

Primer Workflow: del diseño al resultado

Un ejercicio pequeño para recorrer Mapping, Session, Workflow y Monitor.

Antes de comenzar

Prepara una ubicación de laboratorio en el tenant seleccionado y permisos para crear, publicar y ejecutar artefactos. Necesitas un worker compatible con los archivos y una salida que puedas revisar. El objetivo es conservar clientes activos, no probar un volumen alto. Usa únicamente datos sintéticos.

Los tres artefactos tienen funciones diferentes: Mapping transforma; Session fija versión y contexto de ejecución; Workflow coordina la tarea. Revisa cada publicación antes de avanzar al siguiente nivel.

Procedimiento

1. Crea un CSV UTF-8 con cabecera `id,name,status` y tres filas sintéticas. Incluye dos filas con `active` y una con `inactive`.
2. En Mappings, conecta Flat File input → Row filter → Field selector → Flat File output.
3. Configura Row filter con `field=status`, `operator>equals`, `value=active`. En Field selector usa `fields=["id", "name"]`. Activa la cabecera del archivo de salida.
4. Valida, guarda el borrador y publica el Mapping. Crea una Session referenciando esa versión, configura worker y bindings de archivos y publica la Session.
5. Crea Start → Session → End en Workflows. Selecciona la versión publicada de la Session, valida, guarda y publica.
6. Ejecuta manualmente y abre el intento en Monitor.

Ejemplo sintético

Entrada:

```
id,name,status
1,Jorge,active
2,Ana,inactive
3,Bea,active
```

Copiar

Salida esperada con los ajustes indicados:

```
id,name
1,Jorge
3,Bea
```

Copiar

Los nombres e identificadores son ficticios. El archivo debe estar disponible para el runtime elegido; escribir su nombre en la descripción no crea un binding.

Comprobar el resultado

Confirma estado terminal del Workflow y sus tareas, dos filas de datos, cabecera `id, name` y ausencia de `status`. Revisa que la Session use la versión que acabas de publicar. Un archivo de un intento anterior no demuestra el resultado actual; usa una salida exclusiva o un mecanismo de identificación del intento.

Solo tras esta aceptación considera un calendario. Revisa zona horaria, próxima ejecución y permisos. Una programación activa no sustituye la comprobación del siguiente intento.

Resolver problemas

Si no hay filas, compara exactamente `active` con los valores: `equals` distingue mayúsculas. Si aparece `status`, comprueba Field selector y la versión consumida. Si el run espera, revisa capacidad y política del worker. Si falta el archivo, corrige el binding de Session. No repitas escrituras a ciegas tras un timeout: consulta Monitor. DovaAI explica qué componentes usar y cómo configurarlos; tú creas, modificas, publicas y ejecutas los artefactos.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Normalizar clientes con reglas verificables

Limpia texto, reemplaza un valor exacto y limita las columnas de salida.

Antes de comenzar

Define primero la regla: recortar espacios de `name`, cambiar exactamente `jorge` por `Jorge` y conservar únicamente `name` y `amount`. No es una regla general de capitalización. Usa un CSV de laboratorio con cabecera, codificación UTF-8 y un destino independiente. Prepara permisos y un runtime compatible antes de probar.

No necesitas una expresión SQL para ese cambio literal. Para una regla general de primera letra mayúscula, Clean text ofrece `capitalize`; para cada palabra, `title`. Este ejemplo conserva deliberadamente la regla exacta con Replace values. Calculated field interpola texto y no ejecuta `LEFT` ni fórmulas.

Procedimiento

1. Añade Flat File input con `format=csv`, `delimiter=,` y `hasHeader=true`. Revisa nombres y cabecera.
2. Añade Clean text con `fields=["name"]` y `operations=["trim"]`. Así el valor con espacios puede coincidir con la regla exacta posterior.
3. Conecta Replace values y configura `field=name`, `find=jorge`, `replacement=Jorge`, `matchMode=exact`, `caseSensitive=true`.
4. Añade Field selector con `fields=["name", "amount"]`. Este paso elimina las demás columnas del esquema y de cada fila.
5. Conecta Flat File output con `includeHeader=true` y un nombre de prueba. Valida, guarda y publica; prepara la Session y el Workflow para ejecutar.

Ejemplo sintético

Entrada:

```
name,amount,internal_note
jorge ,12.5,sample-a
ana,8,sample-b
```

Copiar

Salida esperada:

```
name,amount
Jorge,12.5
ana,8
```

Copiar

El orden es intencional: Trim antes de Replace values. La regla no cambia `ana` ni convierte importes; si necesitas números tipados, añade Convert type y define su política de error.

Comprobar el resultado

Comprueba dos filas, cabecera con solo `name, amount` , reemplazo de Jorge y conservación de Ana. La columna `internal_note` no debe aparecer. El destino CSV escribe el esquema recibido: su nombre de archivo no selecciona campos.

Para comprobar valores obligatorios puedes añadir Required fields con `fields=["name"]` y `maxInvalidPercent=0` . Esa comprobación falla toda la ejecución ante una fila inválida; no rellena valores ni elimina registros. No añadas controles sin decidir cómo quieres tratar sus errores.

Resolver problemas

Si no reemplaza, revisa espacios, mayúsculas, nombre de campo y `matchMode` . Si salen columnas extra, inspecciona el Mapping guardado y la versión vinculada, no supongas que Field selector conserva campos no seleccionados. Si solicitas capitalización general, especifica esa regla como requisito distinto; no presentes este reemplazo como solución universal. Antes de repetir una carga real, comprueba idempotencia y destinos. Un ejemplo correcto no establece una política financiera de precisión o redondeo.

Resumen regional con Join y Aggregate

Une dos fuentes sintéticas y verifica un total por región.

Antes de comenzar

Este ejercicio utiliza dos fuentes: transacciones y catálogo de clientes. El resultado debe sumar únicamente transacciones aprobadas por región. Prepara un runtime que admita dos entradas y Join; el adaptador genérico de streaming lineal no ejecuta ese grafo. Ambos archivos deben estar disponibles mediante bindings autorizados.

Usa las tablas pequeñas de este capítulo, sin descargar paquetes antiguos. Antes de diseñar, confirma la unicidad de `customer_id` en clientes: duplicados pueden multiplicar filas y aumentar el total.

Procedimiento

1. Crea una entrada TXT delimitada por `|` con `customer_id`, `amount`, `status`. En Flat File input selecciona `format=txt`, `delimiter=|`, `hasHeader=true`.
2. Crea un Excel con hoja Customers y cabecera `customer_id,region`. En Excel input selecciona esa hoja, cabecera en fila 1 y datos desde fila 2.
3. Convierte `customer_id` a `integer` en ambas ramas. En transacciones convierte `amount` a `number` con `outputField=amount_number` y `onError=fail`.
4. En la rama de clientes, limpia `region` con `trim` y `upper`. En transacciones, filtra `status=APPROVED` con Row filter.
5. Une ambas ramas mediante Join `inner`: transacciones a la izquierda, clientes a la derecha; claves `customer_id`, `keyType=integer`, `nullsEqual=false`. Selecciona los identificadores reales de los componentes conectados.
6. Agrupa por `region`: `metric=sum`, `field=amount_number`, `outputField=total`. Ordena por `total` descendente y selecciona `region,total` antes del destino CSV.

Ejemplo sintético

Transacciones:

```
customer_id|amount|status
1|10|APPROVED
1|5|APPROVED
2|7|APPROVED
2|99|REJECTED
```

Copiar

Filas de la hoja Customers:

```
customer_id,region
1,NORTH
2,SOUTH
```

Copiar

Resultado esperado:

```
region, total  
NORTH, 15.0  
SOUTH, 7.0
```

[Copiar](#)

La representación textual del número puede variar según el formato numérico; comprueba los totales numéricos 15 y 7 y el orden configurado.

Comprobar el resultado

Calcula el resultado de forma independiente: ignora la fila REJECTED y suma 10+5 para NORTH y 7 para SOUTH. Espera dos filas y exactamente dos columnas. Confirma que el Join encontró todos los identificadores aprobados y no expandió filas por duplicados. Guarda y publica el Mapping aceptado, configura ambos bindings en la Session y ejecuta un Workflow de laboratorio.

Group aggregate devuelve solo claves de agrupación y métrica; no conserva el detalle original de cada transacción.

Resolver problemas

Si faltan regiones, revisa tipos de claves, datos no coincidentes y tipo de Join. Si los totales aumentan, revisa duplicados en el catálogo. Si una región aparece separada por espacios o mayúsculas, revisa Clean text antes del agregado. Un error de streaming puede ser incompatibilidad de topología, no un dato incorrecto. Para importes reales define precisión, moneda y redondeo por separado; este ejemplo no establece una política contable.

Operación segura: del laboratorio a producción

Comprueba identidad, dependencias y resultados antes de habilitar una ejecución periódica.

Preparar un laboratorio

Trabaja con datos sintéticos, un destino separado y una identidad autorizada. Confirma el tenant seleccionado y el rol activo antes de crear objetos. Una ubicación de Repository organiza elementos; no concede permisos. Usa una Connection de lectura para el origen cuando el ejercicio no necesite modificarlo. Introduce credenciales únicamente mediante el mecanismo protegido de Connections, nunca en nombres, descripciones, parámetros de ejemplo o mensajes a DovaAI.

Construir una cadena verificable

1. Crea un Mapping pequeño y configura origen, transformación y destino. Para probar una sustitución exacta, usa Replace values con `field=name`, `find=jorge`, `replacement=Jorge`, `matchMode=exact` y `caseSensitive=true`.
2. Valida y guarda el borrador. Publica la versión aceptada: guardar y publicar son acciones distintas.
3. Crea una Session con esa versión del Mapping. Revisa conexiones, entradas, salidas, perfil de ejecución y capacidad autorizada.
4. Crea un Workflow Start → Session → End. Guarda, valida y publica deliberadamente.
5. Ejecuta un solo intento de laboratorio; compara Monitor y el archivo producido con el resultado esperado.

Una ruta de archivo configurada para ejecución debe existir donde el worker pueda utilizarla. Adjuntar un archivo a DovaAI para consultar su esquema no establece una entrada permanente del worker.

Aceptar antes de programar

Para el ejemplo, comprueba que `jorge` se convierte en `Jorge` y que `ana` permanece igual. Si agregas Field selector con `fields=["name", "amount"]`, verifica que el esquema y la cabecera final contienen únicamente esos campos. Un estado succeeded no sustituye esta comparación funcional.

Activa un horario solo después de aceptar el resultado. Comprueba zona horaria, dependencias publicadas y efectos del destino. Un worker público ofrece capacidad elegible: no publica la Session ni los datos de otro tenant. La disponibilidad de slots tampoco garantiza acceso a una Connection.

Resolver incidencias sin duplicar efectos

Si el cliente agota el tiempo de espera, consulta el estado del intento antes de repetirlo. Cancelar, terminar, reintentar y recuperar son acciones diferentes. Un reintento puede repetir escrituras externas; revisa idempotencia y estado persistido. Si cambia una dependencia, inspecciona las versiones consumidas antes de volver a publicar.

Conserva evidencia autorizada del resultado y limpia solamente los objetos de laboratorio revisados. Purge contents es destructivo y requiere confirmar su alcance; no es un paso rutinario para solucionar un error. Ante acceso denegado, revisa membresía, rol, plan y recurso antes de solicitar permisos adicionales.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

CLI y API: acceso y comprobación controlados

Usa la distribución oficial y diferencia la clave personal de la sesión web con MFA.

Preparar el cliente

Obtén la distribución correspondiente a tu sistema desde Developer Tools y compara su checksum con el publicado. Usa la ayuda de esa distribución como referencia de sus comandos. La versión de la CLI puede ser distinta de la versión de la plataforma; no deduzcas compatibilidad únicamente por el nombre del archivo.

La CLI se autentica con una clave personal de Dova. Créala mediante la interfaz autorizada de API Keys y conserva el valor en un mecanismo privado. No es una clave OpenAI. Evita colocar claves o contraseñas en argumentos de comandos, capturas, archivos compartidos o historiales. El inicio de sesión interactivo solicita la clave sin mostrarla.

Consultar antes de modificar

Estos comandos no contienen secretos y consultan el contexto actual:

```
dova --version
dova --help
dova login
dova status --output json
dova repository list --output json
dova mapping list --output json
dova workflow list --output json
dova run list --output json
dova logout
```

Copiar

`dova login` usa el servicio público configurado por la distribución. Comprueba que status muestra el tenant esperado antes de ejecutar operaciones con efectos. La CLI guarda su configuración de autenticación local; protege la cuenta del sistema y ejecuta logout cuando corresponda. Cerrar la terminal no equivale a revocar la clave personal.

Ejecutar una prueba autorizada

La ayuda de `dova workflow run --help` describe entradas, parámetros y espera. Selecciona primero un Workflow publicado de laboratorio y sus dependencias aceptadas. La opción `--wait` observa el intento; un tiempo de espera del cliente no demuestra que la ejecución haya terminado. Consulta Monitor antes de reenviar. Los comandos cancel, terminate, rerun y recover tienen consecuencias distintas y comprobaciones de estado.

Distinguir los dos flujos de autenticación

La API con sesión web conserva login, MFA cuando se exige, cookie y protección CSRF para cambios. Si login responde `authenticated:false` junto con `mfaRequired:true`, falta completar MFA; una respuesta HTTP 200 no significa autenticación terminada. Después de verificar MFA, consulta la sesión y utiliza el CSRF vigente para el flujo web correspondiente.

La clave personal Bearer de la CLI no se convierte en una sesión web y no sustituye MFA para configurar el proveedor global de DovaAI. Ese ajuste corresponde al superusuario único autorizado mediante su flujo protegido. No copies cookies del navegador a comandos.

Comprobar y resolver errores

Ante 401, revisa credencial o sesión; ante 403, revisa rol y tenant. Un error de validación requiere corregir el recurso indicado, no repetir escrituras a ciegas. Conserva identificadores públicos del intento para diagnóstico, sin registrar encabezados de autenticación. Una respuesta aceptada debe verificarse en el objeto o intento correspondiente.

Exportar una promoción para Git/DevOps

La interfaz Configuration → Environment promotions permite exportar el manifiesto de una release ya aplicada. El kit GitOps añade `promotion-export --release-id <id> --output dova/promotion-release.json` mediante `app/bin/dova_gitops.py`, usando su configuración y autenticación autorizadas. Este comando consulta la release, verifica su digest y escribe el archivo local; no promueve, ejecuta ni modifica Dova o Git.

Ejecuta el comando desde el repositorio local preparado para el kit. Revisa el manifiesto antes de registrarlo en tu control de versiones: conserva artefactos publicados, dependencias y procedencia; las referencias secretas se exportan como placeholders. Registrar repository, commit y tag no autentica un repositorio remoto ni hace un push. GitHub permanece como repositorio y no se activa GitHub Actions.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

GitOps 1.1.2: fijar origen y destino

Descarga el [kit GitOps 1.1.2](#) desde una sesión autorizada de Developer Tools y compara el SHA-256 publicado. Extrae `dova_gitops.py`, `dova-gitops.example.json` y `README.md` en tu repositorio local. Usa Python 3.12 o posterior y claves personales mediante las variables protegidas indicadas por `apiKeyEnv`; no escribas sus valores en Git.

En `dova-gitops.json`, `source.environment` y `source.domainId` seleccionan el origen de exportación. `target.environment` selecciona el destino y `bindings.domainId` es el Domain de destino autorizado. Si incluyes `target.domainId`, debe coincidir con `bindings.domainId`. El tenant sigue siendo el autorizado por la clave personal y debe coincidir con la configuración; estos campos no cambian de tenant ni conceden acceso.

Los valores omitidos de entorno usan `development`; el Domain por defecto es General (`dom_general`). Declara el contexto de forma explícita cuando trabajes en otro Domain. La versión 1.1.2 fija el Domain de `bindings` antes de la primera consulta y conserva el mismo Tenant, Domain y Environment al leer, reservar un draft, guardar, publicar y solicitar o liberar el lease de edición. No cambies el contexto de un cliente que ya realizó solicitudes.

Ejemplo sintético: Finance en Test

Este fragmento complementa el ejemplo del kit. Sustituye los IDs por un Domain y Folder existentes y autorizados; no crea Finance ni sus recursos:

```
{
  "source": {"environment": "development", "domainId": "dom_finance"},
  "target": {"environment": "test", "domainId": "dom_finance"},
  "bindings": {"domainId": "dom_finance", "folderId":
"fld_REPLACE_WITH_EXISTING_FINANCE_FOLDER"},
  "statePath": "dova/target-state-finance-test.json"
}
```

Copiar

Completa también URL, tenant, variables de claves y los bindings de Connections, workers, perfiles y archivos que necesite el bundle. Desde el repositorio local, valida y revisa antes de ejecutar una importación autorizada:

```
python dova_gitops.py --config dova-gitops.json validate
python dova_gitops.py --config dova-gitops.json plan --plan-file dova/plan.json
python dova_gitops.py --config dova-gitops.json apply --plan-file dova/plan.json
```

Copiar

`validate` revisa el bundle local. `plan` consulta el destino sin escribir. `apply` sí crea o actualiza definiciones y publica las versiones revisadas en Development, Test o Stage. Las versiones publicadas del origen y sus etiquetas de entorno permanecen intactas. El autorado directo en Production se rechaza antes de solicitudes con `production_authoring_requires_environment_promotion`; usa la promoción gobernada de Dova para ese destino.

Conservar el estado y resolver un contexto incorrecto

Usa un `statePath` distinto para cada destino. El estado liga origen, tenant, Domain y Environment; reutilizarlo en otro workspace se rechaza antes de consultar Dova. Un estado antiguo vacío puede adoptar el contexto elegido. Un estado anterior poblado sin alcance exige reconciliación manual de sus IDs y versiones publicadas. Conserva el estado y el journal ante un resultado incierto: no los borres ni cambies sus campos de contexto para forzar otra importación.

Si una importación de Finance intenta usar General, revisa `source.domainId`, `target.environment` y `bindings.domainId`, instala el kit 1.1.2 y vuelve a generar el plan. Los kits históricos 1.1.0 y 1.1.1 conservan sus bytes; 1.1.2 es la distribución con alcance de workspace para Dova 24.

Overview: estado del espacio de trabajo

Interpreta los indicadores y encuentra el módulo adecuado para investigar.

Overview resume los objetos, la actividad de ejecución y la salud operativa del tenant activo. Sus tarjetas ayudan a decidir dónde investigar; no demuestran por sí solas que una transformación sea correcta ni que una operación comercial haya terminado. Cada cifra pertenece al contexto autorizado que aparece en el menú de cuenta.

Antes de empezar

Usa una cuenta autorizada y un espacio de laboratorio. Comprueba tenant y rol antes de interpretar indicadores. No cambies al espacio de un cliente solamente para obtener tarjetas con datos.

Procedimiento

1. Abre el menú de cuenta y confirma identidad, espacio y rol activo.
1. Abre Overview y revisa los resúmenes de objetos. Sigue una tarjeta hasta su inventario para comprobar nombres, filtros y estados.
1. Revisa la actividad reciente e identifica el intento relevante por su identificador y hora. Continúa en Monitor para ver las tareas y mensajes seguros.
1. Consulta Workers o Configuration cuando una señal de capacidad o runtime requiera diagnóstico. Comprueba compatibilidad y disponibilidad antes de cambiar políticas.
1. Después de una modificación autorizada, vuelve al resumen y compara con el registro del módulo correspondiente. Usa Refresh si necesitas confirmar el estado actual.

Comprobar el resultado

Puedes explicar el alcance de cada indicador y seguirlo hasta un objeto o intento concreto. Compara un Workflow sintético con su último intento en Monitor. Conserva la diferencia entre definición guardada, versión publicada y ejecución.

Errores y recuperación

Un contador en cero puede indicar un espacio vacío o un alcance limitado. Un error de carga no equivale a cero. Si parece faltar información, comprueba filtros, permisos y tenant. No repitas Save o Run basándote solamente en una tarjeta que no se actualizó; inspecciona primero la operación existente.

Ejemplo sintético

En Training, localiza w_CustomerNames desde Overview, abre su definición y compara el identificador del último intento con Monitor. El ejercicio es de lectura y no necesita lanzar otra ejecución.

Alcance de la comprobación

Trabaja con datos sintéticos y con el rol autorizado para la tarea. Una confirmación visual debe contrastarse con el registro guardado o el intento correspondiente. Conserva únicamente identificadores y mensajes seguros para soporte; no compartas secretos, información personal ni datos del tenant. Si el resultado de una operación queda incierto, comprueba su estado antes de repetirla.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Disponibilidad de workers y carga de la vista

Una referencia a un worker retirado o con una identidad inválida no impide consultar Overview, Monitor o Connections. Esa referencia no queda disponible para ejecutar; los workers válidos continúan apareciendo según tus permisos. Un fallo real de almacenamiento o de política se informa como error y no se convierte en un contador cero.

Repository: organización y purga de contenido

Organiza Domains y Folders y elimina contenido mediante una revisión explícita.

Repository organiza Mappings, Sessions, Workflows y Connections. Un Domain es una agrupación superior; una Folder pertenece a un Domain. La ubicación no concede permisos por sí misma. Purge contents y Delete son operaciones diferentes: la primera vacía el alcance revisado y conserva la ubicación seleccionada; la segunda elimina una ubicación cuando no tiene bloqueos.

Antes de empezar

Confirma tenant y permisos de administración de ubicaciones. Revisa dependencias y editores abiertos antes de una eliminación. Usa una estructura sintética y conserva la información que necesites fuera del alcance que vas a purgar.

Procedimiento

1. Selecciona el Domain correcto. Crea una Folder con un nombre reconocible y comprueba la ruta antes de guardar objetos.
1. Usa las vistas All, Mappings, Sessions, Workflows, Connections o Shared disponibles para identificar el contenido por tipo e identificador.
1. Para vaciar una ubicación, abre Purge contents desde su menú. Revisa la vista previa y sus bloqueos: puede incluir Mappings, Sessions, Workflows, borradores, Connections, versiones guardadas y carpetas hijas.
1. Escribe exactamente el nombre solicitado y marca la confirmación. Envía una sola vez y espera el resultado verificado.
1. Si la respuesta queda incierta, usa Check result. No repitas la purga hasta aclarar el resultado anterior. Una revisión desactualizada requiere Refresh y nueva confirmación.
1. Si también quieres eliminar la ubicación, usa Delete por separado. Un Domain con Folders o una ubicación con artefactos continúa bloqueado.

Comprobar el resultado

Comprueba que se eliminó solamente el contenido revisado. La ubicación seleccionada y el historial de ejecución se conservan. Si después eliminaste una ubicación vacía, confirma su ausencia en el árbol. No interpretes un filtro sin filas como prueba de que no existen dependencias.

Errores y recuperación

Las ubicaciones protegidas o carpetas no activas tienen restricciones. Un bloqueo debe resolverse mediante las acciones soportadas, no cambiando registros almacenados. Si falla la carga, distingue error de inventario vacío. Purge no ofrece una promesa de deshacer ni elimina automáticamente el Domain objetivo.

Ejemplo sintético

Crea Training/CustomerDemo y guarda m_CustomerNames allí. En un laboratorio descartable, revisa Purge contents sobre CustomerDemo y cancela antes de confirmar si solo estás aprendiendo.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Recuperar una carga fallida

Si no se puede cargar el contexto o inventario, Repository muestra el error de carga y una acción Retry (Reintentar) para volver a consultar el mismo contexto autorizado. Revisa la conexión y el Tenant, Domain y Environment seleccionados. Un fallo de carga no significa que el repositorio esté vacío ni que el Trial haya perdido artefactos. No recrees ubicaciones ni elimines borradores para resolver un error de lectura.

Los borradores temporales vencidos se excluyen del inventario sin convertir la consulta del contexto en una escritura dentro de su instantánea. Los artefactos guardados y las publicaciones existentes se conservan. Si el problema continúa, facilita a soporte la hora y el estado HTTP, sin credenciales ni datos del tenant.

Retry conserva el contexto Tenant, Domain y Environment de la consulta fallida; no cambia de entorno ni usa otro inventario como sustituto. Si no tienes acceso al workspace, abre uno autorizado desde el menú de usuario antes de continuar.

Mappings: diseñar y publicar transformaciones

Construye el grafo, configura componentes y publica una versión revisada.

Antes de comenzar

Un Mapping define el algoritmo de transformación. La Session aporta el contexto de ejecución y el Workflow coordina tareas. Selecciona el tenant correcto y una ubicación autorizada en Repository. Necesitas permisos de edición; publicar y ejecutar son acciones adicionales. Prepara una muestra sintética con entrada y resultado esperados, sin credenciales ni datos de clientes.

La presencia de un componente en la paleta no garantiza compatibilidad de cualquier worker. Comprueba el runtime, especialmente para conexiones externas y grafos con Join.

Procedimiento

1. Abre Mappings y crea un borrador en el Domain y Folder adecuados. Usa un nombre descriptivo.
2. Añade los componentes desde sus categorías y conecta la salida de cada paso con la entrada siguiente. Selecciona el componente para configurar el inspector.
3. Ajusta cabecera, nombres y tipos del origen antes de configurar transformaciones. Revisa las claves de Join y las columnas que recibirá el destino.
4. Usa Validate para resolver problemas del grafo. Save guarda el borrador persistente. El indicador de guardado local no sustituye esta acción.
5. Publica únicamente después de guardar todos los cambios. Publish requiere un borrador guardado sin cambios pendientes; no ejecuta el Mapping.

Field selector: conservar solo campos elegidos

Para conservar solo dos columnas de un CSV, conecta Flat File input → Field selector → Flat File output. Configura `hasHeader=true`, `delimiter=,` y `fields=["name", "amount"]`. El selector conserva los valores y las filas, pero elimina las demás columnas del esquema. El destino usa ese esquema; `fileName` no controla selección de campos.

Comprobar el resultado

Reabre el borrador para comprobar persistencia. Revisa la versión publicada antes de enlazarla a una Session. Ejecuta una muestra mediante capacidad autorizada y compara cabecera, filas y valores con el resultado esperado. Una validación estructural correcta no demuestra que el archivo exista en el worker ni que el resultado responda a la regla de negocio.

Puedes pedir a DovaAI los componentes, conexiones y valores que convienen al objetivo. El asistente ofrece instrucciones breves; tú realizas la configuración, guardas, publicas y ejecutas desde los controles normales de Dova.

Resolver problemas

Si Publish está desactivado, comprueba guardado, cambios pendientes, ubicación y permisos o bloqueo de edición. Si aparecen columnas extra, revisa la configuración guardada de Field selector, la rama conectada y la versión usada por la Session. Si faltan archivos, corrige los bindings del runtime. Ante incompatibilidad de

streaming, revisa el adaptador: el genérico exige una ruta lineal compatible. No elimines pasos necesarios solo para hacer pasar la validación.

Modificar campos distintos con operaciones distintas

Conecta dos componentes Clean text en serie:

Raw file → Clean text 1 → Clean text 2 → Destino

1. Clean text 1: Fields = `first_name` ; Operations = `trim` .
2. Clean text 2: Fields = `last_name` ; Operations = `upper` .

