



Flujo de trabajo

OnCommand Workflow Automation

NetApp

October 09, 2025

Tabla de contenidos

Flujo de trabajo	1
Descripción general de OnCommand Workflow Automation	1
Funciones DE WFA	1
Información de licencia DE WFA	2
Descripción del diseñador de Workflow Automation	2
Trabajar con los elementos básicos en OnCommand Workflow Automation	2
Qué base de datos de juegos	10
Gestión de flujos de trabajo	11
Personalizar flujos de trabajo predefinidos	11
Crear flujos de trabajo	13
Cree paquetes de flujo de trabajo de WFA	40
Integre los paquetes de flujo de trabajo de WFA con el repositorio SCM	46
Crear elementos básicos para los flujos de trabajo	49
Cree un tipo de origen de datos	50
Crear un comando	51
Crear un buscador	54
Crear un filtro	54
Crear una entrada de diccionario	55
Crear una función	56
Cree una plantilla	57
Cree una consulta de caché	58
Crear programas recurrentes	59
Defina las reglas de filtro	59
Agregar puntos de aprobación	61
Directrices de codificación para WFA	61
Pautas para las variables	61
Directrices para indentación	65
Directrices para comentarios	66
Directrices para el registro	68
Directrices para la gestión de errores	69
PowerShell general y convenciones Perl para WFA	72
Consideraciones sobre la adición de PowerShell y módulos Perl personalizados	73
Cmdlets y funciones DE WFA	74
Módulos de WFA PowerShell y Perl	74
Consideraciones que tener en cuenta al convertir comandos de PowerShell en Perl	77
Directrices para los elementos básicos de WFA	79
Palabras reservadas	92
Cómo se utilizan las API DE REST	93
Referencias al material de aprendizaje	94
Windows PowerShell	94
Kit de herramientas PowerShell de Data ONTAP	95
Perl	95
SDK de gestión de NetApp	96

Lenguaje de consultas estructurado (SQL)	96
MVFLEX Expression Language (MVEL)	96
Expresiones regulares	96
Documentación relacionada para OnCommand Workflow Automation	96
Otras referencias	96
Referencias de herramientas	96

Flujo de trabajo

Descripción general de OnCommand Workflow Automation

OnCommand Workflow Automation (WFA) es una solución de software que ayuda a automatizar tareas de gestión del almacenamiento, como aprovisionamiento, migración, decomisionado, configuraciones de protección de datos, y clonado de almacenamiento. Puede utilizar WFA para crear flujos de trabajo para completar las tareas especificadas en sus procesos.

Un flujo de trabajo es una tarea repetitiva y de procedimiento que consiste en pasos secuenciales, incluidos los siguientes tipos de tareas:

- Aprovisionamiento, migración o decomisión del almacenamiento para bases de datos o sistemas de archivos
- Configuración de un nuevo entorno de virtualización, incluidos los switches de almacenamiento y los almacenes de datos
- Configuración de almacenamiento para una aplicación como parte de un proceso de orquestación completo

Los arquitectos de almacenamiento pueden definir flujos de trabajo para seguir las prácticas recomendadas y cumplir los requisitos de la organización, como los siguientes:

- Utilizar las convenciones de nomenclatura necesarias
- Establecimiento de opciones únicas para los objetos de almacenamiento
- Selección de recursos
- Integrar la base de datos de gestión de la configuración (CMDB) interna y las aplicaciones de emisión de boletos

Funciones DE WFA

- Portal de diseño de flujos de trabajo para crear flujos de trabajo

El portal de diseño del flujo de trabajo incluye varios elementos básicos, como comandos, plantillas, buscadores, filtros, y funciones, que se utilizan para crear flujos de trabajo. El diseñador le permite incluir funcionalidades avanzadas para flujos de trabajo como la selección automatizada de recursos, la repetición de filas (bucle) y los puntos de aprobación.

El portal de diseño de flujos de trabajo también incluye elementos básicos, como entradas de diccionario, consultas de caché y tipos de origen de datos, para almacenar en caché datos de sistemas externos.

- Portal de ejecución para ejecutar flujos de trabajo, verificar el estado de la ejecución del flujo de trabajo y los registros de acceso
- Opción Administration/Settings para tareas como configurar WFA, conectarse a orígenes de datos y configurar credenciales de usuario
- Interfaces de servicio web para invocar flujos de trabajo desde portales externos y software de orquestación del centro de datos
- Almacén de automatización del almacenamiento para descargar paquetes de WFA. El paquete ONTAP

9.7.0 se incluye con WFA 5.1.

Información de licencia DE WFA

No se necesita ninguna licencia para usar el servidor OnCommand Workflow Automation.

Descripción del diseñador de Workflow Automation

Los flujos de trabajo del diseñador de Workflow Automation (WFA) se crean mediante los elementos básicos como buscadores, filtros y comandos. Comprender los elementos básicos y el proceso de creación de flujos de trabajo es importante antes de empezar a crear los flujos de trabajo.

Trabajar con los elementos básicos en OnCommand Workflow Automation

Los flujos de trabajo de Workflow Automation (WFA) constan de varios elementos básicos y WFA incluye una biblioteca de elementos básicos predefinidos. Puede utilizar los elementos básicos que proporciona WFA para crear flujos de trabajo que cumplan los requisitos de su organización.

WFA proporciona la estructura para los procesos de automatización del almacenamiento. La flexibilidad de WFA se basa en cómo se crean los flujos de trabajo mediante los elementos básicos del flujo de trabajo.

Los elementos básicos de WFA son los siguientes:

- Entradas de diccionario
- Comandos
- Filtros
- Buscadores
- Empresarial
- Plantillas

Debe comprender cómo se utilizan los elementos básicos en WFA para ayudarle a crear los flujos de trabajo.

Qué fuentes de datos son

Un origen de datos es un método para establecer una conexión con otros sistemas, archivos y bases de datos con el fin de extraer datos. Por ejemplo, un origen de datos puede ser una conexión con una base de datos Active IQ Unified Manager del tipo de origen de datos Active IQ Unified Manager 9.4.

Puede añadir un origen de datos personalizado a OnCommand Workflow Automation (WFA) para la adquisición de datos después de definir el tipo de origen de datos requerido asociando el esquema de almacenamiento en caché, el puerto requerido y el método de adquisición al tipo de origen de datos.

WFA almacena información en caché a través de varios orígenes de datos. WFA recopila información de recursos de los orígenes de datos y la formatea para el esquema de almacenamiento en caché. Las tablas de caché, que son las tablas dentro de los esquemas de almacenamiento en caché, tienen el formato de forma que coincidan con los objetos de entrada del diccionario. Cuando utiliza un buscador en flujos de trabajo,

devuelve un objeto de diccionario y los datos del objeto de diccionario se rellenan de las tablas de caché. El proceso de adquisición de datos de los orígenes de datos se conoce como *fuentes de datos adquisición*. Puede utilizar un método basado en scripts o un método basado en controladores para la adquisición del origen de datos. Los orígenes pueden ser diferentes entre sí y la adquisición de orígenes de datos puede muestrearlos a intervalos de tiempo diferentes. WFA entonces fusiona esa información en la base de datos y superpone los datos de reserva para mantener la información actualizada de los recursos en la base de datos.

La base de datos de WFA incluye varios esquemas de almacenamiento en caché diferentes. Un esquema de almacenamiento en caché es un conjunto de tablas y cada tabla incluye información de un determinado tipo de entrada de diccionario; sin embargo, las tablas pueden incluir información combinada de varios orígenes de un tipo de origen de datos específico. WFA usa la información de la base de datos para comprender el estado de los recursos, realizar cálculos y ejecutar comandos en los recursos.

Qué entradas de diccionario son

Las entradas del diccionario son uno de los elementos básicos de OnCommand Workflow Automation (WFA). Puede utilizar entradas del diccionario para representar tipos de objetos y sus relaciones en los entornos de almacenamiento y relacionados con el almacenamiento. A continuación, puede utilizar filtros en los flujos de trabajo para devolver el valor de las claves naturales de las entradas del diccionario.

Una entrada de diccionario es la definición de un tipo de objeto compatible con WFA. Cada entrada de diccionario representa un tipo de objeto y su relación en los entornos de almacenamiento y relacionados con el almacenamiento admitidos. Un objeto de diccionario consta de una lista de atributos, que puede ser de tipo verificado. Un objeto de diccionario con valores completos describe una instancia de objeto de un tipo. Además, los atributos de referencia describen la relación del objeto con el entorno; por ejemplo, un objeto de diccionario de volumen tiene muchos atributos, como nombre, tamaño_mb y volumen_garantizado. Además, el objeto de diccionario de volúmenes incluye referencias al agregado y a la matriz que contiene el volumen en forma de array_id y aggregate_id

La tabla de caché de un objeto es una base de datos que contiene algunos o todos los atributos de la entrada de diccionario marcados para el almacenamiento en caché. Para que una entrada de diccionario incluya una tabla de caché, al menos uno de los atributos de la entrada de diccionario debe marcarse para almacenar en caché. Las entradas del diccionario incluyen claves naturales, que son identificadores únicos de los objetos; por ejemplo, los volúmenes de 7-Mode se identifican de forma única por su nombre y la dirección IP de la matriz que los contiene. Los qtrees se identifican mediante el nombre del qtree, el nombre del volumen y la dirección IP de la cabina. Debe identificar los atributos del diccionario que forman parte de las claves naturales de la entrada del diccionario al crear entradas del diccionario.

Cómo funcionan los comandos

Los comandos OnCommand Workflow Automation son los bloques de ejecución para los flujos de trabajo. Puede usar un comando para cada paso del flujo de trabajo.

Los comandos WFA se escriben utilizando scripts PowerShell y Perl. Los comandos de PowerShell utilizan el kit de herramientas de PowerShell de Data ONTAP y PowerCLI de VMware, si el paquete está instalado. Los comandos Perl utilizan la distribución Perl y los módulos Perl instalados en el servidor WFA. Si incluye varios idiomas de scripting en un comando, como PowerShell y Perl, la WFA elige el script adecuado basado en el sistema operativo en el que se instala y el orden de idioma preferido que haya especificado en el menú de configuración de WFA.

Las secuencias de comandos de WFA incluyen varios parámetros. Estos parámetros pueden asignarse a atributos de entrada de diccionario.

Tenga en cuenta que cada comando de WFA puede incluir varios comandos de la Data ONTAP.

Algunos de los comandos de WFA se conocen como comandos de espera porque pueden esperar operaciones de ejecución prolongada y sondear periódicamente --por ejemplo, el comando **Espere para múltiples movimientos de volumen**. El intervalo de espera en el que se ejecuta el comando Polling se puede configurar para comprobar si la operación se ha completado.

WFA inicia un comando WFA mientras el flujo de trabajo está en fase de ejecución. WFA ejecuta los comandos en serie, en orden de izquierda a derecha y de arriba a abajo. La planificación del flujo de trabajo confirma la disponibilidad y validez de los parámetros que se suministran al comando. El servidor WFA proporciona todos los parámetros necesarios para los comandos antes de ejecutarse los comandos.

Los parámetros de los comandos se finalizan durante la planificación del flujo de trabajo. A continuación, el flujo de trabajo pasa estos parámetros a los comandos durante el tiempo de ejecución. Los comandos no pueden devolver los parámetros al flujo de trabajo. Sin embargo, si desea intercambiar información que se obtiene durante el tiempo de ejecución entre comandos de un flujo de trabajo, puede utilizar los cmdlets de WFA PowerShell designados o las funciones Perl.

LOS comandos WFA PowerShell no utilizan el indicador -ErrorAction stop para los cmdlets de PowerShell; por lo tanto, las ejecuciones del flujo de trabajo continúan incluso cuando fallan los cmdlets debido a un error. Si desea que la Marca -ErrorAction stop se incluya en un comando específico, es posible clonar el comando y modificar el script de PowerShell para añadir el indicador.

A continuación se enumeran los cmdlets de PowerShell y las funciones Perl que se incluyen en WFA para permitir el intercambio de información entre comandos:

Cmdlets de PowerShell	Funciones Perl
Add-WfahWorkflowParameter	AddWfahWorkflowParameter
Get-WfahWorkflowParameter	GetWfahWorkflowParameter

Los parámetros agregados por los cmdlets o funciones de "add" a un comando pueden recuperarse mediante un comando que se ejecuta posteriormente y que utiliza los cmdlets o funciones de "get". Por ejemplo, en un comando de WFA de PowerShell, puede utilizar lo siguiente en el código para añadir un parámetro llamado volumeld: `Add-WfaWorkflowParameter -Name "VolumeUUID" -Value "12345"`
`-AddAsReturnParameter $true`. A continuación, puede utilizar lo siguiente en un comando posterior para recuperar el valor de volumeld: `$volumeId = Get-WfaWorkflowParameter -Name volumeId`.

Los comandos WFA pueden consultar la base de datos de WFA y obtener el resultado necesario. Esto permite construir un comando sin utilizar filtros ni buscadores. Puede utilizar las siguientes funciones para consultar la base de datos:

Cmdlet de PowerShell	Función Perl
Invoke-MySQLQuery por ejemplo: <code>Invoke-MySQLQuery -Query "SELECT cluster.name AS 'Cluster Name' FROM cm_storage.cluster"</code>	InvokeMySQLQuery

Información relacionada

[Crear un filtro](#)

[Crear un buscador](#)

[Crear un comando](#)

Qué filtros son

Puede usar filtros de WFA en sus flujos de trabajo para seleccionar los recursos necesarios.

Un filtro de WFA es una consulta basada en SQL que funciona en la base de datos WFA. Cada filtro devuelve una lista de elementos de un tipo de diccionario específico. Los elementos devueltos se basan en los criterios de selección especificados en la consulta SQL. Debe tener en cuenta la sintaxis SQL para crear o editar un filtro.

Lo que son los buscadores

Un buscador es una combinación de uno o más filtros que se utilizan conjuntamente para identificar resultados comunes. Puede usar un buscador en los flujos de trabajo para seleccionar los recursos necesarios para la ejecución del flujo de trabajo.

Los finders pueden aplicar una orden de clasificación para diferenciar los resultados aplicables. Los buscadores devuelven el mejor recurso en función de los criterios de selección y la ordenación.

Los finders devuelven un resultado o ningún resultado; por tanto, pueden utilizarse para verificar la existencia de determinados elementos de almacenamiento. Sin embargo, cuando se utiliza un buscador como parte de una definición de fila de repetición, los conjuntos de resultados se utilizan para formar la lista de miembros del grupo. Los filtros que se utilizan en los codificadores devuelven la clave natural del tipo de diccionario, como mínimo, pero pueden devolver campos adicionales cuyo valor se puede hacer referencia a ellos. Es posible que se aplique un orden de clasificación a cualquier campo devuelto de la consulta SQL de un filtro.

Puede probar los resultados de un buscador. A la hora de probar un buscador, puede ver los resultados comunes de todos los filtros de WFA, donde el resultado efectivo del buscador se resalta en los resultados. Si utiliza un buscador en un flujo de trabajo, puede crear un mensaje de error personalizado para transmitir información significativa al operador de almacenamiento.

Qué funciones son

Puede utilizar una función en los flujos de trabajo para una tarea compleja que debe completarse durante la fase de planificación del flujo de trabajo.

Puede escribir funciones mediante el MVFLEX Expression Language (MVEL). Puede utilizar funciones para unir la lógica más utilizada así como la lógica más compleja en una función con nombre y volver a utilizarla como valores para los parámetros de comando o los parámetros de filtro. Puede escribir una función una vez y utilizarla en todos los flujos de trabajo. Puede utilizar funciones para manejar tareas y tareas repetitivas que pueden ser complejas, como definir una convención de nomenclatura compleja.

Las funciones pueden utilizar otras funciones durante su ejecución.

Qué esquemas son

Un esquema representa el modelo de datos de un sistema. Un modelo de datos es una

colección de entradas de diccionario. Puede definir una combinación y, a continuación, definir un tipo de origen de datos. El origen de datos define cómo se adquieren los datos y se llena el esquema. Por ejemplo, un esquema de vc adquiere datos sobre su entorno virtual, como máquinas virtuales, hosts y almacenes de datos.

Las combinaciones también se pueden rellenar directamente con datos a través de flujos de trabajo personalizados para resolver problemas específicos.

Las entradas del diccionario están asociadas a un esquema existente cuando se crean las entradas del diccionario. Las entradas de diccionario también están asociadas con consultas en caché, y las consultas en caché incluyen consultas SQL.

Los esquemas pueden adquirir datos utilizando el tipo de origen de datos basado en secuencias de comandos o el tipo de origen de datos SQL. Los scripts se definen al crear el tipo de origen de datos y las consultas SQL se definen en las consultas de caché.

WFA incluye los siguientes esquemas:

- **7-Mode (almacenamiento)**

Esquema para adquirir datos a través de Active IQ Unified Manager de Data ONTAP operando en 7-Mode.

- **Data ONTAP en cluster (cm_Storage)**

Esquema para adquirir datos a través de Active IQ Unified Manager de Clustered Data ONTAP.

- **Rendimiento de 7-Mode (rendimiento)**

Esquema para adquirir datos de rendimiento de Data ONTAP operando en 7-Mode a través de Performance Advisor.

- **Rendimiento de Data ONTAP en cluster (cm_rendimiento)**

Esquema para adquirir los datos de rendimiento de Clustered Data ONTAP a través de Performance Advisor.

- **VMware vCenter (vc)**

Esquema para adquirir datos de VMware vCenter.

- **Zona de juegos (patio)**

Esquema que puede rellenar directamente con datos.

Qué tipos de sistemas remotos son

OnCommand Workflow Automation (WFA) se comunica con los tipos de sistema remoto. Un tipo de sistema remoto especifica el tipo de sistemas remotos con los que WFA se puede comunicar. Puede configurar los tipos de sistema remoto en WFA. Por ejemplo, el sistema Data ONTAP se puede configurar como un tipo de sistema remoto.

Un tipo de sistema remoto tiene los siguientes atributos:

- Nombre
- Descripción
- Versión
- Protocolo
- Puerto
- Tiempo de espera

Puede tener un script Perl para cada tipo de sistema remoto para validar las credenciales del sistema remoto. Puede almacenar las credenciales de los sistemas remotos configurados en WFA. Puede agregar o editar un nuevo tipo de sistema remoto personalizado. También puede clonar un tipo de sistema remoto existente. Sólo puede eliminar un tipo de sistema remoto si no hay ningún sistema asociado.

Cómo se utilizan las plantillas

Puede utilizar plantillas de WFA en sus flujos de trabajo como referencia o para adherirse a políticas de uso.

Una plantilla de WFA actúa como guía de la definición de un objeto. Puede definir una plantilla incluyendo las propiedades de un objeto y los valores de las propiedades del objeto. A continuación, puede utilizar la plantilla para rellenar las propiedades de una definición de objeto en los flujos de trabajo.

Cuando utiliza una plantilla, no puede editar los campos que incluyen los valores obtenidos de la plantilla. Por lo tanto, puede utilizar plantillas para configurar políticas de uso y la creación de objetos. Si quita la asociación de una plantilla con el flujo de trabajo después de haber aplicado la plantilla, los valores rellenados de la plantilla se conservan, pero puede editar los campos.

Cómo se utilizan las categorías

Puede clasificar los flujos de trabajo para organizar mejor los flujos de trabajo y aplicar la funcionalidad de control de acceso en los flujos de trabajo.

Puede clasificar los flujos de trabajo de forma que aparezcan en grupos específicos en el portal de WFA. También puede aplicar la funcionalidad de control de acceso en categorías de flujos de trabajo. Por ejemplo, solo puede permitir que determinados operadores o aprobadores de almacenamiento vean determinadas categorías de flujos de trabajo. Los operadores o autorizadores de almacenamiento sólo pueden ejecutar los flujos de trabajo de la categoría para los que se les han concedido derechos de acceso.

Los grupos de Active Directory también se pueden utilizar para controlar el acceso a las categorías.

Cómo funciona el control de versiones de entidades

Las entidades de OnCommand Workflow Automation (WFA), como comandos y flujos de trabajo, tienen versiones. Puede utilizar los números de versión para gestionar fácilmente los cambios en las entidades de WFA.

Cada entidad de WFA incluye un número de versión en `major.minor.revision` formato: por ejemplo, 1.1.20. Puede incluir hasta tres dígitos en cada parte del número de versión.

Antes de modificar el número de versión de una entidad de WFA, debe tener en cuenta las siguientes reglas:

- Los números de versión no se pueden cambiar de la versión actual a una versión anterior.

- Cada parte de la versión debe ser un número del 0 al 999.
- De forma predeterminada, las nuevas entidades de WFA cuentan con las versiones 1.0.0.
- El número de versión de una entidad se conserva al clonar o utilizar **Guardar como** para guardar una copia de la entidad.
- No pueden existir varias versiones de una entidad en una instalación de WFA.

Cuando actualiza la versión de una entidad WFA, la versión de su entidad principal inmediata se actualiza automáticamente. Por ejemplo, la actualización de la versión del comando **Crear volumen** actualiza el flujo de trabajo **Crear un volumen NFS**, porque el flujo de trabajo **Crear un volumen NFS** es una entidad principal inmediata del comando **Crear volumen**. La actualización automática a las versiones se aplica de la siguiente manera:

- Al modificar la versión principal de una entidad, se actualiza la versión secundaria de sus entidades principales inmediatas.
- Al modificar la versión secundaria de una entidad, se actualiza la versión de revisión de sus entidades principales inmediatas.
- La modificación de la versión de revisión de una entidad no actualiza ninguna parte de la versión de sus entidades principales inmediatas.

En la siguiente tabla se enumeran las entidades de WFA y sus entidades principales inmediatas:

Entidad	Entidad padre inmediata
Consulta de caché	<ul style="list-style-type: none"> • Tipo de origen de datos
Plantilla	<ul style="list-style-type: none"> • Flujo de trabajo
Función	<ul style="list-style-type: none"> • Flujo de trabajo • Plantilla <div>  Si una función contiene caracteres especiales o mixtos, es posible que la versión de sus entidades principales inmediatas no se actualice. </div>
Diccionario	<ul style="list-style-type: none"> • Plantilla • Filtro • Consulta de caché • Comando • Tipos de origen de datos que utilizan método de script
Comando	<ul style="list-style-type: none"> • Flujo de trabajo
Filtro	<ul style="list-style-type: none"> • Buscador • Flujo de trabajo

Entidad	Entidad padre inmediata
Buscador	<ul style="list-style-type: none"> • Flujo de trabajo
Tipo de origen de datos	Ninguno
Flujo de trabajo	Ninguno

Puede buscar una entidad en WFA usando las partes del número de versión o el número de versión completo.

Si elimina una entidad padre, las entidades secundarias se conservan y su versión no se actualiza para la eliminación.

Cómo funciona el control de versiones al importar entidades

Si importa entidades de versiones anteriores a Workflow Automation 2.2, las entidades se versionan como 1.0.0, de forma predeterminada. Si la entidad importada ya está presente en el servidor de WFA, la entidad existente se sobrescribe con la entidad importada.

A continuación se muestran los posibles cambios en las entidades de WFA durante una importación:

- Actualización de entidades

Las entidades se sustituyen por una versión posterior.

- Reversión de entidades

Las entidades se sustituyen por una versión anterior.



Cuando realiza una reversión de una entidad, se actualiza la versión de sus entidades principales inmediatas.

- Importación de nuevas entidades



No se pueden importar entidades selectivamente desde un .dar archivo.

Si se importa una versión posterior de una entidad, se actualiza la versión de sus entidades principales inmediatas.

Si hay varias entidades secundarias en la entidad padre importada, sólo se aplica el grado de cambio más alto (mayor, menor o revisión) a las entidades secundarias a la entidad padre. Los siguientes ejemplos explican cómo funciona esta regla:

- Para una entidad padre importada, si hay una entidad secundaria con un cambio menor y otra entidad secundaria con un cambio de revisión, el cambio menor se aplica a la entidad padre.

Se incrementa la parte de revisión de la versión padre.

- Para una entidad padre importada, si hay una entidad secundaria con un cambio principal y otra entidad secundaria con un cambio menor, el cambio principal se aplica a la entidad padre.

Se incrementa la parte secundaria de la versión padre.

Ejemplo de cómo afectan las versiones de entidades secundarias importadas a la versión del padre

Tenga en cuenta el siguiente flujo de trabajo de WFA: «'Crear volumen y exportar mediante NFS - personalizado'» 1.0.0.

