurando el alias de autenticación en un clúster.

### **serverName**

Nombre del servidor en el que desea actualizar o crear el alias de autenticación JMS. Este parámetro sólo es necesario si especifica un nodo; no es válido si está configurando el alias de autenticación en un clúster.

### **clusterName**

Nombre del clúster en el que desea actualizar o crear el alias de autenticación JMS. Especifique este parámetro sólo si está configurando el alias de autenticación en un clúster; si especifica un nombre de clúster, no especifique un nodo o un nombre de servidor.

## Resultados

El alias de autenticación JMS que utiliza el objeto del servicio de sucesos se actualiza en el ámbito especificado; si la autenticación no existe, se crea utilizando los valores especificados.

---

## Configuración de la base de datos de sucesos

Puede configurar el origen de datos de sucesos utilizando mandatos específicos para cada producto de base de datos soportado.

### Acerca de esta tarea

La base de datos de sucesos es necesaria para ofrecer soporte a la permanencia de los sucesos. Si no ha utilizado el panel de configuración de Common Event Infrastructure en la consola administrativa, todavía tendrá la opción de crear la base de datos de sucesos utilizando los mandatos que se describen aquí.

## Limitaciones de la base de datos de sucesos

Las configuraciones de la base de datos de sucesos que utilizan determinado software de base de datos tendrán algunas limitaciones.

Consulte la siguiente tabla para ver qué limitaciones pueden afectar a su entorno.

Tabla 1. Limitaciones de la base de datos de sucesos

Tipo de base de datos	Limitaciones
Oracle	<ul style="list-style-type: none"> <li>• El controlador fino Oracle 11 JDBC impone algunas restricciones de tamaño para los valores de serie si se utiliza un juego de caracteres Unicode. Puede que reciba un error Oracle ORA-01461 cuando los sucesos que contienen valores grandes (como por ejemplo, un atributo de mensajes largos) se almacenan en la base de datos de sucesos. Para obtener más información sobre esta restricción, consulte la documentación de Oracle 11. Para evitar este problema, utilice el controlador Oracle 11 OCI o el controlador fino Oracle 11.</li> <li>• El software de base de datos Oracle trata una serie en blanco como un valor NULL. Si especifica una serie en blanco como un valor de atributo de suceso, esa serie se convierte en NULL cuando se almacena en una base de datos de sucesos Oracle.</li> </ul>
Informix	<ul style="list-style-type: none"> <li>• Es necesario el controlador JDBC 3.0 (o posterior). Versiones anteriores del controlador JDBC no proporcionan soporte completo para las transacciones XA necesarias.</li> <li>• Los scripts de supresión y configuración de base de datos generados por el mandato administrativo <code>configEventServiceInformixDB</code> requieren el mandato <code>dbaccess</code> con el objeto de ejecutar scripts SQL. Es posible que este mandato esté disponible únicamente en el servidor Informix. Por lo tanto, si el servidor Informix está en un sistema distinto al del servidor WebSphere, es posible que los scripts de configuración de base de datos se deban copiar en el servidor de Informix y ejecutar localmente.</li> </ul>

Tabla 1. Limitaciones de la base de datos de sucesos (continuación)

Tipo de base de datos	Limitaciones
SQL Server	<ul style="list-style-type: none"> <li>• La base de datos de SQL Server debe estar configurada para utilizar la modalidad de autenticación mixta. No se da soporte a conexiones acreditadas.</li> <li>• Los procedimientos almacenados en XA deben estar instalados. Estos procedimientos almacenados se proporcionan con el controlador JDBC de Microsoft® Corporation.</li> <li>• El archivo sqljdbc.dll debe estar disponible en el directorio especificado en la sentencia PATH. Este archivo se proporciona con el controlador JDBC en Microsoft Corporation.</li> <li>• El servicio DTC (Distributed Transaction Coordinator) debe haberse iniciado.</li> </ul>

## Configuración de una base de datos de sucesos Derby

Puede configurar una base de datos de sucesos Derby en el ámbito del servidor o del clúster en un sistema Linux®, UNIX® o Windows®.

### Acerca de esta tarea

Hay dos tipos de bases de datos Derby que puede utilizar para la base de datos de sucesos: Derby Embedded y Derby Network. Ambos tipos se envían con WebSphere Application Server, pero con una funcionalidad limitada que no es adecuada para un entorno de producción. Por tanto, utilice Derby como base de datos de sucesos con finalidades de desarrollo o prueba. Para obtener más información sobre las bases de datos Derby, consulte la documentación de WebSphere Application Server (enlazada al final de esta página).

Derby Embedded sólo se puede utilizar con un servidor autónomo. En consecuencia, si federa el servidor autónomo con un clúster o entorno ND, deberá volver a configurar por completo el origen de datos de sucesos con otro producto de base de datos. Se inicia automáticamente al iniciar el servidor.

Derby Network se puede utilizar en un entorno de clúster o ND, aunque debe evitarse su utilización con sistemas de producción reales. Debe iniciar la base de datos manualmente para utilizarla con el servidor.

Para configurar una base de datos de sucesos Derby:

### Procedimiento

1. Inicie la herramienta wsadmin.
2. Utilice el objeto AdminTask para ejecutar el mandato administrativo **configEventServiceDerbyDB** en modalidad de proceso por lotes o interactiva. Los parámetros mínimos necesarios del mandato **configEventServerDerbyDB** son los siguientes:

#### **createDB**

Indica si el mandato administrativo crea y ejecuta los scripts de configuración de base de datos. Especifique true o false. Si este parámetro

se establece en `false`, los scripts se crean pero no se ejecutan. Para completar la configuración de la base de datos, a continuación deberá ejecutar los scripts de configuración de base de datos.

**nodeName**

Nombre del nodo que contiene el servidor donde se crea el origen de datos del servicio de sucesos. Si especifica un nombre de nodo, también debe especificar un nombre de servidor. Debe especificar uno de los siguientes:

- Nombre de nodo y nombre de servidor
- Nombre de clúster

**serverName**

Nombre del servidor en el que se crea el origen de datos del servicio de sucesos.

**clusterName**

Nombre del clúster en el que se crea el origen de datos del servicio de sucesos. Si especifica un nombre de clúster, no especifique los nombres de nodo y servidor.

Es posible que sean necesarios otros parámetros para su entorno. Para obtener una lista completa de los parámetros y la información de uso, consulte la ayuda del mandato administrativo **configEventServiceDerbyDB**.

## Resultados

El mandato administrativo crea el origen de datos necesario en el ámbito especificado; si ha especificado `true` para el parámetro `createDB`, el mandato también ejecuta el script de configuración de base de datos generado para crear la base de datos.

Los scripts de configuración de base de datos generados por omisión se almacenan en el directorio `raíz_perfil/databases/event/nodo/servidor/dbscripts/derby` (en un entorno de Network Deployment, estos scripts se almacenan en el directorio del perfil del gestor de despliegue). Si especifica un valor para el parámetro `outputScriptDir` opcional, los scripts se almacenarán en esa ubicación. Puede utilizar estos scripts para configurar manualmente la base de datos en cualquier momento.

## Configuración de una base de datos de sucesos DB2 (sistemas Linux, UNIX y Windows )

Puede configurar una base de datos de sucesos externa mediante DB2 Universal Database en un sistema Linux, UNIX o Windows.

### Acerca de esta tarea

Para configurar una base de datos de sucesos DB2 en un sistema Linux, UNIX o Windows:

### Procedimiento

1. Inicie la herramienta `wsadmin`.
2. Utilice el objeto `AdminTask` para ejecutar el mandato administrativo **configEventServiceDB2DB** en modalidad de proceso por lotes o interactiva. Los parámetros mínimos necesarios del mandato **configEventServiceDB2DB** son los siguientes:



**createDB**

Indica si el mandato administrativo crea y ejecuta los scripts de configuración de base de datos. Especifique `true` o `false`. Si este parámetro se establece en `false`, los scripts se crean pero no se ejecutan. Para completar la configuración de la base de datos, a continuación deberá ejecutar los scripts de configuración de base de datos.

**nodeName**

Nombre del nodo que contiene el servidor donde se crea el origen de datos del servicio de sucesos. Si especifica un nombre de nodo, también debe especificar un nombre de servidor. Debe especificar uno de los siguientes:

- Nombre de nodo y nombre de servidor
- Nombre de clúster

**serverName**

Nombre del servidor en el que se crea el origen de datos del servicio de sucesos.

**clusterName**

Nombre del clúster en el que se crea el origen de datos del servicio de sucesos. Si especifica un nombre de clúster, no especifique los nombres de nodo y servidor.

**jdbcClassPath**

La vía de acceso al controlador JDBC. Especifique sólo el archivo de controlador; no especifique el nombre de archivo.

**dbHostName**

Nombre de sistema principal del servidor en el que se ha instalado la base de datos.

**dbUser**

El ID de usuario de DB2 que debe utilizarse al crear la base de datos de sucesos. El ID de usuario especificado debe tener privilegios suficientes para crear y eliminar bases de datos.

**dbPassword**

La contraseña DB2 que se va a utilizar.

Es posible que sean necesarios otros parámetros para su entorno. Para obtener una lista completa de los parámetros y la información de uso, consulte la ayuda del mandato administrativo **configEventServiceDB2DB**.

## Resultados

El mandato administrativo crea el origen de datos necesario en el ámbito especificado; si ha especificado `true` para el parámetro `createDB`, el mandato también ejecuta el script de configuración de base de datos generado para crear la base de datos.

Los scripts de configuración de base de datos generados por omisión se almacenan en el directorio `raíz_perfil/databases/event/nodo/servidor/dbscripts/db2` (en un entorno de Network Deployment, estos scripts se almacenan en el directorio del perfil del gestor de despliegue). Si especifica un valor para el parámetro `outputScriptDir` opcional, los scripts se almacenarán en esa ubicación. Puede utilizar estos scripts para configurar manualmente la base de datos en cualquier momento.

# Configuración de una base de datos DB2 en un sistema z/OS

Puede configurar una base de datos de sucesos en un sistema z/OS utilizando el software de base de datos DB2.

## Antes de empezar

Para configurar la base de datos DB2 desde un cliente remoto, debe tener instalado el producto DB2 Connect con los fixpacks más recientes.

## Acerca de esta tarea

Para configurar la base de datos de sucesos:

## Procedimiento

- LinuxUNIXWindows

 Si configura la base de datos de sucesos de z/OS desde un sistema cliente Linux, UNIX o Windows, siga estos pasos para crear y catalogar la base de datos:
  - En el sistema z/OS, utilice el menú de administración de DB2 para crear un subsistema.
  - Opcional: Cree el grupo de almacenamiento que desea utilizar para la base de datos de sucesos. También puede utilizar un grupo de almacenamiento existente (por ejemplo, `sysdeflt`).
  - Habilite las agrupaciones de almacenamiento intermedio de 4, 8 y 16 K que desea utilizar para la base de datos de sucesos.
  - Otorgue los permisos necesarios al ID de usuario que desea que utilice el origen de datos. Este ID de usuario debe tener derechos de acceso a la base de datos y al grupo de almacenamiento que ha creado; también debe tener permiso para crear nuevas tablas, espacios de tablas e índices para la base de datos.
  - Catalogue la base de datos remota. Ejecute los siguientes mandatos, en un script o en una ventana de línea de mandatos de DB2:

```
catalog tcpip node zosnode remote nombre_sistppal server puerto_IP
system subsistema_bd
catalog database nombre_bd as nombre_bd at node zosnode authentication DCS
```

Para obtener más información sobre cómo catalogar un nodo y las bases de datos, consulte la documentación de DB2 Connect.
  - Compruebe que puede establecer una conexión con el subsistema remoto. Puede ejecutar el mandato siguiente para realizar la verificación:

```
db2 connect to
subsistema user id_usuario using contraseña
```
  - Enlace con la base de datos de sistema principal. Ejecute los siguientes mandatos:

```
db2 connect to nombre_bd user id_usuario using contraseña
db2 bind raíz_db2/bnd7@ddcsmvs.lst blocking all sqlerror continue message
mvs.msg grant public
db2 connect reset
```

Para obtener más información sobre cómo enlazar un cliente a una base de datos de sistema principal, consulte la documentación de DB2 Connect.
- En el sistema WebSphere, inicie la herramienta wsadmin.
- Utilice el objeto AdminTask para ejecutar el mandato administrativo `configEventServiceDB2ZOSDB` en modalidad de proceso por lotes o interactiva. Los parámetros mínimos necesarios del mandato `configEventServiceDB2ZOSDB` son los siguientes:

## createDB

**Linux** **UNIX** **Windows** Indica si el mandato administrativo crea y ejecuta los scripts de configuración de base de datos. Este parámetro sólo se aplica si está ejecutando el mandato administrativo desde un sistema cliente Linux, UNIX o Windows. Especifique true o false.

Si este parámetro se establece en false o si se está ejecutando el mandato en el sistema z/OS, los scripts se crean pero no se ejecutan. Para completar la configuración de la base de datos, a continuación deberá ejecutar los scripts de configuración de base de datos.

### nodeName

Nombre del nodo que contiene el servidor donde se crea el origen de datos del servicio de sucesos. Si especifica un nombre de nodo, también debe especificar un nombre de servidor. Debe especificar uno de los siguientes:

- Nombre de nodo y nombre de servidor
- Nombre de clúster

### serverName

Nombre del servidor en el que se crea el origen de datos del servicio de sucesos.

### clusterName

Nombre del clúster en el que se crea el origen de datos del servicio de sucesos. Si especifica un nombre de clúster, no especifique los nombres de nodo y servidor.

### jdbcClassPath

La vía de acceso al controlador JDBC. Especifique sólo el archivo de controlador; no especifique el nombre de archivo.

### dbHostName

Nombre de sistema principal del servidor en el que se ha instalado la base de datos.

### dbUser

El ID de usuario de DB2 que debe utilizarse al crear la base de datos de sucesos. El ID de usuario especificado debe tener privilegios suficientes para crear y eliminar bases de datos.

### dbPassword

La contraseña DB2 que se va a utilizar.

Es posible que sean necesarios otros parámetros para su entorno. Para obtener una lista completa de los parámetros y la información de uso, consulte la ayuda del mandato administrativo **configEventServiceDB2ZOSDB**.

## Resultados

El mandato administrativo crea el origen de datos necesario en el ámbito especificado; si ejecuta el mandato en un cliente DB2 de Linux, UNIX o Windows y ha especificado true para el parámetro createDB, el mandato también ejecuta el script de configuración de base de datos generado para crear la base de datos. En un sistema z/OS, debe utilizar el recurso SPUFI (SQL Processor Using File Input) para ejecutar los archivos DDL generados. Los archivos DDL se almacenan en el directorio *raíz\_perfil/databases/event/nodo/servidor/db2zos/ddl*.

Los scripts de configuración de base de datos generados por omisión se almacenan en el directorio *raíz\_perfil/databases/event/nodo/servidor/dbscripts/db2zos* (en un

entorno de despliegue de red, estos scripts se almacenan en el directorio del perfil del gestor de despliegue). Si especifica un valor para el parámetro `outputScriptDir` opcional, los scripts se almacenarán en esa ubicación. Puede utilizar estos scripts para configurar manualmente la base de datos en cualquier momento.

## Qué hacer a continuación

Una vez que ha terminado de configurar la base de datos, puede utilizar la consola administrativa del servidor para probar la configuración de la base de datos. Para realizar esta tarea, vaya al origen de datos JDBC apropiado y seleccione la opción **Probar conexión**.

## Configuración de una base de datos DB2 en un sistema iSeries

Puede configurar una base de datos de sucesos en un sistema iSeries utilizando el software de base de datos DB2.

### Acerca de esta tarea

Si utiliza un servidor iSeries local para configurar un servidor iSeries remoto, debe especificar una entrada de base de datos remota en el servidor local como alias de la base de datos de destino. Para configurar la base de datos de sucesos:

### Procedimiento

1. Inicie la herramienta `wsadmin`.
2. Utilice el objeto `AdminTask` para ejecutar el mandato administrativo `configEventServiceDB2iSeriesDB` en modalidad de proceso por lotes o interactiva. Los parámetros mínimos necesarios del mandato `configEventServiceDB2iSeriesDB` son los siguientes:

#### **createDB**

Indica si el mandato administrativo crea y ejecuta los scripts de configuración de base de datos. Especifique `true` o `false`. Si este parámetro se establece en `false`, los scripts se crean pero no se ejecutan. Para completar la configuración de la base de datos, a continuación deberá ejecutar los scripts de configuración de base de datos.

**Limitación:** El mandato administrativo puede ejecutar automáticamente el script de configuración de base de datos sólo en el sistema iSeries. Si se ejecuta el mandato en un sistema cliente, se devolverá un error.

#### **nodeName**

Nombre del nodo que contiene el servidor donde se crea el origen de datos del servicio de sucesos. Si especifica un nombre de nodo, también debe especificar un nombre de servidor. Debe especificar uno de los siguientes:

- Nombre de nodo y nombre de servidor
- Nombre de clúster

#### **serverName**

Nombre del servidor en el que se crea el origen de datos del servicio de sucesos.

#### **clusterName**

Nombre del clúster en el que se crea el origen de datos del servicio de sucesos. Si especifica un nombre de clúster, no especifique los nombres de nodo y servidor.

**toolboxJdbcClassPath**

La vía de acceso al controlador DB2 JDBC de IBM Toolbox para Java. Utilice este parámetro sólo si desea utilizar el controlador de Toolbox para Java en lugar del controlador JDBC nativo. Especifique sólo el archivo de controlador; no incluya el nombre de archivo.

**nativeJdbcClassPath**

La vía de acceso al controlador JDBC nativo de DB2 para iSeries. Utilice este parámetro sólo si desea utilizar el controlador JDBC nativo en lugar del controlador de Toolbox para Java. Especifique sólo el archivo de controlador; no incluya el nombre de archivo.

**dbHostName**

Nombre de sistema principal del servidor en el que se ha instalado la base de datos. Este parámetro es necesario si utiliza el controlador JDBC de Toolbox para Java.

**dbUser**

El ID de usuario de DB2 que debe utilizarse al crear la base de datos de sucesos. El ID de usuario especificado debe tener privilegios suficientes para crear y eliminar bases de datos.

**dbPassword**

La contraseña DB2 que se va a utilizar.

Es posible que sean necesarios otros parámetros para su entorno. Para obtener una lista completa de los parámetros y la información de uso, consulte la ayuda del mandato administrativo **configEventServiceDB2iSeriesDB**.

## Resultados

El mandato administrativo genera scripts para crear la base de datos y el origen de datos necesarios en el ámbito especificado. Por omisión, estos scripts se almacenan en el directorio *raíz\_perfil/databases/event/nodo/servidor/dbscripts/db2iseries*. Si especifica un valor para el parámetro `outputScriptDir` opcional, los scripts se almacenarán en esa ubicación. Puede utilizar estos scripts para configurar manualmente la base de datos en cualquier momento.

## Qué hacer a continuación

Si ejecutó el mandato administrativo de configuración de base de datos en un sistema cliente, debe transferir los scripts generados al sistema iSeries y ejecutarlos para crear los recursos necesarios.

Una vez que ha terminado de configurar la base de datos, puede utilizar la consola administrativa del servidor para probar la configuración de la base de datos. Para probar la configuración, vaya al origen de datos JDBC apropiado y seleccione la opción **Probar conexión**.

## Configuración de una base de datos de sucesos Informix

Puede configurar una base de datos de sucesos externa mediante IBM Informix Dynamic Server en un sistema Linux, UNIX o Windows.

### Acerca de esta tarea

Para configurar una base de datos de sucesos de Informix:

## Procedimiento

1. Inicie la herramienta wsadmin.
2. Utilice el objeto AdminTask para ejecutar el mandato administrativo **configEventServiceInformixDB** en modalidad de proceso por lotes o interactiva. Los parámetros mínimos necesarios del mandato **configEventServiceInformixDB** son los siguientes:

### **createDB**

Indica si el mandato administrativo debe crear y ejecutar los scripts de configuración de base de datos. Especifique true o false. Si este parámetro se establece en false, los scripts se crean pero no se ejecutan. Para completar la configuración de la base de datos, a continuación deberá ejecutar los scripts de configuración de base de datos.

**Privilegios:** Si especifica true para este parámetro, asegúrese de que el ID de usuario tiene suficientes privilegios para crear bases de datos, espacios de base de datos, tablas, vistas, índices y procedimientos almacenados de Informix.

### **nodeName**

El nombre del nodo que contiene el servidor donde debe crearse el origen de datos del servicio de sucesos. Si especifica un nombre de nodo, también debe especificar un nombre de servidor. Debe especificar uno de los siguientes:

- Nombre de nodo y nombre de servidor
- Nombre de clúster

### **serverName**

El nombre del servidor en el que se debe crear el origen de datos del servicio de sucesos.

### **clusterName**

El nombre del clúster en el que se debe crear el origen de datos del servicio de sucesos. Si especifica un nombre de clúster, no especifique los nombres de nodo y servidor.

### **jdbcClassPath**

La vía de acceso al controlador JDBC. Especifique sólo el archivo de controlador; no especifique el nombre de archivo.

### **dbInformixDir**

El directorio en el que se ha instalado el software de base de datos Informix. Este parámetro sólo es necesario si ha especificado true para el parámetro createDB.

### **dbHostName**

Nombre de sistema principal del sistema en el que se ha instalado el servidor de base de datos.

### **dbServerName**

Nombre de servidor Informix (por ejemplo, ol\_servername).

### **dbUser**

ID de usuario de esquema de base de datos de Informix que será propietario de las tablas de base de datos de sucesos. Este debe ser un ID de usuario con suficientes privilegios para crear bases de datos y espacios de base de datos. El origen de datos WebSphere utiliza este ID de usuario para autenticar la conexión de base de datos de Informix.

### **dbPassword**

Contraseña del ID de usuario de esquema especificado.

Es posible que sean necesarios otros parámetros para su entorno. Para obtener una lista completa de los parámetros y la información de uso, consulte la ayuda del mandato administrativo **configEventServiceInformixDB**.

## **Resultados**

El mandato administrativo crea el origen de datos necesario en el ámbito especificado; si ha especificado **true** para el parámetro **createDB**, el mandato también ejecuta el script de configuración de base de datos generado para crear la base de datos.

Los scripts de configuración de base de datos generados por omisión se almacenan en el directorio *raíz\_perfil/databases/event/nodo/servidor/dbscripts/informix* (en un entorno de Network Deployment, estos scripts se almacenan en el directorio del perfil del gestor de despliegue). Si especifica un valor para el parámetro **outputScriptDir** opcional, los scripts se almacenarán en esa ubicación. Puede utilizar estos scripts para configurar manualmente la base de datos en cualquier momento.

**Ejecución de scripts:** Los scripts de supresión y configuración de base de datos generados por el mandato administrativo **configEventServiceInformixDB** requieren el mandato **dbaccess** con el objeto de ejecutar scripts SQL. Es posible que este mandato esté disponible únicamente en el servidor Informix. Por lo tanto, si el servidor Informix está en un sistema distinto al del servidor, es posible que los scripts de configuración de base de datos se deban copiar en el servidor de Informix y ejecutar localmente.

## **Configuración de una base de datos de sucesos Oracle**

Puede configurar una base de datos de sucesos externa mediante una base de datos Oracle en un sistema Linux, UNIX o Windows.

### **Antes de empezar**

Antes de configurar una base de datos de sucesos Oracle, primero debe crear la base de datos. El SID de Oracle ya debe existir antes de ejecutar el mandato de configuración de base de datos de sucesos. El SID por omisión para la base de datos de sucesos es **event**.

### **Acerca de esta tarea**

Para configurar una base de datos Oracle:

### **Procedimiento**

1. Inicie la herramienta **wsadmin**.
2. Utilice el objeto **AdminTask** para ejecutar el mandato administrativo **configEventServiceOracleDB** en modalidad de proceso por lotes o interactiva. Los parámetros mínimos necesarios del mandato **configEventServiceOracleDB** son los siguientes:

#### **createDB**

Indica si el mandato administrativo debe crear y ejecutar los scripts de configuración de base de datos. Especifique **true** o **false**. Si este parámetro se establece en **false**, los scripts se crean pero no se ejecutan. Para

completar la configuración de la base de datos, a continuación deberá ejecutar los scripts de configuración de base de datos.

**nodeName**

El nombre del nodo que contiene el servidor donde debe crearse el origen de datos del servicio de sucesos. Si especifica un nombre de nodo, también debe especificar un nombre de servidor. Debe especificar uno de los siguientes:

- Nombre de nodo y nombre de servidor
- Nombre de clúster

**serverName**

El nombre del servidor en el que se debe crear el origen de datos del servicio de sucesos.

**clusterName**

El nombre del clúster en el que se debe crear el origen de datos del servicio de sucesos. Si especifica un nombre de clúster, no especifique los nombres de nodo y servidor.

**jdbcClassPath**

La vía de acceso al controlador JDBC. Especifique sólo el archivo de controlador; no especifique el nombre de archivo.

**oracleHome**

El directorio ORACLE\_HOME. Este parámetro sólo es necesario si ha especificado true para el parámetro createDB.

**dbPassword**

Contraseña que se utiliza para el ID de usuario de esquema durante la configuración de base de datos (el ID de usuario por omisión es ceuser). Esta contraseña se utiliza para autenticar la conexión de base de datos Oracle.

**sysUser**

ID de usuario SYSUSER de Oracle. Este ID de usuario debe tener privilegios SYSDBA.

**sysPassword**

Contraseña para el ID de usuario SYSUSER especificado.

Es posible que sean necesarios otros parámetros para su entorno. Para obtener una lista completa de los parámetros y la información de uso, consulte la ayuda del mandato administrativo **configEventServiceOracleDB**.

## Resultados

El mandato administrativo crea el origen de datos necesario en el ámbito especificado; si ha especificado true para el parámetro createDB, el mandato también ejecuta el script de configuración de base de datos generado para crear la base de datos.

Los scripts de configuración de base de datos generados por omisión se almacenan en el directorio *raíz\_perfil/databases/event/nodo/servidor/dbscripts/oracle* (en un entorno de Network Deployment, estos scripts se almacenan en el directorio del perfil del gestor de despliegue). Si especifica un valor para el parámetro outputScriptDir opcional, los scripts se almacenarán en esa ubicación. Puede utilizar estos scripts para configurar manualmente la base de datos en cualquier momento.



## Configuración de una base de datos de sucesos de SQL Server

Puede configurar una base de datos de sucesos externa mediante Microsoft SQL Server Enterprise en un sistema Windows.

### Acerca de esta tarea

Para configurar una base de datos de sucesos de SQL Server:

### Procedimiento

1. En el sistema de servidor de bases de datos de SQL Server, cree el directorio utilizado para contener los archivos de base de datos. Por omisión, los archivos se graban en el directorio `c:\Archivos de programa\ibm\event\ceiinst1\sqlserver_data`. Si es necesario especificar una ubicación distinta, debe editar el script de configuración de base de datos generado para modificar el valor del parámetro `ceiInstancePrefix` y, a continuación, ejecutar el script manualmente.
2. En el sistema del servidor, inicie la herramienta `wsadmin`.
3. Utilice el objeto `AdminTask` para ejecutar el mandato administrativo `configEventServiceSQLServerDB` en modalidad de proceso por lotes o interactiva. Los parámetros mínimos necesarios del mandato `configEventServiceSQLServerDB` son los siguientes:

#### **createDB**

Indica si el mandato administrativo debe crear y ejecutar los scripts de configuración de base de datos. Especifique `true` o `false`. Si este parámetro se establece en `false`, los scripts se crean pero no se ejecutan. Para completar la configuración de la base de datos, a continuación deberá ejecutar los scripts de configuración de base de datos.

#### **nodeName**

El nombre del nodo que contiene el servidor donde debe crearse el origen de datos del servicio de sucesos. Si especifica un nombre de nodo, también debe especificar un nombre de servidor. Debe especificar uno de los siguientes:

- Nombre de nodo y nombre de servidor
- Nombre de clúster

#### **serverName**

El nombre del servidor en el que se debe crear el origen de datos del servicio de sucesos. Si especifica un nombre de servidor, también debe especificar un nombre de nodo.