Ambos campos viajan juntos. Cada componente modifica únicamente el campo indicado y conserva los demás. No dividas el flujo ni añadas Field mapper para aplicar estas dos operaciones.

Eliminar una selección del canvas

Selecciona el elemento y presiona Supr (Delete) con el foco dentro del canvas. El atajo no debe borrar componentes mientras escribes en un input, textarea u otro editor. Revisa las conexiones afectadas y valida el grafo después de eliminar; el cambio pertenece al borrador hasta que lo guardes y publiques.

Columnas para revisar dependencias y publicación

En Columns puedes elegir Used by (Sessions), Location, Last published y Description. Used by (Sessions) muestra la cantidad de Sessions que utilizan el Mapping; pulsa el número para consultar sus nombres y las versiones del Mapping referenciadas. Location combina Domain y Folder. Last published muestra fecha y responsable de la última publicación, por separado de Updated, que puede corresponder a cambios del borrador. Description resume el propósito del Mapping. Si falta evidencia histórica, la tabla lo indica en lugar de inventar una fecha o un usuario.

Para una configuración nueva, la tabla muestra Mapping, Version, Status, Used by (Sessions), Location y Updated. Conserva las columnas que ya hubieras elegido; activa las adicionales desde Columns. El filtro Status ofrece Draft y Published, de acuerdo con el ciclo de edición y publicación.

Pantalla completa sin perder el trabajo

Full Screen amplía el canvas de Mappings o de Workflows en el mismo editor. La entrada y la salida actualizan el diseño de inmediato y conservan selección, zoom y posición. Usa el mismo botón o Escape para volver. DovaAI continúa disponible en esta vista; ampliar el canvas no guarda ni publica el borrador.

Acciones de la tabla

Al iniciar una opción de Actions, el indicador de progreso aparece en el control y bloquea las demás acciones de esa tabla mientras la solicitud está pendiente. Espera el resultado antes de repetir. Que la solicitud haya comenzado no demuestra que el guardado, la publicación o la operación hayan terminado.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta

sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Lista compacta y botones con íconos

La lista reúne Search, Status y las acciones en una barra compacta. Refresh, Columns, Export CSV, Delete selected, New Mapping y Clear filters usan íconos; coloca el puntero sobre un botón o llévale el foco con Tab para identificar su acción. Las acciones siguen respetando selección, permisos y estados deshabilitados.

La lista de Mappings no incluye filtros Environment ni Runtime. El entorno procede del contexto activo de la barra inferior; el worker y su capacidad se configuran en la Session. Usa Search, Status y los filtros de columnas para revisar los objetos de ese contexto.

Diseñar, guardar y publicar desde el editor

Advanced, Validate, Save draft y Publish se presentan como botones con íconos y nombres accesibles. La ayuda del botón identifica la acción y, cuando corresponde, el entorno del guardado o la publicación. El encabezado no repite el distintivo DEV: confirma Tenant, Domain y Environment en la barra inferior.

Mappings no ofrece un botón Run. Un Mapping define la transformación; para ejecutarla, guarda y publica la versión revisada, selecciónala en una Session y configura sus entradas, destinos y worker compatible. Ejecuta la Session directamente o mediante un Workflow autorizado. Publicar el Mapping no inicia una ejecución ni asigna capacidad.

Configuration del componente

Selecciona un componente para revisar sus secciones y campos en Configuration. Las etiquetas, controles y ayudas se distribuyen dentro del ancho disponible. En los orígenes con Header mapping, los botones para cargar una muestra, agregar un campo y eliminar mantienen su propio espacio junto al título. Revisa Source, Output y Type en la tabla; desplázala horizontalmente si sus columnas no caben en el panel. Una muestra de encabezados configura el esquema y no sustituye el archivo que la Session leerá en el worker.

Sessions: vincular el Mapping al runtime

Configura versiones, parámetros, conexiones y archivos para ejecutar el Mapping.

Antes de comenzar

Una Session reutiliza un Mapping publicado en un contexto de ejecución. No redefine su algoritmo. Prepara una versión aceptada, una ubicación autorizada y capacidad compatible. Mantén separadas las autorizaciones para editar, publicar y ejecutar. Los archivos locales del navegador no aparecen automáticamente en el sistema de archivos del worker.

Elige un entorno y un destino de laboratorio antes de probar escrituras. Registrar Production como entorno no concede permiso ni publica el artefacto.

Procedimiento

1. Crea la Session y selecciona el Mapping publicado. Comprueba su identificador y versión, no solo el nombre.
2. Revisa el entorno y la política de worker. Conserva el valor heredado si no necesitas una asignación explícita; cualquier selección debe cumplir capacidades y permisos.
3. Completa conexiones y bindings de entrada/salida. Usa rutas accesibles por el worker elegido y una salida exclusiva para la prueba.
4. Configura parámetros declarados con tipos correctos. Los overrides de tarea pertenecen al Workflow, no a las capas de la Session.
5. Revisa timeout, reintentos y recuperación. Guarda el borrador y publica la versión aceptada antes de consumirla desde un Workflow.

Ejemplo sintético

Un Mapping lee `name, amount` de un CSV y conserva ambas columnas. Crea dos Sessions con la misma referencia de Mapping: una para Test y otra para Stage. Usa entradas equivalentes y salidas diferentes. Las diferencias de rutas pertenecen a los bindings; no necesitas duplicar el algoritmo para cambiar únicamente el archivo.

Comprobar el resultado

Reabre la Session y confirma referencia de Mapping, versión, entorno, política de worker y bindings. Si usas parámetros, revisa el origen del valor resuelto. Crea un Workflow de laboratorio para cada contexto y compara esquema y datos producidos. Un guardado correcto no demuestra que las conexiones o archivos sean accesibles durante la ejecución.

Al actualizar un Mapping, revisa el aviso de dependencia y acepta la nueva versión de forma deliberada. Comprueba después los Workflows que consumen esa Session.

Resolver problemas

Para un archivo ausente, corrige su binding y revisa permisos del worker. Para falta de capacidad, distingue estado del worker, slots, requisitos y políticas del tenant. Un error JSON debe corregirse sin descartar otros ajustes del borrador. Un checkpoint solo ayuda si existe estado utilizable; no revierte automáticamente escrituras externas. Antes de repetir una carga, determina si el destino admite reintentos sin duplicados.

Runtime: perfil, nombres y variables

En Runtime elige Execution profile y revisa sus límites de entrada, memoria, disco y timeout junto con la capacidad del worker. Un perfil automático puede admitir menos bytes que tu archivo; el espacio libre de la VM no aumenta ese límite ni la memoria temporal del ejecutor aislado.

Available runtime variables muestra nombre, significado, Example preview y Last resolved value. El ejemplo ayuda a preparar la configuración; el valor resuelto pertenece al run identificado y su fecha. Los valores sensibles se ocultan. Si aún no hay un intento compatible, la ausencia de valor real no se sustituye por el ejemplo.

Para nombres de salida usa Target filename timestamp y Target filename template. `%RUN_TIMESTAMP%` se resuelve al iniciar el intento. Enabled lo activa para esa Session; Disabled lo desactiva; Inherit sigue la preferencia global. Escribir el token en el template mientras Inherit resulta desactivado no activa el reemplazo. Filename preview · example ilustra el nombre, no prueba la salida real de un run. Guarda y publica para que los consumidores puedan adoptar el cambio.

General: elegir ubicación antes del Mapping

Después de Name, completa Domain → Folder → Mapping y luego Environment y Status. Los mappings disponibles se filtran por Domain y permisos de uso compartido, y se cargan desde la apertura del formulario sin volver a seleccionar Domain. Folder organiza la Session y no obliga al Mapping a estar en la misma carpeta. Al cambiar la ubicación, revisa las opciones y conserva una referencia válida antes de guardar.

TRACE opcional

En Sessions → Runtime, TRACE permite elegir Inherit from Workflow, Enabled o Disabled. Inherit from Workflow toma el ajuste del Workflow que la ejecuta; una ejecución directa de Session sin Workflow lo resuelve como desactivado. Enabled y Disabled sustituyen la herencia para esa Session. Por defecto se usa Inherit from Workflow y los Workflows nuevos tienen TRACE desactivado.

El ajuste controla la generación de detalle TRACE para los intentos nuevos; no elimina evidencia histórica ni oculta INFO, WARNING o ERROR. Guarda, publica y actualiza las versiones del Workflow cuando corresponda para ejecutar el ajuste aceptado.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Worker Unassigned

Antes de volver a ejecutar, selecciona un worker compatible y permitido y revisa el pool del entorno. Unassigned no toma otro worker automáticamente. Puedes aprovisionar una VM nueva y asignarla explícitamente. Los permisos, la propiedad del recurso, el servidor base protegido y las operaciones pendientes siguen verificándose; la retirada no es una forma de eludir esas protecciones.

Barra de la lista y alcance del entorno

En la lista de Sessions, los botones de acciones usan íconos con ayuda y nombres accesibles y se alinean con Search y los filtros. Puedes identificarlos con el puntero o recorriéndolos con Tab. El filtro Environment se elimina de esta lista: se usa el contexto Tenant, Domain y Environment seleccionado en el menú de usuario y visible en la barra inferior. Runtime y los demás filtros permanecen disponibles. Este cambio de la lista no elimina la configuración Runtime de la Session ni sus requisitos de worker.

Workflows: coordinar y ejecutar tareas

Define dependencias, publica versiones y verifica cada intento.

Antes de comenzar

Un Workflow coordina tareas y sus dependencias. Las tareas Session ejecutan el contexto que corresponde al Mapping; Start, Decision, Timer y End expresan la orquestación. Prepara una Session publicada, capacidad compatible y una salida de laboratorio. Revisa permisos por separado para edición, publicación, ejecución y programación.

Trabaja primero con una secuencia pequeña. Añadir paralelismo antes de verificar las salidas dificulta distinguir errores de datos, dependencias y disponibilidad de workers.

Procedimiento

1. Crea un Workflow en el tenant y ubicación autorizados. Añade Start, Session y End.
2. Selecciona la Session y comprueba `sessionVersion`. Conecta Start → Session → End.
3. Configura parámetros de tarea si están declarados y permitidos. Para varias entradas, revisa `joinPolicy=any` o `all`; no confíes solo en la posición visual.
4. Cuando necesites Decision, configura su condición True; revisa la rama False complementaria. Timer usa `delaySeconds`, de 1 a 604800.
5. Valida, guarda y publica. Ejecuta manualmente antes de habilitar un calendario. Guardar y publicar no inician una ejecución.

Ejemplo sintético

Prepara una Session que filtra `status=active` y escribe un CSV de prueba. El Workflow mínimo tiene una tarea de esa Session entre Start y End. Espera una ejecución de la Session, con un archivo que contenga únicamente las filas activas. Para estudiar una espera, añade un Timer de 5 segundos antes de la Session y conserva el mismo resultado de datos.

Comprobar el resultado

En Monitor, localiza el identificador del intento y revisa sus tareas. Confirma la versión de Session consumida y el Mapping correspondiente. Compara salida, esquema y filas con el resultado esperado. Una rama omitida no equivale a una tarea fallida.

Tras la aceptación manual, configura un calendario autorizado, zona horaria y estado habilitado. Comprueba la próxima ejecución y después un intento real. Una programación habilitada expresa intención; no demuestra que una ejecución haya terminado.

Resolver problemas

El validador admite 2–100 nodos y 1–300 enlaces y rechaza ciclos, identificadores repetidos y dependencias inválidas. Para una tarea en espera, inspecciona padres, worker, capacidad y política. Para un fallo, busca el primer error relevante. Antes de reintentar, revisa si la escritura puede repetirse sin duplicados. Al cambiar una dependencia, revisa y republica los consumidores deliberadamente; una versión nueva no garantiza aceptación automática en todas las referencias.

Versiones consumidas y disponibles

La cadena compacta distingue Workflow, Session y Mapping y muestra la versión fijada junto a una versión más reciente disponible. Abrir Session o Mapping permite inspeccionar el objeto; no adopta por sí mismo la versión nueva. Update versions actualiza las referencias del borrador y muestra lo que cambiará. Después usa Save draft → Publish. Reload status solo refresca la disponibilidad. La cadena de una ejecución en Monitor conserva las versiones históricas realmente usadas.

Para promover una versión aceptada utiliza Configuration → Environment promotions. Revisa el paquete exacto y los bindings del ambiente siguiente. Publicar una versión nueva, actualizar referencias, promover entre ambientes y ejecutar son operaciones distintas que realiza el usuario.

TRACE del Workflow y herencia

En el editor de Workflow → Workflow settings, selecciona TRACE = Enabled cuando necesites más detalle de diagnóstico; el valor por defecto es Disabled. Las Sessions con Inherit from Workflow usan ese valor. Una Session configurada como Enabled o Disabled conserva su decisión explícita. Activar una Session no habilita TRACE para las demás Sessions del Workflow.

La ejecución conserva el ajuste de las versiones que realmente consumió. Cambiar el borrador no altera un intento en curso ni añade TRACE retroactivamente a los logs anteriores. INFO, WARNING y ERROR permanecen disponibles sin activar TRACE.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Barra compacta de la lista

La lista de Workflows alinea Search, los filtros restantes y las acciones con íconos. Los botones conservan ayuda, nombres accesibles, permisos y estados de progreso; usa el puntero o Tab para reconocerlos. Environment ya no aparece como filtro de esta lista: confirma el contexto activo en la barra inferior. Runtime y los demás filtros se conservan. Los controles de edición y ejecución del Workflow mantienen su función; este ajuste de la lista no cambia las versiones de sus Sessions ni la capacidad necesaria para ejecutarlas.

Monitor: investigar y verificar ejecuciones

Relaciona estado global, tareas, capacidad y resultados reales.

Antes de comenzar

Monitor presenta evidencia de intentos de ejecución. Un Workflow guardado o publicado es una definición, no un intento. Conserva el identificador del run, la hora aproximada y las versiones involucradas. Selecciona el tenant autorizado y revisa permisos de lectura y de cada acción operativa.

No compartas filas de entrada, credenciales, cookies ni registros sin filtrar al abrir una incidencia. Una descripción útil incluye estado, tarea y mensaje seguro, sin contenido sensible.

Procedimiento

1. Abre Monitor y reduce la búsqueda por nombre, estado o tiempo. Comprueba identificador y fecha: el mismo Workflow puede tener varios intentos.
2. Distingue estados activos de terminales. Una espera de capacidad es distinta de un fallo; cancelación y éxito también son resultados diferentes.
3. Abre el detalle del intento y revisa tareas y dependencias. Localiza el primer error relevante, no solo la última tarea omitida.
4. Para esperas, revisa worker, slots y contexto de ejecución en las superficies autorizadas. No lances otra copia porque la tabla tarde en actualizarse.
5. Antes de cancelar, terminar, recuperar o repetir, actualiza el estado y revisa la confirmación de esa acción.

Ejemplo sintético

Usa un Workflow de laboratorio con entrada CSV sintética. Ejecuta una vez con la ruta correcta y otra con una ruta inexistente del mismo laboratorio. Localiza ambos intentos por identificador. En el fallido, identifica la tarea afectada y el mensaje de archivo ausente. Corrige el binding y ejecuta un intento nuevo; conserva la evidencia anterior según la política aplicable.

Comprobar el resultado

Para el intento correcto, compara el archivo producido con cabecera, cantidad de filas y valores esperados. El estado de éxito indica el resultado de ejecución, pero no comprueba por sí solo la regla de negocio. Para el intento fallido, confirma que la explicación corresponde al primer error y que el nuevo intento usa la configuración corregida.

Mantén separada la limpieza de historial: no cancela trabajos activos ni sustituye una política de retención. Revisa alcance y confirmación antes de eliminar evidencia.

Resolver problemas

Una tabla vacía puede deberse a filtros; no prueba un fallo del servidor. Un timeout del cliente no prueba cancelación: vuelve a consultar el intento. La recuperación depende de checkpoints disponibles y compatibles; no revierte automáticamente escrituras externas. Repetir puede producir otra salida o duplicar una carga. Si el estado cambia mientras un diálogo está abierto, recarga antes de confirmar. Para errores de proveedor, registra únicamente el mensaje seguro y la hora.

Log compacto y progreso durante la ejecución

El log presenta columnas Timestamp y Message en filas compactas de una línea. Expande un registro para ver su detalle estructurado con formato y colores; vuelve a contraerlo para recuperar espacio. Filtra con Search log messages, Level, Component y Time zone; revisa los contadores por nivel, Errors only y Download log. Follow sigue los mensajes nuevos cuando estás al final; Pause/Resume pausa únicamente su presentación y no detiene la ejecución. Wrap ajusta líneas extensas.

Mientras el detalle está visible, la vista consulta los cambios y conserva los datos anteriores durante una demora o fallo temporal. Las tarjetas usan los últimos contadores reportados. El progreso puede mostrar etapa, lotes, filas procesadas/escritas, tasa, duración y bytes cuando se conocen. Staging, procesamiento y escritura son etapas distintas: no interpretes cero filas antes de lectura como prueba de inactividad. No se inventa un porcentaje cuando falta un total fiable.

El worker registra causas seguras como entrada mayor que el máximo del perfil o destino protegido ya existente, además de etapa y límites disponibles. No se publican credenciales ni respuestas privadas del proveedor. Para un intento antiguo sin excepción original, el mensaje genérico sigue siendo evidencia incompleta; no atribuyas una causa no registrada.

Lista y Gantt

La lista actualiza celdas sin sustituir toda la tabla y conserva selección, filtros y desplazamiento. List y Gantt comparten los filtros. El Gantt compacto diferencia Workflow, Session y Mapping mediante tipo y jerarquía; una Session aparece debajo de su Workflow. Abre el intento concreto para revisar estado, duración y versiones. La barra de un Workflow abarca su orquestación; la de una Session representa esa ejecución y no debe sumarse otra vez como tiempo adicional del Workflow.

Lectura compacta y actualización estable

List alinea Execution, Duration y Status en filas compactas. Los distintivos WF en violeta y SES en verde, con los mismos iconos de las pestañas del espacio de trabajo, distinguen Workflow y Session tanto en List como en Gantt; los indicadores de color acompañan el estado escrito y la jerarquía se puede expandir. La fuente del Gantt y sus tooltips sigue la tipografía de Dova, con pesos suaves para etiquetas y valores.

Los cambios actualizan las filas existentes e incorporan nuevas ejecuciones que coincidan con los filtros sin vaciar ni reconstruir visualmente toda la lista. Se conservan selección, filtros y desplazamiento. Una ejecución fuera de los filtros actuales no tiene que aparecer.

TRACE es detalle opcional generado según la configuración publicada de Workflow y Session. El filtro de nivel del Monitor decide qué evidencia existente muestras; no activa la generación de TRACE. Consulta el ajuste de herencia de la Session cuando el Workflow y sus tareas tengan cantidades distintas de registros TRACE.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Conversación y entorno

DovaAI usa Tenant, Domain y Environment autorizados y el artefacto seleccionado. Al cambiar de contexto se cancelan las consultas anteriores y empieza una conversación del nuevo entorno. Una respuesta pendiente del contexto anterior no se incorpora a la nueva conversación. La identidad del entorno y las versiones se verifican en el servidor; los valores de formularios, datos y credenciales no se añaden al contexto.

Disponibilidad de workers y carga de la vista

Una referencia a un worker retirado o con una identidad inválida no impide consultar Overview, Monitor o Connections. Esa referencia no queda disponible para ejecutar; los workers válidos continúan apareciendo según tus permisos. Un fallo real de almacenamiento o de política se informa como error y no se convierte en un contador cero.

Connections: acceso reutilizable y custodia de credenciales

Configura metadatos y autenticación sin exponer secretos.

Una Connection separa el endpoint, el conector, el método de autenticación y la custodia protegida de credenciales de la lógica de un Mapping. Los campos y métodos disponibles dependen del conector elegido. Que una tecnología aparezca en el catálogo no demuestra que un worker tenga el controlador, la conectividad ni los permisos para usarla.

Antes de empezar

Prepara un recurso de laboratorio autorizado y preferiblemente de solo lectura. Identifica el worker que hará la prueba y la identidad que usará. No incluyas contraseñas, claves ni referencias sensibles de Vault en descripciones o capturas.

Procedimiento

1. Abre Connections y crea un registro con un nombre que indique su propósito. Selecciona el conector y revisa sus campos obligatorios.
1. Introduce los metadatos autorizados: host o endpoint, puerto, base de datos, esquema o recurso según corresponda. Confirma los valores con el responsable del entorno.
1. Elige un método de autenticación y una custodia disponibles. Passwordless usa una identidad de ejecución; una custodia externa exige que la referencia protegida sea resoluble. No fuerces métodos ocultos.
1. Completa el flujo protegido, revisa el tenant y guarda. Comprueba los metadatos persistidos antes de probar.
1. Usa Test para la Connection guardada y lee el resultado seguro. Después ejecuta una extracción pequeña desde un Mapping/Session en el worker compatible.
1. Revisa consumidores antes de rotar, revocar o eliminar la Connection. Distingue editar metadatos de sustituir credenciales.

Comprobar el resultado

La Connection muestra solamente los metadatos permitidos; la prueba informa el estado esperado y una lectura pequeña produce el resultado conocido. Compara identificadores con el editor. Una prueba de conexión satisfactoria no certifica todos los modos de escritura, reintento o recuperación.

Errores y recuperación

Ante un error de autenticación, revisa método, custodia, permisos y expiración. Para errores de red, comprueba DNS, TLS y salida del worker. Un controlador faltante requiere revisar capacidades instaladas. Mantén los mensajes seguros y nunca copies secretos al campo descripción para depurar.

Ejemplo sintético

CustomerLabReadOnly apunta a la base sintética training y la tabla customers mediante una identidad de lectura. Selecciónala en el componente de origen correspondiente y verifica diez registros conocidos.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Preparar Connections del destino

Antes de promover hacia Production, prepara allí Connections propias con custodia y permisos válidos. Un administrador autorizado puede registrarlas o modificar sus metadatos con `etl.connections.manage` y `etl.environments.production.edit`. Esta autorización específica no habilita el editor de Mappings, Sessions o Workflows de Production. Las Connections legacy compartidas o ambiguas siguen de solo lectura.

El preview de promoción y Fork to Development solo ofrece Connections compatibles ya existentes en el destino. Selecciona cada reemplazo explícitamente; no se copian contraseñas ni se inventan referencias de secretos. Domain y Environment de la Connection permanecen ligados a su contexto de propiedad.

Disponibilidad de workers y carga de la vista

Una referencia a un worker retirado o con una identidad inválida no impide consultar Overview, Monitor o Connections. Esa referencia no queda disponible para ejecutar; los workers válidos continúan apareciendo según tus permisos. Un fallo real de almacenamiento o de política se informa como error y no se convierte en un contador cero.

Notification channels: destinos y entrega

Separa canales, políticas, plantillas y transporte de correo.

Un canal define un destino reutilizable para eventos autorizados. No define por sí mismo cuándo se genera el evento ni demuestra que el destinatario recibió el mensaje. La política de notificaciones decide eventos y rutas; la plantilla define el contenido del correo; SMTP proporciona el transporte. Estas configuraciones requieren verificaciones diferentes.

Antes de empezar

Usa un destinatario o webhook de laboratorio bajo tu control y un rol autorizado para administrar canales. Trata direcciones y endpoints como información sensible cuando corresponda. No uses datos de clientes para probar una integración.

Procedimiento

1. Abre Notification channels y revisa si ya existe un destino autorizado que puedas reutilizar.
1. Crea el canal con una etiqueta reconocible, el tipo ofrecido y sus campos obligatorios. Usa el flujo protegido si requiere credenciales.
1. Confirma tenant y destino, guarda y vuelve a abrir el registro para verificar los datos permitidos.
1. Usa Test donde esté disponible y envía contenido sintético. Comprueba tanto el estado informado como la recepción en el sistema de laboratorio.
1. Asocia los eventos, severidad y destinatarios en la política pertinente. Revisa supresión, deduplicación, escalamiento, resumen y horarios cuando estén configurados.
1. Para correo, revisa por separado la plantilla y Email delivery. Una vista previa de contenido no prueba SMTP ni recepción.
1. Genera un evento controlado y compara Delivery history con el sistema receptor. Restaura las políticas temporales al terminar.

Comprobar el resultado

El canal permanece guardado, la prueba informa un resultado coherente y el destino recibe el mensaje esperado o existe una supresión explicada. Conserva solo un identificador seguro y la hora. La aceptación local puede indicar entrega al transporte, no recepción final.

Errores y recuperación

Si no llega el mensaje, comprueba primero si el evento fue generado y si la política lo suprimió. Después revisa destino, autenticación y transporte mediante mensajes seguros. No envíes pruebas repetidas mientras el resultado siga pendiente. Revisa consumidores antes de eliminar el canal.

Ejemplo sintético

Crea LabOperations con un buzón de pruebas autorizado. Envía un aviso sintético, localiza su estado en Delivery history y confirma asunto, texto y enlaces en el buzón.

Workers: capacidad elegible y privacidad

Comprueba salud, compatibilidad y disponibilidad antes de asignar trabajo.

Los workers ejecutan Sessions y Workflows según la política del tenant. Heartbeat, estado de energía de la VM y elegibilidad son observaciones diferentes. Un worker público puede ofrecer capacidad compartida; asignarle una Session no publica su definición, sus datos ni sus artefactos para otros usuarios o tenants.

Antes de empezar

Prepara un host autorizado y conoce su propietario. Necesitas capacidad compatible, permisos y acceso de red a los recursos del proceso. Los archivos de entrada deben estar disponibles para el worker; seleccionarlo no los copia automáticamente.

Procedimiento

1. Abre Workers e identifica el registro por identificador, tenant o pool y heartbeat reciente. No te bases únicamente en el nombre.
1. Abre su detalle y revisa versión, recursos, conectores y slots. Compara con los requisitos de la Session y el Execution profile.
1. Si debes registrar un host, usa el procedimiento de enrollment autorizado y conserva el token en el flujo protegido. Comprueba que aparezca un heartbeat válido.
1. Distingue reiniciar el servicio worker de encender una VM en el proveedor. El aprovisionamiento y los controles de proveedor corresponden a Administrator.
1. Selecciona el valor por defecto o una asignación explícita permitida. Comprueba que la identidad pueda resolver Connections y acceder al archivo de entrada.
1. Ejecuta una Session sintética y verifica en Monitor qué worker recibió el intento y cuál fue su resultado.
1. Antes de retirar o sustituir capacidad, revisa trabajos activos, referencias, políticas por defecto y dependencias.

Comprobar el resultado

El registro está actualizado, es elegible para el trabajo y completa una prueba con resultado conocido. Los slots son unidades ponderadas: cuatro slots pueden admitir cuatro trabajos de peso uno o dos de peso dos, sujetos a otros límites. Online no significa capacidad ilimitada.