Los comandos existentes que se incluyen en el flujo de trabajo son los siguientes:

- «Crear Política de exportación - personalizada»» 1.0.0
- "Crear volumen - personalizado" 1.0.0

Los comandos incluidos en la `.dar` el archivo, que se va a importar, es el siguiente:

- «Crear Política de exportación - personalizada»» 1.1.0
- "Crear volumen - personalizado" 2.0.0

Al importar esto `.dar` File, la versión secundaria del flujo de trabajo "Create Volume y export mediante NFS - Custom" se incrementa a 1.1.0.

Qué base de datos de juegos

La base de datos de juegos infantil es una base de datos de MySQL, que está incluida en la instalación del servidor de Workflow Automation (WFA). Puede agregar tablas a la base de datos de juegos para incluir información, que puede ser utilizada por filtros y consultas SQL para entradas de usuario.

La base de datos de juegos es un esquema al que no se puede acceder a través del portal web de WFA. Puede usar un cliente MySQL, como SQLyog, Toad para MySQL, y MySQL Workbench o una interfaz de línea de comandos (CLI), para acceder a la base de datos.

Debe utilizar las siguientes credenciales para acceder a la base de datos de juegos:

- Nombre de usuario: wfa
- Contraseña: Wfa123

Las credenciales proporcionan acceso completo a la base de datos de juegos y acceso de solo lectura a otros esquemas definidos en la base de datos MySQL de WFA. Puede crear las tablas necesarias en la base de datos de juegos.

Puede agregar las etiquetas o metadatos que utiliza para objetos de almacenamiento del entorno a una tabla de la base de datos de juegos. A continuación, las etiquetas o los metadatos se pueden usar junto con la información de otras tablas de caché de WFA mediante filtros de WFA y consultas de entrada de usuario.

Por ejemplo, puede utilizar la base de datos de juegos para los siguientes casos de uso:

- Etiquetado de agregados con el nombre de unidad de negocio (unidad de negocio) y asignación de volúmenes al bus en función de estas etiquetas
- Etiquetado de unidades vFiler con nombres de unidades de negocio
- Adición de detalles geográficos o de ubicación a objetos de almacenamiento
- Definición del acceso de administradores de bases de datos a bases de datos

Por ejemplo, si utiliza el nombre de la unidad de negocio como etiqueta para los objetos de almacenamiento,

como agregados y unidades vFiler, puede crear una tabla en la base de datos de juegos que incluya el nombre de la unidad de negocio. A continuación, el nombre de la unidad de negocio puede ser utilizado por filtros y consultas de entrada de usuario para sus flujos de trabajo.

A continuación se muestra un ejemplo de tabla de base de datos de playground (playground.volume_bu):

array_ip	volume_name	RES
10.225.126.23	data_11	Marketing
10.225.126.28	arch_11	RR. HH

A continuación se muestra un ejemplo de consulta SQL que se puede utilizar para filtrar volúmenes por unidad de negocio:

```
SELECT
    vol.name,
    array.ip AS 'array.ip'
FROM
    storage.volume AS vol,
    storage.array AS array,
    playground.volume_bu AS vol_bu
WHERE
    vol.array_id = array.id
    AND array.ip = vol_bu.array_ip
    AND vol.name = vol_bu.volume_name
    AND vol_bu.bu = '{$bu}'
```

Información relacionada

["SQLyog"](#)

["MySQL Workbench"](#)

["Toad para MySQL"](#)

Gestión de flujos de trabajo

Puede personalizar flujos de trabajo predefinidos o crear nuevos flujos de trabajo como parte de la gestión de los flujos de trabajo. También debe comprender los conceptos relevantes antes de empezar a gestionar los flujos de trabajo.

Personalizar flujos de trabajo predefinidos

Puede personalizar un flujo de trabajo predefinido de Workflow Automation (WFA) si no hay ningún flujo de trabajo predefinido que sea adecuado para sus necesidades.

Lo que necesitará


Debe haber identificado las modificaciones necesarias para el flujo de trabajo predefinido.

Acerca de esta tarea

Las preguntas y la solicitud de soporte de los siguientes deben dirigirse a la comunidad de WFA:

- Cualquier contenido descargado desde la comunidad de WFA
- Contenido personalizado de WFA que ha creado
- WFA contenido que ha modificado

Pasos

1. Haga clic en **Diseño de flujo de trabajo > flujos de trabajo**.
2. Seleccione el flujo de trabajo predefinido que se ajuste a sus necesidades y, a continuación, haga clic en  en la barra de herramientas.
3. En el diseñador de flujos de trabajo, realice los cambios necesarios en las fichas correspondientes, como editar la descripción, agregar o eliminar un comando, modificar los detalles del comando y modificar los datos introducidos por el usuario.
4. Haga clic en **Vista previa**, introduzca las entradas de usuario necesarias para obtener una vista previa de la ejecución del flujo de trabajo y, a continuación, haga clic en **Vista previa** para ver los detalles de planificación del flujo de trabajo.
5. Haga clic en **Aceptar** para cerrar la ventana de vista previa.
6. Haga clic en **Guardar**.

Después de terminar

Puede probar el flujo de trabajo que modificó en su entorno de prueba y marcar el flujo de trabajo como listo para producción.


Personalizar el flujo de trabajo de creación de un volumen y recurso compartido CIFS


Puede personalizar sus flujos de trabajo en función de sus requisitos. Por ejemplo, puede modificar el flujo de trabajo predefinido *Create a Volume y un flujo de trabajo CIFS Share* para incluir la deduplicación y la compresión.

Acerca de esta tarea

La personalización e ilustraciones de esta tarea son ejemplos. Puede modificar los flujos de trabajo de WFA en función de sus requisitos.

Pasos

1. Haga clic en **Diseño de flujo de trabajo > flujos de trabajo**.
2. Seleccione el flujo de trabajo **Crear un volumen y un recurso compartido CIFS** y, a continuación, haga clic en  en la barra de herramientas.
3. Haga clic en la ficha **Detalles** y edite la descripción del flujo de trabajo en el campo **Nombre del flujo de trabajo**.
4. Haga clic en la ficha **flujo de trabajo**, expanda el esquema **almacenamiento** y, a continuación, arrastre y suelte el comando **Configurar deduplicación y compresión** entre los comandos **Crear volumen** y **Crear recurso compartido CIFS**.

5. Coloque el cursor del ratón debajo del comando * Setup deduplication and compression* en la primera fila y, a continuación, haga clic en .
6. En la ficha **volumen** del cuadro de diálogo **parámetros para la deduplicación y compresión**, seleccione la opción **utilizando un volumen** definido previamente y, a continuación, seleccione la opción **share_volume** en el campo **define Volume**, que es la variable de objeto Volume creada por el comando **Create Volume** del flujo de trabajo.
7. Haga clic en la ficha **otros parámetros** y realice los siguientes pasos:
 - a. Seleccione **true** en el campo **StartNow**.
 - b. Seleccione **Inline** en el campo **compresión**.
 - c. Introduzca 'sun-sat@1' Expresión en el campo * Schedule*, que programa la deduplicación y la compresión en todos los días de la semana a la 1 a.m.
8. Haga clic en **Aceptar**.
9. Haga clic en **Vista previa** para asegurarse de que la planificación del flujo de trabajo se ha completado correctamente y, a continuación, haga clic en **Aceptar**.
10. Haga clic en **Guardar**.

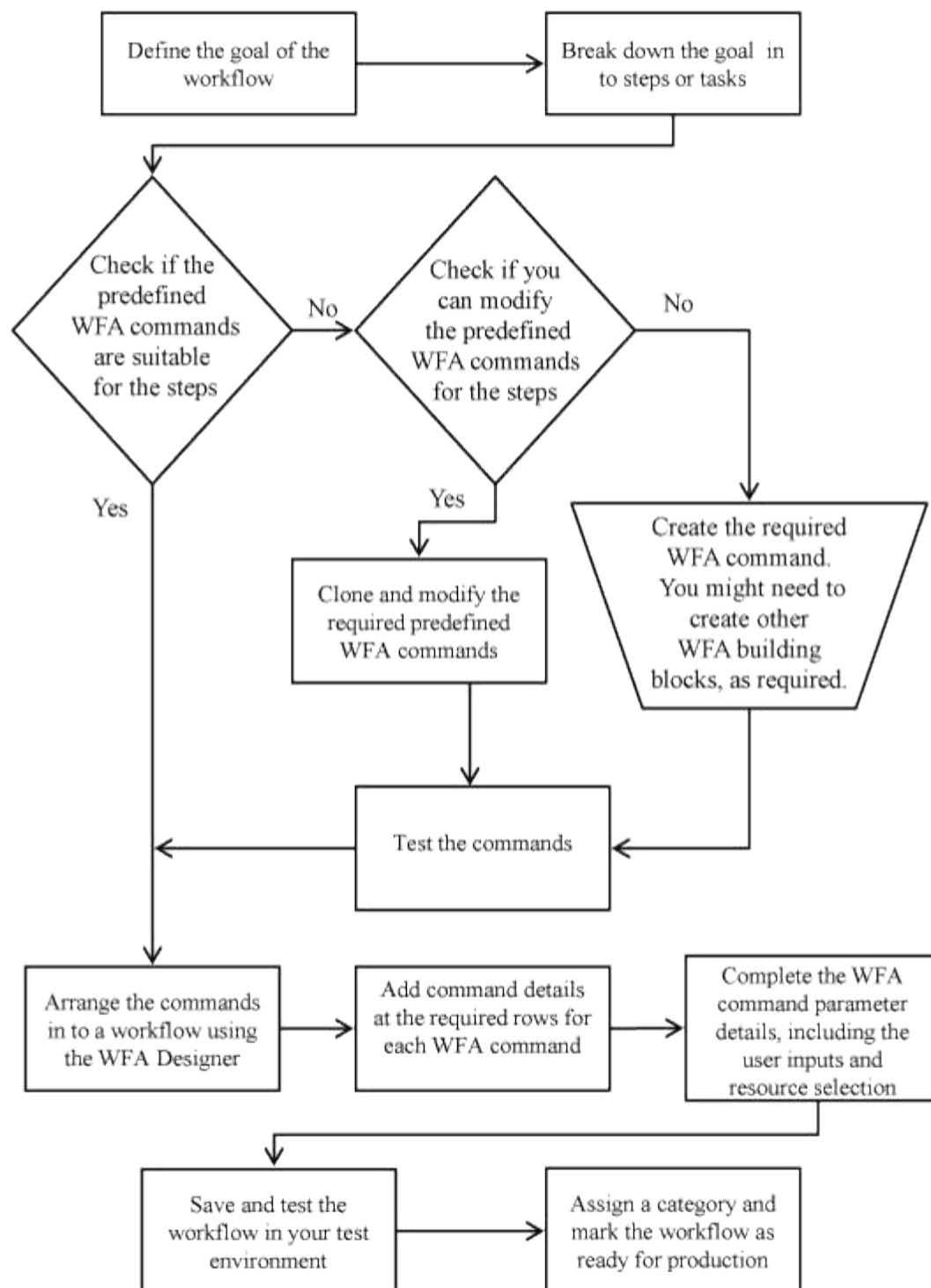
Crear flujos de trabajo

Si los flujos de trabajo predefinidos no coinciden con sus requisitos, puede crear el flujo de trabajo necesario. Antes de crear los flujos de trabajo, debe comprender las funcionalidades disponibles en el diseñador de WFA y crear una lista de comprobación del flujo de trabajo.

Tareas relacionadas con la creación de flujos de trabajo

La creación de flujos de trabajo de automatización del almacenamiento en OnCommand Workflow Automation (WFA) incluye definir los pasos que debe realizar un flujo de trabajo y crear el flujo de trabajo mediante los elementos básicos de WFA, como comandos, buscadores, filtros y entradas del diccionario.

El siguiente diagrama de flujo ilustra el proceso de creación del flujo de trabajo:



Cómo definir los flujos de trabajo

Debe dividir el objetivo de un flujo de trabajo en los pasos que debe ejecutar el flujo de trabajo. A continuación, puede organizar los pasos para completar el flujo de trabajo.

Un flujo de trabajo es un algoritmo que incluye una serie de pasos necesarios para completar un proceso completo. El alcance del proceso puede variar en función del objetivo del flujo de trabajo. El objetivo de un flujo de trabajo puede definirse para manejar solo las operaciones de almacenamiento o los procesos más complejos como gestionar las redes, la virtualización, los sistemas DE TI y otras aplicaciones como parte de un solo proceso. Los flujos de trabajo de OnCommand Workflow Automation (WFA) están diseñados por

arquitectos de almacenamiento y son ejecutados por operadores de almacenamiento.

Para definir el flujo de trabajo, es necesario dividir el objetivo del flujo de trabajo en una serie de pasos, por ejemplo, crear un volumen NFS incluye los siguientes pasos:

1. Crear un objeto de volumen
2. Crear una nueva política de exportación y asociar la política al volumen

Puede usar un comando de WFA o un flujo de trabajo para cada paso del flujo de trabajo. WFA incluye comandos y flujos de trabajo predefinidos, que se basan en casos de uso de almacenamiento comunes. Si no encuentra un comando o un flujo de trabajo predefinidos que puedan utilizarse para un paso en particular, puede realizar una de las siguientes acciones:

- Elija un comando o un flujo de trabajo predefinidos que coincidan con el paso, y luego clone y modifique el comando o el flujo de trabajo predefinidos de acuerdo con sus requisitos.
- Crear un nuevo comando o flujo de trabajo.

A continuación, puede organizar los comandos o flujos de trabajo en un flujo de trabajo nuevo para crear el flujo de trabajo que logra su objetivo.

Al principio de la ejecución del flujo de trabajo, WFA planifica la ejecución y comprueba que el flujo de trabajo se puede ejecutar mediante la entrada en el flujo de trabajo y los comandos. Al planificar el flujo de trabajo, toda la selección de recursos y la entrada del usuario se resuelven para crear un plan de ejecución. Una vez completada la planificación, WFA ejecuta el plan de ejecución, que consiste en un conjunto de comandos WFA con los parámetros aplicables.

Cómo se definen las entradas del usuario

Las entradas del usuario del OnCommand Workflow Automation (WFA) son opciones de entrada de datos que están disponibles durante la ejecución de los flujos de trabajo. Para mejorar la flexibilidad y la facilidad de uso de los flujos de trabajo, debe definir los parámetros de entrada de usuario.

Las entradas del usuario se muestran como campos de entrada, que se pueden rellenar con datos relevantes al previsualizar o ejecutar flujos de trabajo. Puede crear un campo de entrada de usuario al especificar los detalles del comando en un flujo de trabajo mediante la asignación previa de una etiqueta o variable con el signo de dólar (\$). Por ejemplo, \$VolumeName crea un campo de entrada de usuario Volume Name. WFA rellena automáticamente la ficha Entradas de usuario de la ventana Workflow <workflow name> con las etiquetas de entrada de usuario que ha creado. También puede definir el tipo de entrada de usuario y personalizar los campos de entrada modificando los atributos de entrada de usuario, como tipo, nombre para mostrar, valores predeterminados y valores de validación.

Opciones de tipo de entrada de usuario

- **Cadena**

Puede utilizar una expresión regular para valores válidos, por ejemplo, a*.

Cadenas, como 0d y.. 0f, se evalúan como números similares a 0d evaluado como 0 del tipo double.

- **Número**

Puede definir un intervalo numérico que se puede seleccionar, por ejemplo, 1 a 15.

- **Enum**

Puede crear valores de enumeración que se pueden seleccionar al rellenar el campo de entrada del usuario mediante el tipo de enumeración. Opcionalmente, puede bloquear los valores de enumeración que ha creado para garantizar que sólo se seleccionan los valores que ha creado para la entrada del usuario.

- **Consulta**

Puede seleccionar el tipo de consulta cuando desee que la entrada de usuario se seleccione de los valores disponibles en la caché de WFA. Por ejemplo, puede utilizar la siguiente consulta para completar automáticamente los campos de entrada de usuario con la dirección IP y los valores de nombre de la caché WFA: **SELECT ip, name FROM storage.array**. Opcionalmente, puede bloquear los valores recuperados por una consulta para que sólo se seleccionen los resultados reintentados por la consulta.

- **Consulta (selección múltiple)**

El tipo de consulta (selección múltiple), que es similar al tipo de consulta, permite seleccionar varios valores durante la ejecución del flujo de trabajo. Por ejemplo, los usuarios pueden seleccionar varios volúmenes o un volumen junto con sus recursos compartidos y exportaciones. Puede permitir a los usuarios seleccionar varias filas o restringir la selección a una sola fila. Al seleccionar una fila, se seleccionan los valores de todas las columnas de la fila seleccionada.

Puede utilizar las siguientes funciones cuando utilice el tipo de consulta (selección múltiple) de entrada de usuario:

- `GetSize`
- `GetValueAt`
- `GetValueAt2D`
- `GetValueFrom2DByRowKey`

- **Booleano**

Puede utilizar el tipo booleano para mostrar una casilla de verificación en el cuadro de diálogo de entrada de usuario. Debe utilizar el tipo booleano para las entradas de usuario que tengan como valores posibles `"true"` y `"false"`.

- **Tabla**

Puede utilizar el tipo de tabla de entrada de usuario para especificar los encabezados de columna de una tabla que se pueden utilizar para introducir varios valores durante la ejecución del flujo de trabajo. Por ejemplo, una tabla que puede utilizarse para especificar una lista con nombres de nodo y nombres de puerto. También puede especificar uno de los siguientes tipos de entrada de usuario para los encabezados de columna para validar los valores introducidos durante el tiempo de ejecución:

- Cadena
- Número
- Enum
- Booleano
- Consulta `String` es el tipo de entrada de usuario predeterminado para los encabezados de columna. Debe hacer doble clic en la columna Tipo para especificar un tipo de entrada de usuario diferente.

Puede abrir el flujo de trabajo Crear política y reglas de SnapMirror en Designer para ver cómo se utilizan

los tipos de entrada de usuario en la entrada de usuario "MirrorSnapPolicyRule".

Puede utilizar las siguientes funciones cuando utilice el tipo de tabla de entrada de usuario:

- GetSize
- GetValueAt
- GetValueAt2D
- GetValueFrom2DByRowKey puede abrir el flujo de trabajo **Crear y configurar una máquina virtual de almacenamiento con Infinite Volume** en el Diseñador para ver cómo se utiliza el tipo de tabla.
 - **Contraseña**

Puede utilizar el tipo de contraseña para las entradas de usuario que están destinadas a introducir contraseñas. La contraseña introducida por el usuario se cifra y se muestra como una secuencia de caracteres con asterisco en la aplicación WFA y en los archivos de registro. Puede utilizar las siguientes funciones para descifrar la contraseña, que puede utilizar el comando:

- Para comandos Perl: WFAUtil::getWfInputPassword (\$password)
- Para comandos PowerShell: Get-WfInputPassword -EncryptedPassword \$password

Aquí, \$password es la contraseña cifrada que pasa WFA al comando.

- **Diccionario**

Puede agregar los datos de tabla para la entrada de diccionario seleccionada. El atributo de entrada del diccionario selecciona el atributo que se va a devolver. Puede seleccionar un único valor o varios mientras ejecuta el flujo de trabajo. Por ejemplo, puede seleccionar un único volumen o varios volúmenes. De forma predeterminada, se seleccionan valores únicos. También puede seleccionar Reglas para el filtrado. Una regla consta de un atributo de entrada de diccionario, un operador y un valor. El atributo también puede incluir atributos de sus referencias.

Por ejemplo, puede especificar una regla para los agregados enumerando todos los agregados con nombre comenzando por la cadena «'aggr'» y teniendo un tamaño disponible superior a 5 GB. La primera regla del grupo es el atributo name, con el operador starts-with, y el valor aggr. La segunda regla para el mismo grupo es el atributo available_size_mb, con el operador > y el valor 5000.

En la siguiente tabla se enumeran las opciones que puede aplicar a los tipos de entrada de usuario:

Opción	Descripción
Validación	<p>Puede validar el tipo de entradas de usuario de modo que sólo los usuarios introduzcan valores válidos:</p> <ul style="list-style-type: none"> • Los tipos de cadena y número de entrada de usuario se pueden validar con los valores introducidos durante el tiempo de ejecución del flujo de trabajo. • El tipo de cadena también se puede validar con una expresión regular. • El tipo de número es un campo de coma flotante numérico y se puede validar utilizando un intervalo numérico especificado.
Valores de bloqueo	Puede bloquear los valores de los tipos de consulta y enumeración para evitar que el usuario sobrescriba los valores de lista desplegable y para activar la selección sólo de los valores mostrados.
Marcar como obligatorio	Puede marcar las entradas de usuario como obligatorias para que los usuarios deben introducir ciertas entradas de usuario a fin de continuar con la ejecución del flujo de trabajo.
Agrupación	Puede agrupar las entradas de usuario relacionadas y proporcionar un nombre para el grupo de entrada de usuario. Los grupos se pueden expandir y contraer en el cuadro de diálogo de entrada de usuario. Puede seleccionar un grupo que se debe expandir de forma predeterminada.
Aplicación de condiciones	Con la capacidad de entrada de usuario condicional, puede establecer el valor de una entrada de usuario en función del valor que se introduzca para otra entrada de usuario. Por ejemplo, en un flujo de trabajo que configura el protocolo NAS, puede especificar la entrada de usuario necesaria para el protocolo como NFS para activar la entrada del usuario "Leer/escribir listas de host".

Cómo se asignan los parámetros de comandos

Los parámetros de los comandos de Workflow Automation (WFA) se asignan a atributos específicos y referencias de entrada de diccionario en función de ciertas reglas. Debe estar al tanto de las reglas para asignar parámetros de comandos al crear o editar un comando de WFA.

La asignación de parámetros de comandos define cómo se definen los detalles de los comandos en los flujos de trabajo. Los parámetros de comando asignados de un comando se muestran en las pestañas cuando

especifica los detalles del comando para los comandos de los flujos de trabajo. Las fichas se nombran en función del nombre de grupo especificado en la columna Nombre de objeto de la ficha asignación de parámetros. Los parámetros que no están asignados se muestran en la ficha otros parámetros cuando se especifican los detalles del comando en los flujos de trabajo.

Las reglas para la asignación de parámetros de comandos se aplican en función de la categoría de comandos y de cómo se representan los comandos en el editor de flujos de trabajo.

Las siguientes son las categorías de comandos:

- Comandos que crean objetos
- Comandos que actualizan objetos
- Comandos que quitan objetos
- Comandos que tratan objetos padre e hijo opcionales
- Comandos que actualizan asociaciones entre objetos

Las reglas se enumeran a continuación para cada categoría:

Todas las categorías de comandos

Cuando se asigna un parámetro de comando, debe utilizar la ruta natural en función de cómo se utilice el comando en los flujos de trabajo.

Los ejemplos siguientes muestran cómo puede definir un trazado natural:

- Para el parámetro ArrayIP, según el comando, debe utilizar el atributo aggregate.array.ip de la entrada del diccionario Volume y no el atributo array.ip.

Esto es importante cuando un flujo de trabajo crea un volumen y, a continuación, realiza un paso adicional con el volumen creado haciendo referencia a él. Los siguientes son ejemplos similares:

- Volume.aggregate.array.ip de la entrada del diccionario Qtree
- Volume.aggregate.array.ip de la entrada del diccionario LUN
- Para el clúster que se utiliza en los comandos, debe usar uno de los siguientes elementos:
 - Vserver.cluster.primary_address de la entrada de diccionario de volumen
 - Volume.vserver.cluster.primary_address de la entrada del diccionario Qtree

Comandos que crean objetos

Esta categoría de comandos se utiliza para una de las siguientes opciones:

- Buscar un objeto padre y definir nuevos objetos
- Buscar un objeto y crear el objeto si éste no existe

Debe utilizar las siguientes reglas de asignación de parámetros para esta categoría de comandos:

- Asigne los parámetros relevantes del objeto que se crea a la entrada de diccionario del objeto.
- Asigne el objeto primario a través de las referencias de la entrada de diccionario creada.
- Asegúrese de que el atributo relevante está presente en la entrada del diccionario al agregar un nuevo parámetro.

A continuación se muestran los escenarios de excepción de esta regla:

- Algunos objetos que se crean no tienen una entrada de diccionario correspondiente y sólo el objeto primario se asigna a la entrada de diccionario principal correspondiente (por ejemplo, el comando **Crear VIF**) en el que sólo se puede asignar una matriz a una entrada de diccionario de matriz.
- No es necesaria la asignación de parámetros

Por ejemplo, el parámetro ExecutionTimeout del comando **Crear o redimensionar agregado** es un parámetro sin asignar.