#### **clusterName**

El nombre del clúster en el que se debe crear el origen de datos del servicio de sucesos. Si especifica un nombre de clúster, no especifique los nombres de nodo y servidor.

#### **dbServerName**

Nombre de servidor de la base de datos de SQL Server. Este parámetro sólo es necesario si ha especificado `true` para el parámetro `createDB`.

#### **dbHostName**

Nombre de sistema principal del servidor en el que se ha instalado la base de datos de SQL Server.

#### **dbPassword**

Contraseña que se utiliza para el ID de usuario creado para ser propietario

de las tablas de base de datos de sucesos (el ID de usuario por omisión es `ceiuser`). El origen de datos de WebSphere utiliza esta contraseña para autenticar la conexión de base de datos de SQL Server.

**saUser**

ID de usuario con privilegios para crear y eliminar bases de datos y usuarios. Este parámetro sólo es necesario si ha especificado `true` para el parámetro `createDB`.

**saPassword**

Contraseña para el usuario SA especificado.

Es posible que sean necesarios otros parámetros para su entorno. Para obtener una lista completa de los parámetros y la información de uso, consulte la ayuda del mandato administrativo `configEventServiceSQLServerDB`.

## Resultados

El mandato administrativo crea el origen de datos necesario en el ámbito especificado; si ha especificado `true` para el parámetro `createDB`, el mandato también ejecuta el script de configuración de base de datos generado para crear la base de datos.

Los scripts de configuración de base de datos generados por omisión se almacenan en el directorio `raíz_perfil/databases/event/nodo/servidor/dbscripts/dbscripts/sqlserver` (en un entorno de Network Deployment, estos scripts se almacenan en el directorio del perfil del gestor de despliegue). Si especifica un valor para el parámetro `outputScriptDir` opcional, los scripts se almacenarán en esa ubicación. Puede utilizar estos scripts para configurar manualmente la base de datos en cualquier momento.

## Ejecución manual de los scripts de configuración de base de datos

Puede ejecutar manualmente los scripts generados por los mandatos administrativos de configuración de base de datos en cualquier momento.

### Acerca de esta tarea

La configuración de la base de datos es un proceso de dos pasos. El mandato administrativo de configuración de base de datos primero genera un script específico de base de datos para el entorno; a continuación este script generado configurará la base de datos de sucesos y los orígenes de datos. Si especifica `true` para el parámetro `createDB` al ejecutar el mandato administrativo, los dos pasos suceden automáticamente.

Sin embargo, si especifica `false` para el parámetro `createDB`, debe llevar a cabo la configuración de base de datos ejecutando manualmente el script generado en el sistema de destino. Es posible que sea necesario ejecutar manualmente el script en cualquiera de las siguientes situaciones:

- Necesita configurar la base de datos de sucesos en un sistema distinto del sistema en el que se ha ejecutado el mandato administrativo.
- Necesita volver a crear la base de datos de sucesos más adelante.
- Necesita modificar las opciones por omisión utilizadas por el script generado antes de ejecutarlo.

## Creación manual de una base de datos de sucesos Derby

Utilice el mandato `cr_event_derby` para generar manualmente un script de configuración de base de datos para una base de datos de sucesos Derby

### Acerca de esta tarea

Para ejecutar manualmente el script de configuración de base de datos generado para una base de datos de sucesos Derby:

### Procedimiento

1. En el sistema del servidor, vaya al directorio que contiene el script generado. La ubicación por omisión es el directorio `raíz_instalación/profiles/nombre_perfil/dbscripts/CEI_nombreBasededatosCEI`; si ha especificado un valor para el parámetro `outputScriptDir` del mandato administrativo de configuración de base de datos, los scripts se almacenan en esa ubicación.
2. Utilizando un editor de texto ASCII, realice todas las modificaciones necesarias en el script de configuración. El nombre del script varía en función del sistema operativo que se utilice:
  - **Windows** `cr_event_derby.bat`
  - **Linux** **UNIX** `cr_event_derby.sh`
  - **En plataformas i5/OS:** `cr_event_derby`
3. Opcional: Si configura la base de datos en un sistema iSeries, inicie el Intérprete de QShell.
4. Ejecute el script de creación de base de datos con la siguiente sintaxis (recuerde especificar la extensión de archivo, si procede):  
`cr_event_derby -p vía_acceso_perfil [-s nombre_servidor|-c nombre_clúster]`

Los parámetros son los siguientes:

**-p** *vía\_acceso\_perfil*

La vía de acceso al directorio del perfil WebSphere. Este parámetro es obligatorio.

**-s** *nombre\_servidor*

Nombre del servidor. Este parámetro es necesario si está configurando la base de datos en el ámbito del servidor.

**-c** *nombre\_clúster*

Nombre del clúster. Este parámetro es necesario si está configurando la base de datos en el ámbito del clúster.

Por ejemplo, el siguiente mandato creará la base de datos Derby en el ámbito del servidor `server1`, utilizando el perfil `profile1`:

```
cr_event_derby -p c:\WebSphere\appserver\profiles\myprofile -s server1
```

5. Reinicie el servidor. Para un nodo federado, también debe detener y reiniciar el agente de nodo utilizando los mandatos **stopNode** y **startNode**.

### Qué hacer a continuación

Una vez que ha terminado de configurar la base de datos, puede utilizar la consola administrativa para probar la configuración de la base de datos. Para ello, vaya al origen de datos JDBC apropiado y seleccione la opción **Probar conexión**.

## Creación manual de una base de datos de sucesos DB2 en un sistema Linux, UNIX o Windows

Utilice `cr_event_db2` para generar manualmente un script de configuración de base de datos para una base de datos de sucesos DB2 en un servidor Linux, UNIX o Windows.

### Acerca de esta tarea

Para ejecutar manualmente el script de configuración de base de datos generado para una base de datos de sucesos DB2 en un sistema Linux, UNIX o Windows:

### Procedimiento

1. En el sistema del servidor, vaya al directorio que contiene el script generado. La ubicación por omisión es el directorio `raíz_instalación/profiles/nombre_perfil/dbscripts/CEI_nombreBasededatosCEI`; si ha especificado un valor para el parámetro `outputScriptDir` del mandato administrativo de configuración de base de datos, los scripts se almacenan en esa ubicación.
2. Utilizando un editor de texto ASCII, realice todas las modificaciones necesarias en el script de configuración. El nombre del script varía en función del sistema operativo que se utilice:
  - **Windows** Sistemas Windows: `cr_event_db2.bat`
  - **Linux** **UNIX** Sistemas Linux y UNIX: `cr_event_db2.sh`
3. Ejecute el script de creación de base de datos con la siguiente sintaxis (recuerde especificar la extensión de archivo, si procede):  
`cr_event_db2 [client|server] usuario_bd [contraseña_bd]`

Los parámetros son los siguientes:

#### **client|server**

Indica si la base de datos es un cliente o un servidor. Debe especificar **client** o **server**.

#### *usuario\_bd*

ID de usuario de base de datos. Este parámetro es obligatorio.

#### *contraseña\_bd*

Contraseña para el usuario de base de datos. Si no especifica una contraseña para una base de datos de cliente, se le solicitará.

Por ejemplo, el siguiente mandato creará la base de datos de sucesos DB2 para una base de datos de cliente, utilizando el ID de usuario `db2admin` y la contraseña `mypassword`:

```
cr_event_db2 client db2admin mypassword
```

4. Reinicie el servidor. Para un nodo federado, también debe detener y reiniciar el agente de nodo utilizando los mandatos **stopNode** y **startNode**.

### Qué hacer a continuación

Una vez que ha terminado de configurar la base de datos, puede utilizar la consola administrativa para probar la configuración de la base de datos. Para ello, vaya al origen de datos JDBC apropiado y seleccione la opción **Probar conexión**.

## Creación manual de la base de datos de sucesos DB2 en un sistema z/OS

Utilice `cr_event_db2zos` para generar manualmente un script de configuración de base de datos para una base de datos de sucesos DB2 en un sistema z/OS mediante un sistema cliente Linux, UNIX o Windows.

### Acerca de esta tarea

Para ejecutar manualmente el script de configuración de base de datos generado para una base de datos de sucesos DB2 en un sistema z/OS, utilizando un sistema cliente Linux, UNIX o Windows:

### Procedimiento

1. En el sistema del servidor, vaya al directorio que contiene el script generado. La ubicación por omisión es el directorio `raíz_instalación/perfiles/nombre_perfil/dbscripts/CEI_nombreBasededatosCEI`. Si especifica un valor para el parámetro `outputScriptDir` del mandato administrativo de configuración de base de datos, los scripts se almacenan en dicha ubicación.
2. Utilizando un editor de texto ASCII, realice todas las modificaciones necesarias en el script de configuración. El nombre del script varía en función del sistema operativo que se utilice:
  - Sistemas Windows: `cr_event_db2zos.bat`
  - Sistemas Linux y UNIX: `cr_event_db2zos.sh`
3. Ejecute el script de creación de base de datos con la siguiente sintaxis (recuerde especificar la extensión de archivo, si procede):

```
cr_event_db2zos [dbName=nombre_bd] usuario_bd [contraseña_bd]
```

Los parámetros son los siguientes:

*nombre\_bd*

Nombre de base de datos que se va a utilizar. Este parámetro es opcional; si no especifica un nombre de base de datos, se genera un nombre.

*usuario\_bd*

El ID de usuario de la base de datos que se va a utilizar. Este parámetro es obligatorio.

*contraseña\_bd*

Contraseña para el usuario de base de datos. Si no especifica la contraseña, la base de datos DB2 se la solicitará.

Por ejemplo, el siguiente mandato creará la base de datos de sucesos DB2 denominada `event`, utilizando el ID de usuario `db2admin` y la contraseña `mypassword`:

```
cr_event_db2zos dbName=client db2admin mypassword
```

4. Reinicie el servidor. Para un nodo federado, también debe detener y reiniciar el agente de nodo utilizando los mandatos `stopNode` y `startNode`.

### Qué hacer a continuación

Una vez que ha terminado de configurar la base de datos, puede utilizar la consola administrativa para probar la configuración de la base de datos. Para ello, vaya al origen de datos JDBC apropiado y seleccione la opción **Probar conexión**.

## Creación manual de la base de datos de sucesos DB2 en un sistema iSeries

Utilice el mandato `cr_event_db2iseries` para generar manualmente un script de configuración de base de datos para una base de datos de sucesos DB2 en un sistema iSeries

### Acerca de esta tarea

Para ejecutar manualmente el script de configuración de base de datos generado para una base de datos de sucesos DB2 en un sistema iSeries:

### Procedimiento

1. En el sistema del servidor, vaya al directorio que contiene el script generado. La ubicación por omisión es el directorio *raíz\_instalación/profiles/nombre\_perfil/dbscripts/CEI\_nombreBasededatosCEI*. Si especifica un valor para el parámetro `outputScriptDir` del mandato administrativo de configuración de base de datos, los scripts se almacenan en dicha ubicación.
2. Utilizando el editor de texto ASCII, realice todas las modificaciones necesarias en el script `cr_event_db2iseries`.
3. Inicie el Intérprete de QShell.
4. Ejecute el script de creación de base de datos con la siguiente sintaxis:  
`cr_event_db2iseries usuario_bd contraseña_bd`

Los parámetros son los siguientes:

*usuario\_bd*

ID de usuario de base de datos. Este parámetro es obligatorio.

*contraseña\_bd*

Contraseña para el usuario de base de datos. Este parámetro es obligatorio.

Por ejemplo, el siguiente mandato creará la base de datos de sucesos DB2 utilizando el ID de usuario `db2admin` y la contraseña `mypassword`:

```
cr_event_db2iseries db2admin mypassword
```

5. Reinicie el servidor. Para un nodo federado, también debe detener y reiniciar el agente de nodo utilizando los mandatos `stopNode` y `startNode`.

### Qué hacer a continuación

Una vez que ha terminado de configurar la base de datos, puede utilizar la consola administrativa para probar la configuración de la base de datos. Para ello, vaya al origen de datos JDBC apropiado y seleccione la opción **Probar conexión**.

## Creación manual de una base de datos de sucesos Informix

Puede ejecutar manualmente los scripts generados por los mandatos administrativos de configuración de base de datos en cualquier momento.

### Acerca de esta tarea

Para ejecutar manualmente los scripts de configuración de base de datos generados para una base de datos de sucesos Informix:

### Procedimiento

1. En el sistema del servidor, vaya al directorio que contiene el script generado. La ubicación por omisión es el directorio *raíz\_instalación/profiles/nombre\_perfil/dbscripts/CEI\_nombreBasededatosCEI*. Si especifica un valor

para el parámetro `outputScriptDir` del mandato administrativo de configuración de base de datos, los scripts se almacenan en dicha ubicación.

2. Utilizando un editor de texto ASCII, realice todas las modificaciones necesarias en el script de configuración. El nombre del script varía en función del sistema operativo que se utilice:
  - **Windows** Sistemas Windows: `cr_event_informix.bat`
  - **Linux** **UNIX** Sistemas Linux y UNIX: `cr_event_informix.sh`
3. Ejecute el script de creación de base de datos sin parámetros.
4. Reinicie el servidor. Para un nodo federado, también debe detener y reiniciar el agente de nodo utilizando los mandatos `stopNode` y `startNode`.

## Qué hacer a continuación

Una vez que ha terminado de configurar la base de datos, puede utilizar la consola administrativa para probar la configuración de la base de datos. Para ello, vaya al origen de datos JDBC apropiado y seleccione la opción **Probar conexión**.

## Creación manual de una base de datos de sucesos Oracle

Utilice el mandato `cr_event_oracle` para generar manualmente un script de configuración de base de datos para una base de datos de sucesos Oracle.

## Acerca de esta tarea

Para ejecutar manualmente el script de configuración de base de datos generado para una base de datos de sucesos Oracle:

## Procedimiento

1. En el sistema del servidor, vaya al directorio que contiene el script generado. La ubicación por omisión es el directorio `raíz_instalación/profiles/nombre_perfil/dbscripts/CEI_nombreBasededatosCEI`. Si especifica un valor para el parámetro `outputScriptDir` del mandato administrativo de configuración de base de datos, los scripts se almacenan en dicha ubicación.
2. Utilizando un editor de texto ASCII, realice todas las modificaciones necesarias en el script de configuración. El nombre del script varía en función del sistema operativo que se utilice:
  - **Windows** Sistemas Windows: `cr_event_oracle.bat`
  - **Linux** **UNIX** Sistemas Linux y UNIX: `cr_event_oracle.sh`
3. Ejecute el script de creación de base de datos con la siguiente sintaxis (recuerde especificar la extensión de archivo, si procede):

```
cr_event_oracle contraseña usuario_sys
                contraseña_sys [sid=sid]
                [oracleHome=directorio_inicio_oracle]
```

Los parámetros son los siguientes:

*contraseña*

La contraseña para el ID de usuario del esquema. Este parámetro es obligatorio.

*usuario\_sys*

ID de usuario que tiene privilegios SYSDBA en la base de datos Oracle (normalmente el usuario sys). Este parámetro es obligatorio.

*contraseña\_sys*

Contraseña para el ID de usuario sys especificado. Si este ID de usuario no utiliza una contraseña, escriba none.

**sid=***sid*

El identificador del sistema (SID) de Oracle. Este parámetro es opcional.

**oracleHome=***dir\_padre\_oracle*

Directorio padre de Oracle. Este parámetro es opcional; si no especifica un valor, se utiliza la vía de acceso generada.

Por ejemplo, el siguiente mandato creará la base de datos de sucesos Oracle utilizando el ID de usuario de esquema auser y el ID de usuario sys sys:

```
cr_event_oracle auser sys syspassword sid=event oracleHome=c:\oracle
```

4. Reinicie el servidor. Para un nodo federado, también debe detener y reiniciar el agente de nodo utilizando los mandatos **stopNode** y **startNode**.

## Qué hacer a continuación

Una vez que ha terminado de configurar la base de datos, puede utilizar la consola administrativa para probar la configuración de la base de datos. Para ello, vaya al origen de datos JDBC apropiado y seleccione la opción **Probar conexión**.

## Creación manual de una base de datos de sucesos de SQL Server

Utilice el mandato `cr_event_mssql` para generar manualmente un script de configuración de base de datos para una base de datos de sucesos SQL Server.

## Acerca de esta tarea

Para ejecutar manualmente el script de configuración de base de datos generado para una base de datos de sucesos de SQL Server:

## Procedimiento

1. En el sistema del servidor, vaya al directorio que contiene el script generado. La ubicación por omisión es el directorio *raíz\_instalación/perfiles/nombre\_perfil/dbscripts/CEI\_nombreBasededatosCEI*. Si especifica un valor para el parámetro `outputScriptDir` del mandato administrativo de configuración de base de datos, los scripts se almacenan en dicha ubicación.
2. Utilizando el editor de texto ASCII, realice todas las modificaciones necesarias en el script `cr_event_mssql`.
3. Ejecute el script de creación de base de datos con la siguiente sintaxis:  

```
cr_event_mssql id_usuario contraseña [server=servidor] sauser=usuario_sa  
sapassword=contraseña_sa
```

Los parámetros son los siguientes:

*id\_usuario*

ID de usuario de inicio de sesión de SQL Server que será propietario de las tablas creadas. Este ID de usuario se debe crear en SQL Server para que se pueda realizar una conexión JDBC con la base de datos. (Los controladores JDBC no dan soporte a conexiones acreditadas.)

*contraseña*

La contraseña para el nuevo ID de usuario de inicio de sesión que se crea.



**server=***servidor*

El nombre del servidor que contiene la base de datos de SQL Server. Este parámetro es opcional; el valor por omisión es el sistema principal local.

**sauser=***usuario\_sa*

ID de usuario sa. Este ID de usuario debe tener privilegios suficientes para crear bases de datos e inicios de sesión de usuarios.

**sapassword=***contraseña\_sa*

La contraseña sa, si se utiliza la modalidad de autenticación mixta. Si no se ha establecido una contraseña para el ID de usuario, especifique **sapassword=** sin ningún valor. Omita este parámetro si utiliza una conexión acreditada.

Por ejemplo, el siguiente mandato creará la base de datos de sucesos de SQL Server utilizando el ID de usuario de inicio de sesión `userid`:

```
cr_event_mssql userid sapassword server=myserver sauser=sa sapassword=sapassword
```

4. Reinicie el servidor. Para un nodo federado, también debe detener y reiniciar el agente de nodo utilizando los mandatos **stopNode** y **startNode**.

### Qué hacer a continuación

Una vez que ha terminado de configurar la base de datos, puede utilizar la consola administrativa para probar la configuración de la base de datos. Para ello, vaya al origen de datos JDBC apropiado y seleccione la opción **Probar conexión**.

## Actualización de la base de datos de sucesos desde una versión anterior

Si ha migrado desde una versión anterior de Common Event Infrastructure y está utilizando la persistencia de sucesos, es posible que sea necesario actualizar una base de datos de sucesos existente.

### Acerca de esta tarea

La actualización de la base de datos de sucesos es necesaria si está migrando desde Common Event Infrastructure versión 5.1 o anterior.

El proceso de actualización de base de datos actualiza el esquema y los metadatos de la base de datos de sucesos existentes a la versión actual a la vez que conserva los datos de sucesos existentes.

El script de actualización de base de datos actualiza el esquema y los metadatos de la base de datos de sucesos existente a la versión actual.

**Versiones no soportadas:** Si la base de datos de sucesos utiliza una versión de software de base de datos que ya no está soportada por Common Event Infrastructure 6.0, primero debe migrar la base de datos a una versión soportada mediante el procedimiento adecuado para el software de base de datos. A continuación, efectúe el proceso de actualización de base de datos de sucesos para actualizar la base de datos.

### Actualización de una base de datos de sucesos DB2 a partir de una versión anterior

Si tiene una base de datos de sucesos DB2 existente de la versión 5.1 de Common Event Infrastructure en un sistema Linux, UNIX o Windows, debe actualizarlo a la versión actual.

## Acerca de esta tarea

Para actualizar una base de datos de sucesos DB2 en un sistema Linux o UNIX:

### Procedimiento

1. Efectúe una copia de seguridad de la base de datos de sucesos existente.
2. Vaya al directorio *raíz\_perfil/bin*.
3. Ejecute el script de actualización de DB2 correspondiente al sistema operativo:

- **Windows** Sistemas Windows:  
`eventUpgradeDB2 runUpgrade=[true|false] dbUser=usuario  
[dbName=nombre] [dbPassword=contraseña]  
[dbNode=nodo] [scriptDir=dir]`
- **Linux** **UNIX** Sistemas Linux y UNIX:  
`eventUpgradeDB2.sh runUpgrade=[true|false] dbUser=usuario  
[dbName=nombre] [dbPassword=contraseña]  
[dbNode=nodo] [scriptDir=dir]`

Los parámetros necesarios típicos son los siguientes:

#### runUpgrade

Indica si desea actualizar el script para ejecutar automáticamente los scripts DDL generados para completar la actualización de base de datos. Este parámetro es obligatorio. Especifique `false` si desea ejecutar manualmente la actualización de la base de datos más adelante o en un sistema distinto.

#### dbUser

Especifica el ID de usuario de DB2 que se va a utilizar. Este parámetro es obligatorio.

#### dbName

Especifica el nombre de base de datos DB2. El nombre por omisión para la base de datos de sucesos es `event`. Este parámetro es necesario si ha especificado `runUpgrade=true`.

#### dbPassword

Especifica la contraseña para el ID de usuario DB2 especificado. Este parámetro es opcional; si no especifica ninguna contraseña, DB2 le solicita que la especifique.

#### dbNode

Especifique el nombre de nodo de base de datos. Este parámetro es necesario si está ejecutando el script de actualización desde un sistema cliente DB2.

#### scriptDir

Especifica el directorio que desea que contenga los scripts DDL generados. Este parámetro es opcional; si no especifica un directorio, los scripts se almacenan en el directorio `.\eventDBUpgrade\db2`.

Para obtener una lista de parámetros e información de uso completa, ejecute el script `eventUpgradeDB2` sin parámetros.

## Resultados

El script de actualización genera los scripts DDL necesarios para actualizar la base de datos de sucesos. Si ha especificado `runUpgrade=true`, los scripts DDL se ejecutan automáticamente, completando la actualización.

## Ejemplo

El siguiente ejemplo actualiza una base de datos DB2 existente en un sistema Windows:

```
eventUpgradeDB2 runUpgrade=true dbUser=db2inst1 dbName=event
```

## Qué hacer a continuación

Si ha especificado `runUpgrade=false`, debe ejecutar manualmente los scripts DDL en el sistema de base de datos para completar la actualización de base de datos.

## Actualización de una base de datos de sucesos DB2 para z/OS desde una versión anterior

Si tiene una base de datos de sucesos DB2 existente de la versión 5.1 de Common Event Infrastructure en un sistema z/OS, debe actualizarla a la versión actual.

## Acerca de esta tarea

Para actualizar una base de datos de sucesos DB2 en un sistema z/OS:

## Procedimiento

1. Efectúe una copia de seguridad de la base de datos de sucesos existente.
2. Vaya al directorio `raíz_perfil/bin`.
3. Ejecute el script de actualización de DB2 para z/OS para el sistema operativo cliente:

- **Windows** Sistemas Windows:

```
eventUpgradeDB2ZOS runUpgrade=[true|false] dbUser=usuario  
[dbName=nombre] [dbPassword=contraseña]  
[scriptDir=dir] storageGroup=grupo  
bufferPool14K=agr_alm_int_4k bufferPool18k=agr_alm_int_8k  
bufferPool16K=agr_alm_int_16k
```

- **Linux** **UNIX** Sistemas Linux y UNIX:

```
eventUpgradeDB2ZOS.sh runUpgrade=[true|false] dbUser=usuario  
[dbName=nombre] [dbPassword=contraseña]  
[scriptDir=dir] storageGroup=grupo  
bufferPool14K=agr_alm_int_4k bufferPool18k=agr_alm_int_8k  
bufferPool16K=agr_alm_int_16k
```

Los parámetros necesarios típicos son los siguientes:

### runUpgrade

Indica si desea actualizar el script para ejecutar automáticamente los scripts DDL generados para completar la actualización de base de datos. Este parámetro es obligatorio. Especifique `false` si desea actualizar manualmente la base de datos más adelante o en un sistema distinto.

**Sistemas z/OS:** Este parámetro se ignora en un sistema z/OS nativo. La ejecución automática de los scripts DDL generados sólo está soportada en un sistema cliente.

### dbUser

Especifica el ID de usuario de DB2 que se va a utilizar. Este parámetro es obligatorio.

**dbName**

Especifica el nombre de base de datos DB2. El nombre por omisión para la base de datos de sucesos es event. Este parámetro es necesario si ha especificado runUpgrade=true.

**dbPassword**

Especifica la contraseña para el ID de usuario DB2 especificado. Este parámetro es opcional; si no especifica ninguna contraseña, DB2 le solicita que la especifique.

**scriptDir**

Especifica el directorio que desea que contenga los scripts DDL generados. Este parámetro es opcional; si no especifica un directorio, los scripts se almacenan en el directorio .\eventDBUpgrade\db2zos.

**storageGroup**

Especifica el nombre del grupo de almacenamiento. Este parámetro es obligatorio.

**bufferPool4K**

Especifica el nombre de la agrupación de almacenamiento intermedio de 4K. Este parámetro es obligatorio.

**bufferPool8K**

Especifica el nombre de la agrupación de almacenamiento intermedio de 8K. Este parámetro es obligatorio.

**bufferPool16K**

Especifica el nombre de la agrupación de almacenamiento intermedio de 16K. Este parámetro es obligatorio.

Para obtener una lista de parámetros e información de uso completa, ejecute el script **eventUpgradeDB2ZOS** sin parámetros.

**Resultados**

El script de actualización genera los scripts DDL necesarios para actualizar la base de datos de sucesos. Si ha especificado runUpgrade=true en un sistema cliente, los scripts DDL se ejecutan automáticamente, completando la actualización.

**Ejemplo**

En el siguiente ejemplo se actualiza una base de datos de sucesos DB2 para z/OS desde un sistema cliente Windows:

```
eventUpgradeDB2ZOS runUpgrade=true dbUser=db2inst1 dbName=event  
storageGroup=sysdeflt bufferPool4K=BP9 bufferPool8K=BP8K9 bufferPool16K=BP16K9
```

**Qué hacer a continuación**

Si ha especificado runUpgrade=false o si ha ejecutado el script de actualización en el sistema z/OS, debe ejecutar manualmente los scripts DDL generados en el sistema z/OS utilizando el recurso SPUFI (SQL Processor Using File Input). Este paso completa la actualización de la base de datos.

**Actualización de una base de datos de sucesos Oracle desde la versión 5**

Si tiene una base de datos de sucesos Oracle existente de la versión 5.1 de Common Event Infrastructure, debe actualizarla a la versión actual.

## Acerca de esta tarea

Para actualizar una base de datos de sucesos Oracle:

### Procedimiento

1. Efectúe una copia de seguridad de la base de datos de sucesos existente.
2. Vaya al directorio *raíz\_perfil/bin*.
3. Ejecute el script de actualización de Oracle correspondiente al sistema operativo:

- Sistemas Windows:

```
eventUpgradeOracle runUpgrade=[true|false] schemaUser=usuario_esquema  
[oracleHome=dir] [dbName=nombre]  
[dbUser=usuario_sys] [dbPassword=contraseña]  
[scriptDir=dir]
```

- Sistemas Linux y UNIX:

```
eventUpgradeOracle.sh runUpgrade=[true|false] schemaUser=usuario_esquema  
[oracleHome=dir] [dbName=nombre]  
[dbUser=usuario_sys] [dbPassword=contraseña]  
[scriptDir=dir]
```

Los parámetros necesarios típicos son los siguientes:

#### runUpgrade

Indica si desea actualizar el script para ejecutar automáticamente los scripts DDL generados para completar la actualización de base de datos. Este parámetro es obligatorio. Especifique `false` si desea actualizar manualmente la base de datos más adelante o en un sistema distinto.

#### schemaUser

Especifica el ID de usuario Oracle que es propietario de las tablas de base de datos. Este parámetro es obligatorio.