Errores y recuperación

Sin heartbeat, revisa servicio y conectividad permitida. Ante inelegibilidad, inspecciona políticas, capacidades, recursos y ciclo de vida. Provider unavailable no prueba que la VM no exista. Una eliminación fallida debe resolverse mediante el procedimiento soportado; no borres registros para ocultar el problema.

Ejemplo sintético

Asigna `s_CustomerNames` a `lab-worker`, con un CSV sintético disponible allí. Comprueba el identificador del worker y la versión de Mapping utilizados en el intento.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Pool de workers por entorno

En Configuration → Dova Integration Runtime, Environment worker pool muestra el entorno actual. Allowed workers permite asignar workers existentes; Default worker debe pertenecer a esa selección. Pool policy empieza en Dedicated. Selecciona Shared únicamente cuando decidas compartir capacidad y todos los pools participantes también declaren Shared. Un worker físico no se comparte por una coincidencia de nombre.

Save environment pool guarda solamente la política del entorno seleccionado y no inicia una ejecución ni cambia el default global del tenant. La revisión usa una huella de la configuración; si otro usuario la cambia, Refresh pool permite consultar y revisar nuevamente antes de guardar. Los controles requieren administración de configuración y Runtime por entorno. Built-in no se ofrece como worker de ejecución.

Production no toma prestado automáticamente el pool de Development. Los nuevos trabajos requieren un pool explícito, salvo la opción pública de Development para un Trial elegible descrita a continuación. Las ejecuciones legacy solo pueden conservar su worker anterior cuando el servidor verifica la cadena publicada exacta, su configuración y la pertenencia permitida. Test y Stage consumen el bucket Development del plan; Production usa el bucket Production. La pertenencia al pool no sustituye capacidad, estado online, perfil compatible, cuotas ni concurrencia.

Trial público: pool Development gestionado

Para un Trial público elegible, si no existe una política explícita de Development, Dova obtiene un pool público de Development gestionado por la plataforma. Selecciona únicamente workers públicos compartidos con referencias verificadas, autorizados para el Trial y sin pertenencia a otros pools configurados. No incorpora workers privados, crea workers ni guarda cambios en Runtime. El Trial no obtiene permiso para guardar pools.

Una política explícita de Development, incluso con Allowed workers vacío, tiene prioridad. Este mecanismo no se extiende a Test, Stage o Production. Siguen vigentes las comprobaciones de capacidad, perfil, aislamiento, cuotas y concurrencia; el acceso público no garantiza disponibilidad.

Antes de detener o retirar capacidad

Pause VM y Stop VM conservan sus comprobaciones de trabajos activos, worker predeterminado y capacidad compartida; revisa las referencias y el impacto antes de utilizarlos. Pause scheduling detiene nuevas asignaciones y permite terminar el trabajo actual. Para retirar definitivamente la VM, utiliza Remove VM and unlink y revisa su confirmación: después de verificar ausencia cloud, Dova limpia las asignaciones y muestra Unassigned sin borrar definiciones ni versiones publicadas. Antes de ejecutar de nuevo, asigna capacidad compatible explícitamente.

Las acciones requieren el permiso del módulo y Runtime del entorno correspondiente, o un permiso global concedido explícitamente. Una VM compartida requiere autorización sobre todos sus entornos.

Retirada y asignación posterior

Abre Remove VM and unlink, revisa la identidad del recurso y el impacto en las Sessions, ejecuciones y pools, y confirma la eliminación prevista. Para una VM propia compatible, no necesitas desarmar manualmente cada referencia ni apagarla como requisito previo de esta operación. Retirar capacidad puede interrumpir ejecuciones activas; programa la operación según tu necesidad.

Después de confirmar la ausencia en el proveedor, Dova retira el worker activo, limpia referencias de Runtime y muestra Unassigned en los bindings de Sessions que usaban ese worker. Conserva las definiciones, versiones publicadas, configuración e historial de auditoría. Si la VM se elimina directamente en el proveedor, la conciliación aplica el mismo tratamiento cuando comprueba su ausencia. Un error al consultar al proveedor no demuestra que la VM se haya borrado.

Antes de volver a ejecutar, selecciona un worker compatible y permitido y revisa el pool del entorno. Unassigned no toma otro worker automáticamente. Puedes aprovisionar una VM nueva y asignarla explícitamente. Los permisos, la propiedad del recurso, el servidor base protegido y las operaciones pendientes siguen verificándose; la retirada no es una forma de eludir esas protecciones.

Configuration: runtime, perfiles y programación

Separa políticas de capacidad, límites de ejecución, calendarios y salud.

Configuration reúne Dova Integration Runtime, Execution profiles, Workflow Scheduler, Runtime health y Promotions. Las políticas seleccionan capacidad autorizada; los perfiles expresan límites; los calendarios programan orquestaciones futuras; la salud describe observaciones actuales. Guardar una política no demuestra que exista un worker disponible ni que un intento haya terminado correctamente.

Antes de empezar

Confirma tenant y permisos antes de guardar. Registra el estado inicial para restaurar cambios de laboratorio. Diferencia el worker por defecto de un override de Session y la zona horaria de programación de la zona de presentación de Profile.

Procedimiento

1. Abre Configuration e identifica cuál de sus áreas resuelve tu necesidad. Inspecciona antes de editar.
1. En Dova Integration Runtime, revisa el worker por defecto y las opciones de selección o afinidad ofrecidas. Comprueba elegibilidad en Workers.
1. Compara una Session con Use tenant default frente a otra con un override permitido. Un override no cambia el valor por defecto de otras Sessions.
1. En Execution profiles, revisa las unidades y los límites del perfil autorizado. Cambia solo valores con una razón operativa concreta y comprueba cómo se heredan.
1. En Workflow Scheduler, revisa calendario, timezone y enabled. Asócialo al Workflow publicado previsto después de una prueba manual satisfactoria.
1. Guarda los campos revisados, espera aceptación y vuelve a abrir para confirmar persistencia. Si nada cambió, Save puede permanecer deshabilitado.
1. Consulta Runtime health por separado y continúa en Workers o Monitor para comprobar capacidad e intentos reales.

Comprobar el resultado

Dos Sessions sintéticas, una con el valor por defecto y otra con override compatible, muestran el worker esperado en el contexto de ejecución. Para un calendario, registra timezone, próxima hora prevista e identificador de la ejecución resultante; son comprobaciones distintas.

Errores y recuperación

No aumentes límites para ocultar un worker no disponible. Corrige unidades o rangos inválidos. Tras un cambio concurrente, recarga la base y compara antes de reenviar. Deshabilita los calendarios temporales al terminar: cerrar Configuration no los desactiva.

Ejemplo sintético

Usa Training con s_Default y s_Override. Conserva iguales las transformaciones y cambia solamente la selección permitida del worker; compara ambos intentos en Monitor.

Promoción entre ambientes

Abre Configuration → Environment promotions. Selecciona un Mapping, Session o Workflow publicado y la versión de origen. El orden permitido es Development → Test → Production (Stage es opcional cuando está habilitado). Cada promoción conserva las versiones exactas de sus Sessions y Mappings y crea artefactos publicados separados en el destino; no modifica los originales del ambiente de origen.

1. Elige el ambiente siguiente y define los reemplazos de Connections y los ajustes de Session: worker, perfil, parámetros, entradas y salidas. Usa identificadores existentes; las credenciales permanecen protegidas en Dova.
2. Completa Git provenance con repositorio HTTPS y commit SHA completo o etiqueta, si corresponde.
3. Usa Review promotion. Revisa dependencias, cambios de versiones, worker compatible, capacidades del plan, DPU disponible y concurrencia. Development, Test y Stage utilizan la capacidad Development; Production utiliza Production.
4. Aplica únicamente la vista revisada y confirma. Production requiere además permiso de aprobación y un comentario. Si cambia el destino, la definición o el plan, vuelve a revisar.
5. Comprueba el historial y los nuevos artefactos. La promoción no ejecuta el Workflow ni activa sus calendarios; habilita programación solo después de tu aceptación.

Export manifest descarga el manifiesto inmutable y su procedencia para tu repositorio o flujo DevOps. No incluye credenciales ni referencias secretas utilizables. Dova no realiza un push ni activa GitHub Actions.

Restore this release recupera una definición y su configuración histórica mediante nuevas versiones del destino, con autorización, confirmación y controles actuales. No deshace archivos escritos, cargas ni cambios de datos externos. La revisión e idempotencia protegen contra dobles aplicaciones; ante un conflicto, actualiza y compara antes de confirmar nuevamente.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Pool de workers por entorno

En Configuration → Dova Integration Runtime, Environment worker pool muestra el entorno actual. Allowed workers permite asignar workers existentes; Default worker debe pertenecer a esa selección. Pool policy empieza en Dedicated. Selecciona Shared únicamente cuando decidas compartir capacidad y todos los pools participantes también declaren Shared. Un worker físico no se comparte por una coincidencia de nombre.

Save environment pool guarda solamente la política del entorno seleccionado y no inicia una ejecución ni cambia el default global del tenant. La revisión usa una huella de la configuración; si otro usuario la cambia, Refresh pool permite consultar y revisar nuevamente antes de guardar. Los controles requieren administración de configuración y Runtime por entorno. Built-in no se ofrece como worker de ejecución.

Sin un pool explícito no se presta automáticamente el pool de Development a Production ni se ejecutan nuevos trabajos con un default implícito. Las ejecuciones legacy solo pueden conservar su worker anterior cuando el servidor verifica la cadena publicada exacta, su configuración y la pertenencia permitida. Test y Stage consumen el bucket Development del plan; Production usa el bucket Production. La pertenencia al pool no sustituye capacidad, estado online, perfil compatible, cuotas ni concurrencia.

Colores por Tenant y Environment

El color identifica el contexto actual en el resumen de la barra inferior de estado operativo, el distintivo del entorno y el borde del selector Environment. Los valores iniciales son azul para Development, ámbar para Test, violeta para Stage y rojo para Production. El nombre o código del entorno permanece visible; el color no concede acceso ni cambia tu paleta personal.

Para personalizarlo, selecciona primero Tenant y Environment desde el menú de usuario y abre Configuration → Miscellaneous → Environment colors. Elige el color o introduce un valor hexadecimal de seis dígitos, revisa la vista previa y usa Save color. El ajuste se comparte para ese Tenant+Environment, incluso entre Domains. Restore default prepara el valor predeterminado y Save color lo aplica. Refresh permite recuperar la configuración vigente ante un cambio concurrente. Necesitas permiso para administrar la configuración; no existe un color independiente por Domain ni un selector adicional de color general del Tenant.

Perfiles y límites del pool

En Configuration → Dova Integration Runtime, el pool del entorno permite elegir Allowed workers, Default worker y política Dedicated o Shared. Un worker pertenece a varios entornos únicamente cuando todos sus pools participantes declaran Shared. También puedes restringir perfiles de ejecución y su perfil predeterminado, slots concurrentes, timeout e intentos mediante las opciones disponibles. Revisa y guarda la política del entorno seleccionado; estos límites no sustituyen capacidad, heartbeat, compatibilidad ni cuotas del plan.

Planes y suscripciones: oferta efectiva y continuidad

Comprende las condiciones vigentes sin confundir oferta, pedido y suscripción.

Plans and subscriptions muestra las opciones comerciales efectivas para el tenant activo. La disponibilidad depende del tipo de tenant, su ciclo de vida, el plan y las ofertas autorizadas. Un precio mostrado, un pedido de checkout y una suscripción activa son etapas diferentes. Consulta siempre los importes, moneda, periodo y condiciones vigentes en la plataforma; este manual no fija precios.

Antes de empezar

Confirma el tenant y el rol antes de abrir checkout. Para aprender usa un entorno de pruebas de pago autorizado. Una prueba pública debe continuar como el mismo tenant operativo después de una conversión aceptada; no recrees el espacio para simular la conversión.

Procedimiento

1. Abre Plans and subscriptions y revisa el plan, ciclo de vida y capacidades disponibles para tu rol.
1. Lee la versión de oferta, precio, moneda, periodo, cantidad, impuestos y términos mostrados. Las ofertas específicas se basan en el catálogo global versionado.
1. Si debes evaluar una compra autorizada, revisa datos del cliente, documento, nombre legal, correo, país, dirección y método de pago antes de avanzar.
1. Para un ejercicio, utiliza únicamente fixtures o un sandbox independiente autorizado. No envíes tokens sintéticos a un proveedor real.
1. Tras una compra autorizada, consulta el resultado definitivo del pedido y la suscripción. Investiga el pedido existente antes de repetir checkout.
1. Comprueba que el identificador del tenant y los objetos de Repository se conservan y que las capacidades corresponden a la suscripción aceptada.
1. Para renovaciones o cambios, vuelve a revisar vigencia y versión contractual. Un nuevo precio no reescribe pedidos históricos.

Comprobar el resultado

La conversión tiene un resultado comercial aceptado, conserva el tenant operativo y concede las capacidades correspondientes. Los cupos de DovaAI y los límites de usuarios o capacidad dependen del contrato efectivo; no deduzcas acceso ilimitado de un título administrativo.

Errores y recuperación

Si no hay una oferta, revisa elegibilidad, ciclo de vida, rol y plan. Si discrepan los importes, confirma moneda y cantidad antes de enviar. Un resultado pendiente requiere resolver el pedido previo, no crear otro para probar. No marques una factura como pagada para habilitar funciones.

Ejemplo sintético

En un sandbox autorizado, revisa la oferta de Training, cancela antes de pagar si solo estudias el formulario y comprueba que no se creó una suscripción que no pretendías contratar.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Pool de workers por entorno

En Configuration → Dova Integration Runtime, Environment worker pool muestra el entorno actual. Allowed workers permite asignar workers existentes; Default worker debe pertenecer a esa selección. Pool policy empieza en Dedicated. Selecciona Shared únicamente cuando decidas compartir capacidad y todos los pools participantes también declaren Shared. Un worker físico no se comparte por una coincidencia de nombre.

Save environment pool guarda solamente la política del entorno seleccionado y no inicia una ejecución ni cambia el default global del tenant. La revisión usa una huella de la configuración; si otro usuario la cambia, Refresh pool permite consultar y revisar nuevamente antes de guardar. Los controles requieren administración de configuración y Runtime por entorno. Built-in no se ofrece como worker de ejecución.

Production no toma prestado automáticamente el pool de Development. Los nuevos trabajos requieren un pool explícito, salvo la opción pública de Development para un Trial elegible descrita a continuación. Las ejecuciones legacy solo pueden conservar su worker anterior cuando el servidor verifica la cadena publicada exacta, su configuración y la pertenencia permitida. Test y Stage consumen el bucket Development del plan; Production usa el bucket Production. La pertenencia al pool no sustituye capacidad, estado online, perfil compatible, cuotas ni concurrencia.

Trial público: pool Development gestionado

El Trial público elegible puede utilizar automáticamente el pool público de Development gestionado por la plataforma cuando no existe una política explícita de Development. Una política explícita, incluso vacía, prevalece. Esto no concede workers privados ni permisos para guardar pools; la disponibilidad y los límites del plan siguen aplicándose. Consulta [workers y capacidad](#).

Recuperación de la suscripción

La proyección de financiación usa el formato canónico de periodos de Billing y se reconcilia con la suscripción autorizada. La corrección de Subscription funding projection does not match no cambia el periodo contractual, su ciclo, los créditos concedidos ni el consumo; tampoco extiende el Trial. Las comprobaciones de tenant, contrato y financiación siguen aplicándose. Ante un error persistente, revisa la operación y la suscripción existentes antes de repetir registro o checkout.

Developer Tools: CLI y claves personales

Verifica distribuciones y automatiza solo el alcance autorizado.

Developer Tools ofrece distribuciones del CLI por plataforma, checksums, información de firma, instrucciones y notas de versión. También permite acceder a la gestión autorizada de claves personales. La versión del CLI puede diferir de la versión del servidor. Una etiqueta de versión no sustituye la comprobación de integridad ni concede permisos adicionales.

Antes de empezar

Usa un equipo de laboratorio autorizado y una clave personal con propósito y vigencia definidos. No coloques claves en argumentos compartidos, capturas, repositorios o informes. La autorización sigue vinculada a identidad, tenant y permisos; el CLI no evita las reglas de la interfaz.

Procedimiento

1. Abre Developer Tools o Downloads en el menú de cuenta y selecciona la distribución de tu sistema operativo.
1. Lee la versión, checksum, estado de firma e instrucciones. Descarga en una ubicación local autorizada y compara exactamente el SHA-256 con el publicado.
1. Instala según las instrucciones y ejecuta `dova --version`. Comprueba qué ejecutable resuelve tu PATH si existen instalaciones anteriores.
1. Crea una clave personal de laboratorio donde esté permitido y custódiala por el flujo protegido. No incluyas su valor en documentación.
1. Ejecuta `dova login --url` con el servicio de laboratorio autorizado e introduce la clave mediante el prompt interactivo. Comprueba la identidad y el tenant devueltos.
1. Comienza con `status` y listas filtradas de Repository, Mappings o runs. Compara los identificadores con la interfaz antes de automatizar escrituras.
1. Al terminar, distingue `dova logout` de revocar la clave en el servidor. Logout limpia el estado local; revocar impide nuevos usos de esa clave en otros equipos.

Comprobar el resultado

El checksum coincide y el CLI correcto informa su versión. Una consulta de lectura muestra la identidad y objetos esperados. Comprueba ayuda para los comandos y opciones de la distribución instalada, en lugar de asumir que todos coinciden con otra versión.

Errores y recuperación

Un checksum distinto bloquea el uso del archivo hasta verificar su origen. Para 401/403, revisa clave, expiración y permisos. Un timeout puede dejar trabajo ejecutándose: consulta el intento existente antes de reintentar. Protege el perfil local porque puede conservar autenticación.

Ejemplo sintético

En tu host de laboratorio autorizado ejecuta `dova --version` y `dova status --output json`. No copies una clave dentro de la línea de comandos ni uses un dominio de ejemplo como servicio real.

Kit GitOps por workspace

El [kit GitOps 1.1.2](#) fija Tenant, Domain y Environment en consultas, reservas de drafts, publicaciones y leases. Usa `source.environment` y `source.domainId` para exportar; `target.environment` y `bindings.domainId` para importar. Los defaults son Development y General; el estado se conserva por destino. Lee el [ejemplo Finance y la recuperación de estado](#). El autorado directo en Production requiere la promoción gobernada, no el comando de importación.

Menú de cuenta: contexto, apariencia y salida

Confirma identidad y alcance antes de actuar en la plataforma.

El menú de cuenta muestra la identidad, el espacio de trabajo y el rol activo. También ofrece acciones personales como Profile, Downloads, API Keys, Documentation y Sign out según el acceso disponible. Light/Dark y las paletas cambian la presentación; no alteran permisos ni convierten una cuenta en administradora de otros tenants.

Antes de empezar

Conoce el espacio donde debes trabajar y usa solo contextos ofrecidos a tu identidad. Antes de cambiar tenant o cerrar sesión, revisa Open items y cualquier cambio sin guardar. No uses un cambio de contexto para resolver una duda sobre la propiedad de un borrador.

Procedimiento

1. Abre el menú y lee identidad, tenant y rol activo. Si no son los esperados, detén acciones de escritura.
1. Consulta Open items y guarda o descarta deliberadamente los cambios necesarios. Un editor abierto no garantiza persistencia.
1. Selecciona únicamente un workspace autorizado ofrecido por la interfaz. Espera la transición y verifica de nuevo el contexto.
1. Si existe una selección de rol permitida, elige el apropiado y comprueba las acciones resultantes. Una acción oculta puede ser el comportamiento correcto.
1. Ajusta Light/Dark o la paleta disponible. Comprueba legibilidad, foco y estados; un color no debe ser tu única referencia.
1. Abre Profile para preferencias y seguridad personal o Developer Tools para descargas y automatización. Usa Documentation para la ayuda del producto.
1. Al terminar en un equipo compartido, usa Sign out. Para volver, sigue login y MFA normales. Cerrar una pestaña no equivale a cerrar sesión.

Comprobar el resultado

Puedes identificar el tenant de los registros que estás viendo y distinguir apariencia de autorización. Haz un ejercicio reversible cambiando tema y volviendo al original sin modificar datos. Comprueba que el foco del teclado y el texto sigan siendo visibles.

Errores y recuperación

Si falla un cambio, verifica el contexto realmente activo antes de repetir. Si desaparece un objeto, revisa tenant, rol, filtros y permiso; no lo recrees inmediatamente. Si una acción personal no está disponible, revisa el acceso autorizado, no una suposición basada en el nombre del rol.

Ejemplo sintético

En Training, abre Profile desde el menú, revisa la zona horaria y cierra sin cambios. Luego activa Dark y vuelve a Light; ninguna de estas acciones debe publicar o ejecutar un Workflow.

Tenant, Domain y Environment en el menú de usuario

El resumen Tenant, Domain y Environment aparece en la barra inferior de estado operativo de Dova y Administrator. Haz clic en ese resumen para abrir el menú de usuario con los tres selectores. Conserva el distintivo y el color del entorno; no ocupa una franja superior ni duplica los campos.

Abre el botón de usuario del panel izquierdo. En el mismo menú, elige Tenant, Domain y Environment entre los valores autorizados. Espera la carga del contexto y comprueba el entorno antes de editar o ejecutar. Estos tres selectores reemplazan la sección grande del contexto en la pantalla. El menú se abre a la derecha del botón cuando hay espacio y se ajusta al área visible; en pantallas pequeñas puedes desplazarte dentro del menú hasta Sign out.

Después de crear, renombrar o eliminar un Domain correctamente, la lista Domain se actualiza sin recargar toda la aplicación. Si se elimina el Domain seleccionado, Dova verifica un destino disponible y conserva la protección de cambios sin guardar. Si no puedes completar el cambio, revisa el contexto y utiliza Retry context; no sigas escribiendo en un contexto no disponible.

Idioma de la interfaz

En el menú de usuario, selecciona ES, EN o PT para cambiar el idioma de la interfaz. La preferencia se guarda en tu perfil y se utiliza en Dova y Administrator. Si no se puede guardar, la interfaz informa el fallo y restaura la elección anterior. Cambian las etiquetas disponibles de la aplicación; no se renombran artefactos, identidades, campos, código ni datos del usuario. El estilo, la paleta y los permisos mantienen su configuración propia.

Barra inferior compacta

La barra mantiene visible el contexto Tenant, Domain y Environment junto al estado de conexión. El botón Operational details abre un panel sobre la barra con Active, Workers, Queue, Plan, Autosave y Updated. Consulta allí los detalles operativos sin ocupar otra fila permanente. El estado de validación conserva su acceso cuando hay una acción relevante; DovaAI dispone de un espacio propio al acoplarlo. En pantallas pequeñas algunos textos se resumen; el contexto y los nombres accesibles de los controles se conservan.

Profile: preferencias, seguridad y solicitudes personales

Gestiona tu cuenta sin confundirla con la administración de usuarios.

Profile contiene información personal, preferencias, controles de seguridad y solicitudes de privacidad de la identidad autenticada. Es diferente de Administrator → Users y no concede roles ni acceso a otros tenants. Ábrelo desde el menú de cuenta y verifica la identidad antes de editar. Las solicitudes de eliminación requieren el proceso indicado; no borran inmediatamente objetos compartidos ni registros retenidos.

Antes de empezar

Usa una cuenta de laboratorio para comprobar preferencias reversibles. Antes de cambiar contraseña, prepara los factores de autenticación necesarios para volver a iniciar sesión. Trata una exportación de perfil como información personal y guárdala en un lugar autorizado.

Procedimiento

1. Abre Profile y revisa nombre, correo, tenant y rol. Si la identidad es incorrecta, cierra y resuelve el contexto antes de modificarla.
1. Elige una Timezone admitida y usa Save preferences. Vuelve a abrir para comprobar persistencia. Esta preferencia no cambia la zona del calendario de un Workflow.
1. Para Change photo, usa PNG, JPEG o WebP de hasta 10 MB. El flujo recorta al centro a 256 píxeles cuadrados y optimiza la imagen antes de enviarla. Espera la confirmación.
1. Usa Remove photo solo si quieres eliminarla y confirma el resultado aceptado. Un fallo de procesamiento no demuestra que una foto se haya guardado o eliminado.
1. Si deseas cambiar contraseña, proporciona Current password, New password y Confirm new password. La interfaz exige 12–128 caracteres y el servidor puede aplicar reglas adicionales.
1. Ten en cuenta que el cambio de contraseña cierra sesiones y dispositivos de confianza. Auténticate otra vez por el flujo normal.
1. Distingue Download profile export, Request profile data deletion y Request account deletion. Lee la confirmación antes de Submit request.

Comprobar el resultado

Cambia la zona horaria de una cuenta de laboratorio, guarda y vuelve a abrir para verificarla; después restaura el valor inicial. Puedes inspeccionar los diálogos de privacidad y cancelarlos sin enviar una solicitud. Sigue una solicitud enviada como pendiente hasta que el proceso confirme su resultado.

Errores y recuperación

Si falla Save, verifica el valor persistido antes de repetir. Si la sesión termina tras cambiar contraseña, vuelve a iniciar sesión en vez de asumir pérdida de objetos. Corrige formato o tamaño de una foto rechazada. No incluyas exports, factores MFA ni recuperación en capturas compartidas.

Ejemplo sintético

Cambia temporalmente Timezone a America/Lima en una cuenta de laboratorio. Comprueba la presentación de una hora admitida y restaura la preferencia original sin tocar la programación del Workflow.

Open items: editores, borradores y navegación

Mantén claro qué está abierto, guardado o publicado.

Open items reúne los editores de Mapping, Session y Workflow abiertos en el espacio de trabajo. Cada pestaña representa un contexto de edición; no es una versión publicada ni un trabajo en ejecución. El nombre, tipo y marca de cambios ayudan a navegar, pero la identidad del objeto, su tenant y su versión siguen siendo la referencia.

Antes de empezar

Abre objetos sintéticos relacionados con nombres distintos. Confirma el tenant y la ubicación antes de trabajar con varias pestañas. Conoce las diferencias entre guardar un borrador, publicar una versión, ejecutar y descartar un cambio local. La navegación y las optimizaciones de memoria no sustituyen Save.

Procedimiento

1. Abre el Mapping, Session y Workflow de un proceso de laboratorio. Identifica cada pestaña por nombre y tipo.
1. Cambia de pestaña y vuelve al editor. Continuar una edición no demuestra que se haya guardado ni publicado.
1. Haz un cambio reversible y observa la marca de cambios pendientes. Guarda si es intencional o revierte el cambio; espera el resultado aceptado.
1. Abre el menú contextual de una pestaña y lee si una acción afecta un elemento, varios u otras pestañas. Evita cierres masivos con cambios necesarios.
1. Al cerrar un editor modificado, elige conscientemente Save, Discard o Cancel según las opciones presentadas. Cancel conserva el contexto; descartar cambios locales no elimina el objeto guardado.
1. Reabre desde Repository o el gestor y comprueba identificador, contenido guardado y versión. Solo la acción de publicación establece una versión publicada.
1. Antes de cambiar tenant, cerrar sesión o terminar, revisa los pendientes y resuelve cada uno deliberadamente.