Los siguientes comandos certificados son ejemplos de esta categoría:

- Cree el volumen
- Cree la LUN

Comandos que actualizan objetos

Esta categoría de comandos se utiliza para buscar un objeto y actualizar los atributos.

Debe utilizar las siguientes reglas de asignación de parámetros para esta categoría de comandos:

- Asigne los objetos que se actualizan a la entrada del diccionario.
- No asigne los parámetros que se actualizan para el objeto.

Por ejemplo, en el comando **definir estado de volumen**, el parámetro volumen está asignado pero el nuevo estado está desasignado.

Comandos que quitan objetos

Esta categoría de comandos se utiliza para buscar un objeto y eliminarlo.

Debe asignar el objeto eliminado por el comando a su entrada de diccionario. Por ejemplo, en el comando **Quitar volumen**, el volumen que se va a eliminar se asigna a los atributos y referencias pertinentes de la entrada de diccionario de volumen.

Comandos que tratan objetos padre e hijo opcionales

Debe utilizar las siguientes reglas de asignación de parámetros para esta categoría de comandos:

- No asigne ningún parámetro obligatorio de un comando como referencia desde un parámetro opcional del comando.

Esta regla es más relevante cuando un comando trata los objetos secundarios opcionales de un objeto primario específico. En este caso, se debe asignar explícitamente el objeto hijo y el objeto padre. Por ejemplo, en el comando **Stop Deduplication Jobs**, el comando detiene un trabajo de deduplicación en ejecución en un volumen específico cuando se especifica junto con Array o en todos los volúmenes de la matriz indicada. En este caso, el parámetro array debe asignarse directamente a la entrada del diccionario array y no a Volume.Array porque el volumen es un parámetro opcional en este comando.

- Si existe una relación padre-hijo entre las entradas del diccionario en el nivel lógico pero no entre las instancias reales en un comando específico, esos objetos se deben asignar por separado.

Por ejemplo, en el comando **mover volumen**, volumen se mueve de su agregado principal actual a un

nuevo agregado de destino. Por lo tanto, los parámetros de volumen se asignan a una entrada de diccionario de volumen y los parámetros de agregado de destino se asignan por separado a la entrada de diccionario de agregado, pero no como `volume.aggregate.name`.

Comandos que actualizan asociaciones entre objetos

Para esta categoría de comandos, debe asignar tanto la asociación como los objetos a entradas de diccionario relevantes. Por ejemplo, en el comando `Add Volume to vFiler`, los parámetros `Volume` y `vFiler` se asignan a los atributos relevantes de las entradas del diccionario `Volume` y `vFiler`.

Cómo se definen las constantes

Puede crear y utilizar constantes para definir un valor, que se puede utilizar en un único flujo de trabajo. Las constantes se definen a nivel de flujo de trabajo.

Las constantes utilizadas en el flujo de trabajo y su valor se muestran en la ventana de supervisión del flujo de trabajo durante la planificación y la ejecución. Debe utilizar nombres únicos para las constantes.

Puede utilizar las siguientes convenciones de nomenclatura para definir constantes:

- En mayúsculas para la primera letra de cada palabra, sin guiones bajos ni espacios entre palabras

Todos los términos y abreviaturas deben utilizar mayúsculas, por ejemplo, `ActualVolumeSizeInMB`.

- En mayúsculas para todas las letras

Puede utilizar guiones bajos para separar palabras, por ejemplo, `AGGREGATE_USED_SPACE_THRESHOLD`.

Puede incluir los siguientes valores como valores para las constantes de flujo de trabajo:

- Números
- Cadenas
- Expresiones MVEL

Las expresiones se evalúan durante las fases de planificación y ejecución de los flujos de trabajo. En las expresiones, no debe hacer referencia a las variables definidas en un bucle.

- Entradas del usuario
- Variables

Cómo funciona la repetición de la fila

Un flujo de trabajo contiene comandos y detalles de comandos organizados en filas. Puede especificar los comandos de una fila que se van a repetir para un número fijo de iteraciones o un número dinámico de iteraciones basándose en los resultados de los criterios de búsqueda.

Los detalles del comando de una fila se pueden especificar para repetir un número determinado de veces o cuando se diseña el flujo de trabajo. El flujo de trabajo también se puede diseñar de forma que el número de veces que debe repetirse la fila se pueda especificar cuando el flujo de trabajo se ejecute o se programe para una ejecución. Puede especificar criterios de búsqueda para un objeto y los comandos de una fila se pueden

establecer para que se repitan tantas veces como los criterios de búsqueda devuelven los objetos. Las filas también se pueden configurar para que se repitan cuando se cumplan ciertas condiciones.

Variables de repetición de fila

Puede especificar variables en la lista de variables que se pueden manipular durante las iteraciones de fila. Para las variables, puede especificar un nombre, un valor con el que se inicializan las variables y una expresión MVFLEX Expression Language (MVEL) que se evalúa después de cada repetición de la fila.

En la siguiente ilustración se muestran las opciones de repetición de fila y un ejemplo de una variable de repetición de fila:

Row Repetition Details ?

Repeats*

Number of times

Number of Times*

Number of times

For every resource in a group

Index Variable*

Index1

Variables

Name	Initial Value	Expression
size_to_alloc	SIZE_MB	(int)size_to_allocated - getData()

Add

Remove

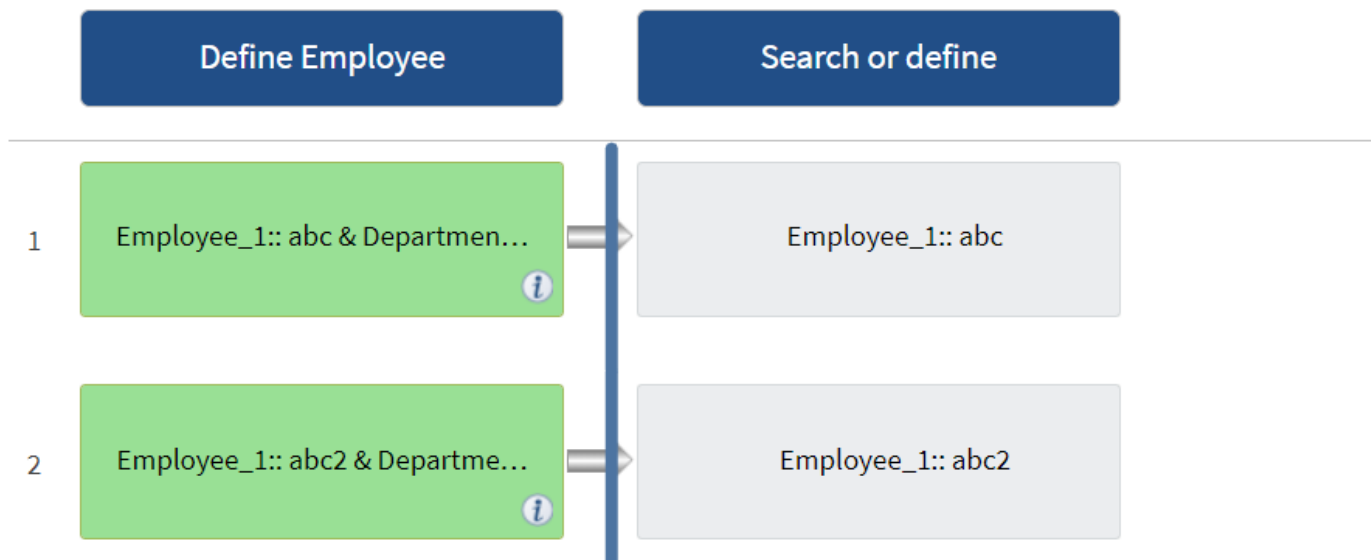
Cancel

OK

Repetición de fila con puntos de aprobación

Si ha especificado iteraciones de filas de repetición para comandos y ha incluido puntos de aprobación, se ejecutan todas las iteraciones de los comandos antes de un punto de aprobación. Después de aprobar el punto de aprobación, la ejecución de todas las iteraciones de los comandos sucesivos continúa hasta el siguiente punto de aprobación.

La siguiente ilustración muestra cómo se ejecutan las iteraciones de filas repetidas cuando se incluye un punto de aprobación en un flujo de trabajo:



Repetir ejemplos de fila en flujos de trabajo predefinidos

Puede abrir los siguientes flujos de trabajo predefinidos en el Diseñador para comprender cómo se utilizan las filas repetidas:

- Cree un volumen NFS de Clustered Data ONTAP
- Crear un almacén de datos NFS de VMware en el almacenamiento de Data ONTAP en clúster
- Establezca la relación entre clústeres
- Quite un volumen de Data ONTAP en clúster

Cómo funciona la selección de recursos

OnCommand Workflow Automation (WFA) utiliza algoritmos de búsqueda para seleccionar recursos de almacenamiento para la ejecución del flujo de trabajo. Debe comprender cómo funciona la selección de recursos para diseñar flujos de trabajo de forma eficiente.

WFA selecciona recursos de entrada del diccionario, como unidades vFiler, agregados y máquinas virtuales, mediante algoritmos de búsqueda. A continuación, los recursos seleccionados se utilizan para ejecutar el flujo de trabajo. Los algoritmos de búsqueda de WFA forman parte de los elementos básicos de WFA, e incluyen buscadores y filtros. Para localizar y seleccionar los recursos necesarios, los algoritmos de búsqueda buscan los datos almacenados en caché desde diferentes repositorios, como Active IQ Unified Manager, VMware vCenter Server y una base de datos. De forma predeterminada, hay un filtro disponible para cada entrada de diccionario para buscar un recurso basado en sus claves naturales.

Debe definir los criterios de selección de recursos para cada comando del flujo de trabajo. Además, puede usar un buscador para definir los criterios de selección de recursos en cada fila del flujo de trabajo. Por ejemplo, cuando se crea un volumen que requiere una cantidad específica de espacio de almacenamiento, puede utilizar el buscador «Buscar agregado por capacidad disponible» del comando «Crear volumen», que selecciona un agregado con una cantidad específica de espacio disponible y crea el volumen en él.

Puede definir un conjunto de reglas de filtro para recursos de entrada de diccionario, como unidades vFiler, agregados y máquinas virtuales. Las reglas de filtro pueden contener uno o más grupos de reglas. Una regla consta de un atributo de entrada de diccionario, un operador y un valor. El atributo también puede incluir atributos de sus referencias. Por ejemplo, puede especificar una regla para los agregados de la siguiente

manera: Enumerar todos los agregados que tienen nombres que comienzan con la cadena "aggr" y tienen más de 5 GB de espacio disponible. La primera regla del grupo es el atributo «nombre», con el operador «mantarritos con» y el valor «aggr». La segunda regla para el mismo grupo es el atributo "available_size_mb", con el operador ">" y el valor "5000". Puede definir un conjunto de reglas de filtro junto con filtros públicos. La opción definir reglas de filtro está desactivada si ha seleccionado un buscador. La opción Guardar como Finder se desactiva si ha seleccionado la casilla de verificación definir reglas de filtro.

Además de los filtros y los buscadores, puede utilizar un comando Search o define para buscar los recursos disponibles. El comando search o define es la opción preferida sobre los comandos no-op. El comando Search and define se puede utilizar para definir recursos tanto del tipo de entrada de diccionario certificado como del tipo de entrada de diccionario personalizado. El comando search o define busca recursos pero no realiza ninguna acción en el recurso. Sin embargo, cuando se utiliza un buscador para buscar recursos, se utiliza en el contexto de un comando y las acciones definidas por el comando se ejecutan en los recursos. Los recursos devueltos por un comando search o define se utilizan como variables para los demás comandos del flujo de trabajo.

En la siguiente ilustración, se muestra un filtro para la selección de recursos:

Ejemplos de selección de recursos en flujos de trabajo predefinidos

Puede abrir los detalles del comando de los siguientes flujos de trabajo predefinidos en el Diseñador para comprender cómo se utilizan las opciones de selección de recursos:

- Cree un volumen NFS de Clustered Data ONTAP
- Establezca la relación entre clústeres
- Quite un volumen de Data ONTAP en clúster

Cómo funciona la reserva

La capacidad de reserva de recursos de OnCommand Workflow Automation reserva los recursos necesarios para garantizar que los recursos estén disponibles para ejecutar correctamente los flujos de trabajo.

Los comandos WFA pueden reservar los recursos necesarios y eliminar la reserva una vez que el recurso está disponible en la base de datos de la caché de WFA, normalmente después de adquirir una memoria caché. La funcionalidad de reserva garantiza que los recursos reservados estén disponibles para el flujo de trabajo hasta el periodo de caducidad de la reserva que se haya configurado en las opciones de configuración de WFA.

Puede utilizar la funcionalidad de reservas para excluir recursos reservados por otros flujos de trabajo durante la selección de recursos. Por ejemplo, si se programa la ejecución de un flujo de trabajo que ha reservado 100 GB de espacio en un agregado después de una semana, Además, está ejecutando otro flujo de trabajo que utiliza el comando **Crear volumen**, el flujo de trabajo que se está ejecutando no consume el espacio reservado por el flujo de trabajo programado para crear un nuevo volumen. Además, la capacidad de reserva permite ejecutar flujos de trabajo en paralelo.

Al ver con antelación un flujo de trabajo para su ejecución, el planificador de WFA considera todos los objetos reservados, incluidos los existentes en la base de datos de la memoria caché. Si ha habilitado la reserva, al planificar el flujo de trabajo se tendrán en cuenta los efectos de los flujos de trabajo programados y los flujos de trabajo que se ejecuten en paralelo, así como la existencia de elementos de almacenamiento.

La flecha de la siguiente ilustración muestra que la reserva está habilitada para el flujo de trabajo:

Workflow 'Abort SnapMirror relationship' ⓘ

Details	Define Workflow	User Inputs	Constants	Return Parameters	Help Content	Advanced
Workflow Name *	Abort SnapMirror relationship					
Entity Version *	1.0.0					
Categories	Data Protection					
Workflow Description	The 'Abort SnapMirror' workflow stops ongoing transfers for a					
Ready For Production	<input checked="" type="checkbox"/>					
Consider Reserved Elements	<input checked="" type="checkbox"/> ←					
Enable Element Existence Validation	<input checked="" type="checkbox"/>					
Minimum Software Versions	Clustered Data ONTAP 8.2.0					

Ejemplos de reserva en flujos de trabajo predefinidos

Puede abrir los siguientes flujos de trabajo predefinidos en el Diseñador para comprender cómo se utiliza la reserva:

- Entorno de clones
- Cree un volumen de Clustered Data ONTAP
- Establezca la relación entre clústeres
- Quite un volumen de Data ONTAP en clúster

Qué es la nomenclatura incremental

La nomenclatura incremental es un algoritmo que permite asignar un nombre a los atributos de un flujo de trabajo en función de los resultados de búsqueda de un parámetro. Puede asignar un nombre a los atributos basado en un valor incremental o en una expresión personalizada. La funcionalidad de nomenclatura incremental le ayuda a implementar una convención de nomenclatura en función de sus necesidades.

Puede usar la funcionalidad de nomenclatura incremental cuando diseñe flujos de trabajo para asignar nombres de forma dinámica a los objetos creados por el flujo de trabajo. La funcionalidad permite especificar criterios de búsqueda para un objeto mediante la característica de selección de recursos y el valor devuelto por los criterios de búsqueda se utiliza para el atributo del objeto. Además, puede especificar un valor para el atributo si no se encontró ningún objeto con los criterios de búsqueda especificados.

Puede utilizar una de las siguientes opciones para asignar nombres a los atributos:

- Proporciona un valor de incremento y un sufijo

Puede proporcionar un valor que se debe utilizar junto con el valor del objeto encontrado por los criterios de búsqueda e incrementar con el número que especifique. Por ejemplo, si desea crear volúmenes con la convención de nomenclatura de `filer name_unique number_environment`, puede usar un buscador para buscar el último volumen por su prefijo de nombre e incrementar el número único en 1, así como añadir el nombre del sufijo al nombre del volumen. Si el último prefijo de nombre de volumen encontrado era `vf_023_prod` y está creando tres volúmenes, los nombres de los volúmenes creados son `vf_024_prod`, `vf_025_prod` y `vf_026_prod`.

- Proporcionar una expresión personalizada

Puede proporcionar un valor que se debe utilizar junto con el valor del objeto encontrado por los criterios de búsqueda y agregar valores adicionales basados en la expresión que especifique. Por ejemplo, si desea crear un volumen con la convención de nomenclatura de `last volume name_environment name padded with 1`, puede introducir la expresión `last_volume.name + ' ' + nextName("lab1")`. Si el último nombre de volumen encontrado era `vf_023`, el nombre del volumen creado es `vf_023_lab2`.

La siguiente ilustración muestra cómo se puede proporcionar una expresión personalizada para especificar una convención de nomenclatura:

Incremental Naming Wizard for Volume : name ?

X

The Incremental Naming wizard allows you to define the value of **name** based on a search for an existing **Volume**

Search criteria for existing Volume

Volume Name : \$VolumeName, Cluster Name or IP Address : \$...

Enter a value for **name** if no **Volume** matches the above search criteria

PRE_8_2_CLUSTER

if **Volume** was found using above search criteria, set value for **name** by

providing a custom expression ▼

Custom expression

last_volume.name

Cancel

Save

Qué es la ejecución condicional

La ejecución condicional ayuda a diseñar flujos de trabajo que puedan ejecutar comandos cuando se cumplan las condiciones especificadas.

La ejecución de comandos en un flujo de trabajo puede ser dinámica. Puede especificar una condición para la ejecución de cada comando o una fila de comandos del flujo de trabajo. Por ejemplo, puede que desee ejecutar el comando «'Agregar volumen al conjunto de datos'» solo cuando se encuentre un conjunto de datos específico y no desea que el flujo de trabajo falle si no se encuentra el conjunto de datos. En este caso, puede activar el comando «'Agregar volumen al conjunto de datos'» para buscar un conjunto de datos específico y, si no se encuentra, puede desactivar el comando en el flujo de trabajo.

Las opciones para la ejecución condicional de comandos están disponibles en la `Dictionary object` Y la ficha Avanzadas de los parámetros de `commands` cuadro de diálogo.

Puede cancelar un flujo de trabajo o deshabilitar un comando específico en el flujo de trabajo. Además, puede configurar un comando para que se ejecute utilizando una de las siguientes opciones:

- Sin ninguna condición

- Cuando se encuentran las variables especificadas
- Cuando no se encuentran las variables especificadas
- Cuando la expresión que ha especificado es verdadera

También puede configurar un comando para que espere un intervalo de tiempo específico.

Ejemplos de ejecución condicional en flujos de trabajo predefinidos

Puede abrir los detalles del comando de los siguientes flujos de trabajo predefinidos en el Diseñador para comprender cómo se utiliza la ejecución condicional de comandos:

- Cree un volumen de Data ONTAP en clúster básico
- Cree un volumen NFS de Clustered Data ONTAP

Cómo funcionan los parámetros de retorno

Los parámetros return son parámetros que están disponibles después de la fase de planificación de un flujo de trabajo. Los valores devueltos por estos parámetros son útiles para depurar un flujo de trabajo. Debe comprender cómo funcionan los parámetros return y qué parámetros se pueden utilizar como parámetros return para depurar flujos de trabajo.

Puede designar un conjunto de parámetros, como atributos de variables, expresiones y valores de entrada de usuario, en un flujo de trabajo como parámetros de retorno. Durante la ejecución del flujo de trabajo, los valores de los parámetros designados se completan en la fase de planificación y se inicia la ejecución del flujo de trabajo. A continuación, los valores de estos parámetros se devuelven de la forma en que se calcularon en esa ejecución específica del flujo de trabajo. Para depurar el flujo de trabajo, puede hacer referencia a los valores que devolvieron los parámetros.

Puede especificar los parámetros de retorno necesarios en un flujo de trabajo cuando desee ver cuáles son los valores calculados o seleccionados para esos parámetros. Por ejemplo, al utilizar la lógica de selección de recursos para seleccionar un agregado en un flujo de trabajo, puede especificar agregado como el parámetro return para ver qué agregado se seleccionó durante la planificación del flujo de trabajo.

Antes de hacer referencia a los valores de los parámetros de retorno para depurar el flujo de trabajo, debe confirmar que se ha completado la ejecución del flujo de trabajo. Los valores del parámetro return se establecen para cada ejecución del flujo de trabajo. Si agregó un parámetro return después de varias ejecuciones de un flujo de trabajo, el valor de ese parámetro solo está disponible para las ejecuciones después de agregar el parámetro.

Parámetros que se pueden utilizar como parámetros de retorno

Parámetros de retorno	Ejemplo
Atributos de variables que son escalares	<code>volume1.name</code> , que es un atributo de la variable "nombre del volumen"
Constantes	<code>MAX_VOLUME_SIZE</code>
Entradas del usuario	<code>\$clustername</code>

Parámetros de retorno	Ejemplo
Expresiones MVEL que implican atributos variables, constantes e entradas de usuario	volume1.name+'-'+\$clusterName
El parámetro return que agrega un comando durante la ejecución	La \$volumeUUID El parámetro se añade como un parámetro return cuando se utiliza la siguiente línea en un comando de PowerShell: Add-WfaWorkflowParameter -Name "VolumeUUID" -Value "12345" -AddAsReturnParameter \$true.

Ejemplos de parámetros de retorno en flujos de trabajo predefinidos

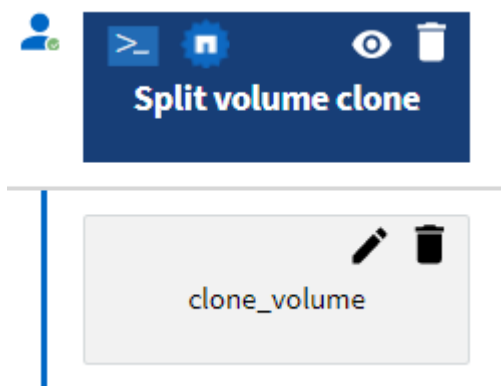
Si desea comprender cómo se especifican los parámetros return, puede abrir los siguientes flujos de trabajo predefinidos en el Diseñador y revisar los parámetros return especificados:

- Cree un volumen NFS en un vFiler
- Cree un recurso compartido CIFS de Qtree en un vFiler
- Cree un recurso compartido CIFS para volúmenes de Clustered Data ONTAP

¿Qué puntos de aprobación son

Los puntos de aprobación son puntos de comprobación utilizados en un flujo de trabajo para pausar la ejecución del flujo de trabajo y reanudarlo basándose en la aprobación del usuario.

La barra vertical azul que se muestra en la siguiente ilustración es un punto de aprobación:



Puede utilizar puntos de aprobación para la ejecución incremental de un flujo de trabajo, donde las secciones del flujo de trabajo sólo se deben ejecutar después de cumplir una condición determinada. Por ejemplo, cuando la siguiente sección tiene que ser aprobada o cuando se valida la ejecución correcta de la primera sección. Los puntos de aprobación no controlan ningún proceso entre la pausa y la reanudación de un flujo de trabajo. Las notificaciones por correo electrónico y SNMP se envían, como se especifica en la configuración de WFA, y se puede pedir al operador de almacenamiento que realice determinadas acciones al recibir la notificación de pausa del flujo de trabajo. Por ejemplo, el operador de almacenamiento puede enviar detalles de planificación a admin, aprobador o operador para su aprobación y reanudar el flujo de trabajo cuando se recibe la aprobación.

Es posible que no se necesiten aprobaciones en todo momento. En algunos casos, la aprobación podría ser necesaria sólo si se cumple una condición determinada y las condiciones se pueden configurar cuando se agrega un punto de aprobación. Por ejemplo, piense en un flujo de trabajo que se ha diseñado para aumentar el tamaño de un volumen. Puede agregar un punto de aprobación al principio del flujo de trabajo para que el operador de almacenamiento obtenga la aprobación de los gestores cuando el aumento del tamaño del volumen tenga como resultado un uso del 85% del espacio en el agregado que contiene el volumen. Durante la ejecución del flujo de trabajo y al seleccionar un volumen que da como resultado esta condición, la ejecución se detiene hasta que se aprueba.