#### oracleHome

Especifica el directorio padre de Oracle. Este parámetro es necesario si ha especificado `runUpgrade=true`.

#### dbName

Especifica el nombre de la base de datos Oracle. El nombre por omisión para la base de datos de sucesos es `event`. Este parámetro es necesario si ha especificado `runUpgrade=true`.

#### dbUser

Especifica el ID de usuario `sys` de Oracle. Este parámetro es necesario si ha especificado `runUpgrade=true`.

#### dbPassword

Especifica la contraseña del ID de usuario `sys`. No especifique este parámetro si el ID de usuario `sys` no tiene contraseña.

#### scriptDir

Especifica el directorio que desea que contenga los scripts DDL generados. Este parámetro es opcional; si no especifica un directorio, los scripts se almacenan en el directorio `.\eventDBUpgrade\oracle`.

Para obtener una lista de parámetros e información de uso completa, ejecute el script `eventUpgradeOracle` sin parámetros.

## Resultados

El script de actualización genera los scripts DDL necesarios para actualizar la base de datos de sucesos. Si ha especificado `runUpgrade=true`, los scripts DDL se ejecutan automáticamente, completando la actualización.

## Ejemplo

El siguiente ejemplo actualiza una base de datos Oracle existente en un sistema Windows:

```
eventUpgradeOracle runUpgrade=true schemaUser=cei  
dbName=event dbUser=sys
```

## Qué hacer a continuación

Si ha especificado `runUpgrade=false`, debe ejecutar manualmente los scripts DDL en el sistema de base de datos para completar la actualización de base de datos.

---

## Configuración de Common Event Infrastructure entre células para WebSphere Business Monitor

Debe configurar la conectividad entre un servidor remoto que produzca sucesos CEI (Common Event Infrastructure) y el servidor WebSphere Business Monitor.

### Acerca de esta tarea

Consulte el tema "Configuración de un servidor CEI remoto para utilizar WebSphere Business Monitor" en el Information Center de IBM WebSphere Business Monitor para obtener información detallada sobre cómo configurar CEI en células de un entorno de varios servidores.

---

## Eliminación de la configuración de Common Event Infrastructure

Si necesita eliminar la configuración de Common Event Infrastructure, en preparación para la desinstalación del servidor, primero debe eliminar las aplicaciones de empresa desplegadas y la configuración de base de datos.

### Acerca de esta tarea

Common Event Infrastructure está instalado en la instalación base del servidor, pero no estará activo a menos que ya esté configurado. Este tema solo indica cómo eliminar una instancia de Common Event Infrastructure configurada anteriormente. Para eliminar la configuración de Common Event Infrastructure, siga estos pasos:

## Suprimir la base de datos de sucesos

Para eliminar la base de datos de sucesos, puede utilizar el mandato administrativo adecuado para el tipo de base de datos.

### Acerca de esta tarea

Para eliminar la base de datos de sucesos:

### Procedimiento

1. Inicie la herramienta wsadmin.

- Utilice el objeto AdminTask para ejecutar el mandato administrativo adecuado para la base de datos de sucesos:

Tipo de base de datos	Mandato
Derby	"Mandato removeEventServiceDerbyDB" en la página 110
DB2 en sistemas Linux, UNIX y Windows	"Mandato removeEventServiceDB2DB" en la página 106
DB2 en sistemas z/OS	"Mandato removeEventServiceDB2ZOSDB" en la página 109
DB2 en sistemas iSeries	"Mandato removeEventServiceDB2iSeriesDB" en la página 108
Informix	"Mandato removeEventServiceInformixDB" en la página 112
Oracle	"Mandato removeEventServiceOracleDB" en la página 113
SQL Server	"Mandato removeEventServiceSQLServerDB" en la página 114

#### Notas específicas de base de datos:

- En un sistema z/OS, el mandato administrativo sólo elimina el origen de datos JDBC. Para eliminar la base de datos, debe utilizar SPUI para ejecutar el script de eliminación de base de datos generado durante la creación de base de datos. Por omisión, este script se almacena en el directorio *raíz\_perfil/databases/event/nodo/servidor/dbscripts/db2zos*.
- En un sistema iSeries, el mandato administrativo sólo elimina el origen de datos JDBC. Para eliminar la base de datos del sistema iSeries, elimine la colección que se ha creado para la base de datos.

Los parámetros necesarios (por ejemplo, los ID de usuario y las contraseñas) varían según el tipo de base de datos. Para obtener una lista completa de parámetros e información de uso, consulte la ayuda relativa al mandato administrativo.

## Eliminación de la aplicación Common Event Infrastructure

Para eliminar manualmente la aplicación empresarial del servicio de sucesos y los recursos del servidor, puede usar el mandato administrativo **removeEventService**.

### Acerca de esta tarea

Para eliminar la aplicación empresarial del servicio de sucesos:

#### Procedimiento

- Inicie la herramienta wsadmin.
- Utilice el objeto AdminTask para ejecutar el mandato administrativo **removeEventService** en modalidad de proceso por lotes o interactiva. Los parámetros del mandato administrativo **removeEventService** son los siguientes:

##### nodeName

Nombre del nodo en el que se despliega el servicio de sucesos. Este parámetro es opcional; si no especifica un nombre de nodo, el valor por

omisión es el nodo actual. Si especifica un nombre de nodo, también debe especificar el nombre de servidor mediante el parámetro **serverName**. Este parámetro no es válido si está eliminando el servicio de sucesos de un clúster.

**serverName**

Nombre del servidor en el que se despliega el servicio de sucesos. Este parámetro sólo es necesario si especifica un nodo; no es válido si está eliminando el servicio de sucesos de un clúster.

**clusterName**

Nombre del clúster en el que se despliega el servicio de sucesos. Este parámetro es opcional y no se debe especificar si está eliminando el servicio de sucesos de un servidor.

## Supresión de la mensajería de sucesos del servidor Common Event Infrastructure

Para eliminar la configuración de mensajería del servicio de sucesos para un proveedor JMS externo, puede utilizar el mandato administrativo `removeEventServiceMdb`.

### Acerca de esta tarea

Este mandato elimina el bean controlado por mensajes desplegado para la configuración de JMS. Para eliminar el bean controlado por mensajes del servicio de sucesos:

### Procedimiento

1. Inicie la herramienta wsadmin.
2. Utilice el objeto `AdminTask` para ejecutar el mandato administrativo `removeEventServiceMdb` en modalidad de proceso por lotes o interactiva. Los parámetros del mandato administrativo `removeEventServiceMdb` son los siguientes:

**applicationName**

Nombre de aplicación del bean controlado por mensajes del servicio de sucesos desplegado.

**nodeName**

Nombre del nodo en el que se despliega el bean controlado por mensajes del servicio de sucesos. Si especifica un nombre de nodo, también debe especificar un nombre de servidor. El nombre del nodo es un parámetro opcional; el valor por omisión es el nodo actual. No especifique este parámetro si está eliminando la aplicación de un clúster.

**serverName**

Nombre del servidor en el que se despliega el bean controlado por mensajes del servicio de sucesos. Este parámetro es necesario si está eliminando la aplicación de un servidor; de lo contrario, es opcional. No especifique un nombre de servidor si está eliminando la aplicación de un clúster.

**clusterName**

Nombre del clúster en el que se despliega el bean controlado por mensajes del servicio de sucesos. Especifique este parámetro sólo si elimina la aplicación de un clúster.



---

## Capítulo 3. Administración de Common Event Infrastructure

Estos temas describen varias tareas administrativas que puede utilizar para controlar la operación de los componentes de Common Event Infrastructure durante la ejecución.

---

### Administración del servicio de sucesos con la consola administrativa

Puede utilizar la consola administrativa basada en Web para administrar el servicio de sucesos.

#### Habilitación e inhabilitación del servicio de sucesos con la consola administrativa

Puede habilitar e inhabilitar el servicio de sucesos modificando las propiedades del servicio de sucesos en la consola administrativa del servidor.

##### Acerca de esta tarea

Si el servicio de sucesos está habilitado, se inicia automáticamente cuando se inicia el servidor.

Para habilitar o inhabilitar el servicio de sucesos desde la consola administrativa:

##### Procedimiento

1. Seleccione uno de los siguientes métodos para ir al panel adecuado de la consola administrativa:
  - Abra el panel Servidor Common Event Infrastructure de la consola administrativa:
    - Para los servidores, seleccione **Servers** → **Tipos de servidores** → **Servidores de aplicaciones WebSphere** → *nombre\_servidor* → **Business Integration** → **Common Event Infrastructure** → **Servidor Common Event Infrastructure**.
    - Para los clústeres, seleccione **Servidores** → **Clústeres** → **Clústeres de WebSphere Application Server** → *nombre\_clúster* → **Business Integration** → **Common Event Infrastructure** → **Servidor Common Event Infrastructure**.
  - Alternativamente, puede abrir los Servicios de contenedor para realizar esta tarea:
    - Para los servidores, pulse **Servidores** → **Tipos de servidores** → **Servidores de aplicaciones WebSphere** → *nombre\_servidor* → **Servicios de contenedor** → **Servicio de Common Event Infrastructure** .
    - Para los clústeres, pulse **Servidores** → **Clústeres** → **Clústeres de WebSphere Application Server** → *nombre\_clúster* → **Miembros de clúster** → *servidor* → **Servicios de contenedor** → **Servicio Common Event Infrastructure**.
2. Seleccione o deseleccione la propiedad **Habilitar el servicio durante el arranque del servidor**. Si el recuadro de selección está seleccionado, el servicio de Common Event Infrastructure se inicia cuando se inicia el servidor.
3. Guarde los cambios de configuración.
4. Sólo en un entorno de despliegue de red, sincronice el nodo.
5. Reinicie los servidores o clúster.

## Creación de una fábrica de emisores de sucesos mediante la consola administrativa

Los orígenes de sucesos utilizan una fábrica de emisores para crear emisores.

### Acerca de esta tarea

Las propiedades de una fábrica de emisores afectan al comportamiento de cualquier emisor creado utilizando esa fábrica de emisores. Puede utilizar la fábrica de emisores por omisión o crear fábricas de emisores adicionales para que las utilicen los orígenes de sucesos. También puede crear una fábrica de emisores adicional para especificar una modalidad de transacciones o una transmisión de sucesos distinta. Para ver el emisor de sucesos que se ha creado después de configurar CEI (Common Event Infrastructure) en el panel Destino de Common Event Infrastructure:

1.
  - Para un solo servidor, seleccione **Servidores > Tipos de servidores > Servidores de aplicaciones WebSphere > nombre\_servidor**.
  - Para un clúster, seleccione **Servidores > Clústeres > Clústeres de WebSphere Application Server nombre\_clúster**.
2. En la pestaña **Configuración**, seleccione **Business Integration > Common Event Infrastructure > Destino de Common Event Infrastructure**.
3. Puede seleccionar una fábrica emisora de sucesos existente del menú de nombres JNDI o especificar una en el campo de texto.

Debe resolver los nombres JNDI en un servidor remoto, si el servidor CEI no es local. Consulte la documentación de WebSphere Application Server para obtener más información acerca de cómo asignar nombres dentro de un entorno de ND. Para crear una fábrica de emisores:

### Procedimiento

1. En la consola administrativa del servidor, pulse **Integración de servicios > Common Event Infrastructure > Fábricas de emisores de sucesos > Nueva**.
2. Especifique las propiedades de la nueva fábrica de emisores. Consulte la ayuda en línea de la página de valores de la fábrica de emisores para obtener información detallada sobre estas propiedades.
3. Guarde los cambios de configuración.
4. Sólo en un entorno de despliegue de red, sincronice el nodo.
5. Reinicie el servidor.

### Resultados

Ahora los orígenes de sucesos pueden utilizar la fábrica de emisores configurada para crear emisores.

## Creación de un grupo de sucesos mediante la consola administrativa

Un grupo de sucesos define una colección lógica de sucesos basada en el contenido de sus datos de propiedades. Se puede utilizar un grupo de sucesos cuando se consultan sucesos del servicio de sucesos y se puede asociar opcionalmente con un destino JMS para la distribución de sucesos asíncrona.

## Acerca de esta tarea

Para crear un grupo de sucesos:

### Procedimiento

1. Opcional: Configure uno o varios destinos JMS para el grupo de sucesos. Un grupo de sucesos puede asociarse a un tema JMS y a una o varias colas JMS. Consulte la documentación de su proveedor JMS para obtener información acerca de cómo crear destinos JMS y fábricas de conexiones y cómo enlazarlas a un espacio de nombres JNDI.

**Seguridad:** Si la seguridad de WebSphere está habilitada, la configuración del destino JMS debe especificar un alias de autenticación.

2. Cree el nuevo grupo de sucesos. En la consola administrativa del servidor, pulse **Integración de servicios > Common Event Infrastructure > Servicio de sucesos > Servicios de sucesos > servicio\_suceso > Grupos de sucesos > Nuevo**.
3. Especifique las propiedades del nuevo grupo de sucesos, incluido el selector de sucesos y los destinos JMS opcionales.
4. Guarde los cambios de la configuración.
5. Sólo en un entorno de despliegue de red, sincronice el nodo.
6. Reinicie el servidor.

### Resultados

Los consumidores de sucesos pueden especificar el grupo de sucesos al consultar sucesos. Si la distribución de sucesos está habilitada en los valores del servicio de sucesos, los sucesos que pertenecen al grupo de sucesos también se publican en cualquiera de los destinos JMS especificados en el grupo de sucesos. A continuación, los consumidores de sucesos pueden recibir sucesos asincrónicamente suscribiéndose a los destinos adecuados.

## Creación de un filtro de sucesos mediante la consola administrativa

Un filtro de sucesos define las propiedades utilizadas por el plug-in del filtro de sucesos que utilizan los emisores para filtrar sucesos en el origen.

### Acerca de esta tarea

Se puede especificar un filtro de sucesos como parte de la configuración de una fábrica de emisores. Cualquier emisor creado con esta fábrica de emisores utilizará el filtro especificado para determinar qué suceso debe enviarse al servicio de sucesos.

Para crear un filtro de sucesos:

### Procedimiento

1. En la consola administrativa del servidor, pulse **Integración de servicios > Common Event Infrastructure > Fábricas de emisores de sucesos > fábrica\_emisores > Filtros de sucesos > Nuevo**.
2. Especifique las propiedades del nuevo filtro de sucesos. Consulte la ayuda en línea de la página de valores del filtro de sucesos para obtener información detallada sobre estas propiedades.
3. Guarde los cambios de la configuración.

4. Sólo en un entorno de despliegue de red, sincronice el nodo.
5. Reinicie el servidor.

## Resultados

Los emisores de sucesos ahora pueden utilizar el filtro configurado para determinar qué sucesos enviar al servicio de sucesos.

---

## Administración del servicio de sucesos mediante scripts

Puede utilizar interfaces de scripts para administrar el servicio de sucesos.

### Habilitación del servicio de sucesos mediante scripts

Puede habilitar el servicio de sucesos ejecutando un mandato administrativo `AdminTask` con la herramienta `wsadmin`.

#### Acerca de esta tarea

Si el servicio de sucesos está habilitado, se inicia automáticamente cuando se inicia el servidor.

Para habilitar el servicio de sucesos mediante la herramienta `wsadmin`:

#### Procedimiento

1. Inicie la herramienta `wsadmin`.
2. Utilice el objeto `AdminTask` para ejecutar el mandato administrativo `enableEventService`:

- Utilizando `Jacl`:

```
# habilite el servidor de sucesos en el ámbito del servidor
$AdminTask enableEventService { -nodeName nodo1 -serverName servidor1 }
```

```
# habilite el servidor de sucesos en el ámbito del clúster
$AdminTask enableEventService { -clusterName clúster1 }
```

- Utilizando `Jython`:

```
# habilite el servidor de sucesos en el ámbito del servidor
AdminTask.enableEventService([' -nodeName nodo1 -serverName servidor1 ]')
```

```
# habilite el servidor de sucesos en el ámbito del clúster
AdminTask.enableEventService([' -clusterName clúster1 ]')
```

Los parámetros del mandato `enableEventService` son los siguientes:

#### **nodeName**

Nombre del nodo en el que se debe habilitar el servicio de sucesos. Este parámetro es opcional; el valor por omisión es el nodo actual. Si especifica un nombre de nodo, también debe especificar un nombre de servidor. No especifique un nodo si está habilitando el servicio de sucesos en un clúster.

#### **serverName**

Nombre del servidor en el que se debe habilitar el servicio de sucesos. Este parámetro es necesario si especifica un nombre de nodo. No especifique un servidor si está habilitando el servicio de sucesos en un clúster.

#### **clusterName**

Nombre del clúster en el que se debe habilitar el servicio de sucesos. Este parámetro es necesario si está habilitando el servicio de sucesos en un clúster.

3. Reinicie el servidor.

## Inhabilitación del servicio de sucesos mediante scripts

Puede inhabilitar el servicio de sucesos ejecutando un mandato administrativo AdminTask con la herramienta wsadmin.

### Acerca de esta tarea

Si el servicio de sucesos está inhabilitado, no se inicia automáticamente cuando se inicia el servidor.

Para inhabilitar el servicio de sucesos mediante la herramienta wsadmin:

### Procedimiento

1. Inicie la herramienta wsadmin.
2. Utilice el objeto AdminTask para ejecutar el mandato administrativo **disableEventService**:

- Utilizando Jacl:

```
# inhabilite el servidor de sucesos en el ámbito del servidor
$AdminTask disableEventService { -nodeName nodo1 -serverName servidor1 }
```

```
# inhabilite el servidor de sucesos en el ámbito del clúster
$AdminTask disableEventService { -clusterName clúster1 }
```

- Utilizando Jython:

```
# inhabilite el servidor de sucesos en el ámbito del servidor
AdminTask.disableEventService([' -nodeName nodo1 -serverName servidor1 ]')
```

```
# inhabilite el servidor de sucesos en el ámbito del clúster
AdminTask.disableEventService([' -clusterName clúster1 ]')
```

Los parámetros del mandato **disableEventService** son los siguientes:

#### nodeName

Nombre del nodo en el que se debe inhabilitar el servicio de sucesos. Este parámetro es opcional; el valor por omisión es el nodo actual. Si especifica un nombre de nodo, también debe especificar un nombre de servidor. No especifique un nodo si está inhabilitando el servicio de sucesos en un clúster.

#### serverName

Nombre del servidor en el que se debe inhabilitar el servicio de sucesos. Este parámetro es necesario si especifica un nombre de nodo. No especifique un servidor si está inhabilitando el servicio de sucesos en un clúster.

#### clusterName

Nombre del clúster en el que se debe inhabilitar el servicio de sucesos. Este parámetro es necesario si está inhabilitando el servicio de sucesos en un clúster.

3. Reinicie el servidor.

## Creación de una fábrica de emisores con scripts

Puede crear una fábrica de emisores mediante un script Jacl o Jython.

## Antes de empezar

Antes de iniciar esta tarea, la herramienta wsadmin debe estar en ejecución. Consulte la documentación de WebSphere Application Server para obtener más información.

## Acerca de esta tarea

Efectúe los siguientes pasos para configurar una nueva fábrica de emisores.

### Procedimiento

1. Identifique el ID del proveedor de Common Event Infrastructure.

- Utilizando Jacl:

```
set providerid [$AdminConfig getid \  
/Cell:mycell/Node:mynode/Server:myserver/ \  
EventInfrastructureProvider:/]
```

- Utilizando Jython:

```
providerid =  
AdminConfig.getid('/Cell:mycell/Node:mynode/Server:myserver/EventInfrastructureProvider/')  
print providerid
```

Ejemplo de salida:

```
EventInfrastructureProvider(cells/mycell/nodes/mynode/servers/myserver|resources-cei.xml#  
EventInfrastructureProvider_1)
```

2. Establezca los atributos necesarios.

- Utilizando Jacl:

```
set Name [list name "EmitterName"]  
set JndiName [list jndiName "Indique aquí el nombre JNDI de la nueva  
fábrica de emisores"]  
set Description [list description "Indique aquí la descripción"]  
set Category [list category "Indique aquí la categoría"]  
  
# establezca TransactionMode en true para enviar cada suceso en una  
nueva transacción  
set TransactionMode [list preferredTransactionMode "false"]  
  
# establezca SynchronizationMode en true para utilizar transmisión de  
sucesos síncrona  
# como modalidad de sincronización preferida  
set SynchronizationMode [list preferredSynchronizationMode "true"]  
  
# déjelo en blanco si no se da soporte a la transmisión síncrona  
set SyncJNDIName [list synchronousTransmissionProfileJNDIName \  
"Indique aquí el nombre JNDI del perfil de transmisión síncrona"]  
  
# déjelo en blanco si no se da soporte a la transmisión asíncrona  
set AsyncJNDIName [list asynchronousTransmissionProfileJNDIName \  
"Indique aquí el nombre JNDI del perfil de transmisión asíncrona"]  
  
set FilteringEnabled [list filteringEnabled "false"]  
  
# déjelo en blanco si el filtrado no está habilitado  
set FilterJNDIName [list filterFactoryJNDIName \  
"Indique aquí el nombre JNDI del filtro de sucesos"]  
  
# propiedades personalizadas incluyen modalidad de compatibilidad  
set CompatibilityMode [list [list name compatibilityMode] \  
[list description ""] \  
[list required false] \  
]
```

```

        [list type java.lang.Boolean] \
        [list value "false" ] \
set resProp [list [list resourceProperties [list \
$CompatibilityMode ]]]

```

- Utilizando Jython:

```

Name = ['name', 'EmitterName']
JndiName = ['jndiName', 'Indique aquí el nombre JNDI de la
nueva fábrica de emisores']
Description = ['description', 'Indique aquí la descripción']
Category = ['category', 'Indique aquí la categoría']

# establezca TransactionMode en true para enviar cada suceso en una
nueva transacción
TransactionMode = ['preferredTransactionMode', 'false']

# establezca SynchronizationMode en true para utilizar transmisión de
sucesos síncrona
# como modalidad de sincronización preferida
SynchronizationMode = ['preferredSynchronizationMode', 'true']

# déjelo en blanco si no se da soporte a la transmisión síncrona
SyncJNDIName = ['synchronousTransmissionProfileJNDIName', \
'Indique aquí el nombre JNDI del perfil de transmisión síncrona']

# déjelo en blanco si no se da soporte a la transmisión asíncrona
AsyncJNDIName = ['asynchronousTransmissionProfileJNDIName', \
'Indique aquí el nombre JNDI del perfil de transmisión asíncrona']

FilteringEnabled = ['filteringEnabled', 'false']

# déjelo en blanco si el filtrado no está habilitado
FilterJNDIName = ['filterFactoryJNDIName', 'Indique aquí el nombre JNDI del
filtro de sucesos filter']
# propiedades personalizadas incluyen modalidad de compatibilidad
compatibilityName = ['name','compatibilityMode']
compatibilityDescription = ['description','']
compatibilityRequired = ['required', 'false']
compatibilityType = ['type', 'java.lang.Boolean']
compatibilityValue = ['value', 'false']
CompatibilityMode = [compatibilityName, compatibilityDescription, compatibilityRequired, \
compatibilityType, compatibilityValue]
customProperties = ['propertySet', [['resourceProperties', [CompatibilityMode]]]]

```

3. Establezca las propiedades para la nueva fábrica de emisores.

- Utilizando Jacl:

```

set properties "[list $Name $JndiName $Description $Category \
$TransactionMode $SynchronizationMode $AsyncJNDIName \
$SyncJNDIName $FilteringEnabled $FilterJNDIName \
[list propertySet $resProp]]"

```

- Utilizando Jython:

```

properties = [Name,JndiName,Description,Category,TransactionMode,
SynchronizationMode,AsyncJNDIName,SyncJNDIName,FilteringEnabled,
FilterJNDIName, customProperties]
print properties

```

Ejemplo de salida:

```

[['name', 'EmitterName'], ['jndiName', 'Indique aquí el nombre JNDI para la nueva
fábrica de emisores'], ['description', 'Indique aquí la descripción'],
['category', 'Indique aquí la categoría'], ['preferredTransactionMode',
'false'], ['preferredSynchronizationMode', 'true'],
['asynchronousTransmissionProfileJNDIName', 'Indique aquí el nombre JNDI del
perfil de transmisión asíncrona '],
['synchronousTransmissionProfileJNDIName', 'Indique aquí el nombre JNDI del
perfil de transmisión síncrona'], ['filteringEnabled',
'false'], ['filterFactoryJNDIName', 'Indique aquí el nombre JNDI del filtro

```

```
de sucesos'], ['propertySet', [['resourceProperties', [[['name',
'compatibilityMode'], ['description', ''], ['required', 'false'],
['type', 'java.lang.Boolean'], ['value', 'false']]]]]]]]]
```

4. Cree la fábrica de emisores.

- Utilizando Jacl:

```
set emitterProf [$AdminConfig create EmitterFactoryProfile \
$providerid $properties]
```

- Utilizando Jython:

```
print AdminConfig.create('EmitterFactoryProfile', providerid, properties)
```

Ejemplo de salida:

```
EmitterName(cells/mycell/nodes/mynode/servers/myserver|resources-cei.xml#EmitterFactoryProfile_1)
```

5. Guarde los cambios de configuración.

6. Sólo en un entorno de despliegue de red, sincronice el nodo.

7. Reinicie el servidor.

## Creación de un grupo de sucesos con scripts

Puede crear un grupo de sucesos mediante un script Jacl o Jython.

### Antes de empezar

Antes de iniciar esta tarea, la herramienta wsadmin debe estar en ejecución. Consulte la documentación de WebSphere Application Server para obtener más información.

### Acerca de esta tarea

Efectúe los siguientes pasos para configurar un nuevo grupo de sucesos.

### Procedimiento

1. Identifique el ID del proveedor de Common Event Infrastructure.

- Utilizando Jacl:

```
set providerid [$AdminConfig getid \
/Cell:mycell/Node:mynode/Server:myserver/ \
EventInfrastructureProvider:/]
```

- Utilizando Jython:

```
providerid =
AdminConfig.getid
('/Cell:mycell/Node:mynode/Server:myserver/EventInfrastructureProvider:/')
print providerid
```

Ejemplo de salida:

```
EventInfrastructureProvider(cells/mycell/nodes/mynode/servers/myserver|
resources-cei.xml#EventInfrastructureProvider_1)
```

2. Obtenga la lista de grupos de sucesos.

- Utilizando Jacl:

```
set eventGroupProfileId [lindex [$AdminConfig list
EventGroupProfileList $providerid] 0]
```

- Utilizando Jython:

```
eventGroupProfileId = AdminConfig.list('EventGroupProfileList',providerid)
```

3. Establezca los atributos necesarios.

- Utilizando Jacl:



```

set name [ list eventGroupName "EventGroupName" ]

# escape todos los caracteres '[' de la serie del selector de sucesos.
set selectorString [ list eventSelectorString "Establezca aquí el selector de
sucesos"]

# déjelo en blanco si no se deben publicar sucesos mediante JMS
set JNDIName [ list topicJNDIName "Establezca aquí el nombre JNDI de tema"]

# déjelo en blanco si no se deben publicar sucesos mediante JMS
set connectionFactoryJNDIName [ list topicConnectionFactoryJNDIName \
"Establezca aquí el nombre JNDI de fábrica de conexiones" ]

set persistEventsFlag [ list persistEvents "true" ]

# propiedades personalizadas incluyen modalidad de compatibilidad
set CompatibilityMode [list [list name compatibilityMode] \
[list description "" ] \
[list required false] \
[list type java.lang.Boolean] \
[list value "false" ] ] \
set resProp [list [list resourceProperties [list \
$CompatibilityMode ]]]

```

- Utilizando Jython:

```

Name = ['eventGroupName', 'EventGroupName' ]
SelectorString = ['eventSelectorString', 'Establezca aquí selector de sucesos']

# déjelo en blanco si no se deben publicar sucesos mediante JMS
JNDIName = [ 'topicJNDIName', 'Establezca aquí el nombre JNDI de tema']

# déjelo en blanco si no se deben publicar sucesos mediante JMS
ConnectionFactoryJNDIName = ['topicConnectionFactoryJNDIName', \
'Establezca aquí el nombre JNDI de fábrica de conexiones']

PersistEventsFlag = ['persistEvents', 'true']

# propiedades personalizadas incluyen modalidad de compatibilidad
compatibilityName = ['name','compatibilityMode']
compatibilityDescription = ['description','']
compatibilityRequired = ['required', 'false']
compatibilityType = ['type', 'java.lang.Boolean']
compatibilityValue = ['value', 'false']
CompatibilityMode = [compatibilityName,
compatibilityDescription,
compatibilityRequired, \
compatibilityType,
compatibilityValue]
customProperties = ['propertySet', [['resourceProperties',
[CompatibilityMode]]]]

```

#### 4. Establezca las propiedades para el nuevo grupo de sucesos.

- Utilizando Jacl:

```

set properties [ list $name $selectorString $JNDIName \
$connectionFactoryJNDIName $persistEventsFlag [list propertySet \
$resProp]]

```

- Utilizando Jython:

```

properties = [Name,SelectorString,JNDIName,ConnectionFactoryJNDIName, \
PersistEventsFlag,customProperties]

```

#### Ejemplo de salida:

```

[['eventGroupName', 'EventGroupName'], ['eventSelectorString',
'Establecer aquí selector de sucesos'], ['topicJNDIName', 'Establecer aquí nombre JNDI
de tema'], ['topicConnectionFactoryJNDIName', 'Establecer nombre JNDI de
fábrica de conexiones'], ['persistEvents','true'],

```

```
['propertySet', [['resourceProperties', [[['name',  
'compatibilityMode'], ['description', ''], ['required', 'false'],  
['type', 'java.lang.Boolean'], ['value', 'false']]]]]]]]
```

5. Cree el grupo de sucesos.

- Utilizando Jacl:

```
set result [ $AdminConfig create EventGroupProfile  
$eventGroupProfileId $properties ]
```

- Utilizando Jython:

```
print AdminConfig.create('EventGroupProfile',  
eventGroupProfileId, properties)
```

Ejemplo de salida:

```
(cells/mycell/nodes/mynode/servers/myserver|resources-cei.xml#EventGroupProfile_1)
```

6. Guarde los cambios de la configuración.

7. Sólo en un entorno de despliegue de red, sincronice el nodo.

8. Reinicie el servidor.

## Creación de un filtro de sucesos con scripts

Puede crear un filtro de sucesos mediante un script Jacl o Jython.