Comprobar el resultado

Modifica una descripción sintética, cambia a otra pestaña, regresa y cancela el cierre. Verifica que el borrador siga presente. Después revierte o guarda el cambio, cierra y reabre para confirmar el estado persistente esperado. Este ejercicio no exige ejecutar el Workflow.

Errores y recuperación

Si el editor no coincide con lo esperado, revisa tipo, identificador, tenant y versión antes de modificarlo. Una pestaña cerrada no demuestra eliminación, y un borrador visible no demuestra publicación. En pantallas compactas usa el foco de teclado y los menús disponibles. Investiga errores de carga antes de repetir acciones.

Ejemplo sintético

Abre m_CustomerNames y s_CustomerNames en Training. Cambia una descripción de laboratorio, visita la Session y vuelve al Mapping; guarda o descarta expresamente antes de cerrar.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

DovaAI: diagnóstico y orientación de solo lectura

Obtén la causa confirmada, el componente adecuado y pasos breves que tú realizas.

DovaAI usa el tenant autorizado, el módulo y la vista actual. Adapta el idioma al último mensaje y da recomendaciones breves sobre componentes, ajustes y ejecuciones. La conversación continúa al cambiar de módulo dentro del contexto permitido.

Antes de empezar

Confirma tenant y permisos. El proveedor debe estar configurado mediante su procedimiento protegido. Nunca pegues claves en el chat. Las consultas consumen los cupos efectivos de plan y usuario; una conversación nueva no reinicia consumo.

Pedir una solución concreta

Describe campo, entrada y resultado esperado. Por ejemplo: «Quiero cambiar exactamente jorge por Jorge en name». La recomendación es Replace values con field=name, find=jorge, replacement=Jorge, matchMode=exact y caseSensitive=true. Esta regla no capitaliza otros nombres. Para conservar únicamente columnas, usa Field selector y su lista fields.

Si adjuntas una muestra sintética compatible, el contexto usa metadatos de esquema. El adjunto no establece un archivo permanente de ejecución ni autoriza cambios. Revisa también nombres de encabezados antes de compartílos: pueden contener información sensible.

Diagnosticar Workflow y Session

Abre la ejecución que falló y pregunta por su causa. DovaAI consulta evidencia segura del intento autorizado: versiones ejecutadas, perfil, etapa, límites y contadores disponibles. Separa causas confirmadas de hipótesis. Un mensaje genérico sin detalle original no demuestra por sí mismo una colisión de destino o falta de memoria.

Espera una respuesta corta: problema confirmado, ruta de pantalla y valores concretos, seguida de pocos pasos manuales. Por ejemplo, si la configuración verificada deja el timestamp heredado y desactivado: Sessions → Session → Runtime → Target filename timestamp → Enabled; conserva el template, guarda y publica. Actualiza después las versiones del Workflow si corresponde.

Quién realiza los cambios

DovaAI solo consulta y recomienda. Nunca crea, modifica, configura, aplica, publica, ejecuta ni elimina Mappings, Sessions, Workflows, Connections o promociones, aunque se lo pidas o confirmes. Los mensajes y controles de guía señalan dónde actuar; tú decides y usas los controles normales de la plataforma.

Comprobar el resultado

Revisa la causa junto al log y la versión realmente ejecutada. Después de tus cambios, guarda y publica deliberadamente, actualiza dependencias cuando corresponda y ejecuta una prueba autorizada con salida identificable. La respuesta del asistente no demuestra que una configuración haya cambiado ni que una ejecución haya terminado.

Formato de una recomendación breve

DovaAI presenta el componente recomendado, un flujo en una línea y pasos con propiedades y valores como código. Explica el efecto sobre los demás campos sin añadir navegación innecesaria. Por ejemplo:

Conecta dos componentes Clean text en serie:

Raw file → Clean text 1 → Clean text 2 → Destino

1. Clean text 1: Fields = `first_name` ; Operations = `trim` .
2. Clean text 2: Fields = `last_name` ; Operations = `upper` .

Ambos campos viajan juntos. Cada componente modifica únicamente el campo indicado y conserva los demás. La redacción puede variar: comprueba los nombres y valores frente al componente actual. El asistente recomienda; tú configuras y ejecutas.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Conversación y entorno

DovaAI usa Tenant, Domain y Environment autorizados y el artefacto seleccionado. Al cambiar de contexto se cancelan las consultas anteriores y empieza una conversación del nuevo entorno. Una respuesta pendiente del contexto anterior no se incorpora a la nueva conversación. La identidad del entorno y las versiones se verifican en el servidor; los valores de formularios, datos y credenciales no se añaden al contexto.

Mover y acoplar DovaAI

Arrastra el icono con el mouse o el dedo: el lanzador sigue el puntero dentro del área visible. Al acercarlo a la barra inferior aparece una vista previa del espacio reservado. Suéltalo allí para acoplarlo con una animación de encaje. Arrástralo fuera de la barra para desprenderlo con una animación breve y volver a colocarlo en la pantalla.

También puedes mover la ventana desde su cabecera. Si arrastras la cabecera hasta la barra inferior, el chat se minimiza y su lanzador queda acoplado; al abrirlo de nuevo conservas la conversación del mismo contexto. La preferencia de reducir movimiento evita estas animaciones.

Con foco en el icono, Alt + End lo acopla, Alt + flechas lo mueve y lo desacopla, y Alt + Home restablece su posición. La posición se guarda en este navegador para el usuario y tenant autenticados; no se comparte con otras identidades.

Seleccionar contexto y captura

1. Abre el chat. Los botones Seleccionar elemento y Seleccionar área están juntos en la parte inferior izquierda del cuadro de mensaje. Usa el primero para señalar un componente y el segundo para dibujar un rectángulo sobre Dova o Administrator.

2. Comprueba el contorno. Las flechas mueven el puntero o el área; en modo área, Shift + flechas ajusta el tamaño. Enter o Capturar selección confirma; Escape o Cancelar cancela. La selección no activa los controles que señala.
3. Revisa la vista previa antes de Enviar. Incluye una captura real del área y un resumen de sus elementos visibles. Dova excluye los valores de formularios y elementos marcados como privados de la captura preparada. Elimina la imagen si el área no es la que deseas compartir.

Imágenes en el mensaje

Pega una imagen o arrastra PNG, JPEG o WebP al campo del chat. No necesitas un botón de adjuntar. Puedes revisar y quitar cada vista previa antes de Enviar. Se admiten hasta cuatro imágenes por mensaje; se preparan con un máximo de 2 MiB y 2048 píxeles por lado. Las imágenes que pegues pueden contener información sensible: revisa su contenido antes de enviarlas. DovaAI continúa siendo de solo lectura y la consulta usa los cupos efectivos del plan. Un cambio de identidad o contexto descarta el envío anterior.

Idioma y orientación de la conversación

Los controles de la conversación siguen el idioma del mensaje en español, inglés o portugués. Un saludo o un agradecimiento no añade un botón Muéstrame fuera de contexto. La orientación aparece cuando es pertinente para una tarea de Dova o cuando la pides; sus pasos continúan limitados a la pantalla y los permisos autorizados.

Abrir el chat y escribir un mensaje

Al abrir el chat aparece Preparando Dova AI... y los controles permanecen deshabilitados hasta terminar la carga y verificar el contexto autorizado. Espera a que se habiliten antes de escribir o seleccionar una captura. Si la preparación falla, revisa el aviso, el contexto y la disponibilidad; después cierra y vuelve a abrir el chat.

El campo de texto ocupa todo el ancho. Debajo, dentro del mismo cuadro, los dos selectores de contexto están juntos a la izquierda y el botón Enviar, con una flecha hacia arriba, está a la derecha. Durante el envío se muestra actividad; un fallo conserva el mensaje para que puedas revisarlo y reintentar.

Resumen de Administrator

Interpreta el estado administrativo antes de modificar la plataforma.

Antes de comenzar

Confirma la cuenta, el tenant y el rol activo en el menú de cuenta. Necesitas permiso de consulta para inspeccionar y el permiso de gestión específico para cambiar registros. Una opción oculta o deshabilitada puede reflejar el alcance autorizado; no cambies de tenant para eludir una denegación. Usa únicamente registros del espacio que tienes permitido administrar.

Procedimiento

1. Abre Overview y revisa las tarjetas Tenants, Users, Subscriptions, Invoices, Usage y Security.
2. Selecciona el worker y un intervalo 1h, 6h o 24h para comparar CPU, memoria y disco.
3. Abre el módulo correspondiente para investigar el registro.

Comportamiento y límites

Administrator tiene diez módulos principales; los dieciséis capítulos de este manual cubren también procedimientos especializados. Los gráficos representan observaciones disponibles y las acciones dependen del rol activo.

Una tarjeta no demuestra pago, ejecución correcta ni capacidad garantizada. Una medición ausente o una carga fallida no equivale a cero.

Compara muestras del mismo worker y período. Revisa la cobertura de mediciones antes de atribuir un cambio a un pipeline. Los controles de zoom cambian el rango visible; Refresh solicita observaciones actuales.

Verificar el resultado

Después de guardar o confirmar, vuelve a abrir el detalle y comprueba el estado informado por el servidor. Distingue una solicitud aceptada de una operación terminada y conserva solo evidencia mínima sin información sensible. Cuando exista una operación en curso, espera su estado final en la vista correspondiente antes de iniciar otra acción sobre el mismo objeto.

Ejemplo de laboratorio

Usa el tenant sintético `documentation-lab` y, cuando corresponda, el worker `worker-demo-a`. Anota el estado anterior, realiza una única acción autorizada y compara el estado posterior. Estos nombres ilustran el procedimiento; no identifican registros de producción ni autorizan crear recursos cloud. Para una revisión sin cambios, consulta el detalle y cancela cualquier formulario abierto.

Si algo falla

Si aparece un conflicto, actualiza el inventario y revisa los valores vigentes antes de repetir. Si caducó la sesión, inicia sesión y completa MFA cuando sea requerido. Ante un error de permisos, revisa el rol y la membresía; ante un error de proveedor, consulta la operación y su mensaje seguro. No alteres almacenamiento, revisiones ni controles para forzar un resultado.

Lecturas relacionadas

- [Administrator](#)
- [Preferences](#)
- [Audit](#)

Actualización continua y lectura estable

Overview consulta cambios cada cinco segundos mientras la vista está activa. Los gráficos y las tarjetas mantienen su estructura y los datos anteriores mientras llega una respuesta; se conservan selección, rango, zoom y desplazamiento. La marca Live muestra la última actualización confirmada. Last confirmed y Refresh pending indican que la lectura no pudo renovarse: los valores visibles siguen siendo la última observación, no una medición nueva.

El intervalo de actualización de la pantalla no cambia la frecuencia de muestreo de la telemetría. Una muestra ausente se presenta como desconocida, sin convertirla en cero. Los controles de selección y las leyendas conservan su lugar mientras se actualizan los valores.

Servidor base de Dova

Los administradores de plataforma ven `vm-cloudata-pe` con la identificación Base server. Es el servidor que aloja Dova y permanece en Overview aunque falten muestras o no se pueda confirmar su estado. No se cuenta como worker de ejecución y no dispone de una acción de eliminación desde esta vista.

Las métricas proceden del muestreo del host local cuando su identidad puede verificarse; no sustituyen una consulta de estado a GCP. Si la identidad o la medición no está disponible, la fila permanece y muestra información desconocida en lugar de atribuirle actividad de otro host.

Lecturas rápidas de Administrator

Administrator carga los datos autorizados del módulo o pestaña al utilizarlos y reutiliza brevemente lecturas en RAM con límites de tamaño y caducidad. Las lecturas simultáneas iguales se agrupan; cambios en los datos, la identidad, el rol o el contexto invalidan la reutilización correspondiente. Una respuesta del contexto anterior no reemplaza la vista actual. Las actualizaciones periódicas conservan la selección y el estado visible cuando los datos siguen siendo válidos. Estas medidas reducen lecturas repetidas; el tiempo real depende de los datos, la red y el proveedor.

En las tablas, la actualización de filas conserva menús abiertos, filtros, selección y desplazamiento. Al iniciar una acción, el control muestra In progress y las demás acciones de esa tabla quedan bloqueadas hasta su resultado; los otros módulos y los controles de lectura siguen disponibles.

Navegación compartida con Dova

Dova y Administrator comparten la presentación y el comportamiento del panel izquierdo: texto, estados activo y hover, foco de teclado, colapso y alineación del usuario. El menú del usuario se abre al lado del botón y permanece dentro de la pantalla. Los módulos y las acciones disponibles siguen dependiendo de tus permisos.

Leer CPU, memoria y disco

CPU, Memory y Disk muestran Not available cuando no existe una medición actual y válida, la VM no está disponible o el heartbeat o la observación del proveedor están vencidos. Una lectura no disponible no equivale a cero. El historial anterior puede conservarse con indicación Historical o Last recorded y su fecha; no lo interpretes como uso actual.

Available disk representa espacio libre. Revisa Capacity, Used y el porcentaje para entender qué parte del disco está ocupada. El eje vertical parte de cero y se escala con la capacidad conocida; el eje horizontal usa la hora local indicada. Hover, foco y teclado permiten consultar la muestra exacta y, cuando existe capacidad, sus valores usados y totales. Last sample y la zona horaria identifican cuándo se observó el dato.

La fila Dova base server identifica el host de la plataforma y permanece protegida. Su telemetría local se usa únicamente al verificar la identidad; esos datos no confirman por sí mismos el estado nativo de la VM en la nube.

Cloud workers

Provisiona capacidad externa, consulta el estado real de la VM y gestiona sus acciones desde la tabla.

Antes de comenzar

Confirma la cuenta, el tenant y el rol activo en el menú de cuenta. Necesitas permiso de consulta para inspeccionar y el permiso de gestión específico para cambiar registros. Una opción oculta o deshabilitada puede reflejar el alcance autorizado; no cambies de tenant para eludir una denegación. Usa únicamente registros del espacio que tienes permitido administrar.

Procedimiento

1. En Cloud workers usa Connections para configurar y verificar la cuenta cloud y sus regiones autorizadas.
2. En Profiles define vCPU, Memory y Disk.
3. En Workers elige Provision worker, revisa tenant, conexión, región y perfil y confirma una vez.
4. Consulta Provisioning history para el resultado. En Workers usa Refresh para actualizar la observación del proveedor.
5. Abre More actions en la fila o el menú contextual de DOVA_DATA_GRID para las acciones autorizadas. El acceso directo de la fila abre View details; Columns permite mostrar la columna opcional Runtime.

Comportamiento y límites

Hay cuatro pestañas: Workers, Provisioning history, Connections y Profiles. Google Cloud, AWS y Azure son tipos de conexión; cada operación depende de las capacidades efectivas del proveedor y de tus permisos.

Status muestra el estado de la VM observado en la nube: Running, Stopped o Paused. En GCP, Paused corresponde a SUSPENDED. También puede mostrar transiciones como Starting o Stopping. Unknown o Unavailable indican que no hay una observación verificable; Last known identifica un estado anterior que no pudo actualizarse. No interpretes esos estados como un heartbeat reciente ni como capacidad disponible.

La columna opcional Runtime muestra el proceso del worker y su admisión de trabajos: Online, Offline, Draining o Disabled. Actívala desde Columns. Es independiente del estado de la VM: Running no garantiza que el worker esté Online. Pause scheduling detiene nuevas asignaciones y permite terminar las activas; Resume scheduling vuelve a admitirlas. Estas acciones no suspenden ni reanudan la VM en el proveedor.

El menú de la fila y los controles junto a Worker name ofrecen acciones según el estado observado y los permisos. Start VM inicia una VM Stopped y Resume VM retoma una VM Paused cuando el proveedor lo admite. Un worker predeterminado, ocupado o público puede bloquear Stop VM. Un worker público comparte capacidad; no vuelve públicos la Session consumidora ni sus datos.

Reset VM reinicia de forma forzada una VM GCP en estado Running y requiere confirmación. Antes de enviarlo al proveedor, el servidor comprueba nuevamente que no existan trabajos, ejecuciones ni reservas activas en ningún tenant autorizado a usar ese worker. Un worker predeterminado o público puede reiniciarse si esta comprobación completa confirma que está inactivo; conserva sus referencias y su política de compartición. La acción se muestra solo para proveedores y estados compatibles.

Mientras un Reset no tenga resultado confirmado, Dova bloquea nuevas asignaciones y otro Reset. El estado principal sigue mostrando la VM; Reset pending o Check reset informa por separado de la operación. Usa Refresh para consultar su resultado sin volver a enviarla. Solo una operación nativa terminada y una observación válida permiten resolver el bloqueo; una VM Running por sí sola no demuestra que el Reset haya terminado. Un Reset confirmado no garantiza que el proceso esté listo: Runtime espera el siguiente heartbeat.

Revisa Tenant, Public, Status, Slots y Last seen. Los slots configurados y los disponibles pueden diferir por admisión física. Una estimación de precio no es garantía de factura ni una confirmación de capacidad disponible.

Red de workers GCP

Los nuevos workers GCP se provisionan solo con IP privada; no se ofrece un selector de IP pública. Configura Cloud NAT en la región de la subred para que el worker acceda por HTTPS a Dova y descargue paquetes y dependencias de Debian, PyPI y GitHub. Private Google Access facilita el acceso a APIs de Google, pero no sustituye a Cloud NAT para esos destinos externos. Revisa también DNS y las reglas de salida necesarias para los conectores autorizados.

Para mantenimiento, usa IAP SSH con OS Login cuando estén configurados los roles IAM y las reglas de firewall correspondientes. El worker obtiene trabajos y envía heartbeat mediante conexiones salientes; no necesita un puerto de aplicación abierto para recibirlos.

La política `public_shared` comparte capacidad entre tenants autorizados; es independiente de que la VM tenga IP pública. Conserva el aislamiento de cada trabajo y el alcance de sus credenciales y datos.

Verificar el resultado

Después de guardar o confirmar, vuelve a abrir el detalle y comprueba el estado informado por el servidor. Distingue una solicitud aceptada de una operación terminada y conserva solo evidencia mínima sin información sensible. Cuando exista una operación en curso, espera su estado final en la vista correspondiente antes de iniciar otra acción sobre el mismo objeto.

Ejemplo de laboratorio

Usa el tenant sintético `documentation-lab` y, cuando corresponda, el worker `worker-demo-a`. Anota el estado anterior, realiza una única acción autorizada y compara el estado posterior. Estos nombres ilustran el procedimiento; no identifican registros de producción ni autorizan crear recursos cloud. Para una revisión sin cambios, consulta el detalle y cancela cualquier formulario abierto.

Si algo falla

Si aparece un conflicto, usa Refresh y revisa los valores vigentes antes de iniciar otra acción. Si aparece Reset pending o Check reset, usa Refresh para consultar la operación; no repitas Reset para intentar resolverla. Si el proveedor confirma un fallo, revisa su mensaje seguro y el estado actualizado antes de decidir otra acción. Si caducó la sesión, inicia sesión y completa MFA cuando sea requerido. Ante un error de permisos, revisa el rol y la membresía. No alteres almacenamiento, revisiones ni controles para forzar un resultado.

Lecturas relacionadas

- [Administrator](#)
- [Preferences](#)
- [Audit](#)

Controles de la VM sincronizados con el proveedor

En el menú de la fila, Pause VM suspende una VM GCP Running y produce el estado Paused (SUSPENDED). Resume VM reanuda una VM Paused; Start VM inicia una VM Stopped y Stop VM la detiene. Reset VM reinicia una VM Running y conserva su registro. Dova comprueba el estado y la titularidad vigentes antes de enviar la acción y consulta el resultado nativo; una solicitud aceptada no equivale a una operación terminada.

Pause VM es diferente de Pause scheduling: el primero suspende la VM; el segundo deja de admitir trabajos y permite finalizar los activos. Tras Start, Resume o Reset, Runtime necesita un heartbeat autenticado para confirmar que el proceso está Online. Las acciones incompatibles se deshabilitan durante una operación o mientras existan trabajos y dependencias que impidan realizarla.

GCP dispone de estos controles nativos según el estado. Azure ofrece Start, Stop y Deprovision; Pause y Reset no se presentan como operaciones equivalentes. AWS puede figurar como tipo de conexión, pero su ciclo de vida de VM no está disponible en esta edición.

Retirar una VM y recuperar las asignaciones

Abre Remove VM and unlink, revisa la identidad del recurso y el impacto en las Sessions, ejecuciones y pools, y confirma la eliminación prevista. Para una VM propia compatible, no necesitas desarmar manualmente cada referencia ni apagarla como requisito previo de esta operación. Retirar capacidad puede interrumpir ejecuciones activas; programa la operación según tu necesidad.

Después de confirmar la ausencia en el proveedor, Dova retira el worker activo, limpia referencias de Runtime y muestra Unassigned en los bindings de Sessions que usaban ese worker. Conserva las definiciones, versiones publicadas, configuración e historial de auditoría. Si la VM se elimina directamente en el proveedor, la conciliación aplica el mismo tratamiento cuando comprueba su ausencia. Un error al consultar al proveedor no demuestra que la VM se haya borrado.

Antes de volver a ejecutar, selecciona un worker compatible y permitido y revisa el pool del entorno. Unassigned no toma otro worker automáticamente. Puedes aprovisionar una VM nueva y asignarla explícitamente. Los permisos, la propiedad del recurso, el servidor base protegido y las operaciones pendientes siguen verificándose; la retirada no es una forma de eludir esas protecciones.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Pool de workers por entorno

En Configuration → Dova Integration Runtime, Environment worker pool muestra el entorno actual. Allowed workers permite asignar workers existentes; Default worker debe pertenecer a esa selección. Pool policy empieza en Dedicated. Selecciona Shared únicamente cuando decidas compartir capacidad y todos los pools participantes también declaren Shared. Un worker físico no se comparte por una coincidencia de nombre.

Save environment pool guarda solamente la política del entorno seleccionado y no inicia una ejecución ni cambia el default global del tenant. La revisión usa una huella de la configuración; si otro usuario la cambia, Refresh pool permite consultar y revisar nuevamente antes de guardar. Los controles requieren administración de configuración y Runtime por entorno. Built-in no se ofrece como worker de ejecución.

Sin un pool explícito no se presta automáticamente el pool de Development a Production ni se ejecutan nuevos trabajos con un default implícito. Las ejecuciones legacy solo pueden conservar su worker anterior cuando el servidor verifica la cadena publicada exacta, su configuración y la pertenencia permitida. Test y Stage consumen el bucket Development del plan; Production usa el bucket Production. La pertenencia al pool no sustituye capacidad, estado online, perfil compatible, cuotas ni concurrencia.

Antes de detener o retirar capacidad

Pause VM y Stop VM conservan sus comprobaciones de trabajos activos, worker predeterminado y capacidad compartida; revisa las referencias y el impacto antes de utilizarlos. Pause scheduling detiene nuevas asignaciones y permite terminar el trabajo actual. Para retirar definitivamente la VM, utiliza Remove VM and unlink y revisa su confirmación: después de verificar ausencia cloud, Dova limpia las asignaciones y muestra Unassigned sin borrar definiciones ni versiones publicadas. Antes de ejecutar de nuevo, asigna capacidad compatible explícitamente.

Las acciones requieren el permiso del módulo y Runtime del entorno correspondiente, o un permiso global concedido explícitamente. Una VM compartida requiere autorización sobre todos sus entornos.

Recursos y permisos por entorno

El Tenant y Environment seleccionados en el menú de usuario delimitan Cloud workers, sus conexiones cloud, perfiles de aprovisionamiento, operaciones y backups de dependencias. Un Domain declarado en el recurso también debe coincidir. Overview, la historia de recursos y los eventos operativos de Audit respetan ese alcance. Usuarios, roles, tenants y facturación conservan su autoridad propia: no se crean copias independientes por entorno.

Para gestionar una VM, además del permiso del módulo se requiere `etl.environments.<entorno>.runtime.manage`, o el permiso global de Runtime cuando fue concedido explícitamente. Si varios entornos comparten la VM, necesitas autorización en todos los entornos afectados. Cambiar el selector no mueve la VM ni amplía tus permisos.

Controles compactos según el estado cloud

Junto a Worker name, los iconos Pause, Start/Resume, Stop y Reset muestran las acciones de la VM. Revisa el texto de ayuda de cada icono. Start está disponible para una VM Stopped y Resume para una VM Paused cuando el proveedor y tus permisos lo permiten. El estado Offline del proceso del worker, por sí solo, no impide iniciar una VM apagada.

Dova comprueba el estado cloud actual y las acciones autorizadas. Un estado sin actualizar, una operación pendiente, trabajos activos o una restricción de permisos puede deshabilitar un control. Después de Start, Resume o Reset, espera el resultado del proveedor y el heartbeat del worker; VM Running y Runtime Online son verificaciones diferentes. Si el resultado es incierto, consulta Refresh antes de repetir la acción.

Observar los cambios del proveedor

GCP ofrece Start, Stop, Pause/Suspend, Resume, Reset y retirada de la VM según su estado. Azure ofrece Start, Stop/Deallocate y retirada; no presenta Pause/Resume/Reset como operaciones equivalentes. AWS no tiene un ciclo de vida de VM habilitado en esta edición. Las capacidades visibles dependen del proveedor y de la autorización efectiva.

La sincronización con el proveedor es periódica. Los cambios realizados directamente en la nube se reflejan al obtener una nueva observación válida; no es una notificación instantánea para toda la flota. Comprueba la hora de observación y utiliza Refresh para consultar. Una operación aceptada no prueba que el cambio haya terminado, y una VM Running todavía necesita heartbeat para mostrar Runtime Online.

Usuarios y acceso

Administra identidades, membresías y roles dentro del alcance autorizado.

Antes de comenzar

Confirma la cuenta, el tenant y el rol activo en el menú de cuenta. Necesitas permiso de consulta para inspeccionar y el permiso de gestión específico para cambiar registros. Una opción oculta o deshabilitada puede reflejar el alcance autorizado; no cambies de tenant para eludir una denegación. Usa únicamente registros del espacio que tienes permitido administrar.

Procedimiento

1. Abre Users, busca la cuenta y elige View details.
2. En New user o Edit user introduce Name, Email, Tenant y el estado apropiado.
3. Usa Select roles para asignar las responsabilidades necesarias; guarda y vuelve a revisar el registro.

Comportamiento y límites

El estado de identidad, el estado de membresía y la preparación de acceso son conceptos diferentes. Cambiar membresías requiere administración de plataforma. El actor no puede delegar permisos que no posee.

No se permite desactivar o degradar la propia cuenta mediante estas acciones, ni eliminar el último administrador de plataforma activo. No publiques enlaces de activación ni material de recuperación.

Comprueba también Access y Last login. Never indica ausencia de un inicio observado, no una ejecución fallida. El indicador de presencia no concede acceso y una cuenta activa puede seguir necesitando activación o MFA.