La condición configurada para el punto de aprobación puede tener una de las siguientes opciones:

- Sin ninguna condición
- Cuando se encuentre la variable especificada
- Cuando no se encuentra la variable especificada
- Cuando la expresión que ha especificado se evalúa como TRUE

No hay limitación en el número de puntos de aprobación de un flujo de trabajo. Puede insertar puntos de aprobación antes de comandos en un flujo de trabajo y establecer los comandos después del punto de aprobación para esperar la aprobación antes de la ejecución. Los puntos de aprobación proporcionan información, como el tiempo de cambio, el usuario y los comentarios, lo que permite ver cuándo y por qué se puso en pausa o se reanudó la ejecución del flujo de trabajo. Los comentarios del punto de aprobación pueden incluir expresiones MVEL.

Ejemplos de puntos de aprobación en flujos de trabajo predefinidos

Puede abrir los siguientes flujos de trabajo predefinidos en el Diseñador para comprender cómo se utilizan los puntos de aprobación:

- Quite un volumen de Data ONTAP en clúster
- Actualización de la controladora y de la bandeja de un par de alta disponibilidad
- Migrar volúmenes

Cómo ejecutar puntos finales de REST personalizados

OnCommand Workflow Automation (WFA) proporciona un mecanismo para configurar los extremos DE REST personalizados para ejecutar los flujos de trabajo. Los extremos de REST personalizados ayudan a un arquitecto a configurar identificadores de recursos (URI) fáciles de entender, intuitivos y uniformes para ejecutar flujos de trabajo, que siguen las convenciones DE REST de POST, PUT o DELETE en función de la semántica del flujo de trabajo. Estos URI facilitan el desarrollo de código de cliente para los desarrolladores de clientes.

WFA permite configurar una ruta de URI personalizada para la ejecución del flujo de trabajo a través de las llamadas API. Cada segmento de la ruta URI puede ser una cadena o un nombre válido de la entrada del usuario del flujo de trabajo entre paréntesis, por ejemplo, `/devops/{ProjectName}/clone`. El flujo de trabajo se puede invocar como una llamada a `https://WFAServer:`

`HTTPS_PORT/rest/devops/Project1/clone/jobs`.

La validación de la ruta de URI es la siguiente:

- La vía DE DESCANSO deberá empezar por «»/».
- Los caracteres permitidos son alfabetos, dígitos y subrayado.
- El nombre de entrada del usuario deberá estar rodeado por «»{}».



Deberá comprobar que el valor rodeado por «»{}» es un nombre de entrada de usuario válido.

- No debería haber segmentos de ruta vacíos, por ejemplo, //, /{}/, etc.
- La configuración del método HTTP y la configuración de la ruta de URI personalizada deben estar configuradas o no configurarse.

¿Cómo funciona continuar con el fallo

La función continuar con el fallo ayuda a configurar un paso en un flujo de trabajo para que la ejecución del flujo de trabajo pueda continuar incluso si el paso falla. Puede abordar los pasos con errores y resolver el problema que provocó el error al acceder al `wfa.log` o haciendo clic en .

Un flujo de trabajo con uno o más de estos pasos fallidos se encuentra en el estado parcialmente correcto después de que se completa la ejecución. Puede configurar un paso para que la ejecución del flujo de trabajo continúe aunque el paso falle seleccionando la opción necesaria en la ficha Avanzadas del cuadro de diálogo parámetros para `<command_name>`.

Si un paso no está configurado para continuar en caso de fallo, la ejecución del flujo de trabajo se cancelará si el paso falla.

Si un paso configurado para continuar en caso de fallo falla, puede establecer el flujo de trabajo para que se ejecute utilizando una de las siguientes opciones:

- Cancelar la ejecución del flujo de trabajo (opción predeterminada)
- Continúe la ejecución desde el siguiente paso
- Continúe la ejecución desde la fila siguiente

Lista de verificación de requisitos de flujo de trabajo de ejemplo

Una lista de verificación de requisitos de flujo de trabajo incluye requisitos detallados, como comandos, entradas de usuario y recursos, para un flujo de trabajo planificado. Puede utilizar la lista de comprobación para planificar los flujos de trabajo e identificar las lagunas de los requisitos.

Ejemplo de lista de requisitos

En la siguiente lista de verificación de requisitos de flujo de trabajo de ejemplo se enumeran los requisitos para el flujo de trabajo «'Crear un volumen de Clustered Data ONTAP'». Puede utilizar esta lista de comprobación de ejemplo como plantilla para enumerar los requisitos del flujo de trabajo.

Requisito	Descripción
Nombre del flujo de trabajo	Cree un volumen de Clustered Data ONTAP

Requisito	Descripción
Categoría	Aprovisionamiento de almacenamiento
Descripción	El flujo de trabajo crea un nuevo volumen en una SVM específica. Este flujo de trabajo está pensado para un escenario en el que se aprovisiona un volumen y se delega para un uso posterior.
Descripción de alto nivel del funcionamiento del flujo de trabajo	<ul style="list-style-type: none"> • El usuario especifica la SVM que contiene el volumen (clúster, nombres de SVM). • Se crea un volumen según el tamaño especificado. • La configuración del volumen se describe en una plantilla.

Requisito	Descripción
Detalles	<ul style="list-style-type: none"> • Utilice el comando Crear volumen cm • Detalles del comando para Crear volumen cm: <ul style="list-style-type: none"> ◦ La ejecución se define como siempre ◦ Los detalles del volumen se especifican rellenando los atributos del volumen ◦ Utilice la plantilla Space Congarantizado Settings para configurar el volumen ◦ El usuario proporciona el nombre de volumen y el tamaño. <p>El volumen se montará en el espacio de nombres de SVM tal como /volname (en el espacio de nombres raíz).</p> <ul style="list-style-type: none"> ◦ Utilice la función actualVolumeSize porque la reserva snap será del 5%. ◦ La referencia de SVM se define con la siguiente lógica de selección de recursos: <ul style="list-style-type: none"> ▪ CM SVM por clave: Busca SVM por nombre y clúster, que proporciona el usuario ▪ SVM DE CM por tipo — solo SVM de datos (tipo = clúster) ▪ SVM por estado — (estado = en ejecución) ◦ La referencia de agregado se define con la lógica de selección de recursos como un buscador predefinido (agregado cm por umbrales de espacio y tipo RAID): <ul style="list-style-type: none"> ▪ Agregado DE CM por capacidad disponible (capacidad = tamaño de volumen que se va a aprovisionar, clúster determinado por el usuario) ▪ Agregado DE CM por delegación a SVM ▪ Agregado DE CM por tipo de RAID (RAID-DP) ▪ El agregado DE CM no es aggr0 ▪ CM de agregado en % de tamaño usado (umbral = 90, spaceToBeProvisioned = tamaño proporcionado, ya que la garantía es volumen) ▪ Agregado CM por encima del compromiso (umbral = 300, spaceToBeAllocalizado = tamaño del volumen que se aprovisiona) ▪ Seleccione el agregado con el máximo espacio libre

Nombre	Tipo	Descripción (valores de datos, validación, etc.)
Clúster	Consulta bloqueada (tabular)	<ul style="list-style-type: none"> • Clúster que aloja la SVM • La consulta puede aparecer en una tabla de resultados con la dirección primaria y el nombre del clúster • Ordenar alfabéticamente por nombre
SVM	Consulta bloqueada	<ul style="list-style-type: none"> • SVM en la que se aprovisiona el volumen • La consulta solo debe mostrar los nombres de las SVM que pertenecen al clúster seleccionado en la entrada anterior <p>Muestra solo las SVM de tipo de clúster, no las de administrador ni nodo (tipo columna de cm_Storage.Vserver)</p> <ul style="list-style-type: none"> • Ordenar alfabéticamente
Volumen	Cadena	<ul style="list-style-type: none"> • Nombre del volumen que se creará
El tamaño en GB	Entero	<ul style="list-style-type: none"> • El tamaño del volumen que se va a aprovisionar • Tamaño de los datos (se debe tener en cuenta la reserva snap)

Comandos

Nombre	Descripción	Estado
Crear volumen cm	Crea un volumen en la SVM	Existente

Parámetros de retorno

Nombre	Valor
Nombre del volumen	El nombre del volumen aprovisionado

Nombre	Valor
Nombre del agregado	Nombre del agregado seleccionado
Nombre del nodo	El nombre del nodo
Nombre del clúster	El nombre del clúster

Lagunas y problemas

1.	
2.	
3.	
4.	
5.	

Crear un flujo de trabajo

Puede usar Workflow Automation (WFA) para crear flujos de trabajo para tareas como el aprovisionamiento, la migración y el decomisionado del almacenamiento para bases de datos o sistemas de archivos. Debe crear flujos de trabajo cuando los flujos de trabajo de WFA predefinidos no coincidan con sus requisitos.


Lo que necesitará



- Debe haber entendido los conceptos básicos de WFA.
- Debe haber comprendido las capacidades como la fila de repetición, los puntos de aprobación y la selección de recursos que se requieren para el flujo de trabajo.
- Debe haber completado la planificación necesaria para el flujo de trabajo, incluida la lista de comprobación de requisitos del flujo de trabajo.
- Debe haber creado el contenido de ayuda, que proporciona información sobre el flujo de trabajo a los operadores de almacenamiento.

Acerca de esta tarea


La construcción de cada flujo de trabajo puede variar en función del objetivo y los requisitos del flujo de trabajo. Esta tarea no proporciona instrucciones para un flujo de trabajo específico, pero proporciona instrucciones generales para crear un flujo de trabajo.

Pasos


1. Haga clic en **Diseño de flujo de trabajo > flujos de trabajo**.
2. Haga clic en  en la barra de herramientas.
3. En la ficha **flujo de trabajo**, realice los siguientes pasos:

- a. Expanda el esquema requerido y, a continuación, haga doble clic en el requerido  (comando) o.  (Flujo de trabajo) en la lista **pasos disponibles**.

Puede repetir este paso según sea necesario. Puede arrastrar y soltar pasos para reorganizar los pasos en el editor de flujos de trabajo.

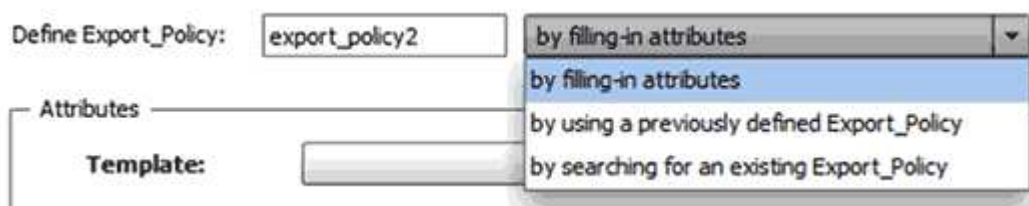
- b. **Opcional:** haga clic  para agregar el número requerido de filas, que se utilizan para especificar detalles para la ejecución de pasos.

Cada paso se ejecuta en función de los detalles de paso especificados en la fila y la columna especificadas. Los pasos se ejecutan de izquierda a derecha y de arriba a abajo.


- c. Coloque el cursor debajo del paso que ha añadido y haga clic en  para agregar detalles de pasos para la ejecución de pasos, en la fila requerida.


Para este paso...	Realice lo siguiente...
Flujo de trabajo	Introduzca las entradas de usuario necesarias en la ficha flujo de trabajo y la condición necesaria en la ficha Avanzado .
Comando	En la ficha parámetros para <command> , haga clic en cada ficha de objeto, seleccione la opción necesaria para definir los atributos del objeto y, a continuación, introduzca los detalles necesarios en la ficha Opciones avanzadas y en la ficha otros parámetros .
Buscar o definir	Seleccione el objeto de entrada de diccionario que se debe buscar o definir.

En la ilustración siguiente se muestran las opciones disponibles para definir los atributos del objeto:



Elija la acción adecuada:

Durante...	Realice lo siguiente...
rellenando atributos	<p>Introduzca el valor para los atributos utilizando las siguientes opciones:</p> <ul style="list-style-type: none"> • Expresiones • Variables • Entradas del usuario • Selección de recursos • Nomenclatura incremental <p>Debe colocar el cursor sobre los campos de atributo y hacer clic en  para usar las capacidades de selección de recursos o de nomenclatura incremental.</p>
utilizando un definido previamente <i>object</i>	<p>Seleccione el definido previamente <i>object</i> en el cuadro situado delante de la lista de opciones.</p>
buscando una existente <i>object</i>	<ol style="list-style-type: none"> Haga clic en introducir criterios de búsqueda para buscar el objeto utilizando la capacidad de selección de recursos. Seleccione una de las opciones necesarias para la ejecución si no se encuentra el objeto necesario: <ul style="list-style-type: none"> ◦ Aborte el flujo de trabajo <p>Esta opción anula la ejecución del flujo de trabajo si no se encuentra el objeto específico.</p> <ul style="list-style-type: none"> ◦ Deshabilite este comando <div> <p>This option disables only the current step and executes the workflow.</p> <p>**** Atributos de relleno para `object` y ejecute el comando</p> </div> <div> <p>This option enables you to enter the required attributes and execute the workflow.</p> </div>

4. Si desea insertar un punto de aprobación, haga clic en  e introduzca el comentario necesario para el punto de aprobación.

Los comentarios de puntos de aprobación pueden incluir expresiones MVEL.

5. Haga clic en  que se encuentra junto a los números de fila para realizar lo siguiente:

- Insertar una fila.
- Copie la fila.
- Repita la fila.

Puede utilizar una de las siguientes opciones para especificar la repetición de los parámetros del comando:

- Número de veces

Puede utilizar esta opción para repetir la ejecución del comando para el número de repeticiones que especifique. Por ejemplo, puede especificar que el comando «'Create Qtree» se debe repetir tres veces para crear tres qtrees.

También puede usar esta opción para un número dinámico de ejecuciones de comandos. Por ejemplo, puede crear una variable de entrada de usuario para el número de LUN que se van a crear y utilizar el número especificado por el operador de almacenamiento cuando el flujo de trabajo se ejecuta o se programa.

- Para cada recurso de un grupo

Puede utilizar esta opción y especificar un criterio de búsqueda para un objeto. El comando se repite tantas veces como los criterios de búsqueda devuelven el objeto. Por ejemplo, puede buscar los nodos de un clúster y repetir el comando «'Crear interfaz lógica iSCSI'» para cada nodo.

- Agregar una condición para la ejecución de la fila.
- Retire la fila.

6. En la ficha **Detalles**, realice los siguientes pasos:

- a. Especifique la información necesaria en los campos **Nombre del flujo de trabajo** y **Descripción del flujo de trabajo**.

El nombre y la descripción del flujo de trabajo deben ser únicos para cada flujo de trabajo.

- b. **Opcional:** especifique la versión de la entidad.
- c. **Opcional:** desactive la casilla de verificación **considerar elementos reservados** si no desea utilizar la capacidad de reserva.
- d. **Opcional:** desactive la casilla de verificación **Activar validación de existencia de elemento** si no desea habilitar la validación de elementos que existen con el mismo nombre.

7. Si desea editar las entradas del usuario, realice los siguientes pasos:

- a. Haga clic en la ficha **Entradas de usuario**.
- b. Haga doble clic en la entrada de usuario que desee editar.
- c. En el cuadro de diálogo **Editar variable: <user input>**, edite la entrada del usuario.

8. Si desea agregar constantes, realice los siguientes pasos

- a. Haga clic en la ficha **constantes** y, a continuación, agregue las constantes necesarias para el flujo de trabajo mediante el botón **Agregar**.

Puede definir constantes cuando se utiliza un valor común para definir los parámetros para varios comandos. Por ejemplo, consulte la constante AGGREGATE_OVERDMENT_THRESHOLD utilizada en el flujo de trabajo "Crear, asignar y proteger LUN con SnapVault".

- b. Introduzca el nombre, la descripción y el valor de cada constante.

9. Haga clic en la ficha **parámetros de retorno** y, a continuación, agregue los parámetros necesarios para el flujo de trabajo mediante el botón **Agregar**.

Puede utilizar parámetros devueltos cuando la planificación y ejecución del flujo de trabajo deben devolver algunos valores calculados o seleccionados durante la planificación. Puede ver los valores calculados o seleccionados en la ficha parámetros de retorno de la ventana de supervisión de la vista previa del flujo de trabajo o una vez completada la ejecución del flujo de trabajo.

Agregado: Puede especificar agregado como parámetro de retorno para ver qué agregado se seleccionó utilizando la lógica de selección de recursos.

Si ha incluido un flujo de trabajo secundario en el flujo de trabajo y si los nombres de los parámetros de retorno del flujo de trabajo secundario contienen un espacio, símbolo de dólar (\$), o una función, debe especificar el nombre del parámetro de retorno entre corchetes en el flujo de trabajo principal para ver el valor del parámetro de retorno del flujo de trabajo secundario en el flujo de trabajo principal.

Si el nombre del parámetro es...	Especificar como...
ChildWorkflow1.abc\$value	ChildWorkflow1["abc\$"+"value"]
ChildWorkflow1.\$value	ChildWorkflow1["\$"+"value"]
ChildWorkflow1.value\$	ChildWorkflow1.value\$
ChildWorkflow1.P N	ChildWorkflow1["P N"]
ChildWorkflow1.return_string("HW")	ChildWorkflow1["return_string(\"HW\")"]

10. **Opcional:** haga clic en la ficha **Contenido de la Ayuda** para agregar el archivo de contenido de ayuda que ha creado para el flujo de trabajo.
11. Haga clic en **Vista previa** y asegúrese de que la planificación del flujo de trabajo se ha completado correctamente.
12. Haga clic en **Aceptar** para cerrar la ventana de vista previa.
13. Haga clic en **Guardar**.

Después de terminar

Pruebe el flujo de trabajo en su entorno de prueba y, a continuación, marque el flujo de trabajo como listo para la producción en **WorkflowName > Details**.

Crear contenido de ayuda del flujo de trabajo

Los administradores y arquitectos de OnCommand Workflow Automation (WFA) que diseñan flujos de trabajo pueden crear contenido de ayuda para los flujos de trabajo e incluirlo en el flujo de trabajo.

Lo que necesitará

Debe ser consciente de cómo crear páginas Web mediante HTML.

Acerca de esta tarea

La ayuda debe proporcionar información sobre el flujo de trabajo y las entradas del usuario para el flujo de trabajo al operador de almacenamiento que ejecuta el flujo de trabajo.

Pasos

1. Cree una carpeta con el siguiente nombre: Workflow-help.
2. Cree el contenido de la ayuda con un editor HTML o un editor de texto y guárdelo como un `index.htm` en la `workflow-help` carpeta.

No debe incluir archivos JavaScript como parte del contenido de ayuda. Las siguientes son las extensiones de archivo compatibles:

- .jpg
- .jpeg
- .gif
- .png
- .xml
- .thmx
- .htm
- .html
- .css

También puede incluir el `Thumbs.db` Archivo, creado por Windows.

3. Compruebe que el `index.htm` el archivo y otros archivos asociados con el contenido de ayuda, como las imágenes, están disponibles en la `workflow-help` carpeta.
4. Cree un `.zip` archivo de la carpeta y asegúrese de que el tamaño de `.zip` El archivo no es superior a 2 MB.

```
Create an NFS volume-help.zip
```

5. Edite el flujo de trabajo para el que ha creado el contenido de la ayuda y, a continuación, haga clic en **Configuración > Contenido de la ayuda > examinar** para cargar el `.zip` archivo.

Cree paquetes de flujo de trabajo de WFA

Puede crear paquetes de flujo de trabajo en OnCommand Workflow Automation (WFA) para sus requisitos de automatización e integración del almacenamiento.

Pasos

1. Inicie sesión en la ventana de WFA mediante un navegador web.
2. Haga clic en **Administración de contenido > Paquetes**.
3. Haga clic en el icono **Nuevo paquete**.
4. En el cuadro de diálogo **Nuevo paquete**, introduzca valores para los campos **Nombre**, **Autor**, **Versión** y **Descripción**.
5. Haga clic en **Guardar**.
6. Compruebe que el nuevo paquete se ha creado en la ventana **Paquetes**.

Añada entidades a los paquetes de flujo de trabajo de WFA

Puede añadir una o varias entidades a un paquete de flujos de trabajo en OnCommand Workflow Automation (WFA) para satisfacer sus requisitos de automatización e integración del almacenamiento.

Acerca de esta tarea

Puede quitar un paquete de las siguientes entidades:

- Flujo de trabajo
- Buscadores
- Filtros
- Comandos
- Empresarial
- Plantillas
- Esquemas
- Diccionario
- Tipos de orígenes de datos
- Tipos de sistemas remotos
- Consultas en caché
- Categorías

Pasos

1. Inicie sesión en la ventana **WFA** a través de un navegador web.
2. En el portal de diseño de **flujo de trabajo**, desplácese hasta la entidad que desee agregar y haga clic en **<Entity>**.
3. En la ventana **entidad**, seleccione la entidad que desea agregar al paquete.
4. Haga clic en el icono **Agregar al paquete**.

"Agregar al paquete" sólo está activado para las entidades para las que la certificación se ha establecido en **ninguna**.
5. En el cuadro de diálogo **Agregar a paquete <Entity>**, en la lista desplegable **Paquetes disponibles**, seleccione el paquete al que desea agregar la entidad.
6. Haga clic en **Aceptar**.

Elimine los paquetes OnCommand Workflow Automation

Puede eliminar un paquete de OnCommand Workflow Automation (WFA) si ya no lo necesita. Si elimina un paquete, se eliminarán todas las entidades asociadas al paquete.

Acerca de esta tarea


- No puede eliminar un paquete si hay dependencias en las entidades que forman parte del paquete.

Por ejemplo, si intenta eliminar un paquete que incluye un comando que forma parte de un flujo de trabajo personalizado, la operación de eliminación fallará porque el flujo de trabajo personalizado depende del paquete. Solo puede eliminar el paquete después de eliminar el flujo de trabajo personalizado.

- Las entidades que forman parte de un paquete no se pueden eliminar individualmente.

Para eliminar una entidad que forma parte de un paquete, debe eliminar el paquete que contiene esa entidad. Si una entidad forma parte de varios paquetes, la entidad se eliminará del servidor WFA solo cuando se eliminen todos los paquetes que contienen esa entidad.

Pasos

1. Inicie sesión en WFA a través de un navegador web como administrador.
2. Haga clic en **Administración de contenido > Paquetes**.
3. Seleccione el paquete que desea eliminar y haga clic en .
4. En el cuadro de diálogo de confirmación **Eliminar paquete**, haga clic en **Aceptar**.

Exporte el contenido OnCommand Workflow Automation




Puede guardar contenido de OnCommand Workflow Automation (WFA) creado por el usuario como un `.dar` archivar y compartir el contenido con otros usuarios. El contenido de WFA puede incluir todo el contenido creado por el usuario o elementos específicos como flujos de trabajo, buscadores, comandos y términos del diccionario.

Lo que necesitará

- Debe tener acceso al contenido de WFA que desee exportar.
- Si el contenido que se va a exportar contiene referencias a contenido certificado, los paquetes de contenido certificado correspondientes deben estar disponibles en el sistema cuando se importe el contenido.

Estos paquetes se pueden descargar del almacén de automatización del almacenamiento.

Acerca de esta tarea

- No puede exportar los siguientes tipos de contenido certificado:
 -  - Contenido certificado por NetApp
 -  - Contenido desarrollado por Servicios profesionales (PS), que está disponible sólo en instalaciones personalizadas realizadas por PS
 -  - packs desarrollados por usuarios
- También se exportan todos los objetos que dependen del objeto exportado.

Por ejemplo, la exportación de un flujo de trabajo también exporta los comandos, filtros y buscadores


dependientes del flujo de trabajo.

- Puede exportar objetos bloqueados.

Los objetos permanecen en estado bloqueado cuando otros usuarios los importan.

Pasos

1. Inicie sesión en WFA mediante un navegador web.
2. Exporte el contenido necesario:

Si desea...	Realice lo siguiente...
Exporte todo el contenido creado por el usuario como un solo elemento .dar archivo	<ol style="list-style-type: none">a. Haga clic en Configuración y en Mantenimiento haga clic en Exportar todos los flujos de trabajo.b. Especifique un nombre de archivo para .dar Y, a continuación, haga clic en Exportar.
Exportar contenido específico	<ol style="list-style-type: none">a. Desplácese hasta la ventana desde la que desea exportar contenido.b. Seleccione uno o más elementos de la ventana y, a continuación, haga clic en .c. En el cuadro de diálogo Exportar como, especifique un nombre de archivo para .dar Y, a continuación, haga clic en Exportar.

3. En el cuadro de diálogo **Guardar como**, especifique la ubicación en la que desea guardar .dar Y, a continuación, haga clic en **Guardar**.