### Antes de empezar

Antes de iniciar esta tarea, la herramienta wsadmin debe estar en ejecución. Consulte la documentación de WebSphere Application Server para obtener más información.

### Acerca de esta tarea

Efectúe los siguientes pasos para configurar un nuevo filtro de sucesos.

### Procedimiento

1. Identifique el ID del proveedor de Common Event Infrastructure.

- Utilizando Jacl:

```
set providerid [$AdminConfig getid \  
/Cell:mycell/Node:mynode/Server:myserver/ \  
EventInfrastructureProvider:/]
```

- Utilizando Jython:

```
providerid =  
AdminConfig.getid('/Cell:mycell/Node:mynode/Server:myserver/EventInfrastructureProvider:')  
print providerid
```

Ejemplo de salida:

```
EventInfrastructureProvider(cells/mycell/nodes/mynode/servers/myserver|resources-cei.xml#  
EventInfrastructureProvider_1)
```

2. Establezca los atributos necesarios.

- Utilizando Jacl:

```
set Name [list name "EventFilterName"]  
set JndiName [list jndiName "Indique aquí el nombre JNDI del nuevo filtro de sucesos"]  
set Description [list description "Indique aquí la descripción del filtro de sucesos"]  
set Category [list category "Indique aquí la categoría del filtro de sucesos"]
```

```
# escape todos los caracteres '[' en la serie de configuración  
# (por ejemplo, "CommonBaseEvent\[@severity=50\  
set filterConfigurationString [list filterConfigurationString \  
"Indique aquí la serie de configuración de filtro"]
```

- Utilizando Jython:

```
Name = ['name', 'EventFilterName']
JndiName = ['jndiName', 'Indique aquí el nombre JNDI del nuevo filtro de sucesos']
Description = ['description', 'Indique aquí la descripción del filtro de sucesos']
Category = ['category', 'Indique aquí la categoría del filtro de sucesos']
FilterConfigurationString = ['filterConfigurationString', \
    'Indique aquí la serie de configuración de filtro']
```

3. Establezca las propiedades para el nuevo filtro de sucesos.

- Utilizando Jacl:

```
set properties [list $name $jndiName $description $category \
    $filterConfigurationString]
```

- Utilizando Jython:

```
properties = [Name,JndiName,Description,Category,FilterConfigurationString]
print properties
```

Ejemplo de salida:

```
[['name', 'EventFilterName'], ['jndiName', 'Indique aquí el nombre JNDI para el nuevo
filtro de sucesos'], ['description', 'Establezca aquí la descripción del
filtro de sucesos'], ['category', 'Establezca aquí la categoría del
filtro de sucesos'], ['filterConfigurationString', 'Establezca aquí la serie
de configuración del filtro']]
```

4. Cree el filtro de sucesos.

- Utilizando Jacl:

```
set filterProf [$AdminConfig create FilterFactoryProfile \
    $providerid $properties]
```

- Utilizando Jython:

```
print AdminConfig.create('FilterFactoryProfile', providerid, properties)
```

Ejemplo de salida:

```
EventFilterName(cells/mycell/nodes/mynode/servers/myserver|resources-cei.xml#FilterFactoryProfile_1)
```

5. Guarde los cambios de la configuración.

6. Sólo en un entorno de despliegue de red, sincronice el nodo.

7. Reinicie el servidor.

---

## Anotación cronológica y rastreo de los componentes de Common Event Infrastructure

Puede habilitar la anotación cronológica y el rastreo para depurar problemas con una aplicación mediante Common Event Infrastructure.

### Acerca de esta tarea

Los componentes de Common Event Infrastructure utilizan la infraestructura de anotación cronológica Java JSR47, que está disponible en el servidor Common Event Infrastructure y en entornos de cliente. Para obtener más información acerca del uso de la infraestructura de anotación cronológica, consulte la documentación sobre resolución de problemas de WebSphere Application Server Network Deployment, versión 6.1.

En la tabla siguiente se indican los nombres de registrador que utilizan los componentes de Common Event Infrastructure.

*Tabla 2. Nombres de registrador*

Componente	Nombre de registrador
Nombre de registrador raíz	com.ibm.events
Catálogo de sucesos	com.ibm.events.catalog

Tabla 2. Nombres de registrador (continuación)

Componente	Nombre de registrador
Subcomponentes del servicio de sucesos	com.ibm.events.access com.ibm.events.bus com.ibm.events.distribution com.ibm.events.server
Plug-in de almacén de datos por omisión	com.ibm.events.datastore
Emisor de sucesos	com.ibm.events.emitter
Ayuda de notificación	com.ibm.events.notification
Configuración	com.ibm.events.configuration com.ibm.events.admintask
Configuración de base de datos	com.ibm.events.install.db
Migración	com.ibm.events.migration
Programas de utilidad varios	com.ibm.events.util

## Mantenimiento de la base de datos de sucesos

Si utiliza una base de datos de sucesos externa, ejecute periódicamente los scripts proporcionados para el mantenimiento de la base de datos.

### Actualización de las estadísticas de base de datos de sucesos DB2

Para habilitar una base de datos DB2 en Linux, UNIX o Windows para optimizar consultar y encontrar espacio libre, puede actualizar las estadísticas de base de datos ejecutando el script **runstats**.

#### Acerca de esta tarea

Se recomienda actualizar las estadísticas de base de datos DB2 con regularidad, y en especial en cualquiera de las siguientes circunstancias:

- Se han suprimido sucesos de la base de datos, mediante las interfaces de supresión de sucesos del servicio de sucesos o mediante el programa de utilidad de depuración de sucesos del plug-in de almacén de datos por omisión
- Un gran número de sucesos se han insertado en la base de datos
- Las tablas se han reorganizado con el script **reorg**
- Se han añadido o eliminado índices de la tabla

El script **runstats** se encuentra en el directorio *raíz\_perfil/event/nombre\_nodo/nombre\_servidor* o *nombre\_clúster/dbscripts/db2*.

#### Procedimiento

Para actualizar las estadísticas de la base de datos, ejecute uno de los siguientes mandatos:

- **Windows** En los sistemas Windows:  
runstats.bat *usuario\_bd* [password=*contraseña\_bd*]
- **Linux** **UNIX** En sistemas Linux y UNIX:  
runstats.sh *usuario\_bd* [password=*contraseña\_bd*]

Los parámetros son los siguientes:

*usuario\_bd*

El ID de usuario de la base de datos que se va a utilizar. Este parámetro es obligatorio.

*contraseña\_bd*

La contraseña de la base de datos. Este parámetro es opcional; si no especifica la contraseña en la línea de mandatos, la base de datos DB2 se la solicitará.

## Ejemplo

Por ejemplo, al ejecutar el siguiente mandato se actualizarán las estadísticas de base de datos de DB2 en un sistema Windows, utilizando el ID de usuario de base de datos dbadmin y la contraseña mypassword:

```
runstats.bat dbadmin mypassword
```

## Reorganización de las tablas de base de datos de sucesos DB2

Después de depurar o suprimir sucesos de una base de datos de sucesos DB2 en Linux, UNIX o Windows, puede volver a organizar las tablas de base de datos mediante el script **reorg**.

### Acerca de esta tarea

El script **reorg** se encuentra en el directorio *raíz\_perfil/event/nombre\_nodo/nombre\_servidor* o *nombre\_clúster/dbscripts/db2*.

### Procedimiento

Para reorganizar las tablas de base de datos de sucesos, ejecute uno de los siguientes mandatos:

- **Windows** En los sistemas Windows:  
`reorg.bat alias_bd usuario_bd [password=contraseña_bd]`
- **Linux** **UNIX** En sistemas Linux y UNIX:  
`reorg.sh alias_bd usuario_bd [password=contraseña_bd]`

Los parámetros son los siguientes:

*alias\_bd*

El alias de la base de datos. La base de datos de sucesos se debe catalogar en el cliente DB2; si ejecuta el script en el servidor DB2, la base de datos ya está catalogada.

*usuario\_bd*

El ID de usuario de la base de datos que se va a utilizar. Este parámetro es obligatorio.

*contraseña\_bd*

La contraseña de la base de datos. Este parámetro es opcional; si no especifica la contraseña en la línea de mandatos, la base de datos DB2 se la solicitará.

## Ejemplo

Por ejemplo, el siguiente mandato reorganiza las tablas de base de datos de sucesos en un sistema Windows utilizando el alias de base de datos eventdb, el ID de usuario dbadmin y la contraseña mypassword:

```
reorg.bat eventdb dbadmin mypassword
```

## Qué hacer a continuación

Después de ejecutar el script **reorg**, debe actualizar las estadísticas de base de datos mediante el script **runstats**. Para obtener más información, consulte “Actualización de las estadísticas de base de datos de sucesos DB2” en la página 54.

## Depuración de sucesos de la base de datos de sucesos

Puede ejecutar los scripts proporcionados para depurar un gran número de sucesos de la base de datos de sucesos.

### Acerca de esta tarea

El plug-in del almacén de datos por omisión proporciona un conjunto de programas de utilidad que puede ejecutar para depurar periódicamente un gran número de sucesos antiguos de la base de datos de sucesos. Estos programas de utilidad son distintos del mandato de servicio de sucesos eventpurge, que suprime los sucesos que coinciden con determinados criterios.

**Bases de datos Derby:** El programa de utilidad de depuración de base de datos no está soportado para una base de datos de sucesos Derby.

La capacidad de depuración de base de datos utiliza el concepto de *cubetas*. Una cubeta es un conjunto de tablas que se utiliza para almacenar sucesos en la base de datos de sucesos. El plug-in del almacén de datos por omisión utiliza dos cubetas:

- La cubeta activa es la cubeta que contiene los sucesos más recientes; los nuevos sucesos se almacenan en la cubeta activa. La cubeta activa no se puede depurar utilizando el programa de utilidad de depuración de base de datos.
- La cubeta inactiva contiene los sucesos más antiguos. Los sucesos almacenados en la cubeta inactiva se pueden consultar, suprimir o modificar, pero normalmente no se almacenan sucesos nuevos en la cubeta inactiva. La cubeta inactiva se puede depurar mediante el programa de utilidad de depuración de base de datos.

Cada suceso se almacena sólo en una cubeta. Desde el punto de vista del consumidor de sucesos, no hay diferencia entre las cubetas activa e inactiva; un consumidor puede consultar, modificar o suprimir un determinado suceso sin saber en qué cubeta está almacenado. La ventaja de este enfoque es que la cubeta inactiva se puede depurar utilizando interfaces específicas de la base de datos sin que esto afecte a la cubeta activa; el tráfico de sucesos normal puede continuar aunque se esté realizando la operación de depuración.

Una vez depurada la cubeta inactiva, puede intercambiar las cubetas para que la cubeta activa pase a ser la cubeta inactiva y viceversa. El intercambio de cubetas sólo se puede realizar cuando la cubeta inactiva está vacía.

Aunque los nuevos sucesos se almacenan sólo en la cubeta activa, en determinadas circunstancias se pueden almacenar en la cubeta inactiva inmediatamente después

de intercambiar las cubetas. El plug-in de almacén de datos comprueba periódicamente qué cubeta está marcada actualmente como activa, pero hasta que se realiza la siguiente comprobación, algunos sucesos pueden continuar almacenándose en la cubeta inactiva. De forma parecida, los sucesos enviados como parte de un lote se almacenan todos en la misma cubeta, aunque esa cubeta se inactive mientras se está procesando el lote.

Si desea utilizar la posibilidad de depuración rápida, es su responsabilidad determinar con qué frecuencia se intercambian las cubetas o se depura la cubeta inactiva, dependiendo de tráfico de sucesos, el espacio de almacenamiento, los requisitos de archivado u otras consideraciones.

## **Visualización o cambio del estado de la cubeta activa de la base de datos de sucesos**

El estado de la cubeta activa indica qué cubeta está disponible y qué cubeta no está disponible actualmente.

### **Procedimiento**

Para ver o cambiar el estado de la cubeta activa, utilice el mandato `eventbucket`:  
`eventbucket [-status] [-change]`

Este mandato tiene las siguientes opciones:

#### **-status**

Utilice esta opción para ver información sobre la configuración de la cubeta actual, incluido el valor de cubeta activa y el intervalo de comprobación de cubetas (la frecuencia con la que el plug-in del almacén de datos realiza comprobaciones para determinar qué cubeta está activa).

#### **-change**

Utilice esta opción para intercambiar las cubetas activas e inactivas. La cubeta inactiva debe estar vacía para poder utilizar esta opción.

## **Depuración de la cubeta inactiva de base de datos de sucesos**

El método que se utiliza para depurar la cubeta inactiva varía dependiendo del software de la base de datos.

### **Acerca de esta tarea**

**Bases de datos Derby:** El programa de utilidad de depuración rápida no está soportado para una base de datos de sucesos Derby.

### **Depuración de la cubeta inactiva para una base de datos de sucesos DB2 (sistemas Linux, UNIX o Windows):**

En los sistemas Linux, UNIX y Windows, el programa de utilidad de depuración de base de datos para una base de datos DB2 se implementa como un script de shell o un archivo de proceso por lotes.

### **Procedimiento**

Para depurar la cubeta inactiva, ejecute uno de los siguientes mandatos:

- Para depurar la cubeta inactiva, ejecute el siguiente mandato:  
`fastpurge alias_bd usuario_bd [password=contraseña_bd] [copydir=dir_copia]`

Los parámetros de este mandato son los siguientes:

#### *alias\_bd*

El alias de la base de datos. La base de datos de sucesos se debe catalogar en el cliente DB2; si ejecuta el script en el servidor DB2, la base de datos ya está catalogada.

#### *usuario\_bd*

El ID de usuario de base de datos que debe utilizarse al conectarse a la base de datos de sucesos.

#### *contraseña\_bd*

La contraseña que se utiliza para el ID de usuario especificado. Este parámetro es opcional; si no especifica la contraseña, la base de datos DB2 se la solicitará.

#### *dir\_copia*

La vía de acceso a un directorio que se utilizará para los archivos generados por el programa de utilidad de carga. Este parámetro sólo es necesario si ha habilitado la recuperación hacia adelante de la base de datos de sucesos (con los valores de la configuración de base de datos LOGRETAIN o USEREXIT habilitados). Por omisión, la base de datos de sucesos no utiliza la recuperación hacia adelante.

### **Depuración de la cubeta inactiva para una base de datos de sucesos DB2:**

El programa de utilidad de depuración de base de datos para una base de datos de sucesos DB2 se implementa utilizando el programa de utilidad de carga de DB2.

#### **Acerca de esta tarea**

Para depurar la cubeta inactiva:

#### **Procedimiento**

1. Ejecute el mandato eventbucket para identificar la cubeta inactiva (cubeta 0 o cubeta 1).
2. Cargue el archivo de control del programa de utilidad adecuado. Estos archivos se generan durante la configuración de la base de datos y se encuentran en el directorio /event/dbscripts/db2zos. Cargue uno de los siguientes archivos:
  - Si la cubeta 0 está inactiva, cargue el archivo fastpurge00.ctl.
  - Si la cubeta 1 está inactiva, cargue el archivo fastpurge01.ctl.

**Formato de carga:** El archivo de control se debe cargar con un formato de registro fijo y una longitud de registro lógico de 80.

3. Vaya al menú de opciones primarias de DB2I de ISPF y seleccione la opción **Utilities** (Programas de utilidad).
4. Especifique la siguiente información:

<b>Campo</b>	<b>Valor</b>
<b>Función</b>	EDITJCL
<b>Programa de utilidad</b>	LOAD
<b>Conjunto de datos de sentencia</b>	El nombre del conjunto de datos que contiene el archivo de control cargado
<b>LISTDEF</b>	NO
<b>Plantilla</b>	NO

5. Pulse Intro para continuar en el siguiente panel.



6. En el campo de entrada recdsn, escriba el nombre del conjunto de datos que contiene el archivo de control cargado.
7. Pulse Intro. Se genera el script JCL para depurar la cubeta inactiva.
8. Pulse Intro para borrar los mensajes de salida.
9. Edite el script JCL generado según sea necesario.
10. Someta el script JCL.

### Resultados

El script JCL generado se puede volver a utilizar. Después de crear los scripts para depurar las dos cubetas (0 y 1), no es necesario repetir todo este procedimiento.

### Depuración de la cubeta inactiva para una base de datos de sucesos DB2 (sistemas iSeries):

En sistemas iSeries, el programa de utilidad de depuración de base de datos para una base de datos de sucesos DB2 se implementa como un procedimiento almacenado.

#### Acerca de esta tarea

Para depurar la cubeta inactiva:

#### Procedimiento

1. Conéctese al sistema iSeries mediante una aplicación de terminal que emule un terminal 5250, como IBM Personal Communications.
2. Inicie la sesión con un ID de usuario que tenga suficientes privilegios para ejecutar el procedimiento almacenado de depuración rápida. Se recomienda el ID de usuario utilizado para crear la base de datos de sucesos.
3. Ejecute el mandato **strsql**.
4. Escriba el siguiente mandato SQL:  

```
call colección/fast_purge()
```

*colección* es el nombre de la colección que contiene la base de datos de sucesos.  
La colección por omisión es event:  

```
call event/fast_purge()
```

### Depuración de la cubeta inactiva para una base de datos de sucesos Oracle:

El programa de utilidad de depuración de base de datos para una base de datos de sucesos Oracle se implementa como un procedimiento almacenado.

#### Antes de empezar

Para poder utilizar el programa de utilidad de depuración de base de datos para una base de datos Oracle, SQL\*Plus debe estar instalado en el cliente Oracle y el cliente debe estar configurado para comunicarse con la base de datos Oracle (el archivo tnsnames.ora debe estar configurado correctamente). Consulte la documentación de Oracle para obtener más información.

#### Procedimiento

Para depurar la cubeta inactiva, ejecute el procedimiento almacenado utilizando SQL\*Plus:

```
sqlplus serie_conexión
@fastpurge.sql
```

El archivo `fastpurge.sql` se almacena en la misma ubicación que los otros scripts generados durante la configuración de la base de datos (la ubicación por omisión es `raíz_perfil/databases/event/nombre_nodo/nombre_servidor` o `nombre_clúster/dbscripts/oracle`).

El parámetro `serie_conexión` es la serie de conexión de Oracle. Utilice el mismo ID de usuario de base de datos utilizado para crear la base de datos de sucesos.

### Depuración de la cubeta inactiva para una base de datos de sucesos Informix:

El programa de utilidad de depuración de base de datos para una base de datos de sucesos Informix se implementa como un procedimiento almacenado.

#### Antes de empezar

Para utilizar el programa de utilidad de una base de datos Informix, debe ejecutar el mandato `dbaccess` de Informix en un entorno cuyo origen se haya establecido correctamente para las variables de entorno de Informix. Para obtener más información, consulte la documentación de Informix.




#### Procedimiento

1. Para depurar la cubeta inactiva, ejecute el procedimiento almacenado utilizando `dbaccess`:

```
dbaccess - fastpurge.sql
```

El archivo `fastpurge.sql` se almacena en la misma ubicación que los demás scripts generados durante la configuración de la base de datos (la ubicación por omisión es `nombre_perfil/databases/event/nodo/servidor/dbscripts/informix`).

2. Ejecute el script `eventpurgepool` para depurar la agrupación de conexiones:

-  En sistemas Windows: `eventpurgepool.bat`
-   En sistemas Linux o UNIX: `eventpurgepool.sh`

### Depuración de la cubeta inactiva para una base de datos de sucesos de SQL Server:

El programa de utilidad de depuración de base de datos para una base de datos de sucesos de SQL Server se implementa como un procedimiento almacenado.

#### Antes de empezar

Para utilizar este programa de utilidad para una base de datos de SQL Server, debe tener instalado el programa de utilidad `osql` en el cliente de SQL Server. Consulte la documentación de SQL Server para obtener más información.

**Nota:** No utilice el programa de utilidad `isql` para ejecutar el procedimiento almacenado de depuración de base de datos.

#### Procedimiento

Para depurar la cubeta inactiva, ejecute el procedimiento almacenado utilizando el programa de utilidad `osql`:

```
osql -Snombre_servidor -Uusuario_bd -Pcontraseña_bd -dbd_sucesos -Q"fast_purge"
```

El archivo `fast_purge` se almacena en la misma ubicación que los otros scripts generados durante la configuración de la base de datos (la ubicación por omisión es `raíz_perfil/databases/event/nodo/servidor/dbscripts/sqlserver`). Los parámetros de este mandato son los siguientes:

*nombre\_servidor*

Nombre de servidor de bases de datos.

*usuario\_bd*

El ID de usuario de base de datos que debe utilizarse al conectarse a la base de datos de sucesos.

*contraseña\_bd*

La contraseña que se utiliza para el ID de usuario especificado. Este parámetro es opcional; si no especifica la contraseña, el programa de utilidad `osql` se la solicitará.

*bd\_sucesos*

Nombre de la base de datos de sucesos (normalmente `event`).

## **Cambio del intervalo de comprobación de cubeta de base de datos de sucesos**

El intervalo de comprobación de cubeta especifica con qué frecuencia el plug-in del almacén de datos realiza comprobaciones para determinar qué cubeta está activa. Este valor se especifica como una propiedad personalizada en los valores del almacén de datos.

### **Acerca de esta tarea**

El intervalo de comprobación de cubeta por omisión es 5 minutos (300 segundos). Un intervalo más corto reduce la probabilidad de que los sucesos se almacenen en la cubeta inactiva después del intercambio, pero puede disminuir el rendimiento.

Para cambiar el intervalo de comprobación de cubetas:

### **Procedimiento**

1. En la consola administrativa, pulse **Integración de servicios > Common Event Infrastructure > Servicio de sucesos > Servicios de sucesos > servicio\_sucesos > Almacén de datos de sucesos**.
2. Modifique el valor de la propiedad `BucketCheckInterval` para especificar el intervalo de comprobación de cubetas en segundos.

### **Resultados**

Los cambios entrarán en vigor la próxima vez que se inicie el servidor.



---

## Capítulo 4. Protección del acceso a las funciones de Common Event Infrastructure

Puede utilizar la seguridad declarativa de nivel de método de WebSphere para proteger el acceso a las funciones de Common Event Infrastructure.

Common Event Infrastructure define seis roles de seguridad, cada uno de ellos está asociado a un grupo relacionado de funciones. Estos roles de seguridad controlan el acceso a los mandatos y las interfaces de programación.

En la tabla siguiente se describen los roles de seguridad y los tipos de usuarios asociados con cada rol.

*Tabla 3. Roles de seguridad y tipos de usuario*

Rol de seguridad	Tipos de usuario
eventCreator	<p>Orígenes de sucesos que necesitan someter sucesos a un emisor utilizando llamadas de EJB síncronas. Este rol proporciona acceso a las siguientes interfaces:</p> <ul style="list-style-type: none"><li>• Emitter.sendEvent()</li><li>• Emitter.sendEvents()</li><li>• Mandato <b>eventemit</b></li></ul> <p>El rol eventCreator restringe el acceso para someter sucesos sólo si se configura el emisor para utilizar llamadas de EJB síncronas en la transmisión de sucesos. Si el emisor utiliza la mensajería JMS asíncrona para la transmisión de sucesos, debe utilizar la seguridad JMS para restringir el acceso al destino utilizado para someter sucesos.</p>
eventUpdater	<p>Consumidores de sucesos que necesitan actualizar los sucesos almacenados en la base de datos de sucesos. Este rol proporciona acceso a las siguientes interfaces:</p> <ul style="list-style-type: none"><li>• EventAccess.updateEvents()</li><li>• EventAccess.eventExists()</li><li>• EventAccess.queryEventByGlobalInstanceId()</li><li>• EventAccess.queryEventsByAssociation()</li><li>• EventAccess.queryEventsByEventGroup()</li><li>• Mandato <b>eventquery</b></li></ul>
eventConsumer	<p>Consumidores de sucesos que necesitan consultar los sucesos almacenados en la base de datos de sucesos. Este rol proporciona acceso a las siguientes interfaces:</p> <ul style="list-style-type: none"><li>• EventAccess.eventExists()</li><li>• EventAccess.queryEventByGlobalInstanceId()</li><li>• EventAccess.queryEventsByAssociation()</li><li>• EventAccess.queryEventsByEventGroup()</li><li>• Mandato <b>eventquery</b></li></ul>

Tabla 3. Roles de seguridad y tipos de usuario (continuación)

Rol de seguridad	Tipos de usuario
eventAdministrator	<p>Consumidores de sucesos que necesitan consultar, actualizar y suprimir los sucesos almacenados en la base de datos de sucesos. Este rol proporciona acceso a las siguientes interfaces:</p> <ul style="list-style-type: none"> <li>• EventAccess.purgeEvents()</li> <li>• EventAccess.eventExists()</li> <li>• EventAccess.queryEventByGlobalInstanceId()</li> <li>• EventAccess.queryEventsByAssociation()</li> <li>• EventAccess.queryEventsByEventGroup()</li> <li>• EventAccess.updateEvents()</li> <li>• Emitter.sendEvent()</li> <li>• Emitter.sendEvents()</li> <li>• Mandato <b>eventquery</b></li> <li>• Mandato <b>eventpurge</b></li> <li>• Mandato <b>eventemit</b></li> <li>• Mandato <b>eventbucket</b></li> </ul>
catalogReader	<p>Aplicaciones de catálogo de sucesos que necesitan recuperar definiciones de sucesos del catálogo de sucesos. Este rol proporciona acceso a las siguientes interfaces:</p> <ul style="list-style-type: none"> <li>• EventCatalog.getAncestors()</li> <li>• EventCatalog.getChildren()</li> <li>• EventCatalog.getDescendants()</li> <li>• EventCatalog.getEventDefinition()</li> <li>• EventCatalog.getEventDefinitions()</li> <li>• EventCatalog.getEventExtensionNamesForSourceCategory()</li> <li>• EventCatalog.getEventExtensionToSourceCategoryBindings()</li> <li>• EventCatalog.getParent()</li> <li>• EventCatalog.getRoot()</li> <li>• EventCatalog.getSourceCategoriesForEventExtension()</li> <li>• Mandato <b>eventcatalog</b> (opción -listdefinitions)</li> <li>• Mandato <b>eventcatalog</b> (opción -listcategories)</li> <li>• Mandato <b>eventcatalog</b> (opción -exportdefinitions)</li> </ul>
catalogAdministrator	<p>Aplicaciones de catálogo de sucesos que necesitan crear, actualizar, suprimir o recuperar definiciones de sucesos en el catálogo de sucesos. Este rol proporciona acceso a todos los métodos de la interfaz EventCatalog y a todas las funciones del mandato <b>eventcatalog</b>. Como los cambios en el catálogo de sucesos pueden provocar la generación de sucesos, este rol también proporciona acceso a las interfaces para someter sucesos.</p>

El bean controlado por mensajes del servidor de sucesos se ejecuta utilizando la identidad de usuario. Si utiliza una transmisión JMS asíncrona para someter sucesos al servicio de sucesos y ha habilitado la seguridad basada en métodos, debe correlacionar esta identidad de usuario con el rol eventCreator.