Verificar el resultado

Después de guardar o confirmar, vuelve a abrir el detalle y comprueba el estado informado por el servidor. Distingue una solicitud aceptada de una operación terminada y conserva solo evidencia mínima sin información sensible. Cuando exista una operación en curso, espera su estado final en la vista correspondiente antes de iniciar otra acción sobre el mismo objeto.

Ejemplo de laboratorio

Usa el tenant sintético `documentation-lab` y, cuando corresponda, el worker `worker-demo-a`. Anota el estado anterior, realiza una única acción autorizada y compara el estado posterior. Estos nombres ilustran el procedimiento; no identifican registros de producción ni autorizan crear recursos cloud. Para una revisión sin cambios, consulta el detalle y cancela cualquier formulario abierto.

Si algo falla

Si aparece un conflicto, actualiza el inventario y revisa los valores vigentes antes de repetir. Si caducó la sesión, inicia sesión y completa MFA cuando sea requerido. Ante un error de permisos, revisa el rol y la membresía; ante un error de proveedor, consulta la operación y su mensaje seguro. No alteres almacenamiento, revisiones ni controles para forzar un resultado.

Lecturas relacionadas

- [Administrator](#)
- [Preferences](#)
- [Audit](#)

Identidad y entornos

El Tenant y Environment seleccionados en el menú de usuario delimitan Cloud workers, sus conexiones cloud, perfiles de aprovisionamiento, operaciones y backups de dependencias. Un Domain declarado en el recurso también debe coincidir. Overview, la historia de recursos y los eventos operativos de Audit respetan ese alcance. Usuarios, roles, tenants y facturación conservan su autoridad propia: no se crean copias independientes por entorno.

Tenants y capacidad

Separa espacios operativos, acceso y condiciones comerciales.

Antes de comenzar

Confirma la cuenta, el tenant y el rol activo en el menú de cuenta. Necesitas permiso de consulta para inspeccionar y el permiso de gestión específico para cambiar registros. Una opción oculta o deshabilitada puede reflejar el alcance autorizado; no cambies de tenant para eludir una denegación. Usa únicamente registros del espacio que tienes permitido administrar.

Procedimiento

1. Abre Tenants y revisa View details del espacio.
2. Para una modificación autorizada, usa New tenant o Edit tenant y completa Name, Type, Parent reseller, Status y los límites permitidos.
3. Comprueba después la suscripción efectiva.

Comportamiento y límites

El formulario distingue Organization, Individual y Reseller. Cambiar de tenant revalida la membresía activa. Un nombre de plan no describe por sí solo todos los derechos contratados. Crear un tenant y crear un usuario son operaciones separadas.

El vencimiento comercial conserva los objetos y permite las lecturas seguras previstas. Una reactivación comercial no anula una suspensión de seguridad. Los valores actuales proceden del plan y contrato vigentes.

Separa Seats, Monthly jobs y Storage MB: son recursos con unidades distintas. En un trial vencido o estado comercial restringido, revisa el aviso y Plans & billing sin asumir que puedes crear, editar o ejecutar objetos.

Verificar el resultado

Después de guardar o confirmar, vuelve a abrir el detalle y comprueba el estado informado por el servidor. Distingue una solicitud aceptada de una operación terminada y conserva solo evidencia mínima sin información sensible. Cuando exista una operación en curso, espera su estado final en la vista correspondiente antes de iniciar otra acción sobre el mismo objeto.

Ejemplo de laboratorio

Usa el tenant sintético `documentation-lab` y, cuando corresponda, el worker `worker-demo-a`. Anota el estado anterior, realiza una única acción autorizada y compara el estado posterior. Estos nombres ilustran el procedimiento; no identifican registros de producción ni autorizan crear recursos cloud. Para una revisión sin cambios, consulta el detalle y cancela cualquier formulario abierto.

Si algo falla

Si aparece un conflicto, actualiza el inventario y revisa los valores vigentes antes de repetir. Si caducó la sesión, inicia sesión y completa MFA cuando sea requerido. Ante un error de permisos, revisa el rol y la membresía; ante un error de proveedor, consulta la operación y su mensaje seguro. No alteres almacenamiento, revisiones ni controles para forzar un resultado.

Lecturas relacionadas

- [Administrator](#)
- [Preferences](#)
- [Audit](#)

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Roles y permisos

Agrupar capacidades según la responsabilidad y el tenant.

Antes de comenzar

Confirma la cuenta, el tenant y el rol activo en el menú de cuenta. Necesitas permiso de consulta para inspeccionar y el permiso de gestión específico para cambiar registros. Una opción oculta o deshabilitada puede reflejar el alcance autorizado; no cambies de tenant para eludir una denegación. Usa únicamente registros del espacio que tienes permitido administrar.

Procedimiento

1. En Roles & permissions abre Roles y View details.
2. Consulta Permissions para identificar la capacidad requerida.
3. En New role o Edit role define Key, Label y los permisos mínimos; guarda y verifica el acceso con una cuenta de laboratorio autorizada.

Comportamiento y límites

El catálogo describe permisos, pero no demuestra que el usuario los tenga asignados. Los roles de plataforma pertenecen al alcance de plataforma y los roles personalizados permanecen ligados a su tenant.

El administrador maestro único tiene una comprobación específica para la exención de cuota DovaAI. No todos los administradores reciben acceso ilimitado. Un permiso visible no permite saltar límites de tenant ni crear superusuarios adicionales.

Comprueba la asignación efectiva en Users después de guardar. La exención DovaAI del maestro requiere identidad persistida activa, tenant de plataforma activo, membresía y rol seleccionado válidos; la disponibilidad del proveedor sigue siendo necesaria.

Verificar el resultado

Después de guardar o confirmar, vuelve a abrir el detalle y comprueba el estado informado por el servidor. Distingue una solicitud aceptada de una operación terminada y conserva solo evidencia mínima sin información sensible. Cuando exista una operación en curso, espera su estado final en la vista correspondiente antes de iniciar otra acción sobre el mismo objeto.

Ejemplo de laboratorio

Usa el tenant sintético `documentation-lab` y, cuando corresponda, el worker `worker-demo-a`. Anota el estado anterior, realiza una única acción autorizada y compara el estado posterior. Estos nombres ilustran el procedimiento; no identifican registros de producción ni autorizan crear recursos cloud. Para una revisión sin cambios, consulta el detalle y cancela cualquier formulario abierto.

Si algo falla

Si aparece un conflicto, actualiza el inventario y revisa los valores vigentes antes de repetir. Si caducó la sesión, inicia sesión y completa MFA cuando sea requerido. Ante un error de permisos, revisa el rol y la membresía; ante un error de proveedor, consulta la operación y su mensaje seguro. No alteres almacenamiento, revisiones

ni controles para forzar un resultado.

Lecturas relacionadas

- [Administrator](#)
- [Preferences](#)
- [Audit](#)

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Permisos de entorno

La autorización combina permisos del módulo, acceso al Domain y capacidades del entorno. El catálogo actual contiene 23 permisos de entorno: `etl.environments.view`, `etl.environments.{development,test,stage,production}.{view,edit,publish,run,runtime.manage}`, `etl.environments.promote` y `etl.environments.runtime.manage`. Los permisos globales de lectura y Runtime conservan su alcance explícito; un permiso de un entorno no concede acceso a todos los demás. Administrar un módulo no sustituye estas capacidades.

ETL Developer puede trabajar en Development, Test y Stage y no tiene acceso a Production por defecto. Release Manager puede revisar los cuatro entornos y publicar en Test, Stage y Production para promover versiones autorizadas; no obtiene edición ni ejecución de Production. La promoción conserva el preview, las dependencias publicadas exactas y la aprobación explícita de Production. Los roles personalizados permiten asignar las capacidades necesarias por entorno. La identidad del usuario no se duplica por ambiente; verifica el rol activo, la membresía y sus permisos efectivos.

Los editores de Mapping, Session y Workflow de Production permanecen de solo lectura: las versiones aceptadas llegan mediante promoción. Configurar Connections existentes o nuevas de Production es una capacidad separada que exige `etl.connections.manage` y `etl.environments.production.edit`; no habilita edición de esos tres tipos ni modifica Connections legacy de solo lectura.

Suscripciones, DPU y facturación

Comprende el consumo y verifica condiciones comerciales vigentes.

Antes de comenzar

Confirma la cuenta, el tenant y el rol activo en el menú de cuenta. Necesitas permiso de consulta para inspeccionar y el permiso de gestión específico para cambiar registros. Una opción oculta o deshabilitada puede reflejar el alcance autorizado; no cambies de tenant para eludir una denegación. Usa únicamente registros del espacio que tienes permitido administrar.

Procedimiento

1. Abre Subscription & billing y selecciona Tenant.
2. Revisa Subscription y el perfil de facturación.
3. Usa DPU & calculator para estimaciones y Payments & invoicing para la configuración comercial permitida.
4. Consulta Licenses, Commerce, Global catalog y Tenant offers según tu responsabilidad.

Comportamiento y límites

Las siete pestañas separan contratos, uso, pagos y catálogo. DPU mide capacidad; DPU-h multiplica capacidad por tiempo activo. Development y Production tienen fondos y concurrencia separados. Las tareas agregan el consumo; el Workflow no cobra otra vez.

La cola y la validación estructural no son facturables. La concurrencia contratada no crea workers. Las consultas aceptadas DovaAI tienen cuotas por tenant y usuario, separadas del DPU. Un retorno del navegador o estado manual de factura no demuestra pago confirmado.

Ejemplo de cálculo: 4 DPU durante 15 minutos consumen 1 DPU-h. Los créditos siguen períodos financiados verificados, no necesariamente meses calendario. Un sandbox de pagos o documentos fiscales no acredita aceptación productiva.

Verificar el resultado

Después de guardar o confirmar, vuelve a abrir el detalle y comprueba el estado informado por el servidor. Distingue una solicitud aceptada de una operación terminada y conserva solo evidencia mínima sin información sensible. Cuando exista una operación en curso, espera su estado final en la vista correspondiente antes de iniciar otra acción sobre el mismo objeto.

Ejemplo de laboratorio

Usa el tenant sintético `documentation-lab` y, cuando corresponda, el worker `worker-demo-a`. Anota el estado anterior, realiza una única acción autorizada y compara el estado posterior. Estos nombres ilustran el procedimiento; no identifican registros de producción ni autorizan crear recursos cloud. Para una revisión sin cambios, consulta el detalle y cancela cualquier formulario abierto.

Si algo falla

Si aparece un conflicto, actualiza el inventario y revisa los valores vigentes antes de repetir. Si caducó la sesión, inicia sesión y completa MFA cuando sea requerido. Ante un error de permisos, revisa el rol y la membresía; ante un error de proveedor, consulta la operación y su mensaje seguro. No alteres almacenamiento, revisiones ni controles para forzar un resultado.

Lecturas relacionadas

- [Administrator](#)
- [Preferences](#)
- [Audit](#)

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Pool de workers por entorno

En Configuration → Dova Integration Runtime, Environment worker pool muestra el entorno actual. Allowed workers permite asignar workers existentes; Default worker debe pertenecer a esa selección. Pool policy empieza en Dedicated. Selecciona Shared únicamente cuando decidas compartir capacidad y todos los pools participantes también declaren Shared. Un worker físico no se comparte por una coincidencia de nombre.

Save environment pool guarda solamente la política del entorno seleccionado y no inicia una ejecución ni cambia el default global del tenant. La revisión usa una huella de la configuración; si otro usuario la cambia, Refresh pool permite consultar y revisar nuevamente antes de guardar. Los controles requieren administración de configuración y Runtime por entorno. Built-in no se ofrece como worker de ejecución.

Sin un pool explícito no se presta automáticamente el pool de Development a Production ni se ejecutan nuevos trabajos con un default implícito. Las ejecuciones legacy solo pueden conservar su worker anterior cuando el servidor verifica la cadena publicada exacta, su configuración y la pertenencia permitida. Test y Stage consumen el bucket Development del plan; Production usa el bucket Production. La pertenencia al pool no sustituye capacidad, estado online, perfil compatible, cuotas ni concurrencia.

Proyección de financiación y periodo

La proyección de financiación usa el formato canónico de periodos de Billing y se reconcilia con la suscripción autorizada. La corrección de Subscription funding projection does not match no cambia el periodo contractual, su ciclo, los créditos concedidos ni el consumo; tampoco extiende el Trial. Las comprobaciones de tenant, contrato y financiación siguen aplicándose. Ante un error persistente, revisa la operación y la suscripción existentes antes de repetir registro o checkout.

Auditoría administrativa

Investiga acciones dentro del alcance permitido.

Antes de comenzar

Confirma la cuenta, el tenant y el rol activo en el menú de cuenta. Necesitas permiso de consulta para inspeccionar y el permiso de gestión específico para cambiar registros. Una opción oculta o deshabilitada puede reflejar el alcance autorizado; no cambies de tenant para eludir una denegación. Usa únicamente registros del espacio que tienes permitido administrar.

Procedimiento

1. Abre Audit y utiliza la búsqueda y los filtros para encontrar el evento.
2. En View details revisa acción, actor, fecha y objeto relacionado.
3. Abre la pantalla correspondiente para confirmar su estado actual y registra únicamente la evidencia necesaria.

Comportamiento y límites

Los eventos ayudan a correlacionar cambios administrativos. La paginación y las columnas modifican la presentación, no el registro. Las exportaciones deben limitarse al propósito autorizado.

Una entrada de auditoría no demuestra por sí sola que el proveedor terminara una VM, que un pago se liquidara o que una ejecución produjera el resultado correcto. Protege identidades y contenido sensible.

Para investigar una modificación de rol, correlaciona el evento con Users y Roles & permissions. Para infraestructura consulta Cloud workers; para ejecuciones consulta Monitor. Conserva la diferencia entre estado histórico del evento y estado actual del objeto.

Verificar el resultado

Después de guardar o confirmar, vuelve a abrir el detalle y comprueba el estado informado por el servidor. Distingue una solicitud aceptada de una operación terminada y conserva solo evidencia mínima sin información sensible. Cuando exista una operación en curso, espera su estado final en la vista correspondiente antes de iniciar otra acción sobre el mismo objeto.

Ejemplo de laboratorio

Usa el tenant sintético `documentation-lab` y, cuando corresponda, el worker `worker-demo-a`. Anota el estado anterior, realiza una única acción autorizada y compara el estado posterior. Estos nombres ilustran el procedimiento; no identifican registros de producción ni autorizan crear recursos cloud. Para una revisión sin cambios, consulta el detalle y cancela cualquier formulario abierto.

Si algo falla

Si aparece un conflicto, actualiza el inventario y revisa los valores vigentes antes de repetir. Si caducó la sesión, inicia sesión y completa MFA cuando sea requerido. Ante un error de permisos, revisa el rol y la membresía; ante un error de proveedor, consulta la operación y su mensaje seguro. No alteres almacenamiento, revisiones ni controles para forzar un resultado.

Lecturas relacionadas

- [Administrator](#)
- [Preferences](#)
- [Audit](#)

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Revisión de registros

Revisa solicitudes pendientes sin confundir aprobación con provisión.

Antes de comenzar

Confirma la cuenta, el tenant y el rol activo en el menú de cuenta. Necesitas permiso de consulta para inspeccionar y el permiso de gestión específico para cambiar registros. Una opción oculta o deshabilitada puede reflejar el alcance autorizado; no cambies de tenant para eludir una denegación. Usa únicamente registros del espacio que tienes permitido administrar.

Procedimiento

1. Abre Registrations, encuentra la solicitud y revisa View details.
2. Para una solicitud pendiente elegible, usa Approve registration o la acción de rechazo, verifica el solicitante y completa la razón requerida.
3. Guarda y actualiza el inventario antes de otra decisión.

Comportamiento y límites

Ver solicitudes y decidir las son permisos diferentes. El resultado registra la revisión administrativa; la creación y preparación de tenant y usuario deben comprobarse por separado.

Si otro administrador ya decidió o el registro cambió, refresca y revisa el estado más reciente. No anuncies que una cuenta está lista únicamente porque la solicitud figura aprobada.

Revisa nombre, organización, tipo, fecha recibida y revisión anterior. La razón ayuda a explicar la decisión, pero no debe incluir datos innecesarios. Una solicitud no pendiente deja de ser elegible para una nueva decisión ordinaria.

Verificar el resultado

Después de guardar o confirmar, vuelve a abrir el detalle y comprueba el estado informado por el servidor. Distingue una solicitud aceptada de una operación terminada y conserva solo evidencia mínima sin información sensible. Cuando exista una operación en curso, espera su estado final en la vista correspondiente antes de iniciar otra acción sobre el mismo objeto.

Ejemplo de laboratorio

Usa el tenant sintético `documentation-lab` y, cuando corresponda, el worker `worker-demo-a`. Anota el estado anterior, realiza una única acción autorizada y compara el estado posterior. Estos nombres ilustran el procedimiento; no identifican registros de producción ni autorizan crear recursos cloud. Para una revisión sin cambios, consulta el detalle y cancela cualquier formulario abierto.

Si algo falla

Si aparece un conflicto, actualiza el inventario y revisa los valores vigentes antes de repetir. Si caducó la sesión, inicia sesión y completa MFA cuando sea requerido. Ante un error de permisos, revisa el rol y la membresía; ante un error de proveedor, consulta la operación y su mensaje seguro. No alteres almacenamiento, revisiones ni controles para forzar un resultado.

Lecturas relacionadas

- [Administrator](#)
- [Preferences](#)
- [Audit](#)

Developer Tools, CLI y acceso

Descarga herramientas verificadas y limita las credenciales.

Antes de comenzar

Confirma la cuenta, el tenant y el rol activo en el menú de cuenta. Necesitas permiso de consulta para inspeccionar y el permiso de gestión específico para cambiar registros. Una opción oculta o deshabilitada puede reflejar el alcance autorizado; no cambies de tenant para eludir una denegación. Usa únicamente registros del espacio que tienes permitido administrar.

Procedimiento

1. Abre Developer Tools y selecciona el sistema operativo de destino.
2. Revisa disponibilidad, versión, checksum y firma publicados para el paquete exacto.
3. Descarga, comprueba integridad y sigue Quick start en un entorno autorizado.
4. En API Keys crea únicamente el alcance necesario y revoca la clave cuando deje de utilizarse.

Comportamiento y límites

La versión del paquete CLI puede diferir de la versión de Dova. Descargar una herramienta no concede permisos. Las claves conservan el alcance autorizado de tenant y rol; no sustituyen las comprobaciones del servidor.

No inventes disponibilidad de plataformas, URLs o firmas. No pongas claves en documentación, historial de terminal ni código. Una clave revocada o sin vínculo verificable no debe recuperarse mediante cambios manuales del almacenamiento.

Distingue disponibilidad, integridad y firma: son propiedades diferentes. La máquina de destino puede usar otro sistema operativo que el navegador. Revisa el alcance del comando antes de ejecutarlo y utiliza las instrucciones incluidas con ese paquete exacto.

Verificar el resultado

Después de guardar o confirmar, vuelve a abrir el detalle y comprueba el estado informado por el servidor. Distingue una solicitud aceptada de una operación terminada y conserva solo evidencia mínima sin información sensible. Cuando exista una operación en curso, espera su estado final en la vista correspondiente antes de iniciar otra acción sobre el mismo objeto.

Ejemplo de laboratorio

Usa el tenant sintético `documentation-lab` y, cuando corresponda, el worker `worker-demo-a`. Anota el estado anterior, realiza una única acción autorizada y compara el estado posterior. Estos nombres ilustran el procedimiento; no identifican registros de producción ni autorizan crear recursos cloud. Para una revisión sin cambios, consulta el detalle y cancela cualquier formulario abierto.

Si algo falla

Si aparece un conflicto, actualiza el inventario y revisa los valores vigentes antes de repetir. Si caducó la sesión, inicia sesión y completa MFA cuando sea requerido. Ante un error de permisos, revisa el rol y la membresía; ante un error de proveedor, consulta la operación y su mensaje seguro. No alteres almacenamiento, revisiones ni controles para forzar un resultado.

Lecturas relacionadas

- [Administrator](#)
- [Preferences](#)
- [Audit](#)

Preferencias y alcance

Configura la política adecuada sin mezclarla con la apariencia personal.

Antes de comenzar

Confirma la cuenta, el tenant y el rol activo en el menú de cuenta. Necesitas permiso de consulta para inspeccionar y el permiso de gestión específico para cambiar registros. Una opción oculta o deshabilitada puede reflejar el alcance autorizado; no cambies de tenant para eludir una denegación. Usa únicamente registros del espacio que tienes permitido administrar.

Procedimiento

1. Abre Preferences y elige Notifications, Authentication, Limits, Email delivery o Data Cleanup.
2. Confirma el alcance antes de cambiar valores.
3. En Limits distingue Platform defaults de la configuración del tenant.
4. Guarda solamente los ajustes permitidos y verifica su persistencia.

Comportamiento y límites

Las cinco categorías tienen autoridades diferentes. El menú de cuenta permite apariencia, paleta y cambios de membresía autorizados; estos ajustes no cambian permisos ni contrato. Las interfaces admiten tema claro y oscuro.

Poder consultar Preferences no permite modificar autenticación global o SMTP. La disponibilidad de un botón depende del rol activo. No cambies límites para evitar una restricción comercial o de seguridad.

Para Notifications distingue Preview, Test channels y Save preferences. Para Authentication prueba la configuración guardada. Para Data Cleanup revisa el inventario y la confirmación destructiva. Cada acción informa un resultado diferente y requiere su propia verificación.

Verificar el resultado

Después de guardar o confirmar, vuelve a abrir el detalle y comprueba el estado informado por el servidor. Distingue una solicitud aceptada de una operación terminada y conserva solo evidencia mínima sin información sensible. Cuando exista una operación en curso, espera su estado final en la vista correspondiente antes de iniciar otra acción sobre el mismo objeto.

Ejemplo de laboratorio

Usa el tenant sintético `documentation-lab` y, cuando corresponda, el worker `worker-demo-a`. Anota el estado anterior, realiza una única acción autorizada y compara el estado posterior. Estos nombres ilustran el procedimiento; no identifican registros de producción ni autorizan crear recursos cloud. Para una revisión sin cambios, consulta el detalle y cancela cualquier formulario abierto.

Si algo falla

Si aparece un conflicto, actualiza el inventario y revisa los valores vigentes antes de repetir. Si caducó la sesión, inicia sesión y completa MFA cuando sea requerido. Ante un error de permisos, revisa el rol y la membresía; ante un error de proveedor, consulta la operación y su mensaje seguro. No alteres almacenamiento, revisiones ni controles para forzar un resultado.

Lecturas relacionadas

- [Administrator](#)
- [Preferences](#)
- [Audit](#)

Idioma de la interfaz

En el menú de usuario, selecciona ES, EN o PT para cambiar el idioma de la interfaz. La preferencia se guarda en tu perfil y se utiliza en Dova y Administrator. Si no se puede guardar, la interfaz informa el fallo y restaura la elección anterior. Cambian las etiquetas disponibles de la aplicación; no se renombran artefactos, identidades, campos, código ni datos del usuario. El estilo, la paleta y los permisos mantienen su configuración propia.

Autenticación externa y MFA

Configura identidad sin reducir la verificación ni la autorización.

Antes de comenzar

Confirma la cuenta, el tenant y el rol activo en el menú de cuenta. Necesitas permiso de consulta para inspeccionar y el permiso de gestión específico para cambiar registros. Una opción oculta o deshabilitada puede reflejar el alcance autorizado; no cambies de tenant para eludir una denegación. Usa únicamente registros del espacio que tienes permitido administrar.

Procedimiento

1. En Preferences → Authentication selecciona Google, Microsoft, Okta, Auth0, GitHub o GitLab.
2. Registra la Authorized callback URL exacta en el proveedor y completa sus campos.
3. Reemplaza el secreto solamente si corresponde, guarda y prueba la configuración guardada con una identidad autorizada.

Comportamiento y límites

La identidad externa no concede roles ni elimina MFA. El autorregistro nuevo acepta identidades Google verificadas según la política actual; no se debe generalizar a todos los proveedores. Si el inicio indica mfaRequired, authenticated=false es esperado hasta completar el segundo factor.

Una verificación vencida exige iniciar sesión de nuevo y usar el desafío actual. No publiques códigos, secretos ni desafíos reales. Conserva una vía de recuperación autorizada antes de cambiar la configuración global.

Microsoft utiliza Tenant ID; Okta y Auth0 utilizan Issuer URL. GitHub requiere que el correo primario verificado corresponda a un usuario Dova activo. Usa las instrucciones de cada tarjeta en lugar de generalizar campos entre proveedores.

Verificar el resultado

Después de guardar o confirmar, vuelve a abrir el detalle y comprueba el estado informado por el servidor. Distingue una solicitud aceptada de una operación terminada y conserva solo evidencia mínima sin información sensible. Cuando exista una operación en curso, espera su estado final en la vista correspondiente antes de iniciar otra acción sobre el mismo objeto.

Ejemplo de laboratorio

Usa el tenant sintético `documentation-lab` y, cuando corresponda, el worker `worker-demo-a`. Anota el estado anterior, realiza una única acción autorizada y compara el estado posterior. Estos nombres ilustran el procedimiento; no identifican registros de producción ni autorizan crear recursos cloud. Para una revisión sin cambios, consulta el detalle y cancela cualquier formulario abierto.

Si algo falla

Si aparece un conflicto, actualiza el inventario y revisa los valores vigentes antes de repetir. Si caducó la sesión, inicia sesión y completa MFA cuando sea requerido. Ante un error de permisos, revisa el rol y la membresía; ante un error de proveedor, consulta la operación y su mensaje seguro. No alteres almacenamiento, revisiones

ni controles para forzar un resultado.

Lecturas relacionadas

- [Administrator](#)
- [Preferences](#)
- [Audit](#)

Acceso a Administrator

Una cuenta autenticada sin permisos de Administrator que abre la página de Administrator vuelve a Dova. Esta navegación no concede acceso administrativo: las llamadas a las API protegidas continúan sujetas a autorización y pueden responder 403. Si no tienes una sesión válida, sigue el inicio de sesión y MFA normales. Utiliza siempre el destino conocido de Dova; una URL de retorno externa no amplía el acceso.

Coherencia de la suscripción al iniciar sesión

La proyección de financiación usa el formato canónico de periodos de Billing y se reconcilia con la suscripción autorizada. La corrección de Subscription funding projection does not match no cambia el periodo contractual, su ciclo, los créditos concedidos ni el consumo; tampoco extiende el Trial. Las comprobaciones de tenant, contrato y financiación siguen aplicándose. Ante un error persistente, revisa la operación y la suscripción existentes antes de repetir registro o checkout.

Catálogo global y ofertas

Publica condiciones versionadas sin alterar contratos anteriores.

Antes de comenzar

Confirma la cuenta, el tenant y el rol activo en el menú de cuenta. Necesitas permiso de consulta para inspeccionar y el permiso de gestión específico para cambiar registros. Una opción oculta o deshabilitada puede reflejar el alcance autorizado; no cambies de tenant para eludir una denegación. Usa únicamente registros del espacio que tienes permitido administrar.