Importe el contenido de OnCommand Workflow Automation

Puede importar contenido de OnCommand Workflow Automation (WFA) creado por el usuario, como flujos de trabajo, buscadores y comandos. También puede importar contenido que se exporta desde otra instalación de WFA, contenido que se descarga desde el almacén de automatización del almacenamiento o la comunidad de WFA, así como paquetes, incluidos los kits de herramientas PowerShell de Data ONTAP y los kits de herramientas Perl de NMSDK.

Lo que necesitará

- Debe tener acceso al contenido de WFA que desee importar.
- El contenido que desea importar debe haberse creado en un sistema que ejecute la misma versión o una versión anterior de WFA.

Por ejemplo, si está ejecutando WFA 2.2, no puede importar contenido que se haya creado con WFA 3.0.

- Puede importar contenido desarrollado en versiones N-2 de WFA solo a WFA 5.1.
- Si la .dar Referencias de archivo el contenido certificado por NetApp, deben importarse los paquetes de contenido certificados por NetApp.

Los paquetes de contenido certificados por NetApp se pueden descargar en la Storage Automation Store. Debe consultar la documentación del paquete para verificar que se cumplen todos los requisitos.

Pasos

1. Inicie sesión en WFA mediante un navegador web.
2. Haga clic en **Configuración** y en **Mantenimiento** haga clic en **flujos de trabajo de importación**.
3. Haga clic en **elegir archivo** para seleccionar .dar Archivo que desea importar y, a continuación, haga clic en **Importar**.
4. En el cuadro de diálogo **éxito de importación**, haga clic en **Aceptar**.

Importe los paquetes de flujo de trabajo de WFA

Puede importar paquetes de flujos de trabajo del servidor a OnCommand Workflow Automation (WFA) para sus requisitos de automatización e integración de almacenamiento.

Lo que necesitará

Debe tener acceso al contenido de WFA en el servidor que desea importar.

Pasos

1. Inicie sesión en la ventana **WFA** a través de un navegador web.
2. Haga clic en **Administración de contenido > Paquetes**.
3. Haga clic en el icono **Importar desde el servidor**.
4. En el cuadro de diálogo **Importar desde carpeta del servidor**, en el campo **ubicación de carpeta en el sistema del servidor**, escriba la ubicación del paquete en el servidor en un formato de cadena, por ejemplo, C:\work\packs\test.
5. Haga clic en **Aceptar**.
6. Compruebe que el paquete se importa en la ventana **Paquetes**.

Consideraciones que tener en cuenta al importar el contenido de OnCommand Workflow Automation

Debe tener en cuenta determinados aspectos que debe tener en cuenta al importar contenido creado por el usuario, contenido exportado desde otra instalación de OnCommand Workflow Automation (WFA), o contenido que se descarga desde el almacén de automatización del almacenamiento o la comunidad de WFA.

- WFA el contenido se guarda como un .dar file y puede incluir todo el contenido creado por el usuario desde otro sistema o elementos específicos como flujos de trabajo, buscadores, comandos y términos de diccionario.
- Cuando se importa una categoría existente desde una .dar archivo, el contenido importado se combina con el contenido existente de la categoría.

Por ejemplo, tenga en cuenta que existen dos flujos de trabajo WF1 y WF2 en la categoría A del servidor WFA. Si los flujos de trabajo WF3 y WF4 de la categoría A se importan al servidor WFA, la categoría A contendrá flujos de trabajo WF1, WF2, WF3 y WF4 después de la importación.

- Si la .dar el archivo contiene entradas del diccionario y las tablas de caché correspondientes a las

entradas del diccionario se actualizan automáticamente.

Si las tablas de caché no se actualizan automáticamente, se registra un mensaje de error en el `wfa.log` archivo.

- Al importar un `.dar` Archivo que tiene una dependencia de un paquete que no está presente en el servidor WFA, WFA intenta identificar si se cumplen todas las dependencias de las entidades.
 - Si faltan una o más entidades o si se encuentra una versión inferior de una entidad, la importación falla y se muestra un mensaje de error.

El mensaje de error proporciona detalles de los paquetes que se deben instalar para cumplir con las dependencias.

- Si se encuentra una versión superior de una entidad o si la certificación ha cambiado, se muestra un cuadro de diálogo genérico acerca de la discrepancia de versión y se completa la importación.

Los detalles de discrepancia de versiones se registran en un `wfa.log` archivo.

- Las preguntas y las solicitudes de soporte relativas a lo siguiente deben dirigirse a la comunidad de WFA:
 - Cualquier contenido descargado desde la comunidad de WFA
 - Contenido personalizado de WFA que ha creado
 - WFA contenido que ha modificado

Identificación del paquete durante la actualización

Durante el proceso de actualización, OnCommand Workflow Automation (WFA) identifica y clasifica las entidades en un paquete. Si eliminó alguna entidad de un paquete antes de la actualización, el paquete no se identificará durante la actualización.

Durante el proceso de actualización, WFA compara los paquetes de la base de datos con la lista de paquetes que se lanzaron en el almacén de automatización del almacenamiento para identificar los paquetes que se instalaron antes de la actualización. Por lo tanto, la identificación de paquetes clasifica los paquetes existentes en la base de datos.

WFA realiza los siguientes procesos para identificar y clasificar paquetes:

- Mantiene una lista de paquetes publicados en Storage Automation Store para comparar e identificar los paquetes que se instalaron antes de la actualización.
- Clasifica las entidades de un paquete como parte de la sincronización del almacén de automatización del almacenamiento si está activado el almacén de automatización del almacenamiento.
- Clasifica las entidades en paquetes mediante la lista actualizada.

La identificación de paquetes solo se aplica a los paquetes certificados de NetApp que se descargaron de Storage Automation Store.

Si no se identifica un paquete durante la actualización, puede volver a importarlo para que se identifique en WFA. Los archivos `wfa.log` proporcionan detalles acerca de las entidades que no se identificaron como un paquete durante la actualización.

Integre los paquetes de flujo de trabajo de WFA con el repositorio SCM

Puede integrar el paquete OnCommand Workflow Automation (WFA) con el repositorio Gestión de control de origen (SCM).

Debe tener credenciales de administrador o de arquitecto.

Las herramientas SCM como GitHub, Perforce y SVN requieren que asigne un directorio local para extraer el código del servidor de repositorio de SCM. Esta asignación de directorio local se denomina ubicación de cliente *SCM*. debe configurar el cliente SCM con una ubicación de sistema de archivos como área de cliente.

Puede configurar el cliente SCM en un sistema de servidor WFA. Debe tener acceso al sistema del servidor WFA para las operaciones SCM

Registrar un nuevo paquete de flujo de trabajo en SCM

Puede crear un nuevo paquete de flujos de trabajo con OnCommand Workflow Automation (WFA) y check-in en Source Control Management (SCM).

Lo que necesitará

SCM debe estar configurado y debe tener credenciales de administrador o de arquitecto.

Pasos

1. Inicie sesión en la ventana **WFA** a través de un navegador web.
2. Cree un nuevo paquete de flujo de trabajo.

[Cree un paquete de automatización de flujos de trabajo](#)

3. Agregue entidades al paquete creado.

[Agregue entidades a un paquete OnCommand Workflow Automation](#)

4. Haga clic en el icono **Exportar al servidor**.
5. En el cuadro de diálogo Exportar a carpeta del servidor, en el campo ubicación de carpeta en el sistema del servidor*, introduzca la ubicación del sistema de archivos donde se guardará el paquete en el servidor que contiene el cliente SCM.

Para editar o volver a exportar el paquete o el contenido, haga clic en el icono **Desbloquear**.

6. En la ubicación del cliente SCM, compruebe el contenido del paquete en el servidor SCM.

Compruebe una nueva versión de un paquete de flujo de trabajo WFA

Puede actualizar la versión de un paquete en OnCommand Workflow Automation (WFA) y, a continuación, realizar el check-in del paquete actualizado a una nueva ubicación en el servidor de gestión de control de origen (SCM).

Lo que necesitará

El SCM debe estar configurado y debe tener credenciales de administrador o de arquitecto.

Pasos

1. Inicie sesión en la ventana **WFA** a través de un navegador web.
2. Haga clic en **Administración de contenido > Paquetes**.
3. Haga clic en el icono **Editar paquete**.
4. En el cuadro de diálogo **<pack name> Pack**, en el campo **Versión**, actualice la versión del paquete.
5. Haga clic en **Guardar**.
6. Haga clic en el icono **Exportar al servidor** en el nivel de paquete.
7. En el cuadro de diálogo **Exportar a carpeta del servidor**, en el campo **ubicación de carpeta en el sistema del servidor**, introduzca una nueva ubicación del sistema de archivos.

Si el paquete se guardó anteriormente en C:\p4\cdot\1.0.0 ubicación del sistema de archivos, ahora guárdelo en el C:\p4\cdot\2.0.0 ubicación.

8. En la ubicación del cliente SCM, compruebe el contenido del paquete en una nueva ubicación en el servidor SCM.

Si el paquete se guardó anteriormente en //depot/wfa/packs/cdot/1.0.0 Ruta en el servidor SCM, puede guardarlo en otra ubicación, como //depot/wfa/packs/cdot/2.0.0.

Actualice los paquetes de flujos de trabajo de WFA desde el servidor SCM

Puede actualizar un paquete en el servidor de gestión de control de origen (SCM) y, a continuación, importar el paquete actualizado a OnCommand Workflow Automation (WFA).

Lo que necesitará

El SCM debe estar configurado y debe tener credenciales de administrador o de arquitecto

Acerca de esta tarea

Si realiza cambios o actualizaciones en un paquete en el servidor SCM, el administrador o el arquitecto deben resolver los conflictos, si los hay, mediante las herramientas Diff proporcionadas por SCM. WFA adapta el archivo diff XML para mostrar sólo los cambios relevantes.

Antes de importar el paquete, recibirá una notificación sobre los cambios que se producirán en el contenido del paquete WFA.

Pasos

1. Inicie sesión en la ventana de WFA mediante un navegador web.
2. Importe el paquete actualizado a WFA.

"Importe los paquetes de flujo de trabajo de WFA"



Si la base de datos de WFA ya contiene el mismo paquete, el contenido del paquete se sobrescribirá.

Compruebe los paquetes de flujo de trabajo de WFA existentes en el servidor SCM

Puede realizar el check-in de paquetes ya existentes en el servidor de gestión de control de origen (SCM) desde OnCommand Workflow Automation (WFA).

Lo que necesitará

El SCM debe estar configurado y debe tener credenciales de administrador o de arquitecto.

Pasos

1. Inicie sesión en la ventana **WFA** a través de un navegador web.
2. Haga clic en **Administración de contenido > Paquetes**.
3. Haga clic en el icono **Exportar al servidor**.
4. En el cuadro de diálogo **Exportar a carpeta del servidor**, en el campo **ubicación de carpeta en el sistema del servidor**, introduzca la ubicación de la carpeta del servidor donde se guarda el paquete en el servidor.

De esta forma, exporta el paquete en forma explosionada en el sistema de archivos donde se crea el cliente SCM.

5. En la ubicación del cliente SCM, compruebe el contenido del paquete en el servidor SCM.
6. Utilice las herramientas Diff proporcionadas por SCM para verificar los cambios con respecto a la versión SCM del paquete.

Quite los paquetes de flujo de trabajo de WFA de las entidades

Puede quitar un paquete de las entidades de OnCommand Workflow Automation (WFA) y realizar el check-in del paquete actualizado en el servidor de gestión de control de origen (SCM).

Lo que necesitará

El SCM debe estar configurado y debe tener credenciales de administrador o de arquitecto.

Acerca de esta tarea

Puede quitar un paquete de las siguientes entidades:

- Flujo de trabajo
- Buscadores
- Filtros
- Comandos
- Empresarial
- Plantillas
- Esquemas
- Diccionario
- Tipos de orígenes de datos
- Tipos de sistemas remotos
- Consultas en caché
- Categorías

Pasos

1. Inicie sesión en la ventana **WFA** a través de un navegador web.

2. En el portal de diseño de **flujo de trabajo**, desplácese hasta la entidad que desee quitar y haga clic en **<Entity>**.
3. Haga clic en el icono **Quitar del paquete**.
4. En el cuadro de diálogo **Quitar del paquete <Entity>**, seleccione el paquete que desea eliminar de esa entidad.
5. Haga clic en **Aceptar**.
6. Haga clic en la ficha **Paquetes**.
7. Haga clic en el icono **Exportar al servidor**.
8. En el cuadro de diálogo **Exportar a carpeta del servidor**, en el campo **ubicación de carpeta en el sistema del servidor**, introduzca la ubicación de la carpeta del servidor donde se guarda el paquete en el servidor.

De esta forma, exporta el paquete en forma explosionada en el sistema de archivos donde se crea el cliente SCM.

9. En la ubicación del cliente SCM, compruebe el contenido del paquete en el servidor SCM.
10. Utilice las herramientas Diff proporcionadas por SCM para verificar los cambios con respecto a la versión SCM del paquete.

Revertir un paquete de flujo de trabajo de WFA a su versión anterior en SCM

Puede revertir un paquete a la versión anterior en Gestión de control de origen (SCM) e importarlo a OnCommand Workflow Automation (WFA).

Lo que necesitará

El SCM debe estar configurado y debe tener credenciales de administrador o de arquitecto.

Pasos

1. En la ubicación del cliente de SCM, vuelva a retroceder el paquete a una versión anterior en la ubicación del sistema de archivos mediante las herramientas de SCM.

El cliente SCM se sincroniza con el número de cambio exacto en el que está interesado.

2. Inicie sesión en la ventana de WFA mediante un navegador web.
3. Importe el paquete actualizado a WFA.

["Importe los paquetes de flujo de trabajo de WFA"](#)



De este modo se revierte la base de datos de WFA a la versión anterior.

Crear elementos básicos para los flujos de trabajo

Workflow Automation (WFA) incluye varios elementos básicos, que se utilizan para crear flujos de trabajo. Puede crear los bloques de WFA que sean necesarios para sus flujos de trabajo.


Cree un tipo de origen de datos

Debe crear un tipo de origen de datos para habilitar la adquisición de datos desde un origen de datos, que no está predefinido en OnCommand Workflow Automation (WFA).

Lo que necesitará

- Debe haber creado la entrada y esquema de diccionario necesarios si está creando un tipo de origen de datos personalizado que no está predefinido en WFA.
- Debe conocer los scripts de PowerShell para crear un tipo de origen de datos que utilice el método de script.


Pasos

1. Haga clic en **Diseño de origen de datos > tipos de origen de datos**.
2. Haga clic en  en la barra de herramientas.
3. En el cuadro de diálogo **Tipo de nuevo origen de datos**, introduzca o seleccione los detalles necesarios en los campos **origen de datos**, **Versión de origen de datos** y **Esquema**.
4. En el campo **Puerto predeterminado**, introduzca el número de puerto.

2638

El número de puerto introducido se rellena al agregar este tipo de origen de datos para la adquisición de datos. De forma predeterminada, WFA utiliza el puerto para comunicarse con el origen de datos y el puerto debe estar abierto en el servidor de origen de datos.

5. En la lista **método**, seleccione el método que WFA debe utilizar para adquirir datos:

Si seleccionó...	Realice lo siguiente...
SQL	<p>En la lista Tipo de controlador, seleccione uno de los siguientes controladores que sean adecuados para el origen de datos:</p> <ul style="list-style-type: none">• Conector MySQL/J• Controlador JDBC del servidor MySQL 3.0• Controlador Oracle JDBC 11.2.0.3
GUIÓN	<p>En el campo Script, introduzca la secuencia de comandos de PowerShell que se utiliza para conectar y recuperar datos del origen de datos.</p> <div><p>Los datos de los archivos CSV equivalentes de la entrada del diccionario deben incluir tabulaciones como separador de campo. Por ejemplo, consulte el script de PowerShell para el tipo de origen de datos de VMware vCenter.</p></div>

6. Haga clic en **Guardar**.


Crear un comando

Puede crear un comando de WFA para completar una tarea específica en el flujo de trabajo si no hay ningún comando WFA predefinido que sea adecuado para la tarea.

Lo que necesitará

Debe saber escribir el código necesario para el comando utilizando PowerShell o Perl.

Pasos

1. Haga clic en **Diseño de flujo de trabajo > comandos**.
2. Haga clic en  en la barra de herramientas.
3. En la ficha **Propiedades** del cuadro de diálogo **Nueva definición de comandos**, introduzca o seleccione los detalles necesarios en los campos **Nombre**, **Descripción** y **tiempo de espera**.

- a. En el campo **representación de cadena**, introduzca una representación de cadena para el comando utilizando la sintaxis MVEL.

```
VolumeName + "=>" + SnapshotName
```

La representación de cadena de un comando se utiliza para mostrar la información que se desea ver en el diseño del flujo de trabajo durante la planificación y la ejecución. Sólo debe utilizar los parámetros del comando en la representación de cadena de un comando.

- a. **Opcional:** Si está creando un comando wait , seleccione **wait for condition** en la sección **Command type** y configure el valor requerido en el campo **Waiting INTERVAL (s)** (intervalo de espera).
4. En la ficha **Código**, realice los siguientes pasos:
 - a. Seleccione el lenguaje de secuencias de comandos necesario para el comando en la lista **Idioma de secuencias de comandos**.

Puede hacer clic en + y seleccionar un idioma adicional para el comando.

- b. Introduzca el código adecuado para el comando en la ficha Idioma seleccionado.

Si desea utilizar el tipo de contraseña para las entradas de usuario en el script de PowerShell, debe crear un alias para el parámetro e incluirlo `_Password` en el atributo. Para el script Perl, puede especificar el tipo como **Contraseña** en la ficha **Definición de parámetros**.



El editor de comandos no admite los siguientes caracteres especiales: "<", "&" y etiquetas XML "<" ">".

```
param (
    [parameter(Mandatory=$false, HelpMessage="Specify an AD administrator
password.")]
    [Alias("ADAdminPassword_Password")] [string]$ADAdminPassword
)
```

5. En la ficha **Definición de parámetros**, realice los siguientes pasos:

- a. Haga clic en **detectar parámetros** para rellenar la tabla de definición de parámetros.

Los parámetros y sus atributos se extraen del código y se muestran en la tabla. Por ejemplo, los parámetros Array y VolumeName se extraen del siguiente código:

```
param (
    [parameter(Mandatory=$true, HelpMessage="Array name or IP address")]
    [string]$Array,

    [parameter(Mandatory=$true, HelpMessage="Volume name")]
    [string]$VolumeName,
)
```

- a. Haga clic en la columna Descripción de los parámetros para editar la descripción.

No puede editar ningún otro campo de esta pestaña.

6. En la ficha **asignación de parámetros**, realice los siguientes pasos para cada parámetro:

- a. En la columna **Tipo**, seleccione el objeto de diccionario adecuado.
- b. En la columna **attribute**, introduzca o seleccione los atributos apropiados para el objeto de diccionario de la lista.

Después de introducir un atributo, puede introducir un punto (.) e incluya otro atributo de ese objeto.

Introduzca `cm_storage.volume` como tipo y `aggregate.name` Como atributo para el parámetro `aggregateName`.

- c. En la columna **Nombre de objeto**, introduzca un nombre de objeto.

El nombre del objeto se utiliza para agrupar los parámetros en una pestaña del cuadro de diálogo parámetros para `<command>` cuando se especifican los detalles del comando en un flujo de trabajo.

Los parámetros sin asignar se muestran en la ficha **otros parámetros** del cuadro de diálogo parámetros para `<command>` cuando se especifican los detalles del comando en un flujo de trabajo.

7. En la ficha **Reservación**, introduzca una secuencia de comandos de reserva mediante consultas SQL para reservar los recursos requeridos por el comando durante una ejecución de flujo de trabajo programada:

- a. **Opcional:** en el campo **representación de reserva**, introduzca una representación de cadena para la reserva utilizando la sintaxis MVEL.

"Añadir regla para la etiqueta de SnapMirror"

" + SnapMirrorLabel + "[.code] A la política de SnapMirror "

" + PolicyName + "`"

La representación de cadena se utiliza para mostrar los detalles de los recursos reservados en la ventana Reservas.



El script de reserva no debe realizar ninguna operación en bases de datos excepto cm_Storage, cm_Performance, almacenamiento, rendimiento, vc, y esquemas personalizados.

8. **Opcional:** en la ficha **verificación**, introduzca una consulta SQL para comprobar si el comando ha afectado a los orígenes de datos y a la caché de WFA como se esperaba para poder eliminar la reserva.

La consulta SQL que escriba sólo puede constar de sentencias SELECT SQL.

- a. Haga clic en **verificación de prueba** para probar la secuencia de comandos de verificación.
 - b. En el cuadro de diálogo **verificación**, introduzca los parámetros de prueba necesarios.
 - c. Si no desea utilizar los datos de reserva para probar la secuencia de comandos de verificación, desactive el campo **usar *datos de reserva en test**.
 - d. Haga clic en **Prueba**.
 - e. Tras revisar el resultado de la prueba, cierre el cuadro de diálogo.
9. Haga clic en **Prueba** para probar el comando.
 10. En el cuadro de diálogo **<command name> de comandos de prueba**, haga clic en **Prueba**.

El resultado de la prueba se muestra en la sección Log messages del cuadro de diálogo.

11. Haga clic en **Guardar**.

Pruebe el script de reserva para comandos

Puede probar los scripts de reserva que ha escrito para los comandos OnCommand Workflow Automation (WFA) en la base de datos de juegos para garantizar que los scripts funcionan bien y no afectan a las tablas de la base de datos de WFA.

Acerca de esta tarea

La ruta de instalación predeterminada de WFA se utiliza en este procedimiento. Si ha cambiado la ubicación predeterminada durante la instalación, debe utilizar la ruta de instalación de WFA cambiada.

Pasos

1. Abra un símbolo del sistema en el servidor de WFA y cambie los directorios a la siguiente ubicación:
`c:\Program Files\NetApp\WFA\mysql\bin`
2. Cree un volcado de la base de datos de WFA usando el siguiente comando: `mysqldump -u wfa -pWfa123 --single-transaction --skip-add-drop-table database_tables> dump_location`

Comando para crear un volcado de las tablas de la base de datos cm_Storage: `mysqldump -u wfa -pWfa123 --single-transaction --skip-add-drop-table cm_storage> c:\tmp\cmSt2.sql`
3. Restaure el volcado que ha creado en la base de datos de la zona de juegos de WFA usando el siguiente comando: `mysql -u wfa -pWfa123 playground < dump_location`

`mysql -u wfa -pWfa123 playground < c:\tmp\cmSt2.sql`
4. Cree o edite un comando de WFA y escriba el script de reserva en la pestaña **Reservación**.

Debe asegurarse de que las secuencias de comandos de reserva y verificación sólo utilicen la base de datos de juegos.

5. Cree o edite un flujo de trabajo, incluya el comando en el flujo de trabajo y, a continuación, ejecute el flujo de trabajo.
6. Confirmar que las secuencias de comandos de reserva y verificación funcionan según lo esperado.

El proceso de adquisición del origen de datos de WFA no actualiza la base de datos de juegos. Es necesario quitar manualmente las reservas que ha creado el comando.



Crear un buscador

Puede crear un buscador de WFA que pueda buscar recursos si no hay ningún localizador predefinido WFA que sea adecuado para buscar los recursos necesarios.


Lo que necesitará

Debe haber creado los filtros requeridos que se usan en el buscador.

Pasos

1. Haga clic en **Diseño de flujo de trabajo > Finders**.
2. Haga clic en  en la barra de herramientas.
3. En la ficha **Propiedades** del cuadro de diálogo **Nuevo Finder**, introduzca o seleccione los detalles necesarios en los campos **Nombre**, **Tipo** y **Descripción**.
4. En la ficha **Filtros**, seleccione los filtros necesarios en la lista **Filtros disponibles** y haga clic en .

Puede añadir o eliminar filtros según sus necesidades.

5. En la ficha atributos devueltos, seleccione los atributos necesarios para el filtro en la lista **disponible** y haga clic en .
6. **Opcional:** haga clic en **Prueba** para probar el buscador.
 - a. En el cuadro de diálogo **<FinderName>** del Buscador de pruebas, introduzca los parámetros de prueba necesarios.
 - b. Desactive la casilla de verificación **usar datos de reserva en prueba** si no desea utilizar los datos de reserva para probar el buscador.
 - c. Haga clic en **Prueba**.

Se muestra el resultado de la prueba.

- d. Cierre el cuadro de diálogo.
7. Haga clic en **Guardar**.