**Seguridad:** Si el origen de sucesos se ejecuta con la seguridad de Java habilitada, y desea generar sus propios identificadores exclusivos globalmente (GUID), debe modificar el archivo de política para habilitar el proceso correcto. Añada las siguientes entradas:

```
permission java.io.FilePermission "${java.io.tmpdir}${/}guid.lock",  
    "read,write,delete";  
permission java.net.SocketPermission "*", "resolve";
```





---

## Capítulo 5. Resolución de problemas de Common Event Infrastructure

Estos temas proporcionan información de resolución de problemas para el servicio de sucesos basándose en la tarea o actividad que estaba realizando cuando surgió el problema.

---

### Problemas durante el inicio

Resolución de problemas de inicio del servidor de Common Event Infrastructure.

#### El servicio de sucesos no se inicia (mensaje CEIDS0058E)

*El servicio de sucesos no se inicia e imprime el mensaje CEIDS0058E en el archivo de anotación cronológica de WebSphere.*

##### Causa

El servicio de sucesos utiliza sentencias SQL calificadas con el nombre de usuario. Este error indica que el nombre de usuario que utiliza el servicio de sucesos para conectarse a la base de datos de sucesos no es el mismo que el ID de usuario que se utilizó para crear la base de datos.

##### Solución

El ID de usuario utilizado para conectarse a la base de datos de sucesos debe ser el mismo que el utilizado para crear la base de datos de sucesos. Para corregir este problema:

1.
  - Para un solo servidor, seleccione **Servidores > Servidores de aplicaciones > nombre\_servidor**.
  - Para un clúster, seleccione **Servidores > Clústeres > nombre\_clúster**.
2. En la pestaña **Configuración**, seleccione **Business Integration > Common Event Infrastructure > Servidor Common Event Infrastructure**.
3. Cambie el ID de usuario y la contraseña especificados por los utilizados para crear la base de datos.
4. Guarde los cambios de la configuración.
5. Reinicie el servidor.

---

### Problemas al enviar sucesos

Resolución de problemas de envío de sucesos del servidor de Common Event Infrastructure.

#### Error al enviar suceso (mensaje CEIDS0060E)

*Mi origen de sucesos encuentra un error al intentar enviar un suceso y aparece el mensaje CEIDS0060E en el archivo de anotación cronológica WebSphere.*

## Causa

El servicio de sucesos utiliza los metadatos almacenados en la base de datos de sucesos para correlacionar atributos y elementos de Common Base Event con tablas y columnas de base de datos. Esta información se lee de la base de datos la primera vez que una aplicación intenta utilizar el servicio de sucesos después del inicio.

Las tablas de metadatos se rellenan cuando se crea la base de datos de sucesos. Este error se produce si las tablas no contienen los metadatos necesarios en el tiempo de ejecución.

## Solución

Para corregir este problema, es necesario volver a crear los metadatos necesarios. Al crear la base de datos de sucesos, el mandato administrativo de configuración de base de datos también genera un script de base de datos que se puede utilizar para volver a rellenar los metadatos más adelante. El nombre de este script depende del tipo de base de datos:

Tipo de base de datos	Nombre de script
Derby	ins_metadata.derby
DB2	ins_metadata.db2
Informix	ins_metadata.sql
Oracle	ins_metadata.ora
SQL Server	ins_metadata.mssql
DB2 UDB para iSeries	ins_metadata.db2

Por omisión, el script se crea en el directorio *raíz\_perfil/dbscripts/ basedatos\_CEI\_nombrenodo*. Este script se puede ejecutar en cualquier momento.

Para volver a crear los metadatos, utilice el procesador SQL adecuado para ejecutar el script:

- Windows Linux UNIX DB2: db2
- Windows Linux UNIX Oracle: SQL\*Plus
- Windows Linux UNIX Informix: dbaccess
- Windows SQL Server: osql

Después de volver a llenar los metadatos, reinicie el servidor.

## Error al enviar suceso (ServiceUnavailableException)

*Mi aplicación de origen de sucesos encuentra un error al intentar enviar un suceso al servidor de sucesos. El archivo de anotación cronológica indica una ServiceUnavailableException con el mensaje "Se ha producido una anomalía en la comunicación al intentar obtener un contexto iniciar con el URL de proveedor".*

## Causa

Este problema indica que la aplicación de origen de sucesos no se puede conectar con el servidor de sucesos. Esto se puede deber a una de las siguientes condiciones:

- El servidor de sucesos no se está ejecutando.
- La aplicación de origen de sucesos no está configurada para utilizar el URL de proveedor JNDI correcto.

## Solución

Para resolver este problema, siga estos pasos:

1. Para comprobar el estado del servidor de sucesos, vaya al directorio `raíz_perfil/bin` y ejecute el mandato **serverStatus**:  
`serverStatus nombre_servidor`
2. Si el servidor de sucesos no está en ejecución, utilice el mandato **startServer** para iniciarlo:  
`startServer nombre_servidor`
3. Compruebe el nombre de host y el puerto RMI (Remote Method Invocation) del servidor que contiene la aplicación que no puede conectarse al servidor de sucesos. Confirme que se han especificado los mismos valores en el URL JNDI configurado para la aplicación del origen de sucesos. Si el servidor CEI está en otro servidor, entonces JNDI se ha de resolver con dicho destino de despliegue remoto.

## Error al enviar suceso (NameNotFoundException)

*Mi aplicación de origen de sucesos encuentra un error al intentar enviar un suceso al servicio de sucesos. El archivo de anotación cronológica indica una `NameNotFoundException` con un mensaje parecido a "No se ha encontrado el primer componente en `events/configuration/emitter/Default`".*

## Causa

Este problema indica que el servicio de sucesos no está disponible y que puede ser causado por alguna de las siguientes condiciones:

- El servicio de sucesos no se ha desplegado.
- El servicio de sucesos está inhabilitado.

## Solución

Para desplegar el servicio de sucesos:

1. Inicie la herramienta `wsadmin`.
2. Utilice el objeto `AdminTask` para ejecutar el mandato administrativo `deployEventService`.
3. Reinicie el servidor.

Para habilitar el servicio de sucesos mediante la herramienta `wsadmin`:

1. Inicie la herramienta `wsadmin`.
2. Utilice el objeto `AdminTask` para ejecutar el mandato administrativo `enableEventService`.
3. Reinicie el servidor.

Para habilitar el servicio de sucesos utilizando la consola administrativa:

1. Pulse **Aplicaciones** → **Tipos de aplicaciones** → **Aplicaciones empresariales WebSphere** → *servidor* → **Servicios de contenedor** → **Servicio Common Event Infrastructure**.
2. Seleccione la propiedad **Habilitar el servicio durante el arranque del servidor**.

3. Pulse **Aceptar** para guardar los cambios.
4. Reinicie el servidor.

## Error al enviar suceso (mensaje CEIEM0025E)

*Mi aplicación de origen de sucesos encuentra un error al intentar enviar un suceso al servidor de sucesos. El archivo de anotación cronológica indica una DuplicateGlobalInstanceIdException.*

### Causa

Este problema indica que el emisor ha enviado el suceso, pero el servicio de sucesos lo ha rechazado porque ya existe otro suceso con el mismo identificador de instancia global. Cada suceso debe tener un identificador de instancia global exclusivo, especificado por la propiedad globalInstanceId.

### Solución

Para resolver este problema, realice una de las siguientes opciones:

- Asegúrese de que la aplicación de origen de sucesos genera un identificador de instancia global exclusivo para cada suceso.
- Deje vacía la propiedad globalInstanceId de los sucesos enviados. El emisor generará automáticamente un identificador exclusivo para cada suceso.

## Error al enviar suceso (mensaje CEIEM0034E)

*Mi origen de sucesos encuentra un error al intentar enviar un suceso al servicio de sucesos. El archivo de anotación cronológica indica un EmitterException con el mensaje "La consulta JNDI de una cola JMS ha fallado porque el nombre JNDI definido en el perfil de emisor no está enlazado en la JNDI".*

### Causa

Este problema indica que la configuración de la transmisión JMS que utiliza el emisor especifica uno o más recursos JMS que no están definidos en la configuración JMS.

### Solución

Para corregir este problema:

1. En la consola administrativa pulse **Integración de servicios > Common Event Infrastructure > Fábricas de emisores de sucesos > fábrica\_emisores > Valores de transmisión JMS**. Asegúrese de que está viendo la transmisión JMS para la fábrica de emisores que utiliza la aplicación de origen de sucesos.
2. Compruebe los valores especificados para las propiedades **Nombre JNDI de cola** y **Nombre JNDI de fábrica de conexiones de cola**. Asegúrese de que los nombres JNDI especificados existen en el espacio de nombres JNDI y que son objetos JMS válidos. Si es necesario, modifique estas propiedades o cree los recursos JMS necesarios.

## El suceso no es válido (mensaje CEIEM0027E)

*Mi origen de sucesos está intentando enviar un suceso, pero el emisor no lo envía al servicio de sucesos e imprime el mensaje CEIEM0027E en el archivo de anotación cronológica ("El emisor no ha enviado el suceso al servidor de sucesos porque Common Base Event no es válido").*

## Causa

Este mensaje indica que una o varias propiedades de sucesos contienen datos que no se ajustan a la especificación de Common Base Event. Hay varias formas en las que los datos de sucesos pueden no ser válidos, incluidas las siguientes:

- El identificador de instancia global debe tener como mínimo una longitud de 32 caracteres pero no más de 64 caracteres.
- La gravedad debe estar comprendida entre 0 y 70.

## Solución

Para corregir este problema:

1. Consulte el mensaje de excepción detallado en el archivo de anotación cronológica para determinar qué propiedad de sucesos no es válida. Por ejemplo, este mensaje indica que la longitud del identificador de instancia global (ABC) no es válido:

```
Excepción: org.eclipse.hyades.logging.events.cbe.ValidationException
: IWAT0206E La longitud del identificador en la propiedad de Common
Base Event especificada está fuera del rango válido de 32 a 64
caracteres.
Propiedad: CommonBaseEvent.globalInstanceId
Valor: ABC
```

2. Corrija el contenido de sucesos en el origen de modo que se ajuste a la especificación de Common Base Event.
3. Vuelva a enviar el suceso.

## Modalidad de sincronización no soportada (mensaje CEIEM0015E)

*Mi origen de sucesos está intentando enviar un suceso, pero el emisor no lo envía al servicio de sucesos e imprime el mensaje CEIEM0015E en el archivo de anotación cronológica ("El emisor no da soporte a la modalidad de sincronización especificada").*

## Causa

Este problema indica que los parámetros pasados por el origen de sucesos al enviar el suceso especifican una modalidad de sincronización a la que el emisor no da soporte. Esto puede deberse a cualquiera de las siguientes condiciones:

- El origen de sucesos está especificando una modalidad de sincronización que no es válida. Esto lo indica una `IllegalArgumentException` con el mensaje "La modalidad de sincronización *modalidad* no es válida".
- El origen de sucesos especifica una modalidad de sincronización que el emisor no está configurado para dar soporte. Esto lo indica una `SynchronizationModeNotSupportedException` con el mensaje "El emisor no da soporte a la modalidad de sincronización especificada: *modalidad*."

## Solución

Si el mensaje de excepción indica que el origen de sucesos está especificando una modalidad de sincronización que no es válida (`IllegalArgumentException`), compruebe la llamada al método que está intentando enviar el suceso. Asegúrese de que los parámetros del método especifican una de las modalidades de sincronización válidas:

- `SynchronizationMode.ASYNCHRONOUS`
- `SynchronizationMode.SYNCHRONOUS`

- SynchronizationMode.DEFAULT

Estas constantes están definidas por la interfaz `com.ibm.events.emitter.SynchronizationMode`.

Si el mensaje de excepción indica que la modalidad de sincronización especificada no recibe soporte en el emisor (`SynchronizationModeNotSupportedException`), compruebe la configuración de la fábrica de emisores:

1. En la consola administrativa, pulse **Integración de servicios > Common Event Infrastructure > Fábricas de emisores de sucesos > fábrica\_emisores**. Asegúrese de que está viendo la fábrica de emisores que utiliza la aplicación de origen de sucesos.
2. Compruebe los valores de la fábrica de emisores para ver qué modalidades de sincronización están soportadas:
  - Si se selecciona la propiedad **Dar soporte a la transmisión del servicio de sucesos**, se da soporte a la modalidad síncrona.
  - Si se selecciona **Dar soporte a la transmisión JMS**, se da soporte a la modalidad asíncrona.

**Consultas de las modalidades de transacción:** Un origen de sucesos puede consultar mediante programación las modalidades de transacción soportadas para un emisor concreto utilizando el método `isSynchronizationModeSupported()`. Consulte la documentación de la API de Javadoc para obtener más información.

3. Si el emisor no da soporte a la modalidad de sincronización que está intentando utilizar, debe cambiar la configuración de la fábrica de emisores o modificar el origen de sucesos para utilizar una modalidad de configuración soportada.

## Modalidad de transacción no soportada (mensaje CEIEM0016E)

*Mi origen de sucesos está intentando enviar un suceso, pero el emisor no lo envía al servicio de sucesos e imprime el mensaje CEIEM0016E en el archivo de anotación cronológica ("El emisor no da soporte a la modalidad de transacción especificada").*

### Causa

Este problema indica que los parámetros pasados por el origen de sucesos al enviar el suceso especifican una modalidad de transacción a la que el emisor no da soporte. Esto puede deberse a cualquiera de las siguientes condiciones:

- El origen de sucesos está especificando una modalidad de transacción que no es válida.
- El origen de sucesos está especificando una modalidad de sincronización a la que el entorno del emisor no da soporte. Sólo se da soporte a las transacciones en un contenedor Java EE.

### Solución

Para corregir este problema, compruebe la llamada al método que está intentando enviar el suceso y asegúrese de que los parámetros del método especifican la modalidad de transacción correcta:

- Si el emisor se está ejecutando en un contenedor Java EE, asegúrese de que los parámetros del método especifican una de las modalidades de transacción válidas:

- TransactionMode.NEW
- TransactionMode.SAME
- TransactionMode.DEFAULT

Estas constantes se definen mediante la interfaz `com.ibm.events.emitter.TransactionMode`.

- Si el emisor no se está ejecutando en un contenedor Java EE, asegúrese de que los parámetros del método especifican `TransactionMode.DEFAULT`.

## Problemas al recibir o consultar sucesos

Resolución de problemas relacionados con la recepción y consulta de sucesos del servidor de Common Event Infrastructure.

### Error al consultar sucesos (mensaje CEIDS0060E)

*Mi cliente de sucesos encuentra un error al intentar consultar sucesos del servicio de sucesos y el mensaje CEIDS0060E aparece en el archivo de anotación cronológica de WebSphere.*

#### Causa

El servicio de sucesos utiliza los metadatos almacenados en la base de datos de sucesos para correlacionar atributos y elementos de Common Base Event con tablas y columnas de base de datos. Esta información se lee de la base de datos la primera vez que una aplicación intenta utilizar el servicio de sucesos después del inicio.

Las tablas de metadatos se rellenan cuando se crea la base de datos de sucesos. Este error se produce si las tablas no contienen los metadatos necesarios en el tiempo de ejecución.

#### Solución

Para corregir este problema, es necesario volver a crear los metadatos necesarios. Al crear la base de datos de sucesos, el mandato administrativo de configuración de base de datos también genera un script de base de datos que se puede utilizar para volver a rellenas los metadatos más adelante. El nombre de este script depende del tipo de base de datos:

Tipo de base de datos	Nombre de script
Derby	ins_metadata.derby
DB2	ins_metadata.db2
Informix	ins_metadata.sql
Oracle	ins_metadata.ora
SQL Server	ins_metadata.mssql
DB2 UDB para iSeries	ins_metadata.db2

Por omisión, el script se crea en el directorio `raíz_perfil/dbscripts/ basedatos_CEI_nombrenodo`. Este script se puede ejecutar en cualquier momento.

Para volver a crear los metadatos, utilice el procesador SQL adecuado para ejecutar el script:

- **Windows** **Linux** **UNIX** DB2: db2
- **Windows** **Linux** **UNIX** Oracle: SQL\*Plus
- **Windows** **Linux** **UNIX** Informix: dbaccess
- **Windows** SQL Server: osql

Después de volver a llenar los metadatos, reinicie el servidor.

## Los sucesos no se almacenan en el almacén de datos persistente

*Mi aplicación de origen de sucesos está enviando sucesos satisfactoriamente al emisor, pero cuando un origen de sucesos realiza una consulta en los sucesos, éstos no están en el almacén de datos persistente.*

### Causa

Este problema indica que el emisor no está enviando sucesos al servicio de sucesos, o que el servicio de sucesos no está almacenando los sucesos en el almacén de datos persistente. Esto se debe a cualquiera de las siguientes condiciones:

- El almacén de datos persistente no está habilitado para el servicio de sucesos.
- Los sucesos no pertenecen a un grupo de sucesos que se ha configurado para la persistencia de sucesos.
- El emisor está filtrando los sucesos.

### Solución

Para verificar que el almacén de datos persistente está habilitado para el servicio de sucesos:

1. En la consola administrativa, pulse **Integración de servicios > Common Event Infrastructure > Servicio de sucesos > Servicios de sucesos > servicio\_sucesos**.
2. Asegúrese de que el recuadro de selección **Habilitar almacén de datos de sucesos** se ha seleccionado.
3. Pulse **Aceptar** para guardar los cambios.

Para verificar que el grupo de sucesos está configurado para la persistencia de sucesos:

1. En la consola administrativa, pulse **Integración de servicios > Common Event Infrastructure > Servicio de sucesos > Servicios de sucesos > servicio\_sucesos > Grupos de sucesos > grupo\_sucesos**.
2. Asegúrese de que el recuadro de selección **Persistir sucesos en el almacén de datos de sucesos** se ha seleccionado.
3. Pulse **Aceptar** para guardar los cambios.

**Varios grupos de sucesos:** Un suceso puede pertenecer a varios grupos de sucesos. Si alguno de los grupos de sucesos pertinente está configurado para la persistencia, y el almacén de datos está habilitado, el suceso se almacena en el almacén de datos.

Para comprobar los valores de filtro:

1. En la consola administrativa pulse **Integración de servicios > Common Event Infrastructure > Fábricas de emisores de sucesos > fábrica\_emisores > Filtro de**



- sucesos.** (Asegúrese de que está viendo los valores para la fábrica de emisores que la aplicación de origen de sucesos está utilizando.)
2. Compruebe si la serie de configuración de filtro excluye los sucesos que está intentando enviar a los clientes. Si es así, puede modificar la configuración del filtro o modificar los datos de sucesos para que los sucesos no se filtren.
  3. Pulse **Aceptar** para guardar los cambios.

## Los clientes no reciben los sucesos (ningún mensaje de error)

*Mi aplicación de origen de sucesos está enviando sucesos satisfactoriamente al emisor, pero los clientes no reciben los sucesos mediante la interfaz JMS.*

### Causa

Este problema puede deberse a cualquiera de las siguientes condiciones:

- La distribución de sucesos no está habilitada para el servicio de sucesos.
- El emisor está filtrando los sucesos.
- La ayuda de notificación está filtrando los sucesos.
- El cliente de sucesos no está especificando el grupo de sucesos correcto.
- La conexión JMS no se ha iniciado.

### Solución

La solución para este problema depende de la causa subyacente.

- **Para verificar que la distribución de sucesos está habilitada para el servicio de sucesos:**
  1. En la consola administrativa, pulse **Integración de servicios > Common Event Infrastructure > Servicio de sucesos > Servicios de sucesos > servicio\_sucesos.**
  2. Si la propiedad **Habilitar distribución de sucesos** no está seleccionada, seleccione el recuadro de selección.
  3. Pulse **Aceptar** para guardar los cambios.
- **Para comprobar los valores de filtro de sucesos para el emisor:**
  1. En la consola administrativa pulse **Integración de servicios > Common Event Infrastructure > Fábricas de emisores de sucesos > fábrica\_emisores > Filtro de sucesos.** (Asegúrese de que está viendo los valores para la fábrica de emisores que la aplicación de origen de sucesos está utilizando.)
  2. Compruebe si la serie de configuración de filtro excluye los sucesos que está intentando enviar a los clientes. Si es así, puede modificar la configuración del filtro o modificar los datos de sucesos para que los sucesos no se filtren.
  3. Pulse **Aceptar** para guardar los cambios.
- **Para comprobar los valores de filtro de sucesos para la ayuda de notificación:**
  1. Compruebe si en la aplicación de cliente de sucesos se ha especificado un selector de sucesos para la ayuda de notificación utilizando el método `NotificationHelper.setEventSelector`.
  2. Si se ha especificado un selector de sucesos, asegúrese de que no excluye el suceso que está intentando recibir. (Un selector de sucesos nulo pasa todos los sucesos.)
- **Para comprobar el grupo de sucesos especificado por el cliente de sucesos:**

1. En la consola administrativa, pulse **Integración de servicios > Common Event Infrastructure > Servicio de sucesos > Servicios de sucesos > servicio\_sucesos > Grupos de sucesos**. La tabla muestra una lista de todos los grupos de sucesos definidos para el servicio de sucesos.
  2. Seleccione el grupo de sucesos al que se suscribe el cliente de sucesos.
  3. Busque la propiedad **Serie de selector de sucesos**.
  4. Asegúrese de que el selector de sucesos especificado coincide con el contenido del suceso que está intentando recibir. Si no coincide, es posible que desee efectuar uno de los siguientes cambios:
    - Modifique el selector de sucesos para que se incluya en el grupo de sucesos.
    - Modifique los datos de sucesos para que el suceso coincida con el grupo de sucesos.
    - Modifique el cliente de sucesos de modo que se suscriba a un grupo de sucesos distinto que incluya el suceso.
- **Para iniciar la conexión JMS:**

En el cliente de sucesos, utilice el método `QueueConnection.start()` o el método `TopicConnection.start()` antes de intentar recibir sucesos.

## Los clientes no reciben los sucesos (NameNotFoundException)

*Mi aplicación de origen de sucesos está enviando sucesos satisfactoriamente al emisor, pero los sucesos no se publican en los clientes utilizando la interfaz JMS y el archivo de anotación cronológica indica una NameNotFoundException.*

### Causa

Este problema indica que la configuración del grupo de sucesos especifica uno o más recursos JMS que no existe.

### Solución

Para corregir este problema:

1. En la consola administrativa, pulse **Integración de servicios > Common Event Infrastructure > Servicio de sucesos > Servicios de sucesos > servicio\_sucesos > Grupos de sucesos > grupo\_sucesos**.

**Varios grupos de sucesos:** Un suceso puede pertenecer a más de un grupo de sucesos.

2. Compruebe las propiedades **Nombre JNDI de tema** y **Nombre JNDI de la fábrica de conexiones de tema**. Verifique que los recursos JMS especificados existan. Si es necesario, utilice la interfaz de configuración del proveedor JMS para crear los recursos necesarios.

## El grupo de sucesos con elementos de datos ampliados no contiene sucesos

*He definido un grupo de sucesos que especifica predicados de elementos de datos ampliados, pero las consultas realizadas al grupo de sucesos no devuelven los sucesos esperados.*

## Causa

Los datos de sucesos pueden ser elementos XML válidos pero no se ajustan a la especificación de Common Base Event. Esto puede provocar resultados inesperados sin ningún mensaje de error.

Suponga un suceso con el siguiente contenido:

```
<?xml version="1.0" encoding="ASCII"?>
<!-- Suceso que coincidirá con la expresión XPath CommonBaseEvent
      [@globalInstanceId] -->
<CommonBaseEvent
  xmlns:xsi="http://www.w3.org/TR/xmlschema-1/"
  xmlns="http://www.ibm.com/AC/commonbaseevent1_0_1"
  version="1.0.1"
  creationTime="2005-10-17T12:00:01Z"
  severity="10"
  priority="60"
  >
  <situation categoryName="RequestSituation">
    <situationType xsi:type="RequestSituation"
      reasoningScope="INTERNAL"
      successDisposition="Succeeded"
      situationQualifier="TEST"
    />
  </situation>
  <sourceComponentId
    component="component"
    subComponent="subcomponent"
    componentIdType="componentIdType"
    location="localhost"
    locationType="Hostname"
    componentType="sourceComponentType"
  />
  <extendedDataElement name="color" type="string">
    <values>red</values>
  </extendedDataElement>
</CommonBaseEvent>
```

Este suceso contiene un solo elemento de datos ampliados con un solo elemento hijo.

Ahora suponga una definición de grupo de sucesos configurada con la siguiente serie de selector de sucesos XPath:

```
CommonBaseEvent[extendedDataElements[@name='color' and @type='string'
and @values='red']]
```

Este selector de sucesos no puede coincidir con el suceso porque la definición XML del suceso contiene una palabra mal escrita. En los datos de sucesos, el elemento `extendedDataElements` se ha escrito incorrectamente como `extendedDataElement`. Debido a que es un XML con formato correcto, no causa un error, sino que se trata como un elemento `any`, en el que no se pueden realizar búsquedas.

## Solución

Asegúrese de que los datos XML para los sucesos enviados se ajustan a la especificación Common Base Event.

## Error al consultar un grupo de sucesos (mensaje CEIES0048E)

*Mi aplicación de consumidor de sucesos encuentra un error al intentar consultar sucesos desde un grupo de sucesos. El archivo de anotación cronológica indica una `EventGroupNotDefinedException` y muestra el mensaje CEIES0048E ("El grupo de sucesos no está definido en la lista de grupos de sucesos que utiliza la instancia del servidor de sucesos.")*

### Causa

Este problema indica que la aplicación del consumidor de sucesos ha realizado una consulta utilizando el bean `EventAccess`, pero el consumidor ha especificado un nombre de grupo de sucesos que no se corresponde con ningún grupo de sucesos existente.

### Solución

Para corregir este problema:

1. En la consola administrativa, pulse **Integración de servicios > Common Event Infrastructure > Servicio de sucesos > Servicios de sucesos > servicio\_sucesos > Grupos de sucesos**. La tabla muestra una lista de todos los grupos de sucesos definidos para el servicio de sucesos.
2. Asegúrese de que el origen de sucesos especifica un nombre de grupo de sucesos definido en los parámetros de la llamada al método de consulta.

---

## Problemas varios

Resolución de problemas varios del servidor de Common Event Infrastructure.

### La consulta de patrón del catálogo de sucesos falla en un sistema Windows

*Estoy intentando realizar una consulta del patrón para las definiciones de sucesos en un sistema Windows con el mandato `eventcatalog`. Por ejemplo: `eventcatalog -listdefinitions -name EVENT% -pattern`. No obtengo los resultados esperados.*

### Causa

El carácter de porcentaje (%) es un carácter reservado en la interfaz de línea de mandatos de Windows y no se pasa correctamente al mandato `eventcatalog`.