Procedimiento

1. En Subscription & billing → Global catalog inspecciona productos, planes, versiones y precios.
2. Crea un borrador de condiciones, revísalo y publica únicamente la versión autorizada.
3. Usa New price para el precio correspondiente.
4. En Tenant offers selecciona el tenant, validez y condiciones y revisa antes de publicar.

Comportamiento y límites

El catálogo es global y una oferta está vinculada a su tenant. Los contratos existentes conservan sus términos congelados. Cien unidades menores equivalen a una unidad monetaria; cien puntos básicos equivalen a un uno por ciento.

No reescribas un precio usado como si fuera un borrador. Publicar una oferta no modifica la de otro cliente. El plan y contrato vigentes determinan los importes y límites reales; este manual no constituye una tarifa fija.

Antes de publicar, confirma moneda, intervalo de cobro, tratamiento tributario y vigencia. Un nuevo precio debe referenciar la versión publicada correcta. Revisa el resumen final del pedido, no únicamente un importe aislado.

Verificar el resultado

Después de guardar o confirmar, vuelve a abrir el detalle y comprueba el estado informado por el servidor. Distingue una solicitud aceptada de una operación terminada y conserva solo evidencia mínima sin información sensible. Cuando exista una operación en curso, espera su estado final en la vista correspondiente antes de iniciar otra acción sobre el mismo objeto.

Ejemplo de laboratorio

Usa el tenant sintético `documentation-lab` y, cuando corresponda, el worker `worker-demo-a`. Anota el estado anterior, realiza una única acción autorizada y compara el estado posterior. Estos nombres ilustran el procedimiento; no identifican registros de producción ni autorizan crear recursos cloud. Para una revisión sin cambios, consulta el detalle y cancela cualquier formulario abierto.

Si algo falla

Si aparece un conflicto, actualiza el inventario y revisa los valores vigentes antes de repetir. Si caducó la sesión, inicia sesión y completa MFA cuando sea requerido. Ante un error de permisos, revisa el rol y la membresía; ante un error de proveedor, consulta la operación y su mensaje seguro. No alteres almacenamiento, revisiones ni controles para forzar un resultado.

Lecturas relacionadas

- [Administrator](#)
- [Preferences](#)
- [Audit](#)

Plantillas de correo, SMTP y avisos

Separa contenido, transporte y política de entrega.

Antes de comenzar

Confirma la cuenta, el tenant y el rol activo en el menú de cuenta. Necesitas permiso de consulta para inspeccionar y el permiso de gestión específico para cambiar registros. Una opción oculta o deshabilitada puede reflejar el alcance autorizado; no cambies de tenant para eludir una denegación. Usa únicamente registros del espacio que tienes permitido administrar.

Procedimiento

1. En Preferences → Notifications abre Execution email template, selecciona resultados y prepara asunto, preheader, HTML y texto plano.
2. Previsualiza y guarda.
3. En Email delivery revisa SMTP y remitente, guarda y prueba con un destinatario autorizado.
4. En Notifications revisa eventos, canales, destinatarios, deduplicación y horario silencioso.

Comportamiento y límites

Usa variables del catálogo como %WORKFLOW_NAME%, %STATUS%, %RUN_ID%, %OUTPUT_ROWS% y %EXECUTION_URL%. Un Replace password vacío conserva la contraseña según la indicación del formulario. La previsualización y la recepción real son comprobaciones distintas.

No incluyas credenciales, datos reales de ejecución ni endpoints secretos en ejemplos. Una entrega suprimida por la política no equivale a fallo SMTP; revisa Delivery history antes de repetir un envío.

Selecciona al menos un resultado: succeeded, failed o cancelled. El texto plano es obligatorio y debe comunicar el mismo resultado que HTML. Los horarios silenciosos dependen de su zona horaria y de ambos límites horarios.

Verificar el resultado

Después de guardar o confirmar, vuelve a abrir el detalle y comprueba el estado informado por el servidor. Distingue una solicitud aceptada de una operación terminada y conserva solo evidencia mínima sin información sensible. Cuando exista una operación en curso, espera su estado final en la vista correspondiente antes de iniciar otra acción sobre el mismo objeto.

Ejemplo de laboratorio

Usa el tenant sintético `documentation-lab` y, cuando corresponda, el worker `worker-demo-a`. Anota el estado anterior, realiza una única acción autorizada y compara el estado posterior. Estos nombres ilustran el procedimiento; no identifican registros de producción ni autorizan crear recursos cloud. Para una revisión sin cambios, consulta el detalle y cancela cualquier formulario abierto.

Si algo falla

Si aparece un conflicto, actualiza el inventario y revisa los valores vigentes antes de repetir. Si caducó la sesión, inicia sesión y completa MFA cuando sea requerido. Ante un error de permisos, revisa el rol y la membresía; ante un error de proveedor, consulta la operación y su mensaje seguro. No alteres almacenamiento, revisiones ni controles para forzar un resultado.

Lecturas relacionadas

- [Administrator](#)
- [Preferences](#)
- [Audit](#)

Instalación de workers

Prepara servidores de ejecución externos al plano de control.

Antes de comenzar

Confirma la cuenta, el tenant y el rol activo en el menú de cuenta. Necesitas permiso de consulta para inspeccionar y el permiso de gestión específico para cambiar registros. Una opción oculta o deshabilitada puede reflejar el alcance autorizado; no cambies de tenant para eludir una denegación. Usa únicamente registros del espacio que tienes permitido administrar.

Procedimiento

1. Elige instalar el agente en un servidor autorizado o provisionar infraestructura con Cloud workers.
2. Descarga el paquete actual del sistema elegido y verifica su integridad y requisitos.
3. Usa Register worker, define nombre y concurrencia, genera instrucciones y ejecútalas en el servidor correcto.
4. Comprueba heartbeat, capacidad y asignación de Session.

Comportamiento y límites

Los workers privados admiten sus rutas Linux y Windows. La capacidad pública aislada es Linux, requiere habilitación explícita y verificación del runtime aprobado. Su entorno inicial admite transformaciones de archivos CSV y Excel, sin acceso de red ni identidad cloud ambiental.

Instalar el agente no asigna Sessions existentes ni transfiere archivos. El sandbox compartido no expone archivos del host ni checkpoints persistentes. Una tarea no compatible falla explícitamente y no pasa a un proceso sin aislamiento.

La versión del agente puede diferir de 24.0.0. Usa las instrucciones generadas para el paquete y sistema elegidos. El token de inscripción se introduce mediante el mecanismo seguro previsto, nunca en ejemplos públicos o capturas compartidas.

Verificar el resultado

Después de guardar o confirmar, vuelve a abrir el detalle y comprueba el estado informado por el servidor. Distingue una solicitud aceptada de una operación terminada y conserva solo evidencia mínima sin información sensible. Cuando exista una operación en curso, espera su estado final en la vista correspondiente antes de iniciar otra acción sobre el mismo objeto.

Ejemplo de laboratorio

Usa el tenant sintético `documentation-lab` y, cuando corresponda, el worker `worker-demo-a`. Anota el estado anterior, realiza una única acción autorizada y compara el estado posterior. Estos nombres ilustran el procedimiento; no identifican registros de producción ni autorizan crear recursos cloud. Para una revisión sin cambios, consulta el detalle y cancela cualquier formulario abierto.

Si algo falla

Si aparece un conflicto, actualiza el inventario y revisa los valores vigentes antes de repetir. Si caducó la sesión, inicia sesión y completa MFA cuando sea requerido. Ante un error de permisos, revisa el rol y la membresía; ante un error de proveedor, consulta la operación y su mensaje seguro. No alteres almacenamiento, revisiones ni controles para forzar un resultado.

Lecturas relacionadas

- [Administrator](#)
- [Preferences](#)
- [Audit](#)

Reasignación y retirada de workers

Reubica referencias de ejecución con revisión de dependencias.

Antes de comenzar

Confirma la cuenta, el tenant y el rol activo en el menú de cuenta. Necesitas permiso de consulta para inspeccionar y el permiso de gestión específico para cambiar registros. Una opción oculta o deshabilitada puede reflejar el alcance autorizado; no cambies de tenant para eludir una denegación. Usa únicamente registros del espacio que tienes permitido administrar.

Procedimiento

1. En Cloud workers → Workers inspecciona el origen, usa Pause o Drain worker y espera que termine el trabajo activo.
2. Abre Reassign references, revisa el destino autorizado y compatible y confirma el impacto.
3. Consulta Dependency backups para restaurar o reasignar metadatos.
4. Si necesitas conservar continuidad antes de retirar infraestructura, verifica la capacidad de destino y las referencias migradas. Para eliminar una VM sin migración previa, sigue Remove VM and unlink y revisa su impacto.

Comportamiento y límites

La reasignación modifica referencias, no mueve bases de datos, archivos, sistema operativo o credenciales. Un respaldo de dependencias conserva metadatos e integridad; no es imagen de VM ni copia completa. La retirada pública revisa referencias entre tenants y versiones históricas todavía usadas.

La retirada exige identidad y propiedad verificadas, permisos y un resultado cloud confirmado. Una consulta de estado desconocida no demuestra ausencia. Remove VM and unlink puede limpiar referencias después de confirmar el borrado; Pause/Stop mantienen sus protecciones de trabajo activo y capacidad predeterminada.

Antes de aplicar, confirma que el destino satisface las capacidades de los objetos referenciados. Las versiones históricas inaccesibles se preservan y no bloquean por sí solas. Los archivos y conexiones deben estar disponibles por su procedimiento autorizado separado.

Verificar el resultado

Después de guardar o confirmar, vuelve a abrir el detalle y comprueba el estado informado por el servidor. Distingue una solicitud aceptada de una operación terminada y conserva solo evidencia mínima sin información sensible. Cuando exista una operación en curso, espera su estado final en la vista correspondiente antes de iniciar otra acción sobre el mismo objeto.

Ejemplo de laboratorio

Usa el tenant sintético `documentation-lab` y, cuando corresponda, el worker `worker-demo-a`. Anota el estado anterior, realiza una única acción autorizada y compara el estado posterior. Estos nombres ilustran el procedimiento; no identifican registros de producción ni autorizan crear recursos cloud. Para una revisión sin cambios, consulta el detalle y cancela cualquier formulario abierto.

Si algo falla

Si aparece un conflicto, actualiza el inventario y revisa los valores vigentes antes de repetir. Si caducó la sesión, inicia sesión y completa MFA cuando sea requerido. Ante un error de permisos, revisa el rol y la membresía; ante un error de proveedor, consulta la operación y su mensaje seguro. No alteres almacenamiento, revisiones ni controles para forzar un resultado.

Lecturas relacionadas

- [Administrator](#)
- [Preferences](#)
- [Audit](#)

Retirar una VM y recuperar las asignaciones

Abre Remove VM and unlink, revisa la identidad del recurso y el impacto en las Sessions, ejecuciones y pools, y confirma la eliminación prevista. Para una VM propia compatible, no necesitas desarmar manualmente cada referencia ni apagarla como requisito previo de esta operación. Retirar capacidad puede interrumpir ejecuciones activas; programa la operación según tu necesidad.

Después de confirmar la ausencia en el proveedor, Dova retira el worker activo, limpia referencias de Runtime y muestra Unassigned en los bindings de Sessions que usaban ese worker. Conserva las definiciones, versiones publicadas, configuración e historial de auditoría. Si la VM se elimina directamente en el proveedor, la conciliación aplica el mismo tratamiento cuando comprueba su ausencia. Un error al consultar al proveedor no demuestra que la VM se haya borrado.

Antes de volver a ejecutar, selecciona un worker compatible y permitido y revisa el pool del entorno. Unassigned no toma otro worker automáticamente. Puedes aprovisionar una VM nueva y asignarla explícitamente. Los permisos, la propiedad del recurso, el servidor base protegido y las operaciones pendientes siguen verificándose; la retirada no es una forma de eludir esas protecciones.

Limpieza ETL y ciclo de vida

Revisa el alcance destructivo y confirma explícitamente el tenant.

Antes de comenzar

Confirma la cuenta, el tenant y el rol activo en el menú de cuenta. Necesitas permiso de consulta para inspeccionar y el permiso de gestión específico para cambiar registros. Una opción oculta o deshabilitada puede reflejar el alcance autorizado; no cambies de tenant para eludir una denegación. Usa únicamente registros del espacio que tienes permitido administrar.

Procedimiento

1. En Preferences → Data Cleanup selecciona Tenant y Refresh inventory.
2. Revisa cantidades, bloqueos y respaldo autorizado.
3. Elige Clean tenant ETL data, compara el tenant y resumen, escribe la confirmación requerida y acepta la eliminación irreversible.
4. Ejecuta Delete ETL data una vez y vuelve a consultar el inventario.

Comportamiento y límites

La limpieza ETL incluye las categorías indicadas de mappings, sesiones, workflows, conexiones y credenciales, ejecuciones y Repository. Conserva usuarios, tenant, roles, facturación y auditoría administrativa global. No elimina una VM ni cancela automáticamente una suscripción.

Las ejecuciones activas bloquean la operación. Un tenant individual gestionado puede requerir su proceso separado de retención. El vencimiento comercial no elimina datos automáticamente; nunca cambies tipo o revisión para evitar un bloqueo.

El formulario no genera automáticamente una copia completa del tenant. Conserva por separado el respaldo autorizado y su recuperación comprobada. Si cambió el inventario desde la revisión, vuelve a solicitarlo antes de confirmar una eliminación.

Verificar el resultado

Después de guardar o confirmar, vuelve a abrir el detalle y comprueba el estado informado por el servidor. Distingue una solicitud aceptada de una operación terminada y conserva solo evidencia mínima sin información sensible. Cuando exista una operación en curso, espera su estado final en la vista correspondiente antes de iniciar otra acción sobre el mismo objeto.

Ejemplo de laboratorio

Usa el tenant sintético `documentation-lab` y, cuando corresponda, el worker `worker-demo-a`. Anota el estado anterior, realiza una única acción autorizada y compara el estado posterior. Estos nombres ilustran el procedimiento; no identifican registros de producción ni autorizan crear recursos cloud. Para una revisión sin cambios, consulta el detalle y cancela cualquier formulario abierto.

Si algo falla

Si aparece un conflicto, actualiza el inventario y revisa los valores vigentes antes de repetir. Si caducó la sesión, inicia sesión y completa MFA cuando sea requerido. Ante un error de permisos, revisa el rol y la membresía; ante un error de proveedor, consulta la operación y su mensaje seguro. No alteres almacenamiento, revisiones ni controles para forzar un resultado.

Lecturas relacionadas

- [Administrator](#)
- [Preferences](#)
- [Audit](#)

Conectores y autenticación

Separa capacidades del componente, acceso de la Connection y disponibilidad del worker.

Tres responsabilidades distintas

El componente define cómo leer o escribir datos. La Connection reúne metadatos del endpoint, método de autenticación y custodia protegida de credenciales. El worker aporta el adaptador, acceso de red y capacidad para ejecutar. Que un conector aparezca en el catálogo no demuestra que un worker determinado disponga del controlador ni de los permisos necesarios.

Dova ofrece orígenes de archivo delimitado, JSON, Excel, base de datos, API y almacenamiento de objetos; los destinos disponibles son archivo delimitado, JSON, base de datos, API y objetos. Revisa el tipo concreto en Connections y los campos que ofrece su formulario. No intercambies métodos de autenticación entre proveedores por similitud de nombres.

Preparar una Connection

1. Confirma tenant, rol y permiso para administrar la Connection.
2. Selecciona un conector ofrecido e introduce únicamente los metadatos autorizados del endpoint.
3. Elige un método de autenticación y custodia disponible en ese formulario. Introduce el secreto mediante su control protegido.
4. Guarda y revisa los metadatos permitidos. Prueba la Connection guardada con la acción disponible.
5. Configura y publica un Mapping de laboratorio. Enlázalo a una Session con un worker elegible y ejecuta desde la Session una extracción pequeña.

Por ejemplo, usa una Connection sintética llamada `CustomerLabReadOnly`, la base `training` y la tabla `customers`. Son nombres de ejercicio, no instrucciones para conectarse a un sistema real. Verifica campos esperados y número de filas antes de habilitar escrituras.

Configurar la lectura y el resultado

En modo tabla, las columnas seleccionadas del origen de base de datos delimitan la proyección. En modo consulta, los campos dependen del SQL admitido por el adaptador; no supongas que otra propiedad reescribe consultas arbitrarias. La compatibilidad del editor no implica que todos los modos funcionen en todos los runtimes.

Los destinos de archivo escriben el esquema recibido. Si necesitas excluir campos, agrega Field selector antes del destino. La autenticación correcta no corrige una transformación mal configurada ni determina la precisión de los datos.

Rotar y diagnosticar

Una prueba de conexión satisfactoria no certifica todas las lecturas, escrituras, políticas de reintento ni resultados. Si la prueba pasa pero la ejecución falla, revisa worker, red, permisos del recurso, versión publicada y operación concreta. Revisa dependencias antes de rotar o revocar una credencial.

Dova Vault protege la custodia; no autoriza mostrar secretos ni referencias internas. Evita pegar credenciales en SQL, descripciones, parámetros o conversaciones. El acceso a capacidad pública de un worker tampoco concede acceso a Connections ni vuelve públicos datos o Sessions. Comprueba el resultado final y conserva únicamente evidencia autorizada.

Referencia de componentes de Mapping

Propósito, configuración y límites de los 36 componentes disponibles.

Cómo usar esta referencia

El catálogo contiene 36 componentes: 6 Sources, 17 Transform, 8 Quality y 5 Targets. Los nombres visibles se conservan como aparecen en Dova; los identificadores y las claves JSON no se traducen. Un componente disponible en el editor no demuestra que la conexión esté configurada o que cualquier worker pueda ejecutarlo.

La configuración propuesta en cada ejemplo es parcial y sintética. Completa los campos obligatorios del caso real, las conexiones autorizadas y los bindings del runtime antes de validar. Nunca escribas contraseñas dentro del Mapping.

Sources

Flat File input (`source_csv`)

Lee archivos CSV, TSV o TXT delimitados. La cabecera determina las columnas; `headerMappings` proyecta, renombra y convierte únicamente los campos mapeados. Sin cabecera, `columns` asigna nombres por posición.

Configuración: `format`, `delimiter`, `hasHeader`, `headerRowNumber`, `columns`, `headerMappings`, `encoding`.

`format`: `csv`, `tsv`, `txt`; `encoding`: `utf-8`, `utf-8-bom`.

Ejemplo sintético:

```
{
  "format": "csv",
  "hasHeader": true,
  "delimiter": ",",
}
```

Copiar

Límites: Usa un delimitador de un carácter y UTF-8. Con `hasHeader=true`, `columns` no selecciona campos. Configura la fila de cabecera antes de revisar los datos.

JSON input (`source_json`)

Lee objetos, listas JSON o JSON Lines como registros. `recordPath` localiza los registros; `fieldMappings` selecciona rutas de origen y nombres/tipos de salida. Sin mapeos conserva los campos superiores observados y serializa objetos o listas anidados como texto JSON.

Configuración: `format`, `recordPath`, `fieldMappings`, `encoding`.

`format`: `json`, `jsonl`; `encoding`: `utf-8`, `utf-8-bom`.

Ejemplo sintético:

```
{
  "format": "json",
  "recordPath": "items"
}
```

Copiar

Límites: Las rutas son rutas con puntos, no expresiones JSONPath generales. Un campo mapeado no anulable que falta provoca un error.

Excel input (`source_excel`)

Lee hojas y rangos de un libro Excel. Selecciona hojas, cabecera y filas/columnas; `headerMappings` proyecta, renombra y convierte campos. `includeSheetName` puede añadir la hoja de procedencia.

Configuración: `workbookFormat`, `sheetMode`, `sheetName`, `sheetNames`, `sheetPattern`, `sheetNameRegex`, `includeHiddenSheets`, `hasHeader`, `headerRowNumber`, `dataStartRowNumber`, `startColumn`, `endColumn`, `columns`, `headerMappings`, `skipEmptyRows`, `includeSheetName`, `sheetNameField`, `passwordProtected`.

`workbookFormat`: `auto`, `xlsx`, `xlsm`, `xls`; `sheetMode`: `single`, `list`, `pattern`, `regex`, `all`.

Ejemplo sintético:

```
{
  "sheetMode": "single",
  "sheetName": "Sheet1",
  "hasHeader": true,
  "headerRowNumber": 1,
  "dataStartRowNumber": 2
}
```

Copiar

Límites: Los formatos de lectura son `xlsx`, `xlsm` y `xls`. Las hojas y los rangos deben existir. La contraseña del libro se entrega únicamente para la operación autorizada; no se guarda en la configuración del Mapping.

Database input (`source_database`)

Lee una tabla/vista o una consulta de solo lectura mediante un worker compatible. En modo `table`, `selectedColumns` proyecta campos y `schema` / `objectName` identifican la relación. En modo `query`, las columnas proceden de `sqlQuery`.

Configuración: `connectionId`, `sourceMode`, `schema`, `objectType`, `objectName`, `sqlQuery`, `selectedColumns`, `fetchSize`, `queryTimeoutSeconds`.

`sourceMode`: `table`, `query`; `objectType`: `table`, `view`.

Ejemplo sintético:

```
{
  "sourceMode": "table",
  "objectName": "customers",
  "selectedColumns": [
    "name",
    "amount"
  ]
}
```

Copiar

```
]
}
```

Límites: `selectedColumns` no reescribe SQL arbitrario. El adaptador genérico de streaming admite tabla/vista; las consultas dependen del conector y del runtime. Seleccionar un tipo de conexión no configura sus credenciales.

API input (`source_api`)

Lee registros mediante `GET` con una conexión configurada. `responsePath` localiza los registros y `fieldMappings` proyecta rutas anidadas en campos lógicos tipados. La paginación puede ser `none`, `page`, `offset` o `cursor`.

Configuración: `connectionId`, `method`, `resourcePath`, `responsePath`, `fieldMappings`, `paginationMode`, `pageSize`, `retries`, `timeoutSeconds`.

`method`: `GET`; `paginationMode`: `none`, `page`, `offset`, `cursor`.

Ejemplo sintético:

```
{
  "method": "GET",
  "resourcePath": "/customers",
  "paginationMode": "none"
}
```

Copiar

Límites: La compatibilidad real depende del proveedor. Los conectores SaaS usan sus recursos/entidades; no todos aceptan las mismas opciones del REST genérico.

Object storage input (`source_object`)

Lee CSV o JSONL de almacenamiento de objetos. `resource` identifica el objeto, `selectedColumns` proyecta campos y `delimiter` se aplica a CSV.

Configuración: `connectionId`, `resource`, `format`, `selectedColumns`, `delimiter`.

`format`: `csv`, `jsonl`.

Ejemplo sintético:

```
{
  "resource": "input/customers.csv",
  "format": "csv",
  "selectedColumns": [
    "name",
    "amount"
  ],
  "delimiter": ",",
}
```

Copiar

Límites: Requiere conexión y permisos de lectura compatibles y un esquema tabular estable. No presupone acceso al almacenamiento por mostrar el componente.

Transform

Row filter (filter)

Conserva las filas que cumplen una condición de un campo; mantiene todas sus columnas y valores.

Configuración: `field` , `operator` , `value` .

`operator` : `equals` , `not_equals` , `contains` , `starts_with` , `ends_with` , `gt` , `gte` , `lt` , `lte` , `not_empty` .

Ejemplo sintético:

```
{
  "field": "status",
  "operator": "equals",
  "value": "active"
}
```

Copiar

Límites: `equals` y `not_equals` comparan texto distinguiendo mayúsculas. `contains` , `starts_with` y `ends_with` no las distinguen. Las comparaciones `gt/gte/lt/lte` convierten a número. `not_empty` excluye nulos y texto vacío.

Field selector (select)

Proyecta cada fila y el esquema de salida únicamente sobre los campos existentes indicados, en el orden configurado. Elimina los demás campos y mantiene el número de filas.

Configuración: `fields` .

Ejemplo sintético:

```
{
  "fields": [
    "First_name",
    "Last_name"
  ]
}
```

Copiar

Límites: No transforma valores, renombra ni filtra filas. Debe existir al menos un campo seleccionado. Si aparecen otros campos al final, revisa la versión ejecutada, la rama conectada y los componentes posteriores.

Field mapper (rename)

Renombra los campos incluidos en `mapping` y conserva sus valores, los campos no mapeados y todas las filas.

Configuración: `mapping` .

Ejemplo sintético:

```
{
  "mapping": {
```

Copiar

```
"old_name": "new_name"
}
}
```

Límites: Los nombres de salida deben ser únicos. No reemplaza valores ni elimina los campos no mapeados; añade Field selector cuando necesites proyectar columnas.

Calculated field (`derive`)

Añade o reemplaza un campo con texto literal e interpolación `${field}` o `${param.name}`. Copia las demás columnas y filas.

Configuración: `field`, `template`.

Ejemplo sintético:

```
{
  "field": "label",
  "template": "ID-${id}"
}
```

Copiar

Límites: No ejecuta expresiones: no admite aritmética, `LEFT`, `SUBSTR`, cortes ni capitalización inicial. El resultado es texto limitado a 4.000 caracteres.

Convert type (`convert`)

Convierte un campo a `string`, `integer`, `number`, `boolean`, `date` o `datetime`. Con `outputField` vacío reemplaza el origen; con nombre conserva el origen y escribe el resultado en ese campo.

Configuración: `field`, `targetType`, `outputField`, `onError`, `defaultValue`.

`targetType`: `string`, `integer`, `number`, `boolean`, `date`, `datetime`; `onError`: `fail`, `null`, `keep`, `default`.

Ejemplo sintético:

```
{
  "field": "amount",
  "targetType": "number",
  "outputField": "amount_number",
  "onError": "fail"
}
```

Copiar

Límites: `date` exige `YYYY-MM-DD`; `datetime` usa ISO 8601. No hay formato de fecha personalizado. `onError` admite `fail`, `null`, `keep` o `default`; los números deben ser finitos.

Clean text (`clean`)

Aplica las operaciones, en el orden configurado, únicamente a `fields`. Conserva todas las filas y columnas; los campos no seleccionados mantienen sus valores y tipos, y los nulos siguen siendo nulos.

Configuración: `fields`, `operations`. En `Operations` escribe al menos una letra para ver las coincidencias disponibles y seleccionar una operación.

Operación	Efecto
<code>trim</code>	Quita espacios en blanco de ambos extremos.
<code>ltrim</code>	Quita espacios en blanco del extremo izquierdo.
<code>rtrim</code>	Quita espacios en blanco del extremo derecho.
<code>collapse_whitespace</code>	Reduce grupos de espacios en blanco a un espacio.
<code>lower</code>	Convierte el texto a minúsculas.
<code>upper</code>	Convierte el texto a mayúsculas.
<code>capitalize</code>	Pone el primer carácter en mayúscula y el resto en minúsculas.
<code>title</code>	Capitaliza cada palabra; espacios y puntuación separan palabras.
<code>remove_accents</code>	Elimina marcas de acento.
<code>remove_line_breaks</code>	Sustituye cada grupo de saltos de línea por un espacio, conservando la separación entre palabras.