Crear un filtro

Puede crear un filtro de WFA que pueda buscar recursos si no hay ningún filtro WFA predefinido adecuado para la tarea.

Lo que necesitará

Debe conocer las sintaxis SQL adecuadas para crear el filtro.

Pasos

1. Haga clic en **Diseño de flujo de trabajo > Filtros**.
2. Haga clic en  en la barra de herramientas.
3. En la ficha **Propiedades** del cuadro de diálogo **filtro nuevo**, introduzca o seleccione los detalles necesarios en los campos **Nombre**, **Tipo de diccionario** y **Descripción**.
4. En la ficha **Consulta**, introduzca la consulta SQL adecuada para el filtro.

Debe introducir una única consulta SQL y, opcionalmente, utilizar parámetros de entrada. Debe utilizar la siguiente sintaxis para utilizar un parámetro de entrada: `${ParameterName}`.

```
SELECT
    array.ip
FROM
    storage.array
WHERE
    array.name = '${ArrayName}'
```

5. Haga clic en **Actualizar** para rellenar la tabla **parámetros de entrada** y la lista **atributos devueltos**.

Esta información se obtiene de la consulta SQL que ha introducido. Por ejemplo, si utiliza el ejemplo de consulta SQL del paso anterior, la ip se muestra en atributos devueltos y ArrayName se muestra en parámetros de entrada. Puede editar las entradas en las columnas **Label** y **Description**.

6. **Opcional:** haga clic en **Prueba** para probar el filtro.
 - a. En el cuadro de diálogo **<FilterName> de filtro de prueba**, introduzca los parámetros de prueba necesarios.
 - b. Desactive la casilla de verificación **usar datos de reserva en prueba** si no desea utilizar los datos de reserva para probar el filtro.
 - c. Haga clic en **Prueba**.

Se muestra el resultado de la prueba.


- d. Cierre los cuadros de diálogo.

7. Haga clic en **Guardar**.

Crear una entrada de diccionario

Puede crear una entrada del diccionario de WFA si desea definir un nuevo tipo de objeto y su relación en el entorno de almacenamiento.

Pasos

1. Haga clic en **Diseño de origen de datos > Diccionario**.
2. Haga clic en  en la barra de herramientas.
3. En el cuadro de diálogo **Nueva entrada de diccionario**, introduzca los detalles necesarios en los campos **Nombre del tipo de objeto** y **Descripción**.

4. Para el campo **Esquema**, realice una de las siguientes acciones:
 - Seleccione uno de los esquemas disponibles de la lista.
 - Haga clic en **Agregar nuevo esquema**, introduzca el **Nombre de esquema** requerido en el cuadro de diálogo Nuevo esquema y, a continuación, haga clic en **Agregar**.
5. Haga clic en **Agregar fila** y realice los pasos siguientes para describir el atributo:
 - a. Haga clic en la columna **Nombre** e introduzca el nombre del atributo.
 - b. En la columna **Tipo**, seleccione el tipo requerido.

La columna **longitud de cadena** se rellena y se puede editar si seleccionó cadena como tipo. Además, la columna **valores** se puede editar si ha seleccionado **enumeración** como tipo.
 - c. Seleccione las casillas de verificación adecuadas para el atributo en `Natural Key`, `To be Cached`, y `Can be Null columns`.

Si ha seleccionado la casilla de verificación **clave natural**, no podrá activar la casilla de verificación **puede ser nula**.
 - d. Agregue los atributos necesarios para el objeto de diccionario.
 - e. **Opcional:** Active la casilla de verificación **valores de las columnas de clave natural distinguen entre mayúsculas y minúsculas** si desea que las claves naturales sean sensibles a mayúsculas y minúsculas.
6. Haga clic en **Guardar**.

Crear una función

Puede crear una función de WFA que pueda utilizarse como utilidad si no hay una función WFA predefinida que sea adecuada para su tarea.

Lo que necesitará

Para crear una función, debe conocer las sintaxis de MVFLEX Expression Language (MVEL).

Acercas de esta tarea


Debe incluir lo siguiente para las definiciones de funciones:

- Nombre: Nombre de la función

No debe utilizar una palabra reservada en la sintaxis de MVEL. Cada función debe tener un nombre único.

- Definición de MVEL: Cadena que especifica la sintaxis de MVEL de la definición de función

Pasos

1. Haga clic en **Diseño de flujo de trabajo > funciones**.
2. Haga clic en  en la barra de herramientas.
3. En el cuadro de diálogo **Nueva función**, introduzca o seleccione los detalles necesarios en los campos **Descripción de función** y **definición de función**.

```
def actualVolumeSize(data_size, snap_pct)
{
    if (snap_pct < 0 ) {
        snap_pct = 0;
    } else if (snap_pct > 99) {
        snap_pct = 99;
    }

    div = 1 - (snap_pct/100);
    return (int)(data_size/div);
}
```

El campo **nombre de función** se rellena a partir de los datos que se utilizan en la sintaxis MVEL.

4. **Opcional:** haga clic en **Prueba** para probar la función:

a. En la sección **expresión** del cuadro de diálogo **Prueba**, introduzca la expresión requerida de la función.

```
actualVolumeSize(600, 1)
```

b. Haga clic en **Prueba**.

Se muestra el resultado de la prueba.

c. Cierre el cuadro de diálogo.

5. Haga clic en **Guardar**.

Cree una plantilla

Puede crear una plantilla que se pueda utilizar como guía para rellenar atributos en los detalles del comando.

Pasos

1. Haga clic en **Diseño de flujo de trabajo > Plantillas**.

2. Haga clic en  en la barra de herramientas.

3. En el cuadro de diálogo **Nueva plantilla**, introduzca o seleccione los detalles necesarios en los campos **Nombre**, **Tipo** y **Descripción**.

La tabla atributos se rellena en función del objeto de diccionario seleccionado en el campo **Tipo**.

4. Haga clic en la columna valor de cada atributo y realice una de las siguientes acciones:

- Introduzca o seleccione el valor requerido de la lista.
- Introduzca una entrada de entrada de usuario, por ejemplo: \$size para entrada de usuario de tamaño.

5. Haga clic en **Guardar**.


Cree una consulta de caché

Puede definir una consulta de caché cuando desee almacenar en caché información acerca de un objeto de diccionario en la base de datos de WFA desde un tipo de origen de datos. Puede crear una consulta de caché y asociarla a una entrada de diccionario y a uno o más tipos de origen de datos, como Active IQ Unified Manager 6.1.

Lo que necesitará

Debe conocer las sintaxis SQL adecuadas para crear una consulta de caché.

Pasos

1. Haga clic en **Diseño de origen de datos > consultas de caché**.
2. Haga clic en  en la barra de herramientas.
3. En el cuadro de diálogo **Agregar consulta de caché**, seleccione la entrada de diccionario y el tipo de origen de datos necesarios.
4. En la sección "consulta de selección de SQL", introduzca la consulta SQL adecuada.

La siguiente consulta SQL almacena en caché información sobre el objeto de diccionario de disco del tipo de origen de datos Active IQ Unified Manager 6.1:

```
SELECT
    disk.objId AS id,
    disk.name AS NAME,
    disk.uid AS uid,
    disk.effectiveInterfaceType AS TYPE,
    disk.rpm AS rpm,
    disk.homeNodeId AS home_node_id,
    disk.ownerNodeId AS owner_node_id,
    disk.model AS model,
    disk.serialNumber AS serial_number,
    disk.totalBytes/1024/1024 AS size_mb,
    disk.shelf AS shelf,
    disk.shelfBay AS shelf_bay,
    disk.pool AS pool,
    disk.vendor AS vendor,
    LOWER(disk.raidPosition) AS raid_position,
    disk.containerTypeRaw AS container_type,
    disk.clusterId AS cluster_id
FROM
    netapp_model_view.disk disk
```

5. Si desea probar la consulta SQL, haga clic en **Prueba**.

Si ha seleccionado más de un tipo de origen de datos, se abre el cuadro de diálogo Consulta de caché de prueba y permite seleccionar el tipo de origen de datos necesario.

Se muestra el resultado de la prueba.

6. Cierre el cuadro de diálogo.
7. Haga clic en **Guardar**.


Crear programas recurrentes

OnCommand Workflow Automation (WFA) proporciona dos opciones de programación para los flujos de trabajo. Puede programar un flujo de trabajo para que se ejecute una vez en un momento específico, o bien crear programaciones periódicas y asociar las programaciones a flujos de trabajo para que los flujos de trabajo se ejecuten de forma rutinaria.

Acerca de esta tarea

Una programación que haya creado puede volver a utilizarse y asociarse con varios flujos de trabajo.

Pasos

1. Haga clic en **ejecución > programas**.
2. Haga clic en  en la barra de herramientas.
3. En el cuadro de diálogo **Nueva programación**, introduzca o seleccione un nombre, una descripción y una frecuencia para la programación.

Para frecuencia, debe introducir la hora en formato de 24 horas. La hora del servidor WFA se aplica a las programaciones.

4. Haga clic en **Aceptar**.

Después de terminar

- Puede asociar la programación a un flujo de trabajo cuando ejecute el flujo de trabajo utilizando la opción **Ejecutar de forma recurrente**.
- Para ver los detalles de un flujo de trabajo y su asociación con un programa, haga clic en **ejecución > programas recurrentes**.

La planificación de los recursos y la ejecución de los flujos de trabajo programados para ejecutarse una vez se realiza de inmediato cuando los flujos de trabajo son programados. Sin embargo, la planificación de recursos y ejecución para los flujos de trabajo con programación recurrente tiene lugar a la hora programada y no cuando la programación está asociada a un flujo de trabajo.

Defina las reglas de filtro







Puede definir un conjunto de reglas para filtrar los recursos de entrada del diccionario, como unidades vFiler, agregados y máquinas virtuales. Puede personalizar las reglas para los flujos de trabajo existentes y para nuevos flujos de trabajo mientras los crea.

Pasos

1. Inicie sesión en WFA a través de un navegador web como administrador.
2. Haga clic en **Diseño de flujo de trabajo > flujos de trabajo**.
3. En la ventana **flujos de trabajo**, haga doble clic en el flujo de trabajo que desee modificar.

Aparecerá la ventana Workflow <workflow name> (de flujo de trabajo).

4. Defina un conjunto de reglas eligiendo una de las siguientes opciones:

Si desea...	Realice lo siguiente...
Buscar recursos cuando se repiten los comandos de una fila	<ul style="list-style-type: none">a. Haga clic en un número de fila y seleccione repetir fila.b. En el cuadro de diálogo repeticiones de filas, seleccione la opción para cada recurso del grupo de la lista desplegable repeticiones.c. Seleccione un tipo de recurso.d. Haga clic en el enlace introducir criterios de búsqueda.
Busque los recursos necesarios en las entradas de comandos	<ul style="list-style-type: none">a. Haga clic en .b. En el cuadro de diálogo parámetros para <command_name>, seleccione la opción buscando un <dictionary object> existente en la lista desplegable definir <dictionary object>.c. Haga clic en el enlace introducir criterios de búsqueda.
Busque recursos a los que hacen referencia las variables en las entradas de comandos	<ul style="list-style-type: none">a. Haga clic en .b. En el cuadro de diálogo parámetros para <command_name>, seleccione la opción rellenando atributos en la lista desplegable definir <dictionary object>.c. Haga clic en  para un campo marcado con .
Entradas de comando Name del tipo String	<ul style="list-style-type: none">a. Haga clic en .b. En el cuadro de diálogo parámetros para <command_name>, seleccione la opción rellenando atributos en la lista desplegable definir <dictionary object>.c. Haga clic en  para un campo de cadena.

5. En el cuadro de diálogo **selección de recursos**, active la casilla de verificación **definir reglas de filtro**.

Si ha seleccionado una de las opciones del cuadro de diálogo desplegable Finder de selección de recursos, la casilla de verificación definir reglas de filtro estará desactivada. El valor del buscador debe definirse en «'Ninguno'» para que se activen las reglas de filtro definir.

6. Introduzca el atributo, el operador y el valor de la regla.


El valor debe proporcionarse entre comillas simples. Las reglas de filtro pueden contener uno o más grupos.

7. Haga clic en **Aceptar**.

Agregar puntos de aprobación

Puede agregar un punto de aprobación como punto de control en un flujo de trabajo para pausar la ejecución del flujo de trabajo y reanudarlo en función de su aprobación. Puede utilizar puntos de aprobación para la ejecución incremental de un flujo de trabajo, donde las secciones del flujo de trabajo se ejecutan sólo después de cumplir una determinada condición; por ejemplo, cuando la siguiente sección tiene que ser aprobada o cuando se valida la ejecución correcta de la primera sección.

Pasos

1. Inicie sesión en WFA mediante un navegador web como arquitecto o administrador.
2. Haga clic en **Diseño de flujo de trabajo > flujos de trabajo**.
3. En la ventana **flujos de trabajo**, haga doble clic en el flujo de trabajo que desee modificar.
4. En la ventana **<workflow name> de flujo de trabajo**, haga clic en  icono a la izquierda del paso para el que desea agregar el punto de aprobación.

Puede agregar puntos de aprobación para uno o varios pasos.

5. En el cuadro de diálogo **Nuevo punto de aprobación**, proporcione los detalles del comentario y la condición.
6. Haga clic en **Aceptar**.

Directrices de codificación para WFA

Debería comprender las directrices de codificación general de OnCommand Workflow Automation (WFA), las convenciones de nomenclatura y las recomendaciones sobre la creación de elementos básicos como filtros, funciones, comandos y flujos de trabajo.

Pautas para las variables

Debe tener en cuenta las directrices para PowerShell y las variables Perl de OnCommand Workflow Automation (WFA) cuando cree un comando o un tipo de origen de datos.

Variables de PowerShell

Directrices	Ejemplo
Para parámetros de entrada de script: <ul style="list-style-type: none">• Utilice la caja Pascal.• No utilice guiones bajos.• No utilice abreviaturas.	<code>\$VolumeName</code> <code>\$AutoDeleteOptions</code> <code>\$Size</code>

Directrices	Ejemplo
<p>Para variables internas de script:</p> <ul style="list-style-type: none"> • Utilice la funda Camel. • No utilice guiones bajos. • No utilice abreviaturas. 	<p>\$newVolume</p> <p>\$treeName</p> <p>\$time</p>
<p>Para funciones:</p> <ul style="list-style-type: none"> • Utilice la caja Pascal. • No utilice guiones bajos. • No utilice abreviaturas. 	<p>GetVolumeSize</p>
<p>Los nombres de variables no distinguen entre mayúsculas y minúsculas. Sin embargo, para mejorar la legibilidad, no debe utilizar mayúsculas diferente para el mismo nombre.</p>	<p>\$variable es igual que \$Variable.</p>
<p>Los nombres de variables deben estar en inglés sin formato y deben estar relacionados con la funcionalidad del script.</p>	<p>Uso \$name y no \$a.</p>
<p>Declare explícitamente el tipo de datos de cada variable.</p>	<p>[cadena]nombre</p> <p>tamaño [int]</p>
<p>No utilice caracteres especiales (! @ # & % , .) y espacios de aplicaciones.</p>	<p>Ninguno</p>
<p>No utilice palabras clave reservadas de PowerShell.</p>	<p>Ninguno</p>
<p>Agrupe los parámetros de entrada colocando primero los parámetros obligatorios seguidos de los parámetros opcionales.</p>	<pre>param ([parameter (Mandatory=\$true)] [string] \$Type, [parameter (Mandatory=\$true)] [string] \$Ip, [parameter (Mandatory=\$false)] [string] \$VolumeName)</pre>

Directrices	Ejemplo
Comentar todas las variables de entrada mediante <i>HelpMessage</i> anotación con un mensaje de ayuda significativo.	<pre>[parameter (Mandatory=\$false, HelpMessage="LUN to map")] [string] \$LUNName</pre>
No utilice «'Archivador'» como nombre de variable; utilice «'Array'» en su lugar.	Ninguno
Uso <i>ValidateSet</i> anotación en los casos en los que el argumento obtiene valores enumerados. Esto se traduce automáticamente al tipo de datos Enum para el parámetro.	<pre>[parameter (Mandatory=\$false, HelpMessage="Volume state")] [ValidateSet("online", "offline", "restricted")] [string] \$State</pre>
Agregue un alias a un parámetro que termine con <code>""_Capacity"</code> para indicar que el parámetro es de tipo capacidad.	<p>El comando «'Create Volume'» utiliza alias de la siguiente forma:</p> <pre>[parameter (Mandatory=\$false, HelpMessage="Volume increment size in MB")] [Alias("AutosizeIncrementSize_Capacity")] [int] \$AutosizeIncrementSize</pre>
Agregue un alias a un parámetro que termine con <code>""_Password"</code> para indicar que el parámetro es de tipo de contraseña.	<pre>param ([parameter (Mandatory=\$false, HelpMessage="In order to create an Active Directory machine account for the CIFS server or setup CIFS service for Storage Virtual Machine, you must supply the password of a Windows account with sufficient privileges")] [Alias("Pwd_Password")] [string] \$ADAdminPassword)</pre>

Variables Perl

Directrices	Ejemplo
Para parámetros de entrada de script: <ul style="list-style-type: none"> • Utilice la caja Pascal. • No utilice guiones bajos. • No utilice abreviaturas. 	<pre>\$VolumeName \$AutoDeleteOptions \$Size</pre>
No utilice abreviaturas para las variables internas del script.	<pre>\$new_volume \$qtrees_name \$time</pre>
No utilice abreviaturas para las funciones.	<pre>get_volume_size</pre>
Los nombres de variables distinguen mayúsculas de minúsculas. Para mejorar la legibilidad, no debe utilizar mayúsculas diferente para el mismo nombre.	<pre>\$variable no es lo mismo que \$Variable.</pre>
Los nombres de variables deben estar en inglés sin formato y deben estar relacionados con la funcionalidad del script.	Uso <code>\$name</code> y no <code>\$a</code> .
Agrupe los parámetros de entrada colocando primero los parámetros obligatorios, seguidos de los parámetros opcionales.	Ninguno
En la función <code>GetOptions</code> , declare explícitamente el tipo de datos de cada variable para los parámetros de entrada.	<pre>GetOptions ("Name=s"=>\\$Name, "Size=i"=>\\$Size)</pre>
No utilice «'Archivador'» como nombre de variable; utilice «'Array'» en su lugar.	Ninguno
Perl no incluye la <code>ValidateSet</code> anotación para valores enumerados. Utilice declaraciones explícitas «'if'» para casos en los que el argumento obtenga valores enumerados.	<pre>if (defined\$SpaceGuarantee&&!(\$SpaceGuaranteeeq'none'</pre>
	<pre>\$SpaceGuaranteeeq'volume'</pre>

Directrices	Ejemplo
	<pre>\$SpaceGuaranteeeq'file')) { die'Illegal SpaceGuarantee argument: \''.\$SpaceGuarantee.'\'; } ----</pre>
<p>Todos los comandos Perl WFA deben utilizar el pragma "strict" para desalentar el uso de construcciones inseguras para variables, referencias y subrutinas.</p>	<pre>use strict; # the above is equivalent to use strictvars; use strictsubs; use strictrefs;</pre>
<p>Todos los comandos Perl de WFA deben utilizar los siguientes módulos Perl:</p> <ul style="list-style-type: none"> • Getopt <p>Se utiliza para especificar parámetros de entrada.</p> <ul style="list-style-type: none"> • WFAUtil <p>Esto se utiliza para las funciones de utilidad que se proporcionan para el registro de comandos, la generación de informes sobre el progreso de comandos, la conexión con las controladoras de la cabina, etc.</p>	<pre>use Getopt::Long; use NaServer; use WFAUtil;</pre>

Directrices para indentación

Debe tener en cuenta las directrices para la sangría cuando se escribe un script de PowerShell o Perl para OnCommand Workflow Automation (WFA).

Directrices	Ejemplo
Una pestaña es igual a cuatro espacios vacíos.	

Directrices	Ejemplo
Utilice tabulaciones y llaves para mostrar el principio y el final de un bloque.	<p>Script de PowerShell</p> <pre> if (\$pair.length-ne 2) { throw "Got wrong input data" } </pre> <p>Script Perl</p> <pre> if (defined \$MaxDirectorySize) { # convert from MBytes to Bytes my \$MaxDirectorySizeBytes = \$MaxDirectorySize * 1024 * 1024; } </pre>
Agregue líneas en blanco entre conjuntos de operaciones o fragmentos de código.	<pre> \$options=\$option.trim(); \$pair=\$option.split(" "); Get-WFALogger -Info -messages \$("split options: "+ \$Pair) </pre>

Directrices para comentarios

Debe tener en cuenta las directrices para PowerShell y comentarios Perl en sus secuencias de comandos para OnCommand Workflow Automation (WFA).

Comentarios sobre PowerShell

Directrices	Ejemplo
Utilice el carácter # para un comentario de una sola línea.	<pre> # Single line comment \$options=\$option.trim(); </pre>

Directrices	Ejemplo
Utilice el carácter # para un comentario de fin de línea.	<pre>\$options=\$option.trim(); # End of line comment</pre>
Utilice los caracteres <# y #> para un comentario de bloque.	<pre><# This is a block comment #> \$options=\$option.trim();</pre>

Comentarios Perl

Directrices	Ejemplo
Utilice el carácter # para el comentario de una línea.	<pre># convert from MBytes to Bytes my \$MaxDirectorySizeBytes = \$MaxDirectorySize * 1024 * 1024;</pre>
Utilice el carácter # para el comentario de fin de línea.	<pre>my \$MaxDirectorySizeBytes = \$MaxDirect orySize * 1024 * 1024; # convert to Bytes</pre>
Utilice el carácter # en cada línea con un # vacío al principio y al final para crear un borde de comentario para comentarios de varias líneas.	<pre># # This is a multi-line comment. Perl 5, unlike # Powershell, does not have direct support for # multi-line comments. Please use a '#' in every line # with an empty '#' at the beginning and end to create # a comment border #</pre>

Directrices	Ejemplo
No incluya código inactivo y comentado en los comandos de WFA. Sin embargo, para realizar pruebas, puede utilizar el mecanismo de documentación antigua simple (POD) para comentar el código.	<pre>=begin comment # Set deduplication if (defined \$Deduplication && \$Deduplication eq "enabled") { \$wfaUtil-> sendLog("Enabling Deduplication"); } =end comment =cut</pre>

Directrices para el registro

Debe conocer las directrices para iniciar sesión al escribir un script PowerShell o Perl para OnCommand Workflow Automation (WFA).

Registro de PowerShell

Directrices	Ejemplo
Utilice el cmdlet Get-WFALogger para el registro.	<pre>Get-WFALogger -Info -message "Creating volume"</pre>
Registre todas las acciones que requieran interacción con paquetes internos como Data ONTAP, VMware y PowerCLI. todos los mensajes de registro están disponibles en registros de ejecución en el historial de estado de ejecución de flujos de trabajo.	Ninguno
Registre todos los argumentos relevantes que se pasan a paquetes internos.	Ninguno
Utilice los niveles de registro adecuados cuando utilice el cmdlet Get-WFALogger, en función del contexto de uso. -Info, -error, -warn y -Debug son los distintos niveles de registro disponibles. Si no se especifica un nivel de registro, el nivel de registro predeterminado es Debug.	Ninguno

Registro Perl

Directrices	Ejemplo
<p>Utilice WFAUtil SendLog para el registro.</p>	<pre>my wfa_util = WFAUtil->new(); eval { \$wfa_util->sendLog('INFO', "Connecting to the cluster: \$DestinationCluster"); }</pre>
<p>Registrar cada acción que requiere la interacción con cualquier elemento externo al comando, como Data ONTAP, VMware y WFA. Todos los mensajes de registro que cree con la rutina WFAUtil SendLog se almacenan en la base de datos WFA. Estos mensajes de registro están disponibles para el flujo de trabajo y el comando ejecutados.</p>	<p>Ninguno</p>
<p>Registre cada argumento relevante pasado a la rutina que se llamó.</p>	<p>Ninguno</p>
<p>Use los niveles de registro adecuados. -Info, -error, -warn y -Debug son los distintos niveles de registro disponibles.</p>	<p>Ninguno</p>
<p>Cuando inicie sesión en el nivel -Info, sea preciso y conciso. No especifique detalles de implementación como el nombre de clase y el nombre de función en los mensajes de registro. Describa el paso exacto o el error exacto en inglés normal.</p>	<p>El siguiente fragmento de código muestra un ejemplo de un mensaje correcto y un mensaje erróneo:</p> <pre>\$wfa_util->sendLog('WARN', "Removing volume: '.\$VolumeName); # Good Message</pre> <pre>\$wfa_util->sendLog('WARN', 'Invoking volume- destroy ZAPI: '.\$VolumeName); # Bad message</pre>

Directrices para la gestión de errores

Debe tener en cuenta las directrices para la gestión de errores al escribir un script PowerShell o Perl para OnCommand Workflow Automation (WFA).