### Solución

En sistemas Windows, debe indicarse un carácter de escape con el carácter de porcentaje escribiendo %%:

```
eventcatalog -listdefinitions -name EVENT%% -pattern
```

---

## Capítulo 6. Mandatos de Common Event Infrastructure

Mandatos utilizados por Common Event Infrastructure para crear y eliminar bases de datos de sucesos y servicios

Mandatos para crear orígenes de datos de sucesos en bases de datos específicas:

- Multiplataforma DB2 — “Mandato configEventServiceDB2DB”
- DB2 iSeries — “Mandato configEventServiceDB2iSeriesDB” en la página 82
- DB2 z/OS — “Mandato configEventServiceDB2ZOSDB” en la página 84
- Derby — “Mandato configEventServiceDerbyDB” en la página 87
- Informix — “Mandato configEventServiceInformixDB” en la página 89
- Oracle — “Mandato configEventServiceOracleDB” en la página 91
- SQL Server — “Mandato configEventServiceSQLServerDB” en la página 94

Mandatos de Common Event Infrastructure genéricos utilizados por el servidor para manejar el servicio de sucesos:

- Desplegar la aplicación de servicio de sucesos — “Mandato deployEventService” en la página 96
- Desplegar el bean controlado por mensaje de aplicación de servicio de sucesos — “Mandato deployEventServiceMdb” en la página 98
- Crear el alias de autenticación utilizado por Common Event Infrastructure para la mensajería — “Mandato setEventServiceJmsAuthAlias” en la página 99
- Habilitación del servicio de sucesos al reiniciar el servidor — “Mandato enableEventService” en la página 100
- Inhabilitación del servicio de sucesos al reiniciar el servidor — “Mandato disableEventService” en la página 102
- Devolver el estado de servicio de sucesos — “Mandato showEventServiceStatus” en la página 103
- Eliminar la aplicación de servicio de sucesos — “Mandato removeEventService” en la página 104
- Eliminar el bean controlado por mensaje de aplicación de servicio de sucesos — “Mandato removeEventServiceMdb” en la página 105

Mandatos para eliminar orígenes de datos de sucesos en bases de datos específicas:

- Multiplataforma DB2 — “Mandato removeEventServiceDB2DB” en la página 106
- DB2 iSeries — “Mandato removeEventServiceDB2iSeriesDB” en la página 108
- DB2 z/OS — “Mandato removeEventServiceDB2ZOSDB” en la página 109
- Derby — “Mandato removeEventServiceDerbyDB” en la página 110
- Informix — “Mandato removeEventServiceInformixDB” en la página 112
- Oracle — “Mandato removeEventServiceOracleDB” en la página 113
- SQL Server — “Mandato removeEventServiceSQLServerDB” en la página 114

---

### Mandato configEventServiceDB2DB

Utilice el mandato configEventServiceDB2DB para configurar Common Event Infrastructure utilizando la base de datos DB2.

## Finalidad

El mandato `configEventServiceDB2DB` es un mandato administrativo de Common Event Infrastructure que está disponible para el objeto `AdminTask`. Utilice este mandato para crear en un servidor o un clúster la base de datos del servicio de sucesos y los orígenes de datos para DB2. Para obtener más información sobre el objeto `AdminTask`, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

## Parámetros

### - `createDB`

El mandato genera los scripts de base de datos DDL y crea la base de datos cuando este parámetro se establece en `true`. El mandato sólo genera los scripts de base de datos DDL cuando este parámetro se establece en `false`. Para crear la base de datos, el servidor actual ya debe estar configurado para ejecutar los mandatos de base de datos. Si no se especifica, el valor por omisión es `false`.

### - `overrideDataSource`

Directorio opcional de salida de scripts de base de datos. Cuando se especifica este parámetro, el mandato genera los scripts de base de datos del servicio de sucesos en el directorio especificado. Si el directorio especificado no contiene una vía de acceso completa, el mandato crea el directorio especificado en *raíz\_perfil/bin*. Si este parámetro no se especifica, el directorio de salida de scripts de base de datos por omisión es *raíz\_perfil/databases/event/nodo/servidor/dbscripts/dbtype*.

### - `nodeName`

El nombre del nodo que contiene el servidor donde debe crearse el origen de datos del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro `serverName`. Este parámetro no se debe especificar si se especifica el parámetro `clusterName`.

### - `serverName`

El nombre del servidor en el que se debe crear el origen de datos del servicio de sucesos. Si este parámetro se especifica sin el parámetro `nodeName`, el mandato utilizará el nombre de nodo del perfil actual de WebSphere. Este parámetro no se debe especificar si se especifica el parámetro `clusterName`.

### - `clusterName`

El nombre del clúster en el que se debe crear el origen de datos del servicio de sucesos. Si se especifica este parámetro, no deben especificarse los parámetros `serverName` y `nodeName`. Este parámetro no se debe especificar si se especifican los parámetros `serverName` y `nodeName`.

### - `jdbcClassPath`

La vía de acceso del controlador JDBC. Especifique sólo la vía de acceso del archivo del controlador; no incluya el nombre de archivo en la vía de acceso. Este parámetro es obligatorio.

### - `dbNodeName`

El nombre del nodo DB2 (debe tener como máximo 8 caracteres). Este nodo ya se debe haber catalogado y configurado para comunicarse con el servidor DB2. Este parámetro se debe haber establecido si el servidor actual se ha configurado como cliente de DB2 y el parámetro `createDB` se establece en `true`.

### - `dbHostName`

Nombre de sistema principal del servidor en el que se ha instalado el servidor de base de datos. Este parámetro es obligatorio.

- **dbPort**  
Puerto de instancia de DB2. Si no se especifica, el valor por omisión es 50.000.
- **dbName**  
Nombre de la base de datos que se creará. Si no se especifica, el valor por omisión es event.
- **dbUser**  
ID de usuario de DB2 que tiene privilegios para crear y eliminar las bases de datos. Si no se especifica, el valor por omisión es db2inst1.
- **dbPassword**  
Contraseña de DB2. Este parámetro es obligatorio.
- **outputScriptDir**  
Directorio opcional de salida de scripts de base de datos. Cuando se especifica este parámetro, el mandato genera los scripts de base de datos del servicio de sucesos en el directorio especificado. Si el directorio especificado no contiene una vía de acceso completa, el mandato crea el directorio especificado en *raíz\_perfil/bin*. Si este parámetro no se especifica, el directorio de salida de scripts de base de datos por omisión es *raíz\_perfil/databases/event/nodo/servidor/dbscripts/db2*.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:
 

```
$AdminTask configEventServiceDB2DB {-createDB true
  -overrideDataSource true
  -nodeName nodename
  -serverName servername
  -jdbcClassPath c:\sqllib\java
  -dbUser db2inst1
  -dbPassword dbpassword
  -dbHostName host_name
  -dbPort 50000 }
```
- Utilizando serie Jython:
 

```
AdminTask.configEventServiceDB2DB(['-createDB true
  -overrideDataSource true
  -nodeName nodename
  -serverName servername
  -jdbcClassPath c:\sqllib\java
  -dbUser db2inst1
  -dbPassword dbpassword
  -dbHostName host_name
  -dbPort 50000 '])
```
- Utilizando lista Jython:
 

```
AdminTask.configEventServiceDB2DB(['-createDB', 'true',
  '-overrideDataSource', 'true',
  '-nodeName', 'nodename',
  '-serverName', 'servername',
  '-jdbcClassPath', 'c:\sqllib\java',
  '-dbUser', 'db2inst1',
  '-dbPassword', 'dbpassword ',
  '-dbHostName', 'host_name',
  '-dbPort', '50000 '])
```

### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:
 

```
$AdminTask configEventServiceDB2DB -interactive
```

- Utilizando serie Jython:

**Nota:** **Vista** **Windows 7** El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

```
AdminTask.configEventServiceDB2DB(['-interactive'])
```

- Utilizando lista Jython:

```
AdminTask.configEventServiceDB2DB(['-interactive'])
```

---

## Mandato configEventServiceDB2iSeriesDB

Utilice el mandato configEventServiceDB2iSeriesDB para configurar Common Event Infrastructure utilizando una base de datos DB2 para iSeries.

### Finalidad

El mandato configEventServiceDB2iSeriesDB es un mandato administrativo de Common Event Infrastructure que está disponible para el objeto AdminTask. Utilícelo para generar los scripts de base de datos DDL para su utilización en servidores de bases de datos remotos, cree la base de datos del servicio de sucesos para DB2 iSeries en un servidor local y crea orígenes de datos en un servidor o clúster. Para obtener más información sobre el objeto AdminTask, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** **Vista** **Windows 7** El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

### Parámetros

#### - createDB

El mandato genera los scripts de base de datos DDL y crea la base de datos cuando este parámetro se establece en true. El mandato sólo genera los scripts de base de datos DDL cuando este parámetro se establece en false. Para crear la base de datos, el servidor actual ya debe estar configurado para ejecutar los mandatos de base de datos. Si no se especifica, el valor por omisión es false.

#### - overrideDataSource

Cuando este parámetro se establece en true, el mandato elimina todos los orígenes de datos del servicio de sucesos existentes en el ámbito especificado antes de crear uno nuevo. Cuando este parámetro se establece en false, el mandato no crea un origen de datos del servicio de sucesos en el ámbito especificado si hay otro origen de datos del servicio de sucesos en el mismo ámbito. Si no se especifica, el valor por omisión es false.

#### - nodeName

El nombre del nodo que contiene el servidor donde debe crearse el origen de datos del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

#### - serverName

El nombre del servidor en el que se debe crear el origen de datos del servicio de sucesos. Si este parámetro se especifica sin el parámetro **nodeName**, el mandato utilizará el nombre de nodo del perfil actual de WebSphere. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.



- **clusterName**  
El nombre del clúster en el que se debe crear el origen de datos del servicio de sucesos. Si se especifica este parámetro, no deben especificarse los parámetros **serverName** y **nodeName**. Este parámetro no se debe especificar si se especifican los parámetros **serverName** y **nodeName**.
- **toolboxJdbcClassPath**  
La vía de acceso al controlador DB2 JDBC de IBM Toolbox para Java. Especifique sólo la vía de acceso del archivo del controlador; no incluya el nombre del archivo. Debe especificar este parámetro o el parámetro **jdbcClassPath**.  
  
**Nota:** Debe especificar la vía de acceso de **toolboxJdbcClassPath** si crea la base de datos en un servidor iSeries.
- **nativeJdbcClassPath**  
La vía de acceso al controlador JDBC nativo de DB2 para iSeries. Especifique sólo la vía de acceso del archivo del controlador; no incluya el nombre de archivo en la vía de acceso. Debe especificar este parámetro o el parámetro **toolboxJdbcClassPath**.  
  
**Nota:** Debe especificar **nativeJdbcClassPath** si utiliza un servidor iSeries para crear la base de datos en un servidor no iSeries. Especifique el controlador de DB2 Universal si crea la base de datos en DB2 Universal Database for multiplatforms o DB2 Universal Database para z/OS. Especifique un controlador de tipo 4 si crea la base de datos en Informix, Oracle o Microsoft SQL Server.
- **dbHostName**  
El nombre de sistema principal del servidor en el que se ha instalado el servidor de base de datos DB2 para iSeries. Este parámetro es necesario si utiliza el controlador JDBC de IBM Toolbox para Java DB2.
- **dbName**  
El nombre de base de datos de DB2 para iSeries. Si no se especifica, el valor por omisión es \*LOCAL.
- **collection**  
Colección SQL de bibliotecas DB2 para iSeries. La longitud máxima para el nombre de la colección es de 10 caracteres. Si no se especifica, el valor por omisión es una serie vacía.
- **dbUser**  
ID de usuario de DB2 que tiene privilegios para crear y eliminar las bases de datos. Este parámetro es obligatorio.
- **dbPassword**  
Contraseña del ID de usuario de la base de datos. Este parámetro es obligatorio.
- **outputScriptDir**  
Directorio opcional de salida de scripts de base de datos. Cuando se especifica este parámetro, el mandato genera los scripts de base de datos del servicio de sucesos en el directorio especificado. Si el directorio especificado no contiene una vía de acceso completa, el mandato crea el directorio especificado en *raíz\_perfil/bin*. Si este parámetro no se especifica, el directorio de salida de scripts de base de datos por omisión es *raíz\_perfil/databases/event/nodo/servidor/dbscripts/db2series*.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:

```
$AdminTask configEventServiceDB2iSeriesDB {createDB true
  -overrideDataSource true
  -nodeName nodename
  -serverName servername
  -dbUser db2user
  -dbPassword dbpassword
  -nativeJdbcClassPath /myDB2ClassPath
  -collection event}
```

- Utilizando serie Jython:

```
AdminTask.configEventServiceDB2iSeriesDB(['-createDB true
  -overrideDataSource true
  -nodeName nodename
  -serverName servername
  -nativeJdbcClassPath /myDB2ClassPath
  -collection event'])
```

- Utilizando lista Jython:

```
AdminTask.configEventServiceDB2iSeriesDB(['-createDB', 'true',
  '-overrideDataSource', 'true',
  '-nodeName', 'nodename',
  '-serverName', 'servername',
  '-nativeJdbcClassPath', '/myDB2ClassPath',
  '-collection', 'event'])
```

### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:

```
$AdminTask configEventServiceDB2iSeriesDB -interactive
```

- Utilizando serie Jython:

```
AdminTask.configEventServiceDB2iSeriesDB(['-interactive'])
```

- Utilizando lista Jython:

```
AdminTask.configEventServiceDB2iSeriesDB(['-interactive'])
```

---

## Mandato configEventServiceDB2ZOSDB

Utilice el mandato configEventServiceDB2ZOSDB para configurar Common Event Infrastructure utilizando una base de datos DB2 para z/OS.

### Finalidad

El mandato configEventServiceDB2ZOSDB es un mandato administrativo de Common Event Infrastructure que está disponible para el objeto AdminTask. Utilice este mandato para crear en un servidor o un clúster la base de datos del servicio de sucesos y los orígenes de datos para DB2 z/OS. Para obtener más información sobre el objeto AdminTask, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** **Vista** **Windows 7** El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

## Parámetros

### - createDB

El mandato genera los scripts de base de datos DDL y crea la base de datos cuando este parámetro se establece en true. El mandato sólo genera los scripts de base de datos DDL cuando este parámetro se establece en false. Para crear la base de datos, el servidor actual ya debe estar configurado para ejecutar los mandatos de base de datos. Si no se especifica, el valor por omisión es false.

### - overrideDataSource

Cuando este parámetro se establece en true, el mandato elimina todos los orígenes de datos del servicio de sucesos existentes en el ámbito especificado antes de crear uno nuevo. Cuando este parámetro se establece en false, el mandato no crea un origen de datos del servicio de sucesos en el ámbito especificado si hay otro origen de datos del servicio de sucesos en el mismo ámbito. Si no se especifica, el valor por omisión es false.

### - nodeName

El nombre del nodo que contiene el servidor donde debe crearse el origen de datos del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

### - serverName

El nombre del servidor en el que se debe crear el origen de datos del servicio de sucesos. Si este parámetro se especifica sin el parámetro **nodeName**, el mandato utilizará el nombre de nodo del perfil actual de WebSphere. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

### - clusterName

El nombre del clúster en el que se debe crear el origen de datos del servicio de sucesos. Si se especifica este parámetro, no deben especificarse los parámetros **serverName** y **nodeName**. Este parámetro no se debe especificar si se especifican los parámetros **serverName** y **nodeName**.

### - jdbcClassPath

La vía de acceso del controlador JDBC. Especifique sólo la vía de acceso del archivo del controlador; no incluya el nombre de archivo en la vía de acceso. Este parámetro es obligatorio.

### - dbHostName

Nombre de sistema principal del servidor en el que se ha instalado la base de datos. Este parámetro es obligatorio.

### - dbPort

Puerto de instancia de DB2 para z/OS. Si no se especifica, el valor por omisión es 5027.

### - dbName

Nombre de base de datos de DB2. En el cliente DB2, es el nombre de la base de datos catalogada. En el servidor z/OS nativo, es el nombre del subsistema de base de datos. El valor por omisión es event, si no se especifica ningún valor.

### - dbDiskSizeInMB

Especifique el tamaño del disco en MB para la base de datos del servicio de sucesos. Este valor debe ser como mínimo de 10 MB. Si no se especifica, el valor por omisión es 100 MB.

- **dbUser**  
ID de usuario de DB2 que tiene privilegios para crear y eliminar las bases de datos. Este parámetro es obligatorio.
- **dbPassword**  
Contraseña del ID de usuario de la base de datos. Este parámetro es obligatorio.
- **storageGroup**  
El grupo de almacenamiento de la base de datos de sucesos y la base de datos del catálogo de sucesos. El grupo de almacenamiento ya debe haberse creado y estar activo.
- **bufferPool4K**  
Nombre de la agrupación de almacenamiento intermedio de 4K. Esta agrupación de almacenamiento intermedio debe estar activa para poder activar los scripts DDL de base de datos.
- **bufferPool8K**  
Nombre de la agrupación de almacenamiento intermedio de 8K. Esta agrupación de almacenamiento intermedio debe estar activa para poder activar los scripts DDL de base de datos.
- **bufferPool16K**  
Nombre de la agrupación de almacenamiento intermedio de 16K. Esta agrupación de almacenamiento intermedio debe estar activa para poder activar los scripts DDL de base de datos.
- **outputScriptDir**  
Directorio opcional de salida de scripts de base de datos. Cuando se especifica este parámetro, el mandato genera los scripts de base de datos del servicio de sucesos en el directorio especificado. Si el directorio especificado no contiene una vía de acceso completa, el mandato crea el directorio especificado en *raíz\_perfil/bin*. Si este parámetro no se especifica, el directorio de salida de scripts de base de datos por omisión es *raíz\_perfil/databases/event/nodo/servidor/dbscripts/db2zos*.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:
 

```
$AdminTask configEventServiceDB2ZOSDB {-createDB true -overrideDataSource true
-nodeName nodename -serverName servername -jdbcClassPath c:\sqllib\java
-dbUser db2user -dbPassword dbpassword -dbHostName host_name -dbPort 5027
-storageGroup sysdeflt -bufferPool4K BP9 -bufferPool8K BP8K9
-bufferPool16K BP16K9}
```
- Utilizando serie Jython:
 

```
AdminTask.configEventServiceDB2ZOSDB(['-createDB true -overrideDataSource true
-nodeName nodename -serverName servername -jdbcClassPath c:\sqllib\java
-dbUser db2user -dbPassword dbpassword -dbHostName host_name -dbPort 5027
-storageGroup sysdeflt -bufferPool4K BP9 -bufferPool8K BP8K9
-bufferPool16K BP16K9'])
```
- Utilizando lista Jython:
 

```
AdminTask.configEventServiceDB2ZOSDB(['-createDB', 'true', '-overrideDataSource',
'true', '-nodeName', 'nodename', '-serverName', 'servername', '-jdbcClassPath',
'c:\sqllib\java', '-dbUser', 'db2user', '-dbPassword', 'dbpassword',
'-dbHostName', 'host_name', '-dbPort', '5027', '-storageGroup', 'sysdeflt',
'-bufferPool4K', 'BP9', '-bufferPool8K', 'BP8K9', '-bufferPool16K', 'BP16K9'])
```

### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:  
`$AdminTask configEventServiceDB2ZOSDB -interactive`
- Utilizando serie Jython:  
`AdminTask.configEventServiceDB2ZOSDB(['-interactive'])`
- Utilizando lista Jython:  
`AdminTask.configEventServiceDB2ZOSDB ['-interactive'])`

---

## Mandato configEventServiceDerbyDB

Utilice el mandato configEventServiceDerbyDB para configurar Common Event Infrastructure utilizando una base de datos Derby.

### Finalidad

El mandato configEventServiceDerbyDB es un mandato administrativo de Common Event Infrastructure que está disponible para el objeto AdminTask. Utilice este mandato para crear en un servidor o un clúster la base de datos del servicio de sucesos y los orígenes de datos para Derby. Para obtener más información sobre el objeto AdminTask, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** **Vista** **Windows 7** El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

### Parámetros

#### - createDB

El mandato genera los scripts de base de datos DDL y crea la base de datos cuando este parámetro se establece en true. El mandato sólo genera los scripts de base de datos DDL cuando este parámetro se establece en false. Para crear la base de datos, el servidor actual ya debe estar configurado para ejecutar los mandatos de base de datos. Si no se especifica, el valor por omisión es false.

#### - overrideDataSource

Cuando este parámetro se establece en true, el mandato elimina todos los orígenes de datos del servicio de sucesos existentes en el ámbito especificado antes de crear uno nuevo. Cuando este parámetro se establece en false, el mandato no crea un origen de datos del servicio de sucesos en el ámbito especificado si hay otro origen de datos del servicio de sucesos en el mismo ámbito. Si no se especifica, el valor por omisión es false.

#### - nodeName

El nombre del nodo que contiene el servidor donde debe crearse el origen de datos del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

#### - serverName

El nombre del servidor en el que se debe crear el origen de datos del servicio de sucesos. Si este parámetro se especifica sin el parámetro **nodeName**, el mandato utilizará el nombre de nodo del perfil actual de WebSphere. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

#### - clusterName

El nombre del clúster en el que se debe crear el origen de datos del servicio de sucesos. Si se especifica este parámetro, no deben especificarse los parámetros

**serverName** y **nodeName**. Este parámetro no se debe especificar si se especifican los parámetros **serverName** y **nodeName**.

- **dbHostName**

Nombre de sistema principal de la base de datos Derby de red. Para crear el origen de datos de red Derby, especifique este parámetro y el parámetro **dbPort**. Para crear el origen de datos local Derby, no especifique este parámetro ni el parámetro **dbPort**.

- **dbPort**

Número de puerto de la base de datos Derby de red. Para crear el origen de datos de red Derby, especifique este parámetro y el parámetro **dbHostName**. Para crear el origen de datos local Derby, no especifique este parámetro ni el parámetro **dbHostName**.

- **dbName**

Nombre de la base de datos que se creará. Si no se especifica, el valor por omisión es event.

- **dbUser**

ID de usuario utilizado por el origen de datos para la autenticación de base de datos Derby. Este parámetro es opcional cuando está habilitada la seguridad del dominio de WebSphere. Si especifica este parámetro, también debe especificar el parámetro **dbPassword**. Este parámetro es obligatorio cuando está habilitada la seguridad de domino de WebSphere.

- **dbPassword**

Contraseña utilizada por el origen de datos para la autenticación de base de datos Derby. Este parámetro es opcional cuando está habilitada la seguridad del dominio de WebSphere. Si especifica este parámetro, también debe especificar el parámetro **dbUser**. Este parámetro es obligatorio cuando está habilitada la seguridad de domino de WebSphere.

- **outputScriptDir**

Directorio opcional de salida de scripts de base de datos. Cuando se especifica este parámetro, el mandato genera los scripts de base de datos del servicio de sucesos en el directorio especificado. Si el directorio especificado no contiene una vía de acceso completa, el mandato crea el directorio especificado en *raíz\_perfil/bin*. Si este parámetro no se especifica, el directorio de salida de scripts de base de datos por omisión es *raíz\_perfil/databases/event/nodo/servidor/dbscripts/derby*.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:

```
$AdminTask configEventServiceDerbyDB {-createDB true -overrideDataSource true  
-nodeName nodename -serverName servername}
```

- Utilizando serie Jython:

```
AdminTask.configEventServiceDerbyDB(['-createDB true -overrideDataSource true  
-nodeName nodename -serverName servername'])
```

- Utilizando lista Jython:

```
AdminTask.configEventServiceDerbyDB(['-createDB', 'true', '-overrideDataSource',  
'true', '-nodeName', 'nodename', '-serverName', 'servername'])
```

### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:

```
$AdminTask configEventServiceDerbyDB -interactive
```

- Utilizando serie Jython:  
`AdminTask.configEventServiceDerbyDB(['-interactive'])`
- Utilizando lista Jython:  
`AdminTask.configEventServiceDerbyDB(['-interactive'])`

---

## Mandato `configEventServiceInformixDB`

Utilice el mandato `configEventServiceInformixDB` para configurar Common Event Infrastructure utilizando la base de datos Informix.

### Finalidad

El mandato `configEventServiceInformixDB` es un mandato administrativo de Common Event Infrastructure que está disponible para el objeto `AdminTask`. Utilice este mandato para crear los orígenes de datos y base de datos de servicio de sucesos para Informix en un servidor o clúster. Para obtener más información sobre el objeto `AdminTask`, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** Vista Windows 7 El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

### Parámetros

#### - `createDB`

El mandato genera los scripts de base de datos DDL y crea la base de datos cuando este parámetro se establece en `true`. El mandato sólo genera los scripts de base de datos DDL cuando este parámetro se establece en `false`. Para crear la base de datos, el servidor actual ya debe estar configurado para ejecutar los mandatos de base de datos. Si no se especifica, el valor por omisión es `false`.

#### - `overrideDataSource`

Directorio opcional de salida de scripts de base de datos. Cuando se especifica este parámetro, el mandato genera los scripts de base de datos del servicio de sucesos en el directorio especificado. Si el directorio especificado no contiene una vía de acceso completa, el mandato crea el directorio especificado en `raíz_perfil/bin`. Si este parámetro no se especifica, el directorio de salida de scripts de base de datos por omisión es `raíz_perfil/databases/event/nodo/servidor/dbscripts/dbtype`.

#### - `nodeName`

El nombre del nodo que contiene el servidor donde debe crearse el origen de datos del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro `serverName`. Este parámetro no se debe especificar si se especifica el parámetro `clusterName`.

#### - `serverName`

El nombre del servidor en el que se debe crear el origen de datos del servicio de sucesos. Si este parámetro se especifica sin el parámetro `nodeName`, el mandato utilizará el nombre de nodo del perfil actual de WebSphere. Este parámetro no se debe especificar si se especifica el parámetro `clusterName`.

#### - `clusterName`

El nombre del clúster en el que se debe crear el origen de datos del servicio de sucesos. Si se especifica este parámetro, no deben especificarse los parámetros `serverName` y `nodeName`. Este parámetro no se debe especificar si se especifican los parámetros `serverName` y `nodeName`.

- **jdbcClassPath**  
La vía de acceso del controlador JDBC. Especifique sólo la vía de acceso del archivo del controlador; no incluya el nombre de archivo en la vía de acceso. Este parámetro es obligatorio.
- **dbInformixDir**  
El directorio en el que se ha instalado la base de datos Informix. Este parámetro debe especificarse cuando el parámetro **createDB** se establece en true. Este parámetro es obligatorio.
- **dbHostName**  
Nombre de sistema principal del servidor en el que se ha instalado la base de datos. Este parámetro es obligatorio.
- **dbServerName**  
Nombre de servidor Informix (por ejemplo, ol\_servername). Este parámetro es obligatorio.
- **dbPort**  
Puerto de instancia de Informix. Si no se especifica, el valor por omisión es 1526.
- **dbName**  
Nombre de la base de datos que se creará. Si no se especifica, el valor por omisión es event.
- **dbUser**  
ID de usuario de esquema de base de datos de Informix que será propietario de las tablas de base de datos del servicio de sucesos. El origen de datos WebSphere utiliza este ID de usuario para autenticar la conexión de base de datos de Informix. Este parámetro es obligatorio.
- **dbPassword**  
Contraseña del ID de usuario del esquema que es propietario de las tablas Informix del servicio de sucesos. El origen de datos de WebSphere utiliza esta contraseña para autenticar la conexión de base de datos de Informix. Este parámetro es obligatorio.
- **ceiInstancePrefix**  
El mandato utiliza el nombre de instancia del servicio de sucesos para agrupar los archivos de base de datos en un directorio con nombres exclusivos. Si no se especifica, el valor por omisión es ceiinst1.
- **outputScriptDir**  
Directorio opcional de salida de scripts de base de datos. Cuando se especifica este parámetro, el mandato genera los scripts de base de datos del servicio de sucesos en el directorio especificado. Si el directorio especificado no contiene una vía de acceso completa, el mandato crea el directorio especificado en *raíz\_perfil/bin*. Si este parámetro no se especifica, el directorio de salida de scripts de base de datos por omisión es *raíz\_perfil/databases/event/nodo/servidor/dbscripts/informix*.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:



```
$AdminTask configEventServiceInformixDB {-createDB true
-overriderDataSource true -nodeName nodename -serverName servername
-jdbcClassPath "c:\archivos de programa\ibm\informix\jdbc\lib"
-dbInformixDir "c:\archivos de programa\ibm\informix" -dbUser informix
-dbPassword dbpassword -dbHostName host_name -dbPort 1526
-dbServerName ol_server }
```

- Utilizando serie Jython:

```
AdminTask.configEventServiceInformixDB(['-createDB true
-overriderDataSource true
-nodeName nodename
-serverName servername
-jdbcClassPath "c:\archivos de programa\ibm\informix\jdbc\lib"
-dbInformixDir "c:\archivos de programa\ibm\informix"
-dbUser informix
-dbPassword dbpassword
-dbHostName host_name
-dbPort 1526
-dbServerName ol_server'])
```

- Utilizando lista Jython:

```
AdminTask.configEventServiceInformixDB(['-createDB', 'true',
'-overriderDataSource', 'true', '-nodeName', 'nodename',
'-serverName', 'servername', '-jdbcClassPath',
'c:\archivos de programa\ibm\informix\jdbc\lib', '-dbInformixDir',
'c:\archivos de programa\ibm\informix', '-dbUser', 'informix ',
'-dbPassword', 'dbpassword', '-dbHostName', 'host_name',
'-dbPort', '1526', '-dbServerName', 'ol_server'])
```

#### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:

```
$AdminTask configEventServiceInformixDB -interactive
```

- Utilizando serie Jython:

```
AdminTask.configEventServiceInformixDB(['-interactive'])
```

- Utilizando lista Jython:

```
AdminTask.configEventServiceInformixDB(['-interactive'])
```

---

## Mandato configEventServiceOracleDB

Utilice el mandato configEventServiceOraclesDB para configurar Common Event Infrastructure utilizando una base de datos Oracle.