Para transformar `jORGE` en `Jorge`, configura:

```
{
  "fields": ["first_name"],
  "operations": ["trim", "capitalize"]
}
```

Copiar

El orden importa: `trim` elimina los espacios antes de capitalizar. Los valores seleccionados no nulos se convierten a texto; no uses `capitalize` si necesitas conservar las mayúsculas del resto del nombre.

Replace values (`replace`)

Reemplaza texto literal en un campo y conserva las demás columnas y todas las filas. Es la opción directa para cambiar exactamente `jorge` por `Jorge`.

Configuración: `field`, `find`, `replacement`, `matchMode`, `caseSensitive`.

`matchMode`: `contains`, `exact`, `starts_with`, `ends_with`.

Ejemplo sintético:

```
{
  "field": "name",
  "find": "jorge",
  "replacement": "Jorge",
  "matchMode": "exact",
  "caseSensitive": true
}
```

Copiar

Límites: No usa expresiones regulares. `matchMode` admite `exact`, `contains`, `starts_with`, `ends_with`; `caseSensitive` controla mayúsculas. Ese reemplazo concreto no capitaliza cualquier nombre.

Split field (`split`)

Divide texto mediante un delimitador literal y escribe los fragmentos en `outputFields`. `keepSource` controla conservar el campo original; `trimParts` recorta espacios.

Configuración: `field`, `delimiter`, `outputFields`, `keepSource`, `trimParts`.

Ejemplo sintético:

```
{
  "field": "full_name",
  "delimiter": " ",
  "outputFields": [
    "first_name",
    "last_name"
  ],
  "keepSource": true,
  "trimParts": true
}
```

Copiar

Límites: No extrae caracteres por posición. Divide como máximo una vez menos que el número de salidas: el resto queda en la última. Los fragmentos ausentes son cadenas vacías.

Concatenate fields (`concat`)

Une valores como texto en `outputField` usando `separator`; conserva los campos originales y todas las filas.

Configuración: `fields`, `outputField`, `separator`, `skipEmpty`.

Ejemplo sintético:

```
{
  "fields": [
    "first_name",
    "last_name"
  ],
  "outputField": "full_name",
  "separator": " ",
  "skipEmpty": true
}
```

Copiar

Límites: No suma números. `skipEmpty` omite fragmentos vacíos o con espacios; los nulos se representan como texto vacío.

Join (`join`)

Combina exactamente dos entradas inmediatas por claves de igualdad. Admite `inner`, `left`, `right`, `full`; emite ambos esquemas y añade `rightPrefix` a colisiones del lado derecho.

Configuración: `joinType`, `leftInputId`, `rightInputId`, `leftKey`, `rightKey`, `conditions`, `keyType`, `strategy`, `caseSensitive`, `trimStrings`, `nullsEqual`, `rightPrefix`.

`joinType`: `inner`, `left`, `right`, `full`; `keyType`: `auto`, `string`, `integer`, `number`, `boolean`, `date`, `datetime`; `strategy`: `auto`, `hash`, `sorted`.

Ejemplo sintético:

```
{
  "joinType": "left",
  "leftInputId": "customers",
  "rightInputId": "orders",
  "leftKey": "id",
  "rightKey": "customer_id",
  "keyType": "auto",
  "strategy": "auto",
  "caseSensitive": true,
  "trimStrings": true,
  "nullsEqual": false,
  "rightPrefix": "right_"
}
```

Copiar

Límites: Configura `leftInputId`, `rightInputId` y claves válidas. Las condiciones múltiples se combinan con AND. Coincidencias muchos-a-muchos multiplican filas. Los nulos no coinciden por defecto. Existen límites de expansión y estrategias `auto`, `hash`, `sorted`.

Limit rows (`limit`)

Conserva una ventana contigua de filas después de `offset`, manteniendo columnas y valores.

Configuración: `count`, `offset`.

Ejemplo sintético:

```
{
  "count": 10,
  "offset": 0
}
```

Copiar

Límites: `count` : 1-250.000; `offset` : 0-250.000. No ordena: añade Row sort antes para obtener los primeros N según un campo.

Row sort (`sort`)

Ordena filas por un campo en dirección `asc` o `desc`, conservando todas las columnas y filas.

Configuración: `field`, `direction`.

`direction`: `asc`, `desc`.

Ejemplo sintético:

```
{
  "field": "amount",
```

Copiar

```
"direction": "desc"
}
```

Límites: Los valores convertibles a número se ordenan numéricamente; otros textos sin distinguir mayúsculas. No es ordenación por varias claves. En streaming requiere una etapa de preparación.

Group aggregate (aggregate)

Produce una fila por grupo de `groupBy`, únicamente con las claves del grupo y el campo de métrica. `count` cuenta filas; `sum` y `avg` agregan un campo.

Configuración: `groupBy`, `metric`, `field`, `outputField`.

`metric`: `count`, `sum`, `avg`.

Ejemplo sintético:

```
{
  "groupBy": [
    "region"
  ],
  "metric": "sum",
  "field": "amount",
  "outputField": "total"
}
```

Copiar

Límites: No calcula entre columnas de una misma fila ni ofrece `min` / `max`. `sum` / `avg` ignoran valores no convertibles a números finitos. El resultado de la métrica es texto. `groupBy` vacío agrupa toda la entrada no vacía.

Unpivot (unpivot)

Convierte cada columna de `fields` en una fila: `keyField` recibe el nombre original y `valueField` su valor. Conserva `keepFields`; si está vacío conserva los demás campos de entrada.

Configuración: `fields`, `keepFields`, `keyField`, `valueField`, `dropNulls`.

Ejemplo sintético:

```
{
  "fields": [
    "jan",
    "feb"
  ],
  "keepFields": [
    "id"
  ],
  "keyField": "month",
  "valueField": "amount",
  "dropNulls": false
}
```

Copiar

Límites: Expande filas. Los nombres de salida no pueden sobrescribir columnas existentes. `dropNulls` elimina únicamente nulos: conserva cero, `false` y texto vacío. No realiza pivot.

Run sequence (`sequence`)

Añade una secuencia entera `start + índice de fila × step`, preservando filas, campos y orden.

Configuración: `outputField`, `start`, `step`.

Ejemplo sintético:

```
{
  "outputField": "sequence",
  "start": 1,
  "step": 1
}
```

Copiar

Límites: Se reinicia en cada ejecución: no es un identificador persistente ni globalmente único. `step` no puede ser cero y la salida no puede sobrescribir un campo existente. Se aplican límites de enteros seguros.

Field fingerprint (`hash_fields`)

Añade una huella hexadecimal SHA-256 sobre nombres de campos seleccionados y valores JSON tipados canónicos. Conserva los datos originales.

Configuración: `fields`, `outputField`.

Ejemplo sintético:

```
{
  "fields": [
    "id",
    "name"
  ],
  "outputField": "fingerprint"
}
```

Copiar

Límites: Nulo y texto vacío producen valores distintos. No cifra ni anonimiza. Los campos de entrada deben existir y la salida no puede sobrescribirlos; profundidad JSON máxima 20 y números finitos.

Partition rank (`rank`)

Añade una posición numérica por grupos `partitionBy` y conserva el orden original de las filas. `dense` comparte posiciones consecutivas; `rank` deja huecos; `row_number` asigna posiciones únicas.

Configuración: `partitionBy`, `field`, `direction`, `mode`, `outputField`.

`direction`: `asc`, `desc`; `mode`: `dense`, `rank`, `row_number`.

Ejemplo sintético:

```
{
  "partitionBy": [
```

Copiar

```
"region"
],
"field": "amount",
"direction": "desc",
"mode": "dense",
"outputField": "rank"
}
```

Límites: El campo ordenado debe ser número finito en cada fila; nulo, vacío y booleano fallan. No sobrescribe columnas existentes. `partitionBy` vacío trata toda la entrada como grupo. No ordena la salida final.

Quality

Default value (defaults)

Completa un campo cuando es nulo, vacío o ambos según `when` ; conserva valores presentes, otras columnas y todas las filas. Puede crear un campo ausente.

Configuración: `field` , `value` , `when` .

`when` : `null_or_empty` , `null` , `empty` .

Ejemplo sintético:

```
{
  "field": "name",
  "value": "Unknown",
  "when": "null_or_empty"
}
```

Copiar

Límites: Cero y `false` se conservan. No reemplaza valores literales no vacíos; usa Replace values para esa finalidad.

Deduplicate (dedupe)

Conserva la primera fila de cada combinación de claves y todos sus campos. `fields` vacío compara todas las columnas de entrada.

Configuración: `fields` .

Ejemplo sintético:

```
{
  "fields": [
    "customer_id"
  ]
}
```

Copiar

Límites: Elimina duplicados, pero no combina valores ni elige automáticamente el más reciente. Ordena antes si importa cuál será la primera fila. La comparación utiliza valores JSON tipados.

Quality check (`validate`)

Verifica el porcentaje de valores vacíos de un campo. Falla la ejecución si supera `maxNullPercent` ; de lo contrario deja pasar todas las filas y campos sin cambios.

Configuración: `field` , `maxNullPercent` .

Ejemplo sintético:

```
{
  "field": "name",
  "maxNullPercent": 0
}
```

Copiar

Límites: Nulo y texto con solo espacios cuentan como vacíos; cero y `false` no. El umbral está entre 0 y 100. No completa ni elimina valores.

Required fields (`quality_required`)

Marca una fila inválida si cualquiera de los campos obligatorios es nulo o vacío. Si el porcentaje supera `maxInvalidPercent` , falla la ejecución.

Configuración: `fields` , `maxInvalidPercent` .

Ejemplo sintético:

```
{
  "fields": [
    "id",
    "name"
  ],
  "maxInvalidPercent": 0
}
```

Copiar

Límites: Si pasa el umbral, conserva incluso las filas inválidas. No completa datos ni descarta registros.

Type check (`quality_type`)

Comprueba si los valores se pueden convertir a `expectedType` sin convertirlos. Conserva filas y valores cuando no supera el umbral.

Configuración: `field` , `expectedType` , `allowEmpty` , `maxInvalidPercent` .

`expectedType` : `string` , `integer` , `number` , `boolean` , `date` , `datetime` .

Ejemplo sintético:

```
{
  "field": "amount",
  "expectedType": "number",
  "allowEmpty": false,
  "maxInvalidPercent": 0
}
```

Copiar

Límites: Para transformar usa Convert type. `allowEmpty` controla ausencias. Un texto numérico puede pasar y seguir siendo texto.

Pattern check (`quality_pattern`)

Comprueba coincidencia del campo completo con una expresión regular segura. Conserva los datos si no supera `maxInvalidPercent`.

Configuración: `field`, `pattern`, `allowEmpty`, `maxInvalidPercent`.

Ejemplo sintético:

```
{
  "field": "code",
  "pattern": "[A-Z]{3}",
  "allowEmpty": false,
  "maxInvalidPercent": 0
}
```

Copiar

Límites: Se aplican límites de seguridad a `pattern`. No extrae, reemplaza ni filtra valores. `allowEmpty` controla ausencias por separado.

Range check (`quality_range`)

Comprueba valores numéricos finitos frente a `minimum` y/o `maximum`, con límites inclusivos o exclusivos. Mantiene datos si pasa el umbral.

Configuración: `field`, `minimum`, `maximum`, `inclusive`, `allowEmpty`, `maxInvalidPercent`.

Ejemplo sintético:

```
{
  "field": "age",
  "minimum": "18",
  "maximum": "100",
  "inclusive": true,
  "allowEmpty": false,
  "maxInvalidPercent": 0
}
```

Copiar

Límites: Debe haber al menos un límite. No recorta ni reemplaza valores fuera de rango. `allowEmpty` controla ausencias.

Allowed values (`quality_domain`)

Comprueba pertenencia textual exacta a `allowedValues`, respetando `caseSensitive`. Conserva datos si pasa el umbral.

Configuración: `field`, `allowedValues`, `caseSensitive`, `allowEmpty`, `maxInvalidPercent`.

Ejemplo sintético:

```
{
  "field": "status",
  "allowedValues": [
    "active",
    "inactive"
  ],
  "caseSensitive": true,
  "allowEmpty": false,
  "maxInvalidPercent": 0
}
```

Copiar

Límites: La lista no puede estar vacía. No mapea ni reemplaza valores; usa Replace values para transformarlos. `allowEmpty` controla ausencias.

Targets

Flat File output (`target_csv`)

Escribe filas como CSV o TXT usando las columnas de salida anteriores y su orden. Permite configurar cabecera, delimitador, comillas, nulos, codificación y saltos de línea.

Configuración: `format`, `fileName`, `delimiter`, `includeHeader`, `textQualifier`, `quoteMode`, `escapeCharacter`, `lineEnding`, `encoding`, `nullValue`.

`format`: `csv`, `txt`; `textQualifier`: `double`, `single`, `none`; `quoteMode`: `minimal`, `all`;
`lineEnding`: `lf`, `crlf`; `encoding`: `utf-8`, `utf-8-bom`.

Ejemplo sintético:

```
{
  "fileName": "output.csv",
  "format": "csv",
  "delimiter": ",",
  "includeHeader": true
}
```

Copiar

Límites: No tiene selector `fields` / `columns`. Usa Field selector antes del destino para eliminar campos. `fileName` define el nombre del archivo, no su proyección.

JSON output (`target_json`)

Escribe una lista JSON o JSON Lines usando únicamente las columnas de salida anteriores.

`includeNulls=false` omite campos cuyo valor es nulo.

Configuración: `format`, `fileName`, `pretty`, `includeNulls`, `lineEnding`, `encoding`.

`format`: `json`, `jsonl`; `lineEnding`: `lf`, `crlf`; `encoding`: `utf-8`, `utf-8-bom`.

Ejemplo sintético:

```
{
  "fileName": "output.jsonl",
```

Copiar

```
"format": "jsonl",
"includeNulls": true
}
```

Límites: No permite seleccionar campos arbitrarios: usa Field selector. `pretty` controla el formato de la lista JSON.

Database output (`target_database`)

Escribe campos anteriores en una tabla mediante un worker compatible. `append` añade filas; `replace` y `upsert` dependen de las capacidades del destino. `keyColumns` define claves para upsert.

Configuración: `connectionId`, `schema`, `objectName`, `writeMode`, `batchSize`, `commitIntervalRows`, `keyColumns`, `createIfMissing`, `queryTimeoutSeconds`.

`writeMode`: `append`, `replace`, `upsert`.

Ejemplo sintético:

```
{
  "objectName": "customers",
  "writeMode": "append",
  "batchSize": 5000
}
```

Copiar

Límites: `replace` puede sustituir datos existentes. No presupongas soporte universal de `createIfMissing`, modos de escritura ni tiempos de consulta. Verifica reintentos y claves antes de producción.

API output (`target_api`)

Envía registros mediante `POST`, `PUT` o `PATCH`. Admite `append` / `upsert` con códigos de éxito, lotes, reintentos y tiempo máximo cuando el runtime REST lo soporta.

Configuración: `connectionId`, `method`, `resourcePath`, `writeMode`, `keyColumns`, `uniqueKeysEnforced`, `batchSize`, `successCodes`, `retries`, `timeoutSeconds`.

`method`: `POST`, `PUT`, `PATCH`; `writeMode`: `append`, `upsert`.

Ejemplo sintético:

```
{
  "method": "POST",
  "resourcePath": "/customers",
  "writeMode": "append",
  "batchSize": 100
}
```

Copiar

Límites: Upsert exige exactamente una clave. La idempotencia depende de la API. `uniqueKeysEnforced` describe la garantía del proveedor; no crea una restricción de unicidad.

Object storage output (`target_object`)

Escribe campos anteriores como CSV/JSONL en almacenamiento de objetos mediante sustitución condicional. Exige `idempotencyKey` y exactamente una precondition.

Configuración: `connectionId`, `resource`, `format`, `writeMode`, `delimiter`, `idempotencyKey`, `ifAbsent`, `expectedEtag`, `expectedGeneration`.

`format`: `csv`, `jsonl`; `writeMode`: `replace`.

Ejemplo sintético:

```
{
  "resource": "output/customers.csv",
  "format": "csv",
  "writeMode": "replace",
  "idempotencyKey": "example-run",
  "ifAbsent": true
}
```

Copiar

Límites: Usa `ifAbsent`, `expectedEtag` o `expectedGeneration`, nunca varias. Con ETag/generación configura `ifAbsent=false`. No hay modo append incondicional; verifica conexión y permisos.

Compatibilidad de ejecución

El adaptador genérico de streaming exige una ruta lineal, con un origen y un destino compatibles. Join y los gráficos con varias entradas necesitan una ruta de ejecución que soporte ese grafo; no cambies el algoritmo para silenciar un error de compatibilidad. Row sort, Group aggregate, Partition rank y las verificaciones de calidad pueden preparar datos antes de emitir resultados. Streaming no significa memoria ilimitada ni salida inmediata para cada componente.

Comprobar el resultado

Valida el grafo, guarda y publica deliberadamente. Ejecuta una muestra sintética mediante el runtime autorizado y compara esquema, filas, orden y valores. Si el CSV contiene campos de más tras Field selector, comprueba su configuración guardada, la versión del Mapping usada por la Session y los componentes posteriores: el selector sí elimina campos del esquema y de las filas.

Transformaciones y controles de calidad

Elige el componente por su efecto real sobre valores, columnas y filas.

Elegir por el efecto esperado

Usa Row filter para quitar filas, Field selector para quitar columnas y Field mapper para renombrarlas. Para cambiar el valor exacto `jorge` a `Jorge`, usa Replace values con `field=name`, `find=jorge`, `replacement=Jorge`, `matchMode=exact` y `caseSensitive=true`. No preguntes por otra función cuando esa regla ya está definida.

Para limpiar nombres, usa Clean text con `trim`, `capitalize` o `title`, según el resultado esperado. `capitalize` convierte el primer carácter a mayúscula y el resto a minúsculas; `title` capitaliza cada palabra. Calculated field interpola texto: `LEFT(name,1)` no ejecuta una función.

Contrato de calidad

Quality incluye Default value y Deduplicate, que sí modifican datos. Las otras seis verificaciones son compuertas: Quality check, Required fields, Type check, Pattern check, Range check y Allowed values. Si el porcentaje inválido supera el umbral, falla toda la ejecución. Si no lo supera, pasan todas las filas, incluidas las inválidas. No crean una salida de rechazados. Con umbral cero, una sola fila inválida en una entrada no vacía provoca fallo.

Transformaciones y calidad

Row filter (`filter`)

Conserva las filas que cumplen una condición de un campo; mantiene todas sus columnas y valores.

Configuración: `field`, `operator`, `value`.

`operator`: `equals`, `not_equals`, `contains`, `starts_with`, `ends_with`, `gt`, `gte`, `lt`, `lte`, `not_empty`.

Ejemplo sintético:

```
{
  "field": "status",
  "operator": "equals",
  "value": "active"
}
```

Copiar

Límites: `equals` y `not_equals` comparan texto distinguiendo mayúsculas. `contains`, `starts_with` y `ends_with` no las distinguen. Las comparaciones `gt/gte/lt/lte` convierten a número. `not_empty` excluye nulos y texto vacío.

Field selector (`select`)

Proyecta cada fila y el esquema de salida únicamente sobre los campos existentes indicados, en el orden configurado. Elimina los demás campos y mantiene el número de filas.

Configuración: `fields` .

Ejemplo sintético:

```
{
  "fields": [
    "First_name",
    "Last_name"
  ]
}
```

Copiar

Límites: No transforma valores, renombra ni filtra filas. Debe existir al menos un campo seleccionado. Si aparecen otros campos al final, revisa la versión ejecutada, la rama conectada y los componentes posteriores.

Field mapper (`rename`)

Renombra los campos incluidos en `mapping` y conserva sus valores, los campos no mapeados y todas las filas.

Configuración: `mapping` .

Ejemplo sintético:

```
{
  "mapping": {
    "old_name": "new_name"
  }
}
```

Copiar

Límites: Los nombres de salida deben ser únicos. No reemplaza valores ni elimina los campos no mapeados; añade Field selector cuando necesites proyectar columnas.

Calculated field (`derive`)

Añade o reemplaza un campo con texto literal e interpolación `${field}` o `${param.name}` . Copia las demás columnas y filas.

Configuración: `field` , `template` .

Ejemplo sintético:

```
{
  "field": "label",
  "template": "ID-${id}"
}
```

Copiar

Límites: No ejecuta expresiones: no admite aritmética, `LEFT` , `SUBSTR` , cortes ni capitalización inicial. El resultado es texto limitado a 4.000 caracteres.

Convert type (convert)

Convierte un campo a `string`, `integer`, `number`, `boolean`, `date` o `datetime`. Con `outputField` vacío reemplaza el origen; con nombre conserva el origen y escribe el resultado en ese campo.

Configuración: `field`, `targetType`, `outputField`, `onError`, `defaultValue`.

`targetType`: `string`, `integer`, `number`, `boolean`, `date`, `datetime`; `onError`: `fail`, `null`, `keep`, `default`.

Ejemplo sintético:

```
{
  "field": "amount",
  "targetType": "number",
  "outputField": "amount_number",
  "onError": "fail"
}
```

Copiar

Límites: `date` exige `YYYY-MM-DD`; `datetime` usa ISO 8601. No hay formato de fecha personalizado. `onError` admite `fail`, `null`, `keep` o `default`; los números deben ser finitos.

Clean text (clean)

Aplica las operaciones, en el orden configurado, únicamente a `fields`. Conserva todas las filas y columnas; los campos no seleccionados mantienen sus valores y tipos, y los nulos siguen siendo nulos.

Configuración: `fields`, `operations`. En Operations escribe al menos una letra para ver las coincidencias disponibles y seleccionar una operación.

Operación	Efecto
<code>trim</code>	Quita espacios en blanco de ambos extremos.
<code>ltrim</code>	Quita espacios en blanco del extremo izquierdo.
<code>rtrim</code>	Quita espacios en blanco del extremo derecho.
<code>collapse_whitespace</code>	Reduce grupos de espacios en blanco a un espacio.
<code>lower</code>	Convierte el texto a minúsculas.
<code>upper</code>	Convierte el texto a mayúsculas.
<code>capitalize</code>	Pone el primer carácter en mayúscula y el resto en minúsculas.
<code>title</code>	Capitaliza cada palabra; espacios y puntuación separan palabras.
<code>remove_accents</code>	Elimina marcas de acento.
<code>remove_line_breaks</code>	Sustituye cada grupo de saltos de línea por un espacio, conservando la separación entre palabras.

Para transformar `jORGE` en `Jorge`, configura:

```
{
  "fields": ["first_name"],
  "operations": ["trim", "capitalize"]
}
```

Copiar

El orden importa: `trim` elimina los espacios antes de capitalizar. Los valores seleccionados no nulos se convierten a texto; no uses `capitalize` si necesitas conservar las mayúsculas del resto del nombre.

Replace values (`replace`)

Reemplaza texto literal en un campo y conserva las demás columnas y todas las filas. Es la opción directa para cambiar exactamente `jorge` por `Jorge`.

Configuración: `field`, `find`, `replacement`, `matchMode`, `caseSensitive`.

`matchMode`: `contains`, `exact`, `starts_with`, `ends_with`.

Ejemplo sintético:

```
{
  "field": "name",
  "find": "jorge",
  "replacement": "Jorge",
  "matchMode": "exact",
  "caseSensitive": true
}
```

Copiar

Límites: No usa expresiones regulares. `matchMode` admite `exact`, `contains`, `starts_with`, `ends_with`; `caseSensitive` controla mayúsculas. Ese reemplazo concreto no capitaliza cualquier nombre.

Split field (`split`)

Divide texto mediante un delimitador literal y escribe los fragmentos en `outputFields`. `keepSource` controla conservar el campo original; `trimParts` recorta espacios.

Configuración: `field`, `delimiter`, `outputFields`, `keepSource`, `trimParts`.

Ejemplo sintético:

```
{
  "field": "full_name",
  "delimiter": " ",
  "outputFields": [
    "first_name",
    "last_name"
  ],
  "keepSource": true,
  "trimParts": true
}
```

Copiar

Límites: No extrae caracteres por posición. Divide como máximo una vez menos que el número de salidas: el resto queda en la última. Los fragmentos ausentes son cadenas vacías.

Concatenate fields (concat)

Une valores como texto en `outputField` usando `separator` ; conserva los campos originales y todas las filas.

Configuración: `fields` , `outputField` , `separator` , `skipEmpty` .

Ejemplo sintético:

```
{
  "fields": [
    "first_name",
    "last_name"
  ],
  "outputField": "full_name",
  "separator": " ",
  "skipEmpty": true
}
```

Copiar

Límites: No suma números. `skipEmpty` omite fragmentos vacíos o con espacios; los nulos se representan como texto vacío.

Join (join)

Combina exactamente dos entradas inmediatas por claves de igualdad. Admite `inner` , `left` , `right` , `full` ; emite ambos esquemas y añade `rightPrefix` a colisiones del lado derecho.

Configuración: `joinType` , `leftInputId` , `rightInputId` , `leftKey` , `rightKey` , `conditions` , `keyType` , `strategy` , `caseSensitive` , `trimStrings` , `nullsEqual` , `rightPrefix` .

`joinType` : `inner` , `left` , `right` , `full` ; `keyType` : `auto` , `string` , `integer` , `number` , `boolean` , `date` , `datetime` ; `strategy` : `auto` , `hash` , `sorted` .

Ejemplo sintético:

```
{
  "joinType": "left",
  "leftInputId": "customers",
  "rightInputId": "orders",
  "leftKey": "id",
  "rightKey": "customer_id",
  "keyType": "auto",
  "strategy": "auto",
  "caseSensitive": true,
  "trimStrings": true,
  "nullsEqual": false,
  "rightPrefix": "right_"
}
```

Copiar

Límites: Configura `leftInputId` , `rightInputId` y claves válidas. Las condiciones múltiples se combinan con AND. Coincidencias muchos-a-muchos multiplican filas. Los nulos no coinciden por defecto. Existen límites de expansión y estrategias `auto` , `hash` , `sorted` .

Default value (defaults)

Completa un campo cuando es nulo, vacío o ambos según `when` ; conserva valores presentes, otras columnas y todas las filas. Puede crear un campo ausente.

Configuración: `field` , `value` , `when` .

`when` : `null_or_empty` , `null` , `empty` .

Ejemplo sintético:

```
{
  "field": "name",
  "value": "Unknown",
  "when": "null_or_empty"
}
```

Copiar

Límites: Cero y `false` se conservan. No reemplaza valores literales no vacíos; usa Replace values para esa finalidad.

Limit rows (limit)

Conserva una ventana contigua de filas después de `offset` , manteniendo columnas y valores.

Configuración: `count` , `offset` .

Ejemplo sintético:

```
{
  "count": 10,
  "offset": 0
}
```

Copiar

Límites: `count` : 1-250.000; `offset` : 0-250.000. No ordena: añade Row sort antes para obtener los primeros N según un campo.