Gestión de errores de PowerShell

Directrices	Ejemplo
<p>Los parámetros comunes añadidos a los cmdlets de PowerShell Runtime incluyen parámetros de control de errores como <code>ErrorAction</code> y <code>WarningAction</code>:</p> <ul style="list-style-type: none"> • El parámetro <code>ErrorAction</code> determina la forma en que un cmdlet debe reaccionar ante un error que no es de terminación del comando. • El parámetro <code>WarningAction</code> determina cómo debe reaccionar un cmdlet a una advertencia del comando. • <code>Stop</code>, <code>SilentlyContinue</code>, <code>Inquire</code> y <code>Continue</code> son los valores válidos para los parámetros <code>ErrorAction</code> y <code>WarningAction</code>. <p>Para obtener más información, puede utilizar la <code>Get-Help about_CommonParameters</code> En la CLI de PowerShell.</p>	<p>ErrorAction: El ejemplo siguiente muestra cómo manejar un error que no es de terminación como un error de terminación:</p> <pre>New-NcIgroup-Name \$IgroupName- Protocol \$Protocol-Type\$OSType- ErrorActionstop</pre> <p>WarningAction</p> <pre>New-VM-Name \$VMName-VM \$SourceVM- DataStore\$DataStoreName- VMHost\$VMHost- WarningActionSilentlyContinue</pre>
<p>Utilice la instrucción general <code>"try/catch"</code> si no se conoce el tipo de excepción entrante.</p>	<pre>try { "In Try/catch block" } catch { "Got exception" }</pre>
<p>Utilice la instrucción específica <code>«try/catch'»</code> si conoce el tipo de excepción entrante.</p>	<pre>try { "In Try/catch block" } catch[System.Net.WebException], [System.IO. IOException] { "Got exception" }</pre>

Directrices	Ejemplo
<p>Utilice la declaración «'finally'» para liberar recursos.</p>	<pre>try { "In Try/catch block" } catch { "Got exception" } finally { "Release resources" }</pre>
<p>Utilice variables automáticas de PowerShell para acceder a la información acerca de excepciones.</p>	<pre>try { Get-WFALogger -Info -message \$("Creating Ipspace: " + \$Ipspace) New-NetIPAddress -Name \$Ipspace } catch { Throw "Failed to create Ipspace. Message: " + \$_.Exception.Message; }</pre>

Gestión de errores Perl

Directrices	Ejemplo
<p>Perl no incluye compatibilidad con el idioma nativo para bloques try/catch. Utilice bloques de evaluación para comprobar y manejar errores. Mantenga los bloques de evaluación lo más pequeños posible.</p>	<pre>eval { \$wfa_util->sendLog('INFO', "Quiescing the relationship : \$DestinationCluster://\$Destination Vserver /\$DestinationVolume"); \$server->snapmirror_quiesce('destination-vserver' => \$DestinationVserver, 'destination-volume' => \$DestinationVolume); \$wfa_util->sendLog('INFO', 'Quiesce operation started successfully.');</pre> <pre>}; \$wfa_util->checkEvalFailure("Failed to quiesce the SnapMirror relationship \$DestinationCluster://\$Destination Vserver /\$DestinationVolume", \$_);</pre>

PowerShell general y convenciones Perl para WFA

Debe comprender ciertas convenciones de PowerShell y Perl que se utilizan en WFA para crear scripts consistentes con los scripts existentes.

- Utilice variables que le ayudarán a aclarar lo que desea que haga el script.
- Escriba un código legible que se pueda entender sin comentarios.
- Mantenga las secuencias de comandos y comandos tan simples como sea posible.
- Para scripts de PowerShell:
 - Use los cmdlets siempre que sea posible.
 - Invoque el código .NET cuando no haya ningún cmdlet disponible.
- Para scripts Perl:
 - Termine siempre las declaraciones "da" con caracteres de nueva línea.

En ausencia de un carácter de salto de línea, se imprime el número de línea de script, lo cual no es útil para depurar comandos Perl ejecutados por WFA.

- En el módulo «'getopt'», haga que los argumentos de cadena sean obligatorios en un comando.

Módulos Perl con Windows

Algunos módulos Perl están integrados con la distribución Perl de Windows Active state para OnCommand Workflow Automation (WFA). Puede utilizar estos módulos Perl en su código Perl para escribir comandos, solo si se combinan con Windows.

La siguiente tabla enumera los módulos de base de datos Perl que se combinan con Windows para WFA.

Módulo de base de datos	Descripción
DBD::mysql	Controlador de interfaz de base de datos Perl5 que permite conectarse a la base de datos MySQL.
Pruebe::diminuto	Minimiza los errores comunes con bloques de evaluación.
XML::libxml	Interfaz para libxml2 que proporciona a los analizadores XML y HTML con interfaces DOM, SAX y XMLReader.
DBD::Cassandra	Controlador de interfaz de base de datos Perl5 para Cassandra que utiliza el lenguaje de consulta CQL3.

Consideraciones sobre la adición de PowerShell y módulos Perl personalizados

Debe tener en cuenta diferentes consideraciones antes de agregar PowerShell personalizado y módulos Perl a OnCommand Workflow Automation (WFA). Los módulos Perl y PowerShell personalizados permiten usar comandos personalizados para crear flujos de trabajo.

- Durante la ejecución de comandos de WFA, todos los módulos personalizados de PowerShell se añaden al directorio de instalación de WFA /Posh/modules se importan automáticamente.
- Todos los módulos Perl personalizados se han añadido al WFA/perl el directorio se incluye en la biblioteca @INC.
- No se realiza un backup de los módulos Perl y PowerShell personalizados como parte de la operación de backup de WFA.
- Los módulos Perl y PowerShell personalizados no se restauran como parte de la operación de restauración de WFA.

Debe realizar manualmente la copia de seguridad de los módulos de PowerShell y Perl personalizados para copiarlos en una nueva instalación de WFA.

El nombre de la carpeta del directorio de módulos debe ser el mismo que el del nombre del módulo.

Cmdlets y funciones DE WFA

OnCommand Workflow Automation (WFA) proporciona varios cmdlets de PowerShell, así como funciones PowerShell y Perl que puede utilizar en sus comandos WFA.

Puede ver todos los cmdlets y las funciones de PowerShell que proporciona el servidor de WFA utilizando los siguientes comandos de PowerShell:

- `Get-Command -Module WFAWrapper`
- `Get-Command -Module WFA`

Puede ver todas las funciones Perl proporcionadas por el servidor WFA en el `WFAUtil.pm` módulo. Las secciones de ayuda, los cmdlets de WFA PowerShell ayudan y los métodos Perl de la ayuda del módulo de ayuda WFA Support Links permite el acceso a todos los cmdlets y funciones de PowerShell y a las funciones Perl.

Módulos de WFA PowerShell y Perl

Debe tener en cuenta la PowerShell o los módulos Perl de OnCommand Workflow Automation (WFA) para escribir scripts para sus flujos de trabajo.


Módulos de PowerShell

Directrices	Ejemplo
Utilice el kit de herramientas PS de Data ONTAP para invocar API siempre que el kit de herramientas esté disponible.	La <code>Add VLAN</code> command utiliza el kit de herramientas de la siguiente manera: <code>Add-NaNetVlan-Interface \$Interface-Vlans\$VlanID</code>
Si no hay cmdlets disponibles en el Kit de herramientas de PS de Data ONTAP, utilice <code>Invoke-SSH</code> Comando para invocar la CLI en Data ONTAP.	<code>Invoke-NaSsh-Name \$ArrayName-Command "ifconfig -a"-Credential \$Credentials</code>

Módulos Perl

El módulo `NaServer` se utiliza en los comandos WFA. El módulo `NaServer` permite la invocación de API Data ONTAP, que se utilizan en la administración activa de sistemas Data ONTAP.

Directrices	Ejemplo
<p>Utilice el módulo NaServer para invocar las API siempre que esté disponible el SDK de capacidad de gestión de NetApp.</p>	<p>El siguiente ejemplo muestra cómo se utiliza el módulo NaServer para reanudar la operación SnapMirror:</p> <pre> eval { \$wfa_util->sendLog('INFO', "Connecting to the cluster: \$DestinationCluster"); my \$server = \$wfa_util- >connect(\$DestinationClusterIp, \$DestinationVserver); my \$sm_info = \$server- >snapmirror_get('destination-vserver' => \$DestinationVserver, 'destination-volume' => \$DestinationVolume); my \$sm_state = \$sm_info- >{'attributes'}->{'snapmirror- info'}->{'mirror-state'}; my \$sm_status = \$sm_info- >{'attributes'}->{'snapmirror- info'}->{'relationship-status'}; \$wfa_util->sendLog('INFO', "SnapMirror relationship is \$sm_state (\$sm_status)"); if (\$sm_status ne 'quiesced') { \$wfa_util->sendLog('INFO', 'The status needs to be quiesced to resume transfer.');</pre> <pre> } else { my \$result = \$server- >snapmirror_resume('destination-vserver' => \$DestinationVserver, 'destination-volume' => \$DestinationVolume); \$wfa_util->sendLog('INFO', </pre>

Directrices	Ejemplo
<p>Si no hay disponible una API de Data ONTAP, invoque la CLI de Data ONTAP mediante el método <code>executeSysteminterd</code>.</p> <div>  <p>ExecuteSystemS4 no es compatible y actualmente sólo está disponible para Data ONTAP en 7-Mode.</p> </div>	Ninguno

Consideraciones que tener en cuenta al convertir comandos de PowerShell en Perl

Debe tener en cuenta diferentes consideraciones al convertir los comandos de PowerShell en Perl, porque PowerShell y Perl tienen diferentes funcionalidades.

Tipos de entrada de comandos

OnCommand Workflow Automation (WFA) permite a los diseñadores de flujos de trabajo utilizar cabinas y hash como entrada para el comando al definir un comando. Estos tipos de entrada no se pueden utilizar cuando el comando se define usando Perl. Si desea que un comando Perl acepte entradas array y hash, puede definir la entrada como una cadena en el diseñador. La definición de comandos puede analizar la entrada, que se pasa para crear una matriz o hash según sea necesario. La descripción de la entrada describe el formato en el que se espera la entrada.

```
my @input_as_array = split(',', $InputString); #Parse the input string of
format val1,val2 into an array

my %input_as_hash = split /[:=]/, $InputString; #Parse the input string of
format key1=val1;key2=val2 into a hash.
```

Declaración de PowerShell

Los siguientes ejemplos muestran cómo puede pasar una entrada de cabina a PowerShell y Perl. En los ejemplos se describe la entrada `CronMonth`, que especifica el mes en el que está programado que se ejecute el trabajo cron. Los valores válidos son números enteros -1 a 11. Un valor de -1 indica que la programación se ejecuta cada mes. Cualquier otro valor denota un mes específico, siendo 0 enero y 11 diciembre.

```
[parameter(Mandatory=$false, HelpMessage="Months in which the schedule
executes. This is a comma separated list of values from 0 through 11.
Value -1 means all months.")]
[ValidateRange(-1, 11)]
[array]$CronMonths,
```

Instrucción Perl


```

GetOptions(
    "Cluster=s"           => \$Cluster,
    "ScheduleName=s"      => \$ScheduleName,
    "Type=s"              => \$Type,
    "CronMonths=s"        => \$CronMonths,
) or die 'Illegal command parameters\n';

sub get_cron_months {
    return get_cron_input_hash('CronMonths', \$CronMonths, 'cron-month',
-1,
        11);
}

sub get_cron_input_hash {
    my $input_name = shift;
    my $input_value = shift;
    my $zapi_element = shift;
    my $low = shift;
    my $high = shift;
    my $exclude = shift;

    if (!defined $input_value) {
        return undef;
    }

    my @values = split(',', $input_value);

    foreach my $val (@values) {
        if ($val !~ /^[+-]?[0-9]+$/) {
            die
                "Invalid value '$input_value' for $input_name: $val must
be an integer.\n";
        }
        if ($val < $low || $val > $high) {
            die
                "Invalid value '$input_value' for $input_name: $val must
be from $low to $high.\n";
        }
        if (defined $exclude && $val == $exclude) {
            die
                "Invalid value '$input_value' for $input_name: $val is not
valid.\n";
        }
    }
    # do something
}

```

Definición de comandos

Una expresión de una línea en PowerShell que use un operador de canalización puede tener que ampliarse a múltiples bloques de sentencias en Perl para lograr la misma funcionalidad. En la siguiente tabla se muestra un ejemplo de uno de los comandos de espera.

Declaración de PowerShell	Instrucción Perl
<pre># Get the latest job which moves the specified volume to the specified aggregate. \$job = Get-NcJob -Query \$query</pre>	<pre>where {\$_ .JobDescription -eq "Split" + \$VolumeCloneName}</pre>
Select-Object -First 1 ----	<pre>my \$result = \$server- >job_get_iter('query' => {'job-type' => 'VOL_CLONE_SPLIT'}, 'desired-attributes' => { 'job-type' => '', 'job-description' => '', 'job-progress' => '', 'job-state' => '' }); my @jobarray; for my \$job (@{ \$result- >{'attributes-list'}}) { my \$description = \$job->{'job- description'}; if(\$description =~ /\$VolumeCloneName/) { push(@jobarray, \$job) } }</pre>

Directrices para los elementos básicos de WFA

Debe conocer las directrices para usar los elementos básicos de Workflow Automation.

Directrices para SQL en WFA

Debe estar al tanto de las directrices para usar SQL en OnCommand Workflow Automation (WFA) a fin de escribir consultas SQL para WFA.

SQL se utiliza en las siguientes ubicaciones de WFA:

- Consultas SQL para rellenar las entradas del usuario para su selección
- Consultas SQL para crear filtros para filtrar objetos de un tipo de entrada de diccionario específico
- Datos estáticos en tablas en la base de datos de juegos
- Tipo de origen de datos personalizado de tipo SQL en el que los datos deben extraerse de un origen de datos externo, como una base de datos de administración de configuración personalizada (CMDB).
- Consultas SQL para secuencias de comandos de reserva y verificación

Directrices	Ejemplo
Las palabras clave reservadas de SQL deben tener caracteres en mayúsculas.	<pre>SELECT vserver.name FROM cm_storage.vserver vserver</pre>
Los nombres de tablas y columnas deben tener caracteres en minúsculas.	Tabla: Agregado Columna: Used_Space_mb
Separe las palabras con un carácter de subrayado (_). No se permiten espacios.	rendimiento_de_cabina
El nombre de la tabla se define en singular. Una tabla es un conjunto de una o más entradas.	«'function'», no «'functions'»
Utilice alias de tabla con nombres significativos en consultas SELECCIONADAS.	<pre>SELECT vserver.name FROM cm_storage.cluster cluster, cm_storage.vserver vserver WHERE vserver.cluster_id = cluster.id AND cluster.name = '\${ClusterName}' AND vserver.type = 'cluster' ORDER BY vserver.name ASC</pre>

Directrices	Ejemplo
<p>Si tiene que hacer referencia a un parámetro de entrada de filtro o a un parámetro de entrada de usuario en una consulta de filtro o una consulta de usuario, utilice la sintaxis como <code>'\${inputVariableName}'</code>. también puede utilizar la sintaxis para hacer referencia a un parámetro de definición de comandos en secuencias de comandos de reserva y secuencias de comandos de verificación.</p>	<pre> SELECT volume.name AS Name, aggregate.name as Aggregate, volume.size_mb AS 'Total Size (MB) ', voulme.used_size_mb AS 'Used Size (MB) ', volume.space_guarantee AS 'Space Guarantee' FROM cm_storage.cluster, cm_storage.aggregate, cm_storage.vserver, cm_storage.volume WHERE cluster.id = vserver.cluster_id AND aggregate.id = volume.aggregate_id AND vserver.id = voulme.vserver_id AND vserver.name = '\${VserverName}' AND cluster.name = '\${ClusterName}' ORDER BY volume.name ASC </pre>
<p>Utilice comentarios para consultas complejas. Algunos de los estilos de comentario admitidos en las consultas son los siguientes:</p> <ul style="list-style-type: none"> • «»--» hasta el final de la línea <p>Es obligatorio un espacio después del segundo guión de este estilo de comentario.</p> <ul style="list-style-type: none"> • De un carácter «»#» hasta el final de la línea • De un «»/" to the following "/"secuencia 	<pre> /* multi-line comment */ --line comment SELECT ip as ip, # comment till end of this line NAME as name FROM --end of line comment storage.array </pre>

Directrices para las funciones de WFA

Puede crear funciones para encapsular la lógica más compleja y utilizada comúnmente en una función llamada y, a continuación, reutilizar la función como valores de parámetros de comandos o valores de parámetros de filtro en OnCommand Workflow Automation (WFA).

Directrices	Ejemplo
Utilice la caja Camel para un nombre de función.	CalculateVolumeSize
Los nombres de las variables deben estar en inglés normal y relacionados con la funcionalidad de la función.	SplitByDelimiter
No utilice abreviaturas.	CalculateVolumeSize, <i>not</i> calcVolSize
Las funciones se definen mediante MVFLEX Expression Language (MVEL).	Ninguno
La definición de la función debe especificarse de acuerdo con las directrices oficiales del lenguaje de programación Java.	Ninguno

Directrices para las entradas del diccionario WFA

Debe conocer las directrices para crear entradas de diccionario en OnCommand Workflow Automation (WFA).

Directrices	Ejemplo
Los nombres de las entradas del diccionario sólo deben contener caracteres alfanuméricos y guiones bajos.	Licencia_clúster Switch_23
Los nombres de las entradas del diccionario deben comenzar con un carácter en mayúsculas. Comience cada palabra en el nombre con un carácter en mayúsculas y separe cada palabra con un guión bajo (_).	Volumen Array_License
Los nombres de atributos de entrada de diccionario no deben incluir el nombre de la entrada de diccionario.	Ninguno
Los atributos y las referencias de una entrada de diccionario deben tener caracteres en minúsculas.	agregado, size_mb

Directrices	Ejemplo
Separe las palabras con un guión bajo. No se permiten espacios.	pool_recursos
Las entradas del diccionario no pueden incluir referencias de un esquema diferente. Cuando una entrada de diccionario requiere referencias cruzadas a un objeto de un esquema diferente, asegúrese de que todas las claves naturales del objeto al que se hace referencia estén presentes en la entrada de diccionario.	La entrada del diccionario Array_Performance requiere todas las claves naturales de la entrada del diccionario Array como atributos directos en él.
Utilice los tipos de datos adecuados para los atributos.	Ninguno
Utilice el tipo de datos Long para los atributos relacionados con el tamaño o el espacio.	Size_mb y available_size_mb en la entrada del diccionario Storage.Volume
Utilice Enum cuando un atributo tenga un conjunto fijo de valores.	raid_TYPE en la entrada del diccionario Storage.Volume
Establezca "para ser almacenado en caché" como true para un atributo o referencia cuando un origen de datos proporcione valor para ese atributo o referencia. para el origen de datos de Active IQ Unified Manager, agregue atributos en caché si el origen de datos puede proporcionarlo.	Ninguno
Establezca "puede ser Nulo" como true si el origen de datos que proporciona el valor para este atributo o referencia puede devolver NULL.	Ninguno
Proporcione una descripción significativa de cada atributo y referencia. la descripción se muestra en los detalles del comando al diseñar un flujo de trabajo.	Ninguno
No utilice «id» como nombre de un atributo en las entradas del diccionario. Está reservado para uso interno de WFA.	Ninguno

Información relacionada

[Referencias al material de aprendizaje](#)

Directrices para comandos

Debe tener en cuenta las directrices para crear comandos en OnCommand Workflow Automation (WFA).

Directrices	Ejemplo
Utilice un nombre fácilmente identificable para los comandos.	Create Qtree
Utilice espacios para delimitar palabras y cada palabra debe comenzar con un carácter en mayúscula.	Create Volume
Proporcione una descripción para explicar la funcionalidad del comando, incluido el resultado esperado de los parámetros opcionales.	Ninguno
De manera predeterminada, el tiempo de espera para los comandos estándar es de 600 segundos. El tiempo de espera predeterminado se configura al crear el comando. Cambie el valor predeterminado solo si el comando puede tardar más tiempo en completarse.	Create Volume comando
En caso de operaciones de ejecución prolongada, cree dos comandos, uno para invocar la operación de ejecución prolongada y otro para informar periódicamente sobre el progreso de la operación. El primer comando debería ser un Standard Execution el tipo de comando y el segundo debería ser Wait for Condition tipo de comando.	Create VSM y.. Wait for VSM comandos
Anteponga el Wait for condition Nombres de comandos con "wait" para facilitar la identificación.	Wait for CM Volume Move
Utilizar un intervalo de espera adecuado para los comandos "wait for condition". El valor especificado rige el intervalo en el que se ejecuta el comando Polling para comprobar si la operación de ejecución prolongada ha finalizado.	60 intervalo de muestreo para Wait for VSM comando
Para la Wait for condition comandos, utilice un tiempo de espera apropiado según el tiempo esperado para completar la operación de ejecución prolongada. El tiempo esperado podría ser considerablemente mayor si la operación implica transferencia de datos a través de una red.	Una transferencia de base VSM puede tardar varios días en completarse. Por lo tanto, el tiempo de espera especificado es de 6 días.

Representación de cadena

La representación de cadena de un comando muestra los detalles de un comando en un diseño de flujo de trabajo durante la planificación y ejecución. Sólo se pueden utilizar los parámetros de comando en la representación de cadena de un comando.

Directrices	Ejemplo
Evite utilizar atributos que no tengan ningún valor. Un atributo sin valor se muestra como NA.	VolName 10.68.66.212[NA]aggr1/testVol7
Separar diferentes entradas en la representación de cadena usando los siguientes delimitadores: [] , / :	<i>ArrayName [ArrayIp]</i>
Proporcione etiquetas significativas a todos los valores de la representación de cadenas.	<i>Volume name=VolumeName</i>

Lenguaje de definición de comandos

Los comandos se pueden escribir utilizando los siguientes lenguajes de secuencias de comandos compatibles:

- PowerShell
- Perl

Definición de parámetros de comando

Los parámetros de comando se describen por Nombre, Descripción, Tipo, valor predeterminado del parámetro y si el parámetro es obligatorio. El tipo de parámetro puede ser String, Boolean, Integer, Long, Double, Enumeración, DateTime, capacidad, matriz, Hashtable, Una contraseña o un XmlDocument. Aunque los valores para la mayoría de los tipos son intuitivos, los valores para Array y Hashtable deben tener un formato determinado tal como se describe en la siguiente tabla:

Directrices	Ejemplo
Asegúrese de que el valor de un tipo de entrada Array es una lista de valores, separados por comas.	<pre>[parameter (Mandatory=\$false, HelpMessage="Months in which the schedule executes.")] [array] \$CronMonths</pre> <p>La entrada se pasa como sigue: 0,3,6,9</p>
Asegúrese de que el valor de un tipo de entrada Hashtable es una lista de pares clave=valor, separados por punto y coma.	<pre>[parameter (Mandatory=\$false, HelpMessage="Volume names and size (in MB) ")] [hashtable] \$VolumeNamesAndSize</pre> <p>La entrada se pasa como sigue: Volume1=100;Volume2=250;Volume3=50</p>

Directrices para flujos de trabajo

Debe tener en cuenta las directrices para crear o modificar un flujo de trabajo predefinido para OnCommand Workflow Automation (WFA).