### Finalidad

El mandato configEventServiceOracleDB es un mandato administrativo de Common Event Infrastructure que está disponible para el objeto AdminTask. Utilice este mandato para crear en un servidor o un clúster las tablas del servicio de sucesos y los orígenes de datos para Oracle. El mandato no crea la base de datos; el SID de Oracle ya debe existir. Para obtener más información sobre el objeto AdminTask, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** Vista Windows 7 El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

### Parámetros

#### - createDB

El mandato genera los scripts de base de datos DDL y crea la base de datos

cuando este parámetro se establece en true. El mandato sólo genera los scripts de base de datos DDL cuando este parámetro se establece en false. Para crear la base de datos, el servidor actual ya debe estar configurado para ejecutar los mandatos de base de datos. Si no se especifica, el valor por omisión es false.

- **overrideDataSource**

Directorio opcional de salida de scripts de base de datos. Cuando se especifica este parámetro, el mandato genera los scripts de base de datos del servicio de sucesos en el directorio especificado. Si el directorio especificado no contiene una vía de acceso completa, el mandato crea el directorio especificado en *raíz\_perfil/bin*. Si este parámetro no se especifica, el directorio de salida de scripts de base de datos por omisión es *raíz\_perfil/databases/event/nodo/servidor/dbscripts/dbtype*.

- **nodeName**

El nombre del nodo que contiene el servidor donde debe crearse el origen de datos del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

- **serverName**

El nombre del servidor en el que se debe crear el origen de datos del servicio de sucesos. Si este parámetro se especifica sin el parámetro **nodeName**, el mandato utilizará el nombre de nodo del perfil actual de WebSphere. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

- **clusterName**

El nombre del clúster en el que se debe crear el origen de datos del servicio de sucesos. Si se especifica este parámetro, no deben especificarse los parámetros **serverName** y **nodeName**. Este parámetro no se debe especificar si se especifican los parámetros **serverName** y **nodeName**.

- **jdbcClassPath**

La vía de acceso del controlador JDBC. Especifique sólo la vía de acceso del archivo del controlador; no incluya el nombre de archivo en la vía de acceso. Este parámetro es obligatorio.

- **oracleHome**

El directorio ORACLE\_HOME. Este parámetro debe establecerse cuando el parámetro **createDB** se establece en true.

- **dbHostName**

El nombre de sistema principal del servidor en el que se ha instalado la base de datos Oracle. Si no se especifica, el valor por omisión es localhost.

- **dbPort**

Puerto de instancia de Oracle. Si no se especifica, el valor por omisión es 1521.

- **dbName**

El identificador del sistema (SID) de Oracle. El SID ya debe existir y debe estar disponible para el mandato del servicio de sucesos para crear las tablas y rellenar las tablas con datos. Si no se especifica, el valor por omisión es orcl.

- **dbUser**

ID de usuario de esquema de Oracle que será propietario de las tablas de Oracle del servicio de sucesos. El ID de usuario se creará durante la creación de base de datos; el origen de datos de WebSphere utiliza este ID de usuario para autenticar la conexión de la base de datos Oracle. Si no se especifica, el valor por omisión es ceouser.

- **dbPassword**

Contraseña del ID de usuario del esquema. La contraseña se creará durante la creación de base de datos; el origen de datos de WebSphere utiliza esta contraseña para autenticar la conexión de la base de datos Oracle. Este parámetro es obligatorio.

- **sysUser**

ID de usuario sys de Oracle. Debe ser un usuario que tiene privilegios SYSDBA. Si no se especifica, el valor por omisión es sys.

- **sysPassword**

Contraseña para el usuario especificado por el parámetro **sysUser**. Si no se especifica, el valor por omisión es una serie vacía.

- **ceiInstancePrefix**

El mandato utiliza el nombre de instancia del servicio de sucesos para agrupar los archivos de base de datos en un directorio con nombres exclusivos. Si no se especifica, el valor por omisión es `ceiinst1`.

- **outputScriptDir**

Directorio opcional de salida de scripts de base de datos. Cuando se especifica este parámetro, el mandato genera los scripts de base de datos del servicio de sucesos en el directorio especificado. Si el directorio especificado no contiene una vía de acceso completa, el mandato crea el directorio especificado en *raíz\_perfil/bin*. Si este parámetro no se especifica, el directorio de salida de scripts de base de datos por omisión es *raíz\_perfil/databases/event/nodo/servidor/dbscripts/oracle*.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:

```
$AdminTask configEventServiceOracleDB {-createDB true
  -overrideDataSource true
  -nodeName nodename
  -serverName servername
  -jdbcClassPath c:\oracle\ora92\jdbc\lib
  -oracleHome c:\oracle\ora92
  -dbUser ceiuser
  -dbPassword ceipassword
  -dbHostName host_name
  -dbPort 1521
  -sysUser sys
  -sysPassword syspassword}
```

- Utilizando serie Jython:

```
AdminTask.configEventServiceOracleDB( ['-createDB true
  -overrideDataSource true
  -nodeName nodename
  -serverName servername
  -jdbcClassPath c:\oracle\ora92\jdbc\lib
  -oracleHome c:\oracle\ora92
  -dbUser ceiuser
  -dbPassword ceipassword
  -dbHostName host_name
  -dbPort 1521
  -sysUser sys
  -sysPassword syspassword]' )
```

- Utilizando lista Jython:

```
AdminTask.configEventServiceOracleDB(['-createDB', 'true',
  '-overrideDataSource', 'true',
  '-nodeName', 'nodename',
```

```

-serverName', 'servername',
-jdbcClassPath', 'c:\oracle\ora92\jdbc\lib',
-oracleHome', 'c:\oracle\ora92',
-dbUser', 'ceiuser',
-dbPassword', 'ceipassword',
-dbHostName', 'host_name',
-dbPort', '1521',
-sysUser', 'sys',
-sysPassword', 'syspassword'])

```

#### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:  
\$AdminTask configEventServiceOracleDB -interactive
- Utilizando serie Jython:  
AdminTask.configEventServiceOracleDB(['-interactive'])
- Utilizando lista Jython:  
AdminTask.configEventServiceOracleDB(['-interactive'])

---

## Mandato configEventServiceSQLServerDB

Utilice el mandato configEventServiceSQLServerDB para configurar Common Event Infrastructure utilizando una base de datos SQL Server.

### Finalidad

El mandato configEventServiceSQLServerDB es un mandato administrativo de Common Event Infrastructure que está disponible para el objeto AdminTask. Utilice este mandato para crear en un servidor o un clúster la base de datos del servicio de sucesos y los orígenes de datos para SQL Server. Para obtener más información sobre el objeto AdminTask, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** **Vista** **Windows 7** El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

### Parámetros

#### - createDB

El mandato genera los scripts de base de datos DDL y crea la base de datos cuando este parámetro se establece en true. El mandato sólo genera los scripts de base de datos DDL cuando este parámetro se establece en false. Para crear la base de datos, el servidor actual ya debe estar configurado para ejecutar los mandatos de base de datos. Si no se especifica, el valor por omisión es false.

#### - overrideDataSource

Directorio opcional de salida de scripts de base de datos. Cuando se especifica este parámetro, el mandato genera los scripts de base de datos del servicio de sucesos en el directorio especificado. Si el directorio especificado no contiene una vía de acceso completa, el mandato crea el directorio especificado en *raíz\_perfil/bin*. Si este parámetro no se especifica, el directorio de salida de scripts de base de datos por omisión es *raíz\_perfil/databases/event/nodo/servidor/dbscripts/dbtype*.

#### - nodeName

El nombre del nodo que contiene el servidor donde debe crearse el origen de datos del servicio de sucesos. Si se especifica este parámetro, deberá

especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

- **serverName**

El nombre del servidor en el que se debe crear el origen de datos del servicio de sucesos. Si este parámetro se especifica sin el parámetro **nodeName**, el mandato utilizará el nombre de nodo del perfil actual de WebSphere. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

- **clusterName**

El nombre del clúster en el que se debe crear el origen de datos del servicio de sucesos. Si se especifica este parámetro, no deben especificarse los parámetros **serverName** y **nodeName**. Este parámetro no se debe especificar si se especifican los parámetros **serverName** y **nodeName**.

- **dbServerName**

Nombre de servidor de la base de datos de SQL Server. Este parámetro debe establecerse cuando el parámetro **createDB** se establece en true.

- **dbHostName**

Nombre de sistema principal del servidor en el que se ha instalado la base de datos de SQL Server.

- **dbPort**

Puerto de SQL Server. Si no se especifica, el valor por omisión es 1433.

- **dbName**

Nombre de la base de datos que se creará. Si no se especifica, el valor por omisión es event.

- **dbUser**

ID de usuario de SQL Server que será propietario de las tablas del servicio de sucesos. Si no se especifica, el valor por omisión es ceouser.

- **dbPassword**

La contraseña del ID de usuario de SQL Server especificada por el parámetro **dbUser**. Este parámetro es obligatorio.

- **saUser**

ID de usuario que tiene privilegios para crear y eliminar bases de datos y usuarios. Este parámetro es necesario cuando el parámetro **createDB** se establece en true. Si no se especifica, el valor por omisión es sa.

- **saPassword**

La contraseña sa. Este parámetro no se debe especificar si el ID de usuario sa no tiene una contraseña.

- **ceiInstancePrefix**

El mandato utiliza el nombre de instancia del servicio de sucesos para agrupar los archivos de base de datos en un directorio con nombres exclusivos. Si no se especifica, el valor por omisión es ceiinst1.

- **outputScriptDir**

Directorio opcional de salida de scripts de base de datos. Cuando se especifica este parámetro, el mandato genera los scripts de base de datos del servicio de sucesos en el directorio especificado. Si el directorio especificado no contiene una vía de acceso completa, el mandato crea el directorio especificado en *raíz\_perfil/bin*. Si este parámetro no se especifica, el directorio de salida de scripts de base de datos por omisión es *raíz\_perfil/databases/event/nodo/servidor/dbscripts/sqlserver*.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:

```
$AdminTask configEventServiceSQLServerDB {-createDB true
  -overrideDataSource true
  -nodeName nodename
  -serverName servername
  -dbUser ceiuser
  -dbPassword ceipassword
  -dbServerName sqlservername
  -dbHostName host_name
  -dbPort 1433
  -saUser sa
  -saPassword sapassword}
```
- Utilizando serie Jython:

```
AdminTask.configEventServiceSQLServerDB(['-createDB true
  -overrideDataSource true
  -nodeName nodename
  -serverName servername
  -dbUser ceiuser
  -dbPassword ceipassword
  -dbServerName sqlservername
  -dbHostName host_name
  -dbPort 1433
  -saUser sa
  -saPassword sapassword'])
```
- Utilizando lista Jython:

```
AdminTask.configEventServiceSQLServerDB(['-createDB', 'true',
  '-overrideDataSource', 'true',
  '-nodeName', 'nodename',
  '-serverName', 'servername',
  '-dbUser', 'ceiuser',
  '-dbPassword', 'ceipassword',
  '-dbServerName', 'sqlservername',
  '-dbHostName', 'host_name',
  '-dbPort', '1433',
  '-saUser', 'sa',
  '-saPassword', 'sapassword'])
```

### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:

```
$AdminTask configEventServiceSQLServerDB -interactive
```
- Utilizando serie Jython:

```
AdminTask.configEventServiceSQLServerDB(['-interactive'])
```
- Utilizando lista Jython:

```
AdminTask.configEventServiceSQLServerDB(['-interactive'])
```

---

## Mandato deployEventService

Utilice el mandato deployEventService para desplegar la aplicación del servicio de sucesos en su servidor.

### Finalidad

El mandato deployEventService es un mandato administrativo de Common Event Infrastructure disponible en el objeto AdminTask. Utilice este mandato para desplegar y configurar el servicio de sucesos de un servidor o clúster. Para obtener

más información sobre el objeto AdminTask, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** **Vista** **Windows 7** El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

## Parámetros

### - nodeName

Nombre del nodo en el que se debe desplegar el servicio de sucesos. Si se especifica este parámetro, debe especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

### - serverName

Nombre del servidor en el que se debe desplegar el servicio de sucesos. Este parámetro se debe especificar si se especifica el parámetro **nodeName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

### - clusterName

Nombre del clúster en el que se debe desplegar el servicio de sucesos. Este parámetro no se debe especificar si se especifican los parámetros **nodeName** o **serverName**.

### - enable

Establezca este parámetro en true si desea que el servicio de sucesos se inicie después del siguiente reinicio del servidor. El valor por omisión es true.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:

```
$AdminTask deployEventService {-nodeName nodename  
-serverName servername}  
$AdminTask deployEventService {-clusterName clustername  
-enable false}
```

- Utilizando serie Jython:

```
AdminTask.deployEventService(['-nodeName nodename  
-serverName servername'])  
  
AdminTask.deployEventService(['-clusterName clustername  
-enable false'])
```

- Utilizando lista Jython:

```
AdminTask.deployEventService(['-nodeName', 'nodename',  
'-serverName', '-servername'])  
AdminTask.deployEventService(['-clusterName', 'clustername',  
'-enable', 'false'])
```

### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:

```
$AdminTask deployEventService {-interactive}
```

- Utilizando serie Jython:

```
AdminTask.deployEventService(['-interactive'])
```

- Utilizando lista Jython:

```
AdminTask.deployEventService(['-interactive'])
```

---

## Mandato deployEventServiceMdb

Utilice el mandato `deployEventServiceMdb` para desplegar el bean controlado por mensajes del servicio de sucesos en su servidor.

### Finalidad

El mandato `deployEventServiceMdb` es un mandato administrativo de Common Event Infrastructure disponible para el objeto `AdminTask`. Utilice este mandato para desplegar el MDB de servicio de sucesos en un servidor o clúster. Para obtener más información sobre el objeto `AdminTask`, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** **Vista** **Windows 7** El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

### Parámetros

- **nodeName**

Nombre del nodo en el que se debe desplegar el MDB del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

- **serverName**

Nombre del servidor en el que se debe desplegar el MDB del servicio de sucesos. Este parámetro se debe especificar si se especifica el parámetro **nodeName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

- **clusterName**

Nombre del clúster en el que se debe desplegar el MDB del servicio de sucesos. Este parámetro no se debe especificar si se especifican los parámetros **nodeName** y **serverName**.

- **applicationName**

Nombre de la aplicación MDB del servicio de sucesos que se debe desplegar en un servidor o un clúster.

- **listenerPort**

Nombre del puerto receptor en el que se debe publicar el MDB del servicio de sucesos. El puerto receptor ya debe haberse creado. Este parámetro no se debe especificar si se especifica el parámetro **activationSpec**.

- **activationSpec**

El nombre JNDI de la especificación de activación donde el MDB del servicio de sucesos debe publicar los sucesos. La especificación de activación ya se debe haber creado. Este parámetro no se debe especificar si se especifica el parámetro **listenerPort**.

- **qcfJndiName**

Nombre JNDI del objeto de fábrica de conexiones de cola JMS que debe utilizar el MDB del servicio de sucesos. Este parámetro se debe especificar si se especifica el parámetro **activationSpec**.



## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:

```
$AdminTask deployEventServiceMdb {-applicationName appname -nodeName nodename  
-serverName servername -listenerPort lpname}$AdminTask deployEventServiceMdb  
{-applicationName appname -clusterName clustername -activationSpec asjndiname  
-qcfJndiName qcfjndiname}
```

- Utilizando serie Jython:

```
AdminTask.deployEventServiceMdb(['-applicationName appname -nodeName nodename  
-serverName servername -listenerPort lpname'])AdminTask.deployEventServiceMdb  
(['-applicationName appname -clusterName clustername -activationSpec asjndiname  
-qcfJndiName qcfjndiname'])
```

- Utilizando lista Jython:

```
AdminTask.deployEventServiceMdb(['-applicationName', 'appname', '-nodeName',  
'nodename', '-serverName', '-servername', '-listenerPort', 'lpname'])  
AdminTask.deployEventServiceMdb(['-applicationName', 'appname',  
'-clusterName', 'clustername', '-activationSpec', 'asjndiname',  
'-qcfJndiName', 'qcfjndiname'])
```

### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:

```
$AdminTask deployEventServiceMdb {-interactive}
```

- Utilizando serie Jython:

```
AdminTask.deployEventServiceMdb(['-interactive'])
```

- Utilizando lista Jython:

```
AdminTask.deployEventServiceMdb(['-interactive'])
```

---

## Mandato setEventServiceJmsAuthAlias

Utilice el mandato setEventServiceJmsAuthAlias para establecer o actualizar el alias de autenticación JMS asociado al servicio de sucesos en su servidor.

### Finalidad

El mandato setEventServiceJmsAuthAlias es un mandato administrativo de Common Event Infrastructure disponible para el objeto AdminTask. Utilice este mandato para actualizar el alias de autenticación utilizado por los objetos JMS de servicio de sucesos de un servidor o clúster. Si el alias de autenticación JMS no existe, se crea. Para obtener más información sobre el objeto AdminTask, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** **Vista** **Windows 7** El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

### Parámetros

#### - nodeName

Nombre del nodo en el que se debe actualizar el alias de autenticación JMS del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

- **serverName**  
Nombre del servidor en el que se debe actualizar el alias de autenticación JMS del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.
- **clusterName**  
Nombre del clúster en el que se debe actualizar el alias de autenticación JMS del servicio de sucesos. Este parámetro no se debe especificar si se especifican los parámetros **nodeName** y **serverName**.
- **userName**  
Nombre del usuario que se debe utilizar en la actualización de un alias de autenticación JMS del servicio de sucesos en un servidor o clúster.  
  
**Importante:** Debe especificar un ID de usuario válido; este campo no puede estar vacío.
- **password**  
Contraseña del usuario que se debe utilizar en la actualización de un alias de autenticación JMS del servicio de sucesos en un servidor o clúster.  
  
**Importante:** Debe especificar una contraseña válida; este campo no puede estar vacío.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:
 

```
$AdminTask setEventServiceJmsAuthAlias{-nodeName nodename
-serverName servername username -password pwd}
$AdminTask setEventServiceJmsAuthAlias {-clusterName clustername
-userName username -password pwd}
```
- Utilizando serie Jython:
 

```
AdminTask.setEventServiceJmsAuthAlias('[-nodeName nodename
-serverName servername -userName username -password pwd]')
AdminTask.setEventServiceJmsAuthAlias('[-clusterName clustername
-userName username -password pwd]')
```
- Utilizando lista Jython:
 

```
AdminTask.setEventServiceJmsAuthAlias(['-nodeName', 'nodename',
'-serverName', '-servername', '-userName', 'username', '-password', 'pwd'])
AdminTask.setEventServiceJmsAuthAlias(['-clusterName',
'clustername', '-userName', 'username', '-password', 'pwd'])
```

### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:
 

```
$AdminTask setEventServiceJmsAuthAlias {-interactive}
```
- Utilizando serie Jython:
 

```
AdminTask.setEventServiceJmsAuthAlias( ['-interactive'])
```
- Utilizando lista Jython:
 

```
AdminTask.setEventServiceJmsAuthAlias(['-interactive'])
```

---

## Mandato enableEventService

Utilice el mandato enableEventService para habilitar el servicio de sucesos en el servidor.

## Finalidad

El mandato `enableEventService` es un mandato administrativo de Common Event Infrastructure que está disponible para el objeto `AdminTask`. Utilice este mandato para habilitar el servicio de sucesos para que se inicie después del reinicio siguiente de los servidores designados por los parámetros `nodeName`, `serverName` o `clusterName`. Para obtener más información sobre el objeto `AdminTask`, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** **Vista** **Windows 7** El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

## DITA

### Parámetros

#### - `nodeName`

Nombre del nodo en el que se debe habilitar el servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro `serverName`. Este parámetro no se debe especificar si se especifica el parámetro `clusterName`.

#### - `serverName`

Nombre del servidor en el que se debe habilitar el servicio de sucesos. Este parámetro se debe especificar si se especifica el parámetro `nodeName`. Este parámetro no se debe especificar si se especifica el parámetro `clusterName`.

#### - `clusterName`

Nombre del clúster en el que se debe habilitar el servicio de sucesos. Este parámetro no se debe especificar si se especifican los parámetros `nodeName` y `serverName`.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:

```
$AdminTask enableEventService {-nodeName nodename  
-serverName servername}
```

```
$AdminTask enableEventService {-clusterName clustername}
```

- Utilizando serie Jython:

```
AdminTask.enableEventService(['-nodeName nodename  
-serverName servername'])
```

```
AdminTask.enableEventService(['-clusterName clustername'])
```

- Utilizando lista Jython:

```
AdminTask.enableEventService(['-nodeName', 'nodename',  
'-serverName', '-servername'])
```

```
AdminTask.enableEventService(['-clusterName', 'clustername'])
```

### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:

```
$AdminTask enableEventService {-interactive}
```

- Utilizando serie Jython:

```
AdminTask.enableEventService(['-interactive'])
```

- Utilizando lista Jython:  
`AdminTask.enableEventService(['-interactive'])`

---

## Mandato disableEventService

Utilice el mandato `disableEventService` para inhabilitar el servicio de sucesos en el servidor.

### Finalidad

El mandato `disableEventService` es un mandato administrativo de Common Event Infrastructure que está disponible para el objeto `AdminTask`. Utilice este mandato para inhabilitar el servicio de sucesos para que no se inicie después del reinicio siguiente de los servidores designados por los parámetros `nodeName`, `serverName` o `clusterName`. Para obtener más información sobre el objeto `AdminTask`, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** **Vista** El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003 o Windows Vista.

### Parámetros

- **nodeName**  
Nombre del nodo en el que se debe inhabilitar el servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.
- **serverName**  
Nombre del servidor en el que se debe inhabilitar el servicio de sucesos. Este parámetro se debe especificar si se especifica el parámetro **nodeName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.
- **clusterName**  
Nombre del clúster en el que se debe inhabilitar el servicio de sucesos. Este parámetro no se debe especificar si se especifican los parámetros **nodeName** y **serverName**.

### Ejemplo

#### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:  

```
$AdminTask disableEventService {-nodeName nodename  
-serverName servername}  
  
$AdminTask disableEventService {-clusterName clustername}
```
- Utilizando serie Jython:  

```
AdminTask.disableEventService(['-nodeName nodename  
-serverName servername'])  
  
AdminTask.disableEventService(['-clusterName clustername'])
```
- Utilizando lista Jython:  

```
AdminTask.disableEventService(['-nodeName', 'nodename',  
'-serverName', '-servername'])  
  
AdminTask.disableEventService(['-clusterName', 'clustername'])
```

### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:  
`$AdminTask disableEventService {-interactive}`
- Utilizando serie Jython:  
`AdminTask.disableEventService(['-interactive'])`
- Utilizando lista Jython:  
`AdminTask.disableEventService(['-interactive'])`

---

## Mandato showEventServiceStatus

Utilice el mandato showEventServiceStatus para visualizar estado del servicio de sucesos en el servidor.

### Finalidad

El mandato showEventServiceStatus es un mandato administrativo de Common Event Infrastructure disponible para el objeto AdminTask. Utilice este mandato para devolver el estado del servicio de sucesos de un servidor o clúster. Si la tarea se ejecuta sin parámetros, se visualiza el estado de todos los servicios de sucesos. Para filtrar la lista de servicios de sucesos que se va a visualizar, proporcione nodeName, serverName o clusterName. Para obtener más información sobre el objeto AdminTask, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** **Vista** **Windows 7** El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

### Parámetros

#### - nodeName

Utilice este parámetro para mostrar únicamente el estado de los servicios de sucesos que pertenecen al nodo especificado. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

#### - serverName

Utilice este parámetro para mostrar únicamente el estado de los servicios de sucesos que pertenecen al servidor especificado. Puede utilizar este parámetro con el parámetro **nodeName** para visualizar el estado del servicio de sucesos que pertenece al nodo y servidor especificados. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

#### - clusterName

Utilice este parámetro para mostrar únicamente el estado de los servicios de sucesos que pertenecen al clúster especificado. Este parámetro no se debe especificar si se especifican los parámetros **nodeName** o **serverName**.

### Ejemplo

#### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:  
`$AdminTask showEventServiceStatus {-nodeName nodename  
-serverName servername}`  
`$AdminTask showEventServiceStatus {-clusterName clustername}`
- Utilizando serie Jython:

```
AdminTask.showEventServiceStatus(['-nodeName nodename  
-serverName servername'])
```

```
AdminTask.showEventServiceStatus(['-clusterName clustername'])
```

- Utilizando lista Jython:

```
AdminTask.showEventServiceStatus(['-nodeName', 'nodename',  
'-serverName', '-servername'])
```

```
AdminTask.showEventServiceStatus(['-clusterName', 'clustername'])
```

#### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:

```
$AdminTask showEventServiceStatus {-interactive}
```

- Utilizando serie Jython:

```
AdminTask.showEventServiceStatus(['-interactive'])
```

- Utilizando lista Jython:

```
AdminTask.showEventServiceStatus(['-interactive'])
```



---

## Mandato removeEventService

Utilice el mandato `removeEventService` para suprimir la aplicación del servicio de sucesos de su servidor.

### Finalidad

El mandato `removeEventService` es un mandato administrativo de Common Event Infrastructure disponible para el objeto `AdminTask`. Utilice este mandato para eliminar el servicio de sucesos de un servidor o clúster. Para obtener más información sobre el objeto `AdminTask`, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:**   El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

### Parámetros

#### - nodeName

Nombre del nodo del que debe eliminarse el servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

#### - serverName

Nombre del servidor del que debe eliminarse el servicio de sucesos. Este parámetro se debe especificar si se especifica el parámetro **nodeName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

#### - clusterName

Nombre del clúster del que debe eliminarse el servicio de sucesos. Este parámetro no se debe especificar si se especifican los parámetros **nodeName** y **serverName**.

### Ejemplo

#### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:

```
$AdminTask removeEventService {-nodeName nodename  
-serverName servername}
```

```
$AdminTask removeEventService {-clusterName clustername}
```

- Utilizando serie Jython:

```
AdminTask.removeEventService(['-nodeName nodename  
-serverName servername'])
```

```
AdminTask.removeEventService(['-clusterName clustername'])
```

- Utilizando lista Jython:

```
AdminTask.removeEventService(['-nodeName', 'nodename',  
'-serverName', '-servername'])
```

```
AdminTask.removeEventService(['-clusterName', 'clustername'])
```

#### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:

```
$AdminTask removeEventService {-interactive}
```

- Utilizando serie Jython:

```
AdminTask.removeEventService(['-interactive'])
```

- Utilizando lista Jython:

```
AdminTask.removeEventService(['-interactive'])
```

---

## Mandato removeEventServiceMdb

Utilice el mandato removeEventServiceMdb para suprimir el bean controlado por mensajes del servicio de sucesos de su servidor.

### Finalidad

El mandato removeEventServiceMdb es un mandato administrativo de Common Event Infrastructure disponible para el objeto AdminTask. Utilice este mandato para eliminar el MDB de servicio de sucesos de un servidor o clúster. Para obtener más información sobre el objeto AdminTask, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** **Vista** **Windows 7** El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

### Parámetros

#### - nodeName

Nombre del nodo en el que se debe eliminar el MDB del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

#### - serverName

Nombre del servidor en el que se debe eliminar el MDB del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

- **clusterName**

Nombre del clúster en el que se debe eliminar el MDB del servicio de sucesos. Este parámetro no se debe especificar si se especifican los parámetros **nodeName** y **serverName**.

- **applicationName**

Nombre de la aplicación MDB del servicio de sucesos que se debe eliminar de un servidor o un clúster.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:

```
$AdminTask removeEventServiceMdb {-applicationName appname  
-nodeName nodename -serverName servername}  
$AdminTask removeEventServiceMdb {-applicationName appname  
-clusterName clustername}
```

- Utilizando serie Jython:

```
AdminTask.removeEventServiceMdb(['-applicationName appname  
-nodeName nodename -serverName servername'])  
AdminTask.removeEventServiceMdb(['-applicationName appname  
-clusterName clustername'])
```

- Utilizando lista Jython:

```
AdminTask.removeEventServiceMdb (['-applicationName',  
'appname', '-nodeName', 'nodename', '-serverName', 'servername'])  
AdminTask.removeEventServiceMdb (['-applicationName',  
'appname', '-clusterName', 'clustername'])
```

### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:

```
$AdminTask removeEventServiceMdb {-interactive}
```

- Utilizando serie Jython:

```
AdminTask.removeEventServiceMdb(['-interactive'])
```

- Utilizando lista Jython:

```
AdminTask.removeEventServiceMdb(['-interactive'])
```

---

## Mandato removeEventServiceDB2DB

Utilice el mandato removeEventServiceDB2DB para suprimir el servicio de sucesos y, opcionalmente, la base de datos de sucesos DB2 asociada.

### Finalidad

El mandato removeEventServiceDB2DB es un mandato administrativo de Common Event Infrastructure disponible para el objeto AdminTask. Utilice este mandato para suprimir la base de datos del servicio de sucesos y los orígenes de datos para DB2 de un servidor o clúster. Para obtener más información sobre el objeto AdminTask, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** **Vista** **Windows 7** El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.



## Parámetros

### - **removeDB**

El mandato elimina la base de datos cuando este parámetro se establece en `true` y no la elimina de la base de datos cuando se establece en `false`. Para eliminar la base de datos, el servidor actual ya debe estar configurado para ejecutar los mandatos de base de datos.

### - **nodeName**

Nombre del nodo que contiene el servidor en el que se debe eliminar el origen de datos del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

### - **serverName**

Nombre del servidor en el que se debe eliminar el origen de datos del servidor de sucesos. Si este parámetro se especifica sin el parámetro **nodeName**, el mandato utilizará el nombre de nodo del perfil de WebSphere actual. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

### - **clusterName**

Nombre del clúster en el que se debe eliminar el origen de datos del servidor de sucesos. Si se especifica este parámetro, no deben especificarse los parámetros **serverName** y **nodeName**. Este parámetro no se debe especificar si se especifican los parámetros **serverName** y **nodeName**.

### - **dbUser**

ID de usuario de DB2 que tiene privilegios para crear y eliminar las bases de datos. Este parámetro debe establecerse cuando el parámetro **removeDB** se establece en `true`. Si no se especifica, el valor por omisión es `db2inst1`.

### - **dbPassword**

Contraseña de DB2. Este parámetro debe establecerse cuando el parámetro **removeDB** se establece en `true`.

### - **dbScriptDir**

Directorio que contiene los scripts generados por el mandato de configuración de base de datos del servicio de sucesos. Si se especifica, el mandato ejecutará los scripts de este directorio para eliminar la base de datos del servicio de sucesos. El directorio de salida de script de base de datos por omisión es `raíz_perfil/databases/event/nodo/servidor/dbscripts/db2`.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:

```
$AdminTask removeEventServiceDB2DB {-removeDB true
-nodeName nodename
-serverName servername
-dbUser db2inst1
-dbPassword dbpassword }
```

- Utilizando serie Jython:

```
AdminTask.removeEventServiceDB2DB(['-removeDB true
-nodeName nodename
-serverName servername
-dbUser db2inst1
-dbPassword dbpassword'])
```

- Utilizando lista Jython:

```
AdminTask.removeEventServiceDB2DB(['-removedB', 'true',  
  '-nodeName', 'nodename',  
  '-serverName', 'servername',  
  '-dbUser', 'db2inst1',  
  '-dbPassword', 'dbpassword'])
```

#### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:  
`$AdminTask removeEventServiceDB2DB -interactive`
- Utilizando serie Jython:  
`AdminTask.removeEventServiceDB2DB(['-interactive'])`
- Utilizando lista Jython:  
`AdminTask.removeEventServiceDB2DB(['-interactive'])`



---

## Mandato removeEventServiceDB2iSeriesDB

Utilice el mandato removeEventServiceDB2iSeriesDB para suprimir el servicio de sucesos y, opcionalmente, la base de datos de sucesos DB2 para iSeries asociada.

### Finalidad

El mandato removeEventServiceDB2iSeriesDB es un mandato administrativo de Common Event Infrastructure que está disponible para el objeto AdminTask. Utilice este mandato para eliminar los orígenes de datos DB2 para iSeries desde un servidor o clúster. Para obtener más información sobre el objeto AdminTask, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:**   El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

### Parámetros

#### - nodeName

Nombre del nodo que contiene el servidor en el que se debe eliminar el origen de datos del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

#### - serverName

Nombre del servidor en el que se debe eliminar el origen de datos del servidor de sucesos. Si este parámetro se especifica sin el parámetro **nodeName**, el mandato utilizará el nombre de nodo del perfil de WebSphere actual. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

#### - clusterName

Nombre del clúster en el que se debe eliminar el origen de datos del servidor de sucesos. Si se especifica este parámetro, no deben especificarse los parámetros **serverName** y **nodeName**. Este parámetro no se debe especificar si se especifican los parámetros **serverName** y **nodeName**.

### Ejemplo

#### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:

```
$AdminTask removeEventServiceDB2iSeriesDB {-nodeName nodename  
-serverName servername }
```

- Utilizando serie Jython:

```
AdminTask.removeEventServiceDB2iSeriesDB(['-nodeName nodename  
-serverName servername'])
```

- Utilizando lista Jython:

```
AdminTask.removeEventServiceDB2iSeriesDB(['-nodeName', 'nodename',  
'-serverName', 'servername'])
```

#### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:

```
$AdminTask removeEventServiceDB2iSeriesDB -interactive
```

- Utilizando serie Jython:

```
AdminTask.removeEventServiceDB2iSeriesDB(['-interactive'])
```

- Utilizando lista Jython:

```
AdminTask.removeEventServiceDB2iSeriesDB(['-interactive'])
```



---

## Mandato removeEventServiceDB2ZOSDB

Utilice el mandato removeEventServiceDB2ZOSDB para suprimir el servicio de sucesos y, opcionalmente, la base de datos de sucesos DB2 para z/OS asociada.

### Finalidad

El mandato removeEventServiceDB2ZOSDB es un mandato administrativo de Common Event Infrastructure disponible para el objeto AdminTask. Utilice este mandato para eliminar los orígenes de datos y base de datos de servicio de sucesos para DB2 z/OS desde un servidor o clúster. Para obtener más información sobre el objeto AdminTask, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:**   El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

### Parámetros

#### - removeDB

El mandato elimina la base de datos cuando este parámetro se establece en true y no la elimina de la base de datos cuando se establece en false. Para eliminar la base de datos, el servidor actual ya debe estar configurado para ejecutar los mandatos de base de datos.

#### - nodeName

Nombre del nodo que contiene el servidor en el que se debe eliminar el origen de datos del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

#### - serverName

Nombre del servidor en el que se debe eliminar el origen de datos del servidor de sucesos. Si este parámetro se especifica sin el parámetro **nodeName**, el mandato utilizará el nombre de nodo del perfil de WebSphere actual. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

- **clusterName**  
Nombre del clúster en el que se debe eliminar el origen de datos del servidor de sucesos. Si se especifica este parámetro, no deben especificarse los parámetros **serverName** y **nodeName**. Este parámetro no se debe especificar si se especifican los parámetros **serverName** y **nodeName**.
- **dbName**  
Nombre de base de datos de DB2. En el cliente DB2, es el nombre de la base de datos catalogada. En el servidor z/OS nativo, es el nombre del subsistema de base de datos. Este parámetro debe establecerse cuando el parámetro **removeDB** se establece en true. Si no se especifica, el valor por omisión es event.
- **dbUser**  
ID de usuario de DB2 que tiene privilegios para crear y eliminar las bases de datos. Este parámetro debe establecerse cuando el parámetro **removeDB** se establece en true.
- **dbPassword**  
Contraseña de DB2. Este parámetro debe establecerse cuando el parámetro **removeDB** se establece en true.
- **dbScriptDir**  
Directorio que contiene los scripts generados por el mandato de configuración de base de datos del servicio de sucesos. Si se especifica, el mandato ejecutará los scripts de este directorio para eliminar la base de datos del servicio de sucesos. El directorio de salida de script de base de datos por omisión es *raíz\_perfil/databases/event/nodo/servidor/dbscripts/db2zos*.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:  

```
$AdminTask removeEventServiceDB2ZOSDB {-removeDB true -nodeName nodename -serverName servername -dbUser db2user -dbPassword dbpassword}
```
- Utilizando serie Jython:  

```
AdminTask.removeEventServiceDB2ZOSDB(['-removeDB true -nodeName nodename -serverName servername -dbUser db2user -dbPassword dbpassword'])
```
- Utilizando lista Jython:  

```
AdminTask.removeEventServiceDB2ZOSDB(['-removeDB', 'true', '-nodeName', 'nodename', '-serverName', 'servername', '-dbUser', 'db2user', '-dbPassword', 'dbpassword'])
```

### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:  

```
$AdminTask removeEventServiceDB2ZOSDB -interactive
```
- Utilizando serie Jython:  

```
AdminTask.removeEventServiceDB2ZOSDB(['-interactive'])
```
- Utilizando lista Jython:  

```
AdminTask.removeEventServiceDB2ZOSDB(['-interactive'])
```

---

## Mandato removeEventServiceDerbyDB

Utilice el mandato removeEventServiceDerbyDB para suprimir el servicio de sucesos y, opcionalmente, la base de datos de sucesos Derby asociada.

## Finalidad

El mandato `removeEventServiceDerbyDB` es un mandato administrativo de Common Event Infrastructure disponible para el objeto `AdminTask`. Utilice este mandato para eliminar el origen de datos y base de datos de servicio de sucesos para Derby desde un servidor o clúster. Para obtener más información sobre el objeto `AdminTask`, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** **Vista** **Windows 7** El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

## Parámetros

### - `removeDB`

El mandato elimina la base de datos cuando este parámetro se establece en `true` y no la elimina de la base de datos cuando se establece en `false`. Para eliminar la base de datos, el servidor actual ya debe estar configurado para ejecutar los mandatos de base de datos.

### - `nodeName`

Nombre del nodo que contiene el servidor en el que se debe eliminar el origen de datos del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro `serverName`. Este parámetro no se debe especificar si se especifica el parámetro `clusterName`.

### - `serverName`

Nombre del servidor en el que se debe eliminar el origen de datos del servidor de sucesos. Si este parámetro se especifica sin el parámetro `nodeName`, el mandato utilizará el nombre de nodo del perfil de WebSphere actual. Este parámetro no se debe especificar si se especifica el parámetro `clusterName`.

### - `clusterName`

Nombre del clúster en el que se debe eliminar el origen de datos del servidor de sucesos. Si se especifica este parámetro, no deben especificarse los parámetros `serverName` y `nodeName`. Este parámetro no se debe especificar si se especifican los parámetros `serverName` y `nodeName`.

### - `dbScriptDir`

Directorio que contiene los scripts generados por el mandato de configuración de base de datos del servicio de sucesos. Si se especifica, el mandato ejecutará los scripts de este directorio para eliminar la base de datos del servicio de sucesos. El directorio de salida de script de base de datos por omisión es *raíz\_perfil/databases/event/nodo/servidor/dbscripts/derby*.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:

```
$AdminTask removeEventServiceDerbyDB {-removeDB true -nodeName nodename  
-serverName servername}
```

- Utilizando serie Jython:

```
AdminTask.removeEventServiceDerbyDB('[-removeDB true -nodeName nodename  
-serverName servername]')
```

- Utilizando lista Jython:

```
AdminTask.removeEventServiceDerbyDB(['-removeDB', 'true', '-nodeName',
'nodename', '-serverName', 'servername'])
```

#### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:  
`$AdminTask removeEventServiceDerbyDB -interactive`
- Utilizando serie Jython:  
`AdminTask.removeEventServiceDerbyDB( ['-interactive'])`
- Utilizando lista Jython:  
`AdminTask.removeEventServiceDerbyDB(['-interactive'])`



---

## Mandato `removeEventServiceInformixDB`

Utilice el mandato `removeEventServiceInformixDB` para suprimir el servicio de sucesos y, opcionalmente, la base de datos de sucesos Informix asociada.

### Finalidad

El mandato `removeEventServiceInformixDB` es un mandato administrativo de Common Event Infrastructure disponible para el objeto `AdminTask`. Utilice este mandato para suprimir la base de datos del servicio de sucesos y los orígenes de datos para Informix de un servidor o clúster. Para obtener más información sobre el objeto `AdminTask`, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:**   El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

### Parámetros

#### - `removeDB`

El mandato elimina la base de datos cuando este parámetro se establece en `true` y no la elimina de la base de datos cuando se establece en `false`. Para eliminar la base de datos, el servidor actual ya debe estar configurado para ejecutar los mandatos de base de datos.

#### - `nodeName`

Nombre del nodo que contiene el servidor en el que se debe eliminar el origen de datos del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro `serverName`. Este parámetro no se debe especificar si se especifica el parámetro `clusterName`.

#### - `serverName`

Nombre del servidor en el que se debe eliminar el origen de datos del servidor de sucesos. Si este parámetro se especifica sin el parámetro `nodeName`, el mandato utilizará el nombre de nodo del perfil de WebSphere actual. Este parámetro no se debe especificar si se especifica el parámetro `clusterName`.

#### - `clusterName`

Nombre del clúster en el que se debe eliminar el origen de datos del servidor de sucesos. Si se especifica este parámetro, no deben especificarse los parámetros `serverName` y `nodeName`. Este parámetro no se debe especificar si se especifican los parámetros `serverName` y `nodeName`.

#### - `dbScriptDir`

Directorio que contiene los scripts generados por el mandato de configuración

de base de datos del servicio de sucesos. Si se especifica, el mandato ejecutará los scripts de este directorio para eliminar la base de datos del servicio de sucesos. El directorio de salida de script de base de datos por omisión es *raíz\_perfil/databases/event/nodo/servidor/dbscripts/informix*.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:  

```
$AdminTask removeEventServiceInformixDB {-removedB true -nodeName nodename  
-serverName servername}
```
- Utilizando serie Jython:  

```
AdminTask.removeEventServiceInformixDB(['-removedB true -nodeName nodename  
-serverName servername'])
```
- Utilizando lista Jython:  

```
AdminTask.removeEventServiceInformixDB(['-removedB', 'true', '-nodeName',  
'nodename', '-serverName', 'servername'])
```

### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:  

```
$AdminTask removeEventServiceInformixDB -interactive
```
- Utilizando serie Jython:  

```
AdminTask.removeEventServiceInformixDB(['-interactive'])
```
- Utilizando lista Jython:  

```
AdminTask.removeEventServiceInformixDB(['-interactive'])
```

---

## Mandato `removeEventServiceOracleDB`

Utilice el mandato `removeEventServiceOracleDB` para suprimir el servicio de sucesos y, opcionalmente, la base de datos de sucesos Oracle asociada.

### Finalidad

El mandato `removeEventServiceOracleDB` es un mandato administrativo de Common Event Infrastructure disponible para el objeto `AdminTask`. Utilice este mandato para suprimir los orígenes de datos y las tablas de servicio de sucesos para Oracle de un servidor o clúster. El mandato no elimina la base de datos. Para obtener más información sobre el objeto `AdminTask`, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** **Vista** **Windows 7** El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

### Parámetros

#### - `removeDB`

El mandato elimina las tablas del servicio de sucesos cuando este parámetro se establece en `true` y no las elimina de la base de datos cuando se establece en `false`.

#### - `nodeName`

Nombre del nodo que contiene el servidor en el que se debe eliminar el origen de datos del servicio de sucesos. Si se especifica este parámetro, deberá

especificarse el parámetro **serverName**. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

- **serverName**

Nombre del servidor en el que se debe eliminar el origen de datos del servidor de sucesos. Si este parámetro se especifica sin el parámetro **nodeName**, el mandato utilizará el nombre de nodo del perfil de WebSphere actual. Este parámetro no se debe especificar si se especifica el parámetro **clusterName**.

- **clusterName**

Nombre del clúster en el que se debe eliminar el origen de datos del servidor de sucesos. Si se especifica este parámetro, no deben especificarse los parámetros **serverName** y **nodeName**. Este parámetro no se debe especificar si se especifican los parámetros **serverName** y **nodeName**.

- **sysUser**

ID de usuario del sistema de base de datos Oracle. Si no se especifica, el valor por omisión es sys.

- **sysPassword**

Contraseña para el usuario especificado por el parámetro sysUser.

- **dbScriptDir**

Directorio que contiene los scripts generados por el mandato de configuración de base de datos del servicio de sucesos. Si se especifica, el mandato ejecutará los scripts de este directorio para eliminar la base de datos del servicio de sucesos. El directorio de salida de script de base de datos por omisión es *raíz\_perfil/databases/event/nodo/servidor/dbscripts/oracle*.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:  

```
$AdminTask removeEventServiceOracleDB {-removeDB true -nodeName nodename  
-serverName servername -sysUser sys -sysPassword syspassword}
```
- Utilizando serie Jython:  

```
AdminTask.removeEventServiceOracleDB(['-removeDB true -nodeName nodename  
-serverName servername -sysUser sys -sysPassword syspassword'])
```
- Utilizando lista Jython:  

```
AdminTask.removeEventServiceOracleDB(['-removeDB', 'true', '-nodeName',  
'nodename', '-serverName', 'servername', '-sysUser', 'sys',  
'-sysPassword', 'syspassword'])
```

### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:  

```
$AdminTask removeEventServiceOracleDB -interactive
```
- Utilizando serie Jython:  

```
AdminTask.removeEventServiceOracleDB(['-interactive'])
```
- Utilizando lista Jython:  

```
AdminTask.removeEventServiceOracleDB(['-interactive'])
```

---

## Mandato removeEventServiceSQLServerDB

Utilice el mandato removeEventServiceSQLServerDB para suprimir el servicio de sucesos y, opcionalmente, la base de datos de sucesos SQL Server asociada.



## Finalidad

El mandato `removeEventServiceSQLServerDB` es un mandato administrativo de Common Event Infrastructure disponibles para el objeto `AdminTask`. Utilice este mandato para suprimir los orígenes de datos y base de datos de servicio de sucesos para SQL Server de un servidor o clúster. Para obtener más información sobre el objeto `AdminTask`, consulte la documentación de WebSphere Application Server Network Deployment, versión 6.1.

**Nota:** **Vista** **Windows 7** El producto utiliza una versión de Jython que no da soporte a los sistemas operativos Microsoft Windows 2003, Windows Vista o Windows 7.

## Parámetros

### - `removeDB`

El mandato elimina la base de datos cuando este parámetro se establece en `true` y no la elimina de la base de datos cuando se establece en `false`. Para eliminar la base de datos, el servidor actual ya debe estar configurado para ejecutar los mandatos de base de datos.

### - `nodeName`

Nombre del nodo que contiene el servidor en el que se debe eliminar el origen de datos del servicio de sucesos. Si se especifica este parámetro, deberá especificarse el parámetro `serverName`. Este parámetro no se debe especificar si se especifica el parámetro `clusterName`.

### - `serverName`

Nombre del servidor en el que se debe eliminar el origen de datos del servidor de sucesos. Si este parámetro se especifica sin el parámetro `nodeName`, el mandato utilizará el nombre de nodo del perfil de WebSphere actual. Este parámetro no se debe especificar si se especifica el parámetro `clusterName`.

### - `clusterName`

Nombre del clúster en el que se debe eliminar el origen de datos del servidor de sucesos. Si se especifica este parámetro, no deben especificarse los parámetros `serverName` y `nodeName`. Este parámetro no se debe especificar si se especifican los parámetros `serverName` y `nodeName`.

### - `dbServerName`

Nombre de servidor de la base de datos de SQL Server. Este parámetro debe establecerse cuando el parámetro `removeDB` se establece en `true`.

### - `dbUser`

ID de usuario de SQL Server que es propietario de las tablas del servicio de sucesos. Si no se especifica, el valor por omisión es `ceiuser`.

### - `saUser`

ID de usuario que tiene privilegios para eliminar bases de datos y usuarios. Si no se especifica, el valor por omisión es `sa`.

### - `saPassword`

Contraseña especificada por el parámetro `saUser`. Este parámetro es obligatorio cuando el parámetro `removeDB` se establece en `true`.

### - `dbScriptDir`

Directorio que contiene los scripts generados por el mandato de configuración de base de datos del servicio de sucesos. Si se especifica, el mandato ejecutará los scripts de este directorio para eliminar la base de datos del servicio de

sucesos. El directorio de salida de script de base de datos por omisión es *raíz\_perfil/databases/event/nodo/servidor/dbscripts/sqlserver*.

## Ejemplo

### Uso de ejemplo de la modalidad de proceso por lotes:

- Utilizando Jacl:  

```
$AdminTask removeEventServiceSQLServerDB {-removeDB true -nodeName nodename  
-serverName servername -dbUser ceiuser -saUser sa -saPassword sapassword  
-dbServerName sqlservername}
```
- Utilizando serie Jython:  

```
AdminTask.removeEventServiceSQLServerDB(['-removeDB true -nodeName nodename  
-serverName servername -dbUser ceiuser -saUser sa -saPassword sapassword  
-dbServerName sqlservername'])
```
- Utilizando lista Jython:  

```
AdminTask.removeEventServiceSQLServerDB(['-removeDB', 'true', '-nodeName',  
'nodename', '-serverName', 'servername', '-dbUser', 'ceiuser', '-saUser', 'sa',  
-saPassword', 'sapassword', '-dbServerName', 'sqlservername'])
```

### Uso de ejemplo de la modalidad interactiva:

- Utilizando Jacl:  

```
$AdminTask removeEventServiceSQLServerDB -interactive
```
- Utilizando serie Jython:  

```
AdminTask.removeEventServiceSQLServerDB(['-interactive'])
```
- Utilizando lista Jython:  

```
AdminTask.removeEventServiceSQLServerDB(['-interactive'])
```

---

## Programa de utilidad de línea de mandatos eventbucket

El mandato **eventbucket** visualiza o cambia la configuración de la cubeta de la base de datos de sucesos.

### Finalidad

Visualiza o cambia la configuración de la cubeta de la base de datos de sucesos.

**eventbucket** [-status] [-change]

### Descripción

El mandato **eventbucket** visualiza o cambia la configuración de la cubeta de la base de datos de sucesos. El programa de utilidad de depuración rápida utiliza cubetas para depurar datos antiguos de sucesos en la base de datos de sucesos. Al ejecutar este mandato, puede determinar la configuración de la cubeta actual, o bien puede intercambiar las cubetas activas e inactivas.

**Seguridad:** Si la seguridad de WebSphere está habilitada, el ID de usuario debe estar correlacionado con el rol **eventAdministrator** para ver o cambiar la configuración de la cubeta de base de datos de sucesos.

### Parámetros

#### **-status**

Visualiza información sobre la configuración de la cubeta actual, incluido el

valor de cubeta activa y el intervalo de comprobación de cubetas (la frecuencia con la que el plug-in del almacén de datos realiza comprobaciones para determinar qué cubeta está activa).

#### **-change**

Intercambia las cubetas para que la cubeta activa pase a ser la cubeta inactiva y viceversa. La cubeta inactiva debe estar vacía para poder utilizar esta opción.

### **Ejemplos**

En este ejemplo se visualiza la configuración de la cubeta actual:

```
eventbucket -status
```

En este ejemplo se intercambian las cubetas activas e inactivas:

```
eventbucket -change
```

---

## **Programa de utilidad de línea de mandatos eventpurge**

El mandato **eventpurge** suprime sucesos de la base de datos de sucesos.

### **Finalidad**

Suprime sucesos de la base de datos de sucesos.

**eventpurge** [-seconds *segundos* | -end *hora\_final*] [-group *grupo\_sucesos*] [-severity *gravedad*] [-extensionname *nombre\_extensión*] [-start *hora\_inicial*] [-size *tamaño*]

### **Descripción**

El mandato **eventpurge** suprime sucesos de la base de datos de sucesos. Puede suprimir todos los sucesos de la base de datos de sucesos, o bien puede limitar la supresión a sucesos que cumplan ciertos criterios.

**Seguridad:** Si la seguridad de WebSphere está habilitada, el ID de usuario debe estar correlacionado con el rol `eventAdministrator` para suprimir sucesos.

### **Parámetros**

#### **-seconds** *segundos*

Antigüedad mínima de los sucesos que se han suprimido. El valor *segundos* debe ser un entero. Solo se suprimen los sucesos que tengan una antigüedad mayor que el número de segundos especificados. Este parámetro es necesario si no especifica el parámetro **-end**.

#### **-end** *hora\_final*

Hora final del grupo de sucesos que desea suprimir. Solo se suprimen los sucesos generados antes de la hora especificada. El valor *hora\_final* debe especificarse en el formato `dateTime XML (CCAA-MM-DDThh:mm:ss )`. Por ejemplo, las 12 del mediodía del 1 de enero de 2006 en la hora estándar del Este (EST) sería `2006-01-01T12:00:00-05:00`. Para obtener más información sobre el tipo de datos `dateTime`, consulte la documentación de Esquema XML en [www.w3.org](http://www.w3.org).

Este parámetro es necesario si no especifica el parámetro **-seconds**.

#### **-group** *grupo\_sucesos*

Grupo de sucesos en el que suprimir sucesos. El valor *grupo\_sucesos* debe ser el

nombre de un grupo de sucesos definido en la configuración de Common Event Infrastructure. Este parámetro es opcional.

**-severity** *gravedad*

Gravedad de los sucesos que desea suprimir. El valor *gravedad* debe ser un entero; solo se suprimen los sucesos cuya gravedad sea igual al valor especificado. Este parámetro es opcional.

**-extensionname** *nombre\_extensión*

Nombre de extensión de los sucesos que desea incluir en la supresión. Utilice este parámetro para restringir la supresión a los sucesos de un tipo específico. Solo se suprimen los sucesos cuya propiedad `extensionName` sea igual a *nombre\_extensión*. Este parámetro es opcional.

**-start** *hora\_inicial*

Hora inicial del grupo de sucesos que desea suprimir. Solo se suprimen los sucesos generados después de la hora especificada. El valor *hora\_inicial* debe especificarse en el formato `dateTime XML (CCAA-MM-DDThh:mm:ss)`. Este parámetro es opcional.

**-size** *tamaño*

Número de sucesos que se van a suprimir en una sola transacción. El valor *tamaño* debe ser un entero. Después de suprimir este número de sucesos, el mandato compromete la transacción antes de pasar a una transacción nueva. Este parámetro es opcional.

## Ejemplo

En el ejemplo siguiente se suprimen todos los sucesos de la base de datos cuya gravedad sea 20 (inofensivo) y que se hayan generado antes de hace 10 minutos.

```
eventpurge -group "Todos los sucesos" -severity 20 -seconds 600
```