Directrices generales

Directrices	Ejemplo
Asigne un nombre al flujo de trabajo de modo que refleje la operación que ejecuta el operador de almacenamiento.	Create a CIFS Share
En el caso de los nombres de flujo de trabajo, capitalice la letra inicial de la primera palabra y cada palabra que sea un objeto. Capitalice letras para abreviaturas y acrónimos.	Volumen Qtree Cree un recurso compartido CIFS Qtree de Clustered Data ONTAP
En las descripciones del flujo de trabajo, incluya todos los pasos importantes del flujo de trabajo, incluidos los requisitos previos, el resultado del flujo de trabajo o aspectos condicionales de la ejecución.	Consulte la descripción del flujo de trabajo de ejemplo Create VMware NFS Datastore on Clustered Data ONTAP Storage, lo que incluye los requisitos previos.
Establezca "preparado para la producción" en <code>true</code> sólo cuando el flujo de trabajo esté listo para la producción y se pueda mostrar en la página del portal.	Ninguno
De forma predeterminada, establezca «considerar elementos reservados» en <code>true</code> . Al previsualizar un flujo de trabajo para su ejecución, el planificador de WFA considera todos los objetos que están reservados junto con los objetos existentes en la base de datos de caché. Los efectos de otros flujos de trabajo programados o los flujos de trabajo que se ejecutan en paralelo se tienen en cuenta al planificar un flujo de trabajo específico si esta opción está establecida en <code>true</code> .	<ul style="list-style-type: none">• Situación 1 El flujo de trabajo 1 crea un volumen y está programado para ejecutarse una semana después. El flujo de trabajo 2 crea qtrees o LUN en los volúmenes en los que se busca y, si el flujo de trabajo 2 se ejecuta en un día aproximadamente, debería desactivar «considerar elementos reservados» para el flujo de trabajo 2 para impedir que se considere el volumen que debe crearse en una semana.• Situación 2 El flujo de trabajo 1 utiliza la <code>Create Volume</code> comando. Si hay un flujo de trabajo programado 2 que consume 100 GB de un agregado, el flujo de trabajo 1 debe tener en cuenta los requisitos para el flujo de trabajo 2 durante la planificación.

Directrices	Ejemplo
De forma predeterminada, ""Habilitar validación de existencia de elementos"" se establece en <code>true</code> .	<ul style="list-style-type: none"> Situación 1 <p>Si crea un flujo de trabajo que primero quita un volumen por nombre mediante el comando <code>Remove Volume</code> solo si hay un volumen y se vuelve a crear con otro comando, como <code>Create Volume</code> o <code>Clone Volume</code>, a continuación, el flujo de trabajo no debe utilizar este indicador. El efecto de eliminar el volumen no estará disponible para el <code>Create volume</code> comando, lo que provoca un error en el flujo de trabajo.</p> <ul style="list-style-type: none"> Situación 2 <p>La <code>Create Volume</code> el comando se utiliza en un flujo de trabajo con un nombre específico denominado «'vol198'».</p> <p>Si esta opción está establecida en <code>true</code>, el planificador de WFA comprueba durante la planificación para ver si existe un volumen con ese nombre en la cabina determinada. Si hay un volumen, el flujo de trabajo falla durante la planificación.</p>
Cuando se selecciona el mismo comando más de una vez en un flujo de trabajo, proporcione los nombres de visualización adecuados para las instancias de comandos.	El flujo de trabajo de ejemplo «'Crear, asignar y proteger las LUN con SnapVault'» utiliza <code>Create Volume</code> comando dos veces. Sin embargo, utiliza los nombres de visualización como <code>Create Primary Volume</code> y <code>Create Secondary Volume</code> adecuadamente para el volumen primario y el volumen de destino reflejado.

Entradas del usuario

Directrices	Ejemplo
<p>Nombres:</p> <ul style="list-style-type: none"> Inicie el nombre con el carácter « »\$». Utilice una letra mayúscula al principio de cada palabra. Utilice letras mayúsculas para todos los términos y abreviaturas. No utilice guiones bajos. 	<p>\$Array</p> <p>\$VolumeName</p>

Directrices	Ejemplo
<p>Nombres para mostrar:</p> <ul style="list-style-type: none"> • Utilice una letra mayúscula al principio de cada palabra. • Separe las palabras con espacios. • Si las entradas tienen unidades específicas, especifique la unidad entre paréntesis directamente en el nombre de visualización. 	<p>Volume Name</p> <p>Volume Size (MB)</p>
<p>Descripciones:</p> <ul style="list-style-type: none"> • Proporcione una descripción significativa para cada información de usuario. • Proporcione ejemplos cuando sea necesario. <p>Debe hacer esto especialmente cuando se espera que la entrada del usuario esté en un formato específico.</p> <p>Las descripciones de entrada del usuario se muestran como información sobre herramientas para las entradas del usuario durante la ejecución del flujo de trabajo.</p>	<p>Iniciadores que se van a agregar a un «iGroup». Por ejemplo, IQN o WWPN del iniciador.</p>
<p>Escriba: Seleccione Enum como el tipo si desea restringir la entrada a un conjunto específico de valores.</p>	<p>Protocolo: «iscsi», «fcp», «mixta»</p>
<p>Tipo: Seleccione Query como tipo cuando el usuario pueda seleccionar de entre los valores disponibles en la caché de WFA.</p>	<p>\$Array: Tipo DE CONSULTA con la siguiente consulta:</p> <pre>SELECT ip, name FROM storage.array</pre>
<p>Tipo: Marque la entrada del usuario como bloqueada cuando la entrada del usuario deba restringirse a los valores que se obtienen de una consulta o estar restringida sólo a los tipos de Enum admitidos.</p>	<p>\$Array: Tipo de consulta bloqueado: Sólo se pueden seleccionar las matrices de la caché.\$Protocol: Tipo de Enum bloqueado con valores válidos como iscsi, fcp, mixto. No se admite ningún otro valor distinto del válido.</p>

Directrices	Ejemplo
<p>Tipo: Tipo de query Agregue columnas adicionales como valores devueltos en la consulta cuando ayude al operador de almacenamiento a elegir correctamente la entrada del usuario.</p>	<p>\$aggregate: Proporcione nombre, tamaño total y tamaño disponible para que el operador conozca los atributos antes de seleccionar el agregado.</p>
<p>Tipo: Consulta TypeSQL para entradas de usuario puede hacer referencia a cualquier otra entrada de usuario anterior. Esto puede utilizarse para limitar los resultados de una consulta basada en otras entradas de usuario, como unidades vFiler de una cabina, volúmenes de un agregado o LUN de una máquina virtual de almacenamiento (SVM).</p>	<p>En el flujo de trabajo de ejemplo Create a Clustered Data ONTAP Volume, La consulta para VserverName es la siguiente:</p> <pre data-bbox="820 478 1484 1066"> SELECT vserver.name FROM cm_storage.cluster cluster, cm_storage.vserver vserver WHERE vserver.cluster_id = cluster.id AND cluster.name = '\${ClusterName}' AND vserver.type = 'cluster' ORDER BY vserver.name ASC </pre> <p>La consulta hace referencia a \${ClusterName}, donde \$ClusterName es el nombre de la entrada de usuario que precede a la entrada de usuario \$VserverName.</p>
<p>Tipo: Utilice el tipo booleano con valores como "true, false" para las entradas de usuario que son de naturaleza booleana. Esto ayuda a escribir expresiones internas en el diseño del flujo de trabajo utilizando la entrada del usuario directamente. Por ejemplo, \$UserInputName en lugar de \$UserInputName == "Yes".</p>	<p>\$CreateCIFSShare: Tipo booleano con valores válidos como "true" o "false"</p>
<p>Tipo: para tipo de cadena y número, utilice expresiones regulares en la columna valores cuando desee validar el valor con formatos específicos.</p> <p>Utilice expresiones regulares para las entradas de dirección IP y máscara de red.</p>	<p>La entrada de usuario específica de la ubicación puede expresarse como "[A-Z][A-Z]\-0[1-9]". Esta información del usuario acepta valores como «US-01», «NB-02», pero no «nb-00».</p>
<p>Tipo: Para el tipo de número, se puede especificar una validación basada en rango en la columna valores.</p>	<p>Para el número de LUN que se van a crear, la entrada en la columna valores es 1-20.</p>

Directrices	Ejemplo
Grupo: Agrupar las entradas de usuario relacionadas en los cubos apropiados y nombrar al grupo.	«Información de almacenamiento» para todas las entradas de los usuarios relacionadas con el almacenamiento. «Detalles de datastore» para todas las entradas de usuario relacionadas con VMware.
Obligatorio: Si el valor de cualquier entrada de usuario es necesario para que el flujo de trabajo se ejecute, marque la entrada de usuario como obligatoria. Esto garantiza que la pantalla de entrada del usuario acepte mandatorily esa entrada del usuario.	«»\$VolumeName» en el flujo de trabajo «'Create NFS Volume».
Valor predeterminado: Si una entrada de usuario tiene un valor predeterminado que puede funcionar para la mayoría de las ejecuciones del flujo de trabajo, proporcione los valores. Esto ayuda a permitir al usuario proporcionar menos entradas durante la ejecución, si el valor predeterminado cumple con el propósito.	Ninguno

Constantes, variables y parámetros de retorno

Directrices	Ejemplo
Constantes: Defina constantes cuando se utiliza un valor común para definir parámetros en varios comandos.	<i>AGGREGATE_OVERPROMISO_THRESHOLD</i> en Create, map, and protect LUNs with SnapVault sample workflow.
Constantes: nombres <ul style="list-style-type: none"> • Utilice una letra mayúscula al principio de cada palabra. • Utilice letras mayúsculas para todos los términos y abreviaturas. • No utilice guiones bajos. • Utilice letras mayúsculas para todas las letras de nombres constantes. 	<i>AGGREGATE_USED_SPACE_THRESHOLD</i> <i>ActualVolumeSizeInMB</i>
Variables: Proporcione un nombre a un objeto definido en uno de los cuadros de parámetros de comando. Las variables se generan automáticamente nombres y se pueden cambiar.	Ninguno
Variables: Los nombres utilizan caracteres en minúscula para los nombres de variables.	volume1 recurso_compartido_cifs

<p>Parámetros de retorno: Utilice parámetros de retorno cuando la planificación y ejecución del flujo de trabajo devuelva algunos valores calculados o seleccionados durante la planificación. Los valores se ponen a disposición en el modo de vista previa cuando el flujo de trabajo se ejecuta también desde un servicio web.</p>	<p>Agregado: Si se selecciona el agregado mediante la lógica de selección de recursos, el agregado seleccionado real se puede definir como un parámetro return.</p>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------

Directrices para crear scripts de validación para tipos de sistema remoto

Debe tener en cuenta las directrices para crear scripts de validación que se utilicen para probar los tipos de sistema remoto que defina en OnCommand Workflow Automation (WFA).

- El script Perl que cree debe ser similar al script de ejemplo que se proporciona en la ventana Script de validación.
- El resultado del script de validación debe ser similar al del script de muestra.

Ejemplo de script de validación

```
# Check connectivity.
# Return 1 on success.
# Return 0 on failure and set $message
sub checkCredentials {
my ($host, $user, $passwd, $protocol, $port, $timeout) = @_;
#
# Please add the code to check connectivity to $host using $protocol here.
#
return 1;
}
```

Directrices para crear tipos de origen de datos

Debe tener en cuenta las directrices para crear los tipos de origen de datos que se utilizan para definir orígenes de datos personalizados para OnCommand Workflow Automation (WFA).

Puede definir un tipo de origen de datos mediante uno de los siguientes métodos:

- SQL: Puede utilizar las directrices de WFA SQL para definir consultas seleccionadas de orígenes de datos basadas en una base de datos externa.
- SCRIPT: Puede escribir una secuencia de comandos de PowerShell que proporcione los datos de un esquema específico de entradas del diccionario.

Las directrices para crear tipos de origen de datos son las siguientes:

- Se debe utilizar el idioma de PowerShell para crear un script.
- El script de PowerShell debe proporcionar la salida de cada entrada de diccionario en su directorio de

trabajo actual.

- Se debe dar nombre a los archivos de datos `dictionary_entry.csv`, donde el nombre de la entrada del diccionario debe tener caracteres en minúsculas.

El tipo de origen de datos predefinido que recopila información de Performance Advisor utiliza un tipo de origen de datos basado en SCRIPTS. Se denomina a los archivos de salida `array_performance.csv` y `aggregate_performance.csv`.

- La `.csv` el archivo debe incluir el contenido en el orden exacto de los atributos de entrada del diccionario.

Una entrada de diccionario incluye atributos en el siguiente orden: `Array_ip`, `date`, `day`, `hour`, `cpu_busy`, `total_ops_por_seg`, `disk_throughput_per_s`.

El script de PowerShell añade datos al `.csv` archivar en el mismo orden.

```
$values = get-Array-CounterValueString ([REF]$data)
Add-Content $arrayFile ([byte[]][char[]] "`n"
t$arrayIP't$date't$day't$hour't$values'n")
```

- Debe utilizar codificación para asegurarse de que la salida de datos del script se carga en la caché de WFA de forma precisa.
- Debe utilizar `\N` al introducir un valor Null en `.csv` archivo.

Palabras reservadas

OnCommand Workflow Automation (WFA) incluye algunas palabras reservadas. No debe utilizar las palabras reservadas en los flujos de trabajo para ningún atributo o parámetro como nombres de variables, entradas de usuario, constantes y parámetros de retorno.

A continuación se muestra una lista de las palabras reservadas de WFA:

<ul style="list-style-type: none"> • y.. • Cabina • afirmar • Booleano • booleano • Byte • byte • caract • Carácter • CharSequence • Clase • Cargador de clases • Compilador • contiene • convertible_to • def • sí • Doble • doble • el resto • vacío • falso 	<ul style="list-style-type: none"> • Flotante • flotante • para • foreach • función • si • importar • import_static • pulg • instanceof • int • Entero • es • isdef • Largo • largo • Matemáticas • nuevo • nulo • Número • Objeto • o. 	<ul style="list-style-type: none"> • proto • retorno • Tiempo de ejecución • SecurityManager • Corto • corto • sonido • StrictMath • Cadena • StringBuffer • StringBuilder • la calle • conmutador • Sistema • Proceso • Threadlocal • verdadero • hasta • var • Vacío • aunque • con
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Cómo se utilizan las API DE REST

Puede utilizar las API DE REST que proporciona Workflow Automation (WFA) para iniciar flujos de trabajo desde portales externos y el software de orquestación del centro de datos. WFA admite tipos de contenido XML y JSON para todas las API REST.

WFA permite que los servicios externos accedan a diversas colecciones de recursos, como flujos de trabajo, usuarios, filtros e buscadores, A través de rutas de URI. Los servicios externos pueden utilizar métodos HTTP como GET, PUT, POST y DELETE. En estos URI para realizar operaciones CRUD en los recursos.

Puede realizar varias acciones a través de las API REST de WFA, incluidas las siguientes:

- Acceder a definiciones y metadatos de flujos de trabajo.
- Ejecutar flujos de trabajo y supervisar su ejecución.
- Ver usuarios y roles, y cambiar contraseñas.
- Ejecutar y probar filtros de selección de recursos.
- Ejecutar y probar buscadores de recursos.

- Administrar credenciales de almacenamiento u otros objetos de centros de datos.
- Ver orígenes de datos y tipos de orígenes de datos.

REST Documentation tiene más información sobre las API DE REST:

`https://wfa_server_ip:port/rest/docs`

`wfa_server_ip` Es la dirección IP del servidor y el puerto de WFA es el número de puerto TCP que ha utilizado para el servidor WFA durante la instalación.



WFA comprueba los tokens de falsificación de solicitudes entre sitios (CSRF) para las solicitudes que proceden de la interfaz de usuario web. Sin embargo, WFA no comprueba los tokens CSRF si existen solicitudes REST entrantes que provienen de clientes REST o de software de orquestación.

Referencias al material de aprendizaje

Debe conocer ciertas prácticas de programación y secuencias de comandos para crear flujos de trabajo avanzados de automatización de flujo de trabajo (WFA). Puede utilizar material de referencia para obtener información sobre las opciones necesarias antes de crear elementos básicos o flujos de trabajo de WFA.

Windows PowerShell

WFA usa scripts de PowerShell para operaciones de flujo de trabajo. En la siguiente tabla se incluyen referencias de material de aprendizaje para PowerShell:

Introducción a Windows PowerShell	http://msdn.microsoft.com/en-us/library/windows/desktop/aa973757(v=vs.85).aspx
Desarrollo PowerShell — entorno de scripting integrado (ISE)	https://docs.microsoft.com/en-us/powershell/scripting/windows-powershell/ise/introducing-the-windows-powershell-ise?view=powershell-7.2
<i>.NET Directrices de nomenclatura del marco</i>	http://msdn.microsoft.com/en-us/library/xzf533w0%28v=vs.71%29.aspx
Estilo de código PowerShell	http://get-powershell.com/post/2011/04/13/Extra-Points-for-Style-when-writing-PowerShell-Code.aspx
PowerShell: Intentar/capturar finalmente	http://technet.microsoft.com/en-us/library/dd315350.aspx
Variables automáticas de PowerShell	http://technet.microsoft.com/en-us/library/dd347675.aspx

Generación de informes de error de PowerShell	https://docs.microsoft.com/en-us/powershell/scripting/developer/cmdlet/error-reporting-concepts?view=powershell-7.2
Parámetros comunes de PowerShell	https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_commonparameters?view=powershell-7.2

Kit de herramientas PowerShell de Data ONTAP

El kit de herramientas PowerShell de Data ONTAP se incluye junto con WFA. Puede utilizar los cmdlets del kit de herramientas de PowerShell para invocar comandos de Data ONTAP desde un script de PowerShell. Para obtener más información, consulte la Ayuda_ del _Data ONTAP PowerShell Toolkit, a la que puede acceder desde la siguiente ubicación:

`WFA_install_location\WFA\PoSH\Modules\DataONTAP\webhelp\index.html`.

`WFA_install_location` Es el directorio de instalación de WFA, y `C:\Program Files\NetApp` es el directorio de instalación predeterminado.

En la siguiente tabla se incluyen referencias a información acerca del kit de herramientas de PowerShell para Data ONTAP:

Artículos del kit de herramientas PowerShell de ONTAP	https://community.netapp.com/t5/Tech-OnTap-Articles/The-Data-ONTAP-PowerShell-Toolkit/ta-p/85933
Kit de herramientas PowerShell de ONTAP Comunidad de NetApp	https://community.netapp.com/t5/forums/filteredbylabelpage/board-id/microsoft-cloud-and-virtualization-discussions/label-name/powershell%20toolkit

Perl

WFA admite comandos Perl para operaciones de flujo de trabajo. Al instalar WFA, se instalan los módulos Perl y Perl necesarios en el servidor WFA.

"Guía del usuario de ActivePerl"

También puede acceder a la *ActivePerl User Guide* desde la siguiente ubicación:

`WFA_install_location\WFA\Perl64\html\index.html`.

`WFA_install_location` Es el directorio de instalación de WFA, y `C:\Program Files\NetApp` is the default installation directory. "WFA utiliza scripts Perl para operaciones de flujo de trabajo. La siguiente tabla incluye referencias al material de aprendizaje para Perl:

Perl moderno: 2014	http://modernperlbooks.com/books/modern_perl_2014/index.html
Documentación de programación Perl	http://perldoc.perl.org/

SDK de gestión de NetApp

Los módulos Perl necesarios del SDK para la capacidad de gestión de NetApp se combinan con WFA. Estos módulos Perl son necesarios para utilizar los comandos Perl de WFA. Para obtener más información, consulte la documentación de SDK para facilitar la gestión de NetApp, a la que se puede acceder desde la siguiente ubicación: `WFA_install_location\WFA\perl\NMSDK\html`.

`WFA_install_location` Es el directorio de instalación de WFA, y `C:\Program Files\NetApp` es el directorio de instalación predeterminado.

Lenguaje de consultas estructurado (SQL)

La sintaxis SQL SELECT se utiliza en filtros y para rellenar las entradas del usuario.

["SINTAXIS DE SELECCIÓN de MySQL"](#)

MVFLEX Expression Language (MVEL)

Puede utilizar las sintaxis de expresiones MVEL en flujos de trabajo de WFA, por ejemplo, en funciones y variables.

Para obtener más información, consulte *MVEL Language Guide*.

Expresiones regulares

Puede utilizar la expresión regular (regex) en WFA.

["ActionScript 3.0 mediante expresiones regulares"](#)

Documentación relacionada para OnCommand Workflow Automation

Hay documentos y herramientas adicionales que le ayudarán a aprender a realizar una configuración más avanzada de su servidor OnCommand Workflow Automation (WFA).

Otras referencias

El espacio de Workflow Automation dentro de la comunidad de NetApp proporciona recursos de aprendizaje adicionales, incluidos los siguientes:

- **Comunidad de NetApp**

["Comunidad de NetApp: Automatización de flujos de trabajo \(WFA\)"](#)

Referencias de herramientas

- **Matriz de interoperabilidad**

Enumera las combinaciones compatibles de componentes de hardware y versiones de software.

"Matriz de interoperabilidad"

Información de copyright

Copyright © 2025 NetApp, Inc. Todos los derechos reservados. Imprimido en EE. UU. No se puede reproducir este documento protegido por copyright ni parte del mismo de ninguna forma ni por ningún medio (gráfico, electrónico o mecánico, incluidas fotocopias, grabaciones o almacenamiento en un sistema de recuperación electrónico) sin la autorización previa y por escrito del propietario del copyright.

El software derivado del material de NetApp con copyright está sujeto a la siguiente licencia y exención de responsabilidad:

ESTE SOFTWARE LO PROPORCIONA NETAPP «TAL CUAL» Y SIN NINGUNA GARANTÍA EXPRESA O IMPLÍCITA, INCLUYENDO, SIN LIMITAR, LAS GARANTÍAS IMPLÍCITAS DE COMERCIALIZACIÓN O IDONEIDAD PARA UN FIN CONCRETO, CUYA RESPONSABILIDAD QUEDA EXIMIDA POR EL PRESENTE DOCUMENTO. EN NINGÚN CASO NETAPP SERÁ RESPONSABLE DE NINGÚN DAÑO DIRECTO, INDIRECTO, ESPECIAL, EJEMPLAR O RESULTANTE (INCLUYENDO, ENTRE OTROS, LA OBTENCIÓN DE BIENES O SERVICIOS SUSTITUTIVOS, PÉRDIDA DE USO, DE DATOS O DE BENEFICIOS, O INTERRUPCIÓN DE LA ACTIVIDAD EMPRESARIAL) CUALQUIERA SEA EL MODO EN EL QUE SE PRODUJERON Y LA TEORÍA DE RESPONSABILIDAD QUE SE APLIQUE, YA SEA EN CONTRATO, RESPONSABILIDAD OBJETIVA O AGRAVIO (INCLUIDA LA NEGLIGENCIA U OTRO TIPO), QUE SURJAN DE ALGÚN MODO DEL USO DE ESTE SOFTWARE, INCLUSO SI HUBIEREN SIDO ADVERTIDOS DE LA POSIBILIDAD DE TALES DAÑOS.

NetApp se reserva el derecho de modificar cualquiera de los productos aquí descritos en cualquier momento y sin aviso previo. NetApp no asume ningún tipo de responsabilidad que surja del uso de los productos aquí descritos, excepto aquello expresamente acordado por escrito por parte de NetApp. El uso o adquisición de este producto no lleva implícita ninguna licencia con derechos de patente, de marcas comerciales o cualquier otro derecho de propiedad intelectual de NetApp.

Es posible que el producto que se describe en este manual esté protegido por una o más patentes de EE. UU., patentes extranjeras o solicitudes pendientes.

LEYENDA DE DERECHOS LIMITADOS: el uso, la copia o la divulgación por parte del gobierno están sujetos a las restricciones establecidas en el subpárrafo (b)(3) de los derechos de datos técnicos y productos no comerciales de DFARS 252.227-7013 (FEB de 2014) y FAR 52.227-19 (DIC de 2007).

Los datos aquí contenidos pertenecen a un producto comercial o servicio comercial (como se define en FAR 2.101) y son propiedad de NetApp, Inc. Todos los datos técnicos y el software informático de NetApp que se proporcionan en este Acuerdo tienen una naturaleza comercial y se han desarrollado exclusivamente con fondos privados. El Gobierno de EE. UU. tiene una licencia limitada, irrevocable, no exclusiva, no transferible, no sublicenciable y de alcance mundial para utilizar los Datos en relación con el contrato del Gobierno de los Estados Unidos bajo el cual se proporcionaron los Datos. Excepto que aquí se disponga lo contrario, los Datos no se pueden utilizar, desvelar, reproducir, modificar, interpretar o mostrar sin la previa aprobación por escrito de NetApp, Inc. Los derechos de licencia del Gobierno de los Estados Unidos de América y su Departamento de Defensa se limitan a los derechos identificados en la cláusula 252.227-7015(b) de la sección DFARS (FEB de 2014).

Información de la marca comercial

NETAPP, el logotipo de NETAPP y las marcas que constan en <http://www.netapp.com/TM> son marcas comerciales de NetApp, Inc. El resto de nombres de empresa y de producto pueden ser marcas comerciales de sus respectivos propietarios.