Deduplicate (dedupe)

Conserva la primera fila de cada combinación de claves y todos sus campos. `fields` vacío compara todas las columnas de entrada.

Configuración: `fields` .

Ejemplo sintético:

```
{
  "fields": [
    "customer_id"
  ]
}
```

Copiar

Límites: Elimina duplicados, pero no combina valores ni elige automáticamente el más reciente. Ordena antes si importa cuál será la primera fila. La comparación utiliza valores JSON tipados.

Row sort (sort)

Ordena filas por un campo en dirección `asc` o `desc`, conservando todas las columnas y filas.

Configuración: `field`, `direction`.

`direction`: `asc`, `desc`.

Ejemplo sintético:

```
{
  "field": "amount",
  "direction": "desc"
}
```

Copiar

Límites: Los valores convertibles a número se ordenan numéricamente; otros textos sin distinguir mayúsculas. No es ordenación por varias claves. En streaming requiere una etapa de preparación.

Group aggregate (aggregate)

Produce una fila por grupo de `groupBy`, únicamente con las claves del grupo y el campo de métrica. `count` cuenta filas; `sum` y `avg` agregan un campo.

Configuración: `groupBy`, `metric`, `field`, `outputField`.

`metric`: `count`, `sum`, `avg`.

Ejemplo sintético:

```
{
  "groupBy": [
    "region"
  ],
  "metric": "sum",
  "field": "amount",
  "outputField": "total"
}
```

Copiar

Límites: No calcula entre columnas de una misma fila ni ofrece `min` / `max`. `sum` / `avg` ignoran valores no convertibles a números finitos. El resultado de la métrica es texto. `groupBy` vacío agrupa toda la entrada no vacía.

Unpivot (unpivot)

Convierte cada columna de `fields` en una fila: `keyField` recibe el nombre original y `valueField` su valor. Conserva `keepFields`; si está vacío conserva los demás campos de entrada.

Configuración: `fields`, `keepFields`, `keyField`, `valueField`, `dropNulls`.

Ejemplo sintético:

```
{
  "fields": [
    "jan",
```

Copiar

```
    "feb"
  ],
  "keepFields": [
    "id"
  ],
  "keyField": "month",
  "valueField": "amount",
  "dropNulls": false
}
```

Límites: Expande filas. Los nombres de salida no pueden sobrescribir columnas existentes. `dropNulls` elimina únicamente nulos: conserva cero, `false` y texto vacío. No realiza pivot.

Run sequence (`sequence`)

Añade una secuencia entera `start + índice de fila × step`, preservando filas, campos y orden.

Configuración: `outputField`, `start`, `step`.

Ejemplo sintético:

```
{
  "outputField": "sequence",
  "start": 1,
  "step": 1
}
```

Copiar

Límites: Se reinicia en cada ejecución: no es un identificador persistente ni globalmente único. `step` no puede ser cero y la salida no puede sobrescribir un campo existente. Se aplican límites de enteros seguros.

Field fingerprint (`hash_fields`)

Añade una huella hexadecimal SHA-256 sobre nombres de campos seleccionados y valores JSON tipados canónicos. Conserva los datos originales.

Configuración: `fields`, `outputField`.

Ejemplo sintético:

```
{
  "fields": [
    "id",
    "name"
  ],
  "outputField": "fingerprint"
}
```

Copiar

Límites: Nulo y texto vacío producen valores distintos. No cifra ni anonimiza. Los campos de entrada deben existir y la salida no puede sobrescribirlos; profundidad JSON máxima 20 y números finitos.

Partition rank (rank)

Añade una posición numérica por grupos `partitionBy` y conserva el orden original de las filas. `dense` comparte posiciones consecutivas; `rank` deja huecos; `row_number` asigna posiciones únicas.

Configuración: `partitionBy` , `field` , `direction` , `mode` , `outputField` .

`direction` : `asc` , `desc` ; `mode` : `dense` , `rank` , `row_number` .

Ejemplo sintético:

```
{
  "partitionBy": [
    "region"
  ],
  "field": "amount",
  "direction": "desc",
  "mode": "dense",
  "outputField": "rank"
}
```

Copiar

Límites: El campo ordenado debe ser número finito en cada fila; nulo, vacío y booleano fallan. No sobrescribe columnas existentes. `partitionBy` vacío trata toda la entrada como grupo. No ordena la salida final.

Quality check (validate)

Verifica el porcentaje de valores vacíos de un campo. Falla la ejecución si supera `maxNullPercent` ; de lo contrario deja pasar todas las filas y campos sin cambios.

Configuración: `field` , `maxNullPercent` .

Ejemplo sintético:

```
{
  "field": "name",
  "maxNullPercent": 0
}
```

Copiar

Límites: Nulo y texto con solo espacios cuentan como vacíos; cero y `false` no. El umbral está entre 0 y 100. No completa ni elimina valores.

Required fields (quality_required)

Marca una fila inválida si cualquiera de los campos obligatorios es nulo o vacío. Si el porcentaje supera `maxInvalidPercent` , falla la ejecución.

Configuración: `fields` , `maxInvalidPercent` .

Ejemplo sintético:

```
{
  "fields": [
    "id",
    "name"
  ]
}
```

Copiar

```
],  
  "maxInvalidPercent": 0  
}
```

Límites: Si pasa el umbral, conserva incluso las filas inválidas. No completa datos ni descarta registros.

Type check (`quality_type`)

Comprueba si los valores se pueden convertir a `expectedType` sin convertirlos. Conserva filas y valores cuando no supera el umbral.

Configuración: `field`, `expectedType`, `allowEmpty`, `maxInvalidPercent`.

`expectedType`: `string`, `integer`, `number`, `boolean`, `date`, `datetime`.

Ejemplo sintético:

```
{  
  "field": "amount",  
  "expectedType": "number",  
  "allowEmpty": false,  
  "maxInvalidPercent": 0  
}
```

Copiar

Límites: Para transformar usa `Convert type`. `allowEmpty` controla ausencias. Un texto numérico puede pasar y seguir siendo texto.

Pattern check (`quality_pattern`)

Comprueba coincidencia del campo completo con una expresión regular segura. Conserva los datos si no supera `maxInvalidPercent`.

Configuración: `field`, `pattern`, `allowEmpty`, `maxInvalidPercent`.

Ejemplo sintético:

```
{  
  "field": "code",  
  "pattern": "[A-Z]{3}",  
  "allowEmpty": false,  
  "maxInvalidPercent": 0  
}
```

Copiar

Límites: Se aplican límites de seguridad a `pattern`. No extrae, reemplaza ni filtra valores. `allowEmpty` controla ausencias por separado.

Range check (`quality_range`)

Comprueba valores numéricos finitos frente a `minimum` y/o `maximum`, con límites inclusivos o exclusivos. Mantiene datos si pasa el umbral.

Configuración: `field`, `minimum`, `maximum`, `inclusive`, `allowEmpty`, `maxInvalidPercent`.

Ejemplo sintético:

```
{
  "field": "age",
  "minimum": "18",
  "maximum": "100",
  "inclusive": true,
  "allowEmpty": false,
  "maxInvalidPercent": 0
}
```

Copiar

Límites: Debe haber al menos un límite. No recorta ni reemplaza valores fuera de rango. `allowEmpty` controla ausencias.

Allowed values (`quality_domain`)

Comprueba pertenencia textual exacta a `allowedValues`, respetando `caseSensitive`. Conserva datos si pasa el umbral.

Configuración: `field`, `allowedValues`, `caseSensitive`, `allowEmpty`, `maxInvalidPercent`.

Ejemplo sintético:

```
{
  "field": "status",
  "allowedValues": [
    "active",
    "inactive"
  ],
  "caseSensitive": true,
  "allowEmpty": false,
  "maxInvalidPercent": 0
}
```

Copiar

Límites: La lista no puede estar vacía. No mapea ni reemplaza valores; usa `Replace values` para transformarlos. `allowEmpty` controla ausencias.

Compatibilidad de ejecución

El adaptador genérico de streaming exige una ruta lineal, con un origen y un destino compatibles. Join y los gráficos con varias entradas necesitan una ruta de ejecución que soporte ese grafo; no cambies el algoritmo para silenciar un error de compatibilidad. Row sort, Group aggregate, Partition rank y las verificaciones de calidad pueden preparar datos antes de emitir resultados. Streaming no significa memoria ilimitada ni salida inmediata para cada componente.

Comprobar el resultado

Valida el grafo, guarda y publica deliberadamente. Ejecuta una muestra sintética mediante el runtime autorizado y compara esquema, filas, orden y valores. Si el CSV contiene campos de más tras Field selector, comprueba su configuración guardada, la versión del Mapping usada por la Session y los componentes posteriores: el selector sí elimina campos del esquema y de las filas.

Parámetros, plantillas y condiciones

Distingue valores tipados, interpolación de texto y decisiones del Workflow.

Antes de comenzar

Los parámetros tipados, las plantillas de Calculated field y las condiciones JSON del Workflow son mecanismos distintos. Ninguno convierte texto arbitrario en código ejecutable. Define tipos y alcance antes de introducir overrides. Usa valores sintéticos y nunca pongas secretos en parámetros de texto.

Los tipos admitidos son `string`, `integer`, `number`, `boolean`, `datetime`, `json` y `secret`. Un parámetro sensible debe usar `secret`, con una referencia protegida válida, no un valor claro marcado como sensible.

Procedimiento

1. Declara un parámetro con nombre único, tipo y restricciones. Se admiten hasta 100 definiciones. El nombre comienza por letra y tiene hasta 64 caracteres; permite letras, números, puntos, guiones y guiones bajos. `sys.` y `dova.` están reservados.
2. Revisa `required`, `overridable`, `scopes`, valor por defecto y restricciones. Booleano requiere JSON `true` / `false`; no uses texto equivalente. Los números deben ser finitos.
3. Comprueba la resolución por capas: valor por defecto → global → environment → worker → task. Una capa posterior permitida reemplaza la anterior; no anula restricciones de alcance.
4. Para texto derivado usa `${name}` y `${param.region}`. El resultado es texto de hasta 4.000 caracteres; un identificador reconocido ausente se sustituye por vacío.
5. Para decidir una rama, usa una condición JSON con referencias autorizadas, no una fórmula SQL.

Ejemplo sintético

Plantilla de Calculated field:

```
{
  "field": "label",
  "template": "${name}-${param.region}"
}
```

Copiar

Con `name=Ana` y parámetro `region=NORTH`, produce `Ana-NORTH`. No realiza suma ni extracción de primera letra.

Condición True de una Decision:

```
{
  "ref": "parameters.RUN_MODE",
  "op": "eq",
  "value": "full"
}
```

Copiar

Declara `RUN_MODE` antes de usarlo. Ejecuta intentos separados con `full` y `sample`; revisa la rama False complementaria que muestra la interfaz.

Comprobar el resultado

Las referencias de condición admiten `parameters.name`, `variables.name`, `tasks.id.status` y `tasks.id.outputs.name`. Usa el identificador real de tarea y salidas documentadas; no inventes un contador universal. Operadores: `eq`, `ne`, `gt`, `gte`, `lt`, `lte`, `in`, `not_in`, `contains`, `exists`, `truthy`. Los grupos `all` / `any` reúnen de 1 a 20 expresiones; `not` niega una expresión.

Confirma valores y procedencia en el preview autorizado. Luego comprueba la rama tomada en Monitor. Las comparaciones de tipos incompatibles no prometen conversiones automáticas.

Resolver problemas

Si un override no cambia el resultado, revisa scope, Overridable y la capa de procedencia. Si aparece texto vacío, revisa el nombre del campo o parámetro de la plantilla. Un nombre con puntos no equivale a navegación general por objetos anidados. Los árboles de condiciones rechazan profundidad superior a cinco, más de 40 nodos y valores de comparación de más de 4.096 bytes. Una referencia ausente hace falsas las comparaciones ordinarias. Corrige el contrato, no debilites restricciones para aceptar un valor no autorizado.

Variables operativas de Runtime

Consulta Sessions → Session → Runtime → Available runtime variables para ver tokens disponibles, significado, ejemplo y último valor resuelto autorizado. `%RUN_TIMESTAMP%` sirve para el nombre de salida cuando Target filename timestamp está efectivo como Enabled. El ejemplo del formulario es una vista previa; el valor real se asocia al identificador y fecha del intento. Estas variables operativas no sustituyen `${field}` ni permiten código arbitrario. Los parámetros sensibles permanecen ocultos.

Permisos y acciones

Comprende por qué ver un módulo, editar un objeto y ejecutarlo requieren comprobaciones distintas.

Contexto de autorización

La identidad, su membresía en el tenant y el rol seleccionado determinan el contexto de trabajo. La ubicación de Repository no reemplaza esas comprobaciones. Una cuenta activa puede tener una membresía inactiva; un rol visible no concede automáticamente todos los permisos. El servidor comprueba la autoridad actual para cada acción y puede rechazar una operación que antes estaba disponible.

No confundas acceso a Administrator con la identidad maestra única de la plataforma. La exención de cuota de DovaAI del superusuario requiere la identidad persistida, membresía activa y rol de plataforma seleccionado. No se concede a todos los administradores ni por coincidir un correo. Los controles de proveedor y autorización siguen vigentes.

Revisar una acción

Objetivo	Qué revisar
Consultar	Tenant, membresía y permiso de lectura del módulo u objeto
Editar	Permiso de cambio, versión actual y estado editable
Publicar	Borrador guardado, dependencias y permiso de publicación
Ejecutar	Permiso de ejecución, plan, runtime, capacidad y conexiones
Eliminar o purgar	Alcance, referencias, bloqueos y confirmación actual

La autorización no garantiza recursos disponibles. Un plan válido tampoco concede permisos ausentes. Ver una Session no significa que pueda ejecutarse, y publicar un Workflow no activa por sí solo un horario.

Procedimiento para acceso denegado

1. Consulta tu identidad y tenant desde la cuenta actual.
2. Confirma el rol seleccionado y la membresía activa.
3. Identifica la acción concreta: lectura, edición, publicación, ejecución o eliminación.
4. Pide al administrador que revise esa autorización y su alcance, sin ampliar privilegios de forma indiscriminada.
5. Actualiza la vista tras un cambio aceptado y vuelve a comprobar el recurso.

Para un ejercicio sintético, una persona que solo consulta Monitor debe revisar intentos y resultados autorizados, no recibir permisos para purgar Repository. Delega la acción necesaria, no un conjunto mayor para ocultar un fallo de configuración.

Límites entre tenants y ejecución

Un worker público comparte capacidad elegible; no comparte objetos, conversaciones ni datos de otros tenants. Las Connections mantienen sus propias restricciones. DovaAI consulta y recomienda dentro del alcance autorizado. Nunca crea, modifica, publica, ejecuta ni elimina artefactos, aunque lo solicites o confirmes; una instrucción del chat no concede permisos nuevos.

Si cambian identidad, tenant o permisos durante una operación, el contexto debe volver a comprobarse. No reutilices una confirmación antigua ni interpretes un botón oculto como único control de seguridad. La respuesta del servidor y el estado aceptado del objeto determinan el resultado. Conserva evidencia de auditoría autorizada sin copiar credenciales o información de otro tenant.

Contexto de entorno

Antes de utilizar este módulo confirma Tenant, Domain y Environment en el menú de usuario. El entorno inicial es Development. Las capacidades del módulo, el Domain y el entorno se aplican juntas; Production queda de solo lectura sin autorización explícita. Cambiar el selector no mueve artefactos ni publica o ejecuta sus datos. Consulta [entornos y recuperación](#) y [promoción entre entornos](#).

Permisos de entorno

La autorización combina permisos del módulo, acceso al Domain y capacidades del entorno. El catálogo actual contiene 23 permisos de entorno: `etl.environments.view`, `etl.environments.{development, test, stage, production}.{view, edit, publish, run, runtime.manage}`, `etl.environments.promote` y `etl.environments.runtime.manage`. Los permisos globales de lectura y Runtime conservan su alcance explícito; un permiso de un entorno no concede acceso a todos los demás. Administrar un módulo no sustituye estas capacidades.

ETL Developer puede trabajar en Development, Test y Stage y no tiene acceso a Production por defecto. Release Manager puede revisar los cuatro entornos y publicar en Test, Stage y Production para promover versiones autorizadas; no obtiene edición ni ejecución de Production. La promoción conserva el preview, las dependencias publicadas exactas y la aprobación explícita de Production. Los roles personalizados permiten asignar las capacidades necesarias por entorno. La identidad del usuario no se duplica por ambiente; verifica el rol activo, la membresía y sus permisos efectivos.

Los editores de Mapping, Session y Workflow de Production permanecen de solo lectura: las versiones aceptadas llegan mediante promoción. Configurar Connections existentes o nuevas de Production es una capacidad separada que exige `etl.connections.manage` y `etl.environments.production.edit`; no habilita edición de esos tres tipos ni modifica Connections legacy de solo lectura.

Dova 23.0.2: capacidades y comprobación

Consulta los cambios funcionales y verifica el comportamiento en tu entorno autorizado.

Alcance de esta edición

Este manual describe Dova 23.0.2. Confirma la versión activa de tu instalación antes de seguir un procedimiento. La ruta histórica de esta referencia permanece estable para conservar enlaces anteriores; el título y la edición identifican su contenido actual.

Monitor y diagnóstico operativo

El log compacto reúne hora, nivel, componente y mensaje, con filtros, seguimiento y pausa de presentación. Las tarjetas y el progreso consultan los contadores del intento en curso; etapas, filas, bytes y tasa dependen de la evidencia que reporte el worker. Un total desconocido no se presenta como porcentaje terminado. La lista actualiza sus celdas conservando filtros, selección y desplazamiento.

El Gantt usa filas compactas y distingue Workflow, Session y Mapping mediante tipo y jerarquía. La duración de una Session está dentro de la orquestación; no es tiempo adicional que deba sumarse a la barra del Workflow. Abre el intento para ver las versiones realmente ejecutadas.

Los errores seguros del worker conservan causa clasificada, etapa y límites disponibles. Un archivo mayor que el máximo del perfil y un destino protegido existente son problemas distintos. El disco libre físico no amplía el límite de entrada ni la memoria temporal del ejecutor. Los intentos anteriores sin excepción original no adquieren evidencia retroactivamente.

Session y dependencias

Sessions → Runtime muestra variables disponibles, significado, ejemplo y último valor real asociado a un run. `%RUN_TIMESTAMP%` participa en nombres de salida cuando el timestamp está efectivamente activado. Inherit con preferencia global desactivada no aplica el token, aunque aparezca en el template. El preview ilustra la configuración y no prueba el nombre de una salida real.

La cadena compacta de Workflow conserva nombre, tipo, versión usada y disponibilidad de una versión nueva. Update versions cambia el borrador; Save draft → Publish adopta la revisión para ejecuciones posteriores. Reload status y abrir un objeto solo permiten inspeccionar.

DovaAI de solo lectura

DovaAI sigue el idioma del usuario y recomienda el componente, los campos y los valores adecuados con pasos breves. Para cambiar exactamente jorge por Jorge recomienda Replace values con coincidencia exacta; Field selector conserva únicamente los campos elegidos. Usa el manual público y evidencia autorizada de Workflow y Session para distinguir causas confirmadas de hipótesis.

El asistente nunca crea, modifica, configura, publica, ejecuta ni elimina objetos, incluso con confirmación. Tú realizas los cambios mediante los controles normales. Esta política sustituye la creación asistida de artefactos de versiones anteriores.

Promoción entre ambientes

Configuration → Promotions sigue Development → Test → Stage → Production. Promueve un paquete inmutable con Workflow, Sessions y Mappings publicados y sus versiones exactas, aplica bindings del destino y crea artefactos separados sin alterar los originales. Revisa diff, worker, plan, DPU y concurrencia antes de confirmar. Production requiere aprobación explícita y comentario.

El historial permite exportar un manifiesto con procedencia Git y restaurar definiciones históricas mediante nuevas versiones. No contiene credenciales ni referencias secretas utilizables. No ejecuta automáticamente el Workflow, no activa calendarios, no deshace datos externos ni inicia GitHub Actions. Development, Test y Stage consumen la capacidad Development; Production utiliza la capacidad Production del plan efectivo.

Aceptación

1. Confirma versión, tenant, permisos y ubicación.
2. Inspecciona perfil, variables y dependencias antes de cambiar configuración.
3. Usa un laboratorio y datos sintéticos para probar una salida identificable.
4. Compara log, contadores, versiones, esquema y archivo final.
5. Revisa y confirma tú cada promoción o restauración; conserva su evidencia.

Cambios de 23.0.1

1. Log con Timestamp y Message compactos y detalle estructurado expandible.
2. TRACE opcional por Workflow y Session, con herencia y generación desactivada por defecto.
3. Tipografía y tooltips del Gantt alineados con Dova.
4. List compacta con distintivos W/S, jerarquía y columnas alineadas.
5. Actualización de filas y nuevas ejecuciones sin parpadeo, conservando el contexto de lectura.
6. Supr elimina la selección cuando el foco está en el canvas de Mappings y no en un editor de texto.
7. Clean text sugiere operaciones desde la primera letra y añade `capitalize`, `title`, `ltrim`, `rtrim` y `remove_line_breaks`.
8. Sessions → General ordena Domain → Folder → Mapping y carga sus opciones desde el inicio.
9. DovaAI recomienda con componente directo, flujo breve y pasos con propiedades y valores precisos; mantiene su función de solo lectura.

Cambios de 23.0.2

1. Monitor → List y Gantt muestran los distintivos e iconos WF y SES usados en las pestañas del espacio de trabajo.
2. Full Screen entra y sale del canvas de Mappings y Workflows en el mismo editor, conservando selección, zoom, posición y acceso a DovaAI.
3. Las acciones de DOVA_DATA_GRID muestran progreso y bloquean otras acciones de la misma tabla hasta terminar o fallar.
4. Los filtros usan estados aplicables al módulo; Mappings ofrece Draft y Published.
5. Mappings añade Used by (Sessions), Location, Last published y Description, configurables desde Columns.
6. Las tablas de Administrator conservan menús y posición al actualizarse. Cloud Workers ofrece controles nativos según proveedor y una preparación ordenada para resolver dependencias y desaprovechar primero en la nube y después en Dova.

7. Administrator reutiliza lecturas frecuentes en RAM con caducidad, límites e invalidación, respetando tenant, rol y permisos.
8. Overview actualiza sus gráficos sin desmontarlos, consulta cambios cada cinco segundos y señala la última lectura confirmada cuando falla la actualización.
9. Overview incluye permanentemente `vm-cloudata-pe` como Base server de Dova para administradores de plataforma, separado de los workers de ejecución y visible aunque falten métricas.

Esta edición conserva el worker Linux 1.3.14. Las acciones de VM respetan las capacidades reales del proveedor: Pause y Resume nativos están disponibles para GCP; Azure ofrece Start, Stop y Deprovision; AWS no dispone de gestión del ciclo de vida de VM en esta edición. Los estados y los resultados se verifican con su fuente; refrescar una pantalla no equivale a completar una operación.

Entornos: separar Development, Test y Production

Selecciona Tenant, Domain y Environment y conserva el trabajo de cada contexto.

Elegir el contexto

El resumen Tenant, Domain y Environment aparece en la barra inferior de estado operativo de Dova y Administrator. Haz clic en ese resumen para abrir el menú de usuario con los tres selectores. Conserva el distintivo y el color del entorno; no ocupa una franja superior ni duplica los campos.

El menú de usuario reúne Tenant, Domain y Environment. El contexto inicial es Development. Los entornos canónicos son Development, Test, Stage y Production; Stage aparece según la configuración y los permisos. Una ubicación Domain/Folder organiza la propiedad y el acceso, mientras el entorno separa las definiciones y sus ejecuciones. Compartir un Domain no concede permisos sobre otro entorno.

1. Confirma el Tenant autorizado.
2. Selecciona el Domain desde el menú de usuario.
3. Elige Environment y espera a que termine la carga del contexto.
4. Revisa el distintivo DEV, TEST, STAGE o PROD antes de editar o ejecutar.

Mappings, Sessions, Workflows, Connections y Monitor consultan el contexto seleccionado. Domain del editor identifica al propietario y queda bloqueado; Folder debe pertenecer a ese propietario. Para cambiar la ubicación utiliza la operación Move explícita y sus permisos.

Guardar, publicar y ejecutar

Save draft in DEV guarda el borrador en Development. Publish in DEV publica esa versión en el mismo entorno. Promote to Test abre una revisión separada del destino. Las etiquetas reflejan el entorno actual; guardar o publicar no traslada una definición a otro entorno ni ejecuta sus datos.

La autorización combina el permiso del módulo, el acceso al Domain y las capacidades del entorno y del artefacto. Production se presenta de solo lectura cuando no existe una autorización explícita. Un operador puede tener permiso de ejecución sin permiso de edición. Los controles no otorgan permisos por el nombre del entorno.

Cambiar sin mezclar borradores

Las pestañas, filtros, selección, borradores y posiciones se conservan por Tenant, Domain y Environment. Un cambio cancela lecturas anteriores y descarta respuestas obsoletas. Mientras una escritura está en curso no se puede cambiar de contexto. La restauración comprueba el artefacto y su autorización antes de recuperar un editor con cambios.

Cambiar de entorno conserva el trabajo local sin guardarlo ni publicarlo en el servidor. Los campos de contraseña y los archivos seleccionados no se trasladan a otro contexto. Guarda o cierra una Session con cambios antes de cambiar de Tenant. Si el servidor no confirma el contexto, la plataforma queda de solo lectura y ofrece Retry context.

Definiciones anteriores y recuperación

La actualización no convierte automáticamente todos los artefactos antiguos en Development. Las dependencias compartidas y las versiones publicadas pueden conservarse mediante pertenencia verificada y referencias exactas. Un artefacto ambiguo o compartido puede aparecer de solo lectura; los borradores locales anteriores permanecen pendientes de recuperación.

Para crear una copia editable usa Fork to Development. La revisión incluye las dependencias indirectas de Mapping, Session o Workflow. Selecciona para cada Connection una Connection compatible que ya pertenezca a Development. Dova no copia credenciales. El servidor vuelve a comprobar la revisión y rechaza cambios de origen o destino posteriores al preview. La copia se crea como borrador; Open in Development es una acción posterior explícita. Revisa y publica la copia antes de ejecutarla.

Consulta [la guía de promoción](#) para el flujo hacia Test y Production.

Permisos de entorno

La autorización combina permisos del módulo, acceso al Domain y capacidades del entorno. El catálogo actual contiene 23 permisos de entorno: `etl.environments.view`, `etl.environments.`

`{development, test, stage, production}. {view, edit, publish, run, runtime.manage}`, `etl.environments.promote` y `etl.environments.runtime.manage`. Los permisos globales de lectura y Runtime conservan su alcance explícito; un permiso de un entorno no concede acceso a todos los demás. Administrar un módulo no sustituye estas capacidades.

ETL Developer puede trabajar en Development, Test y Stage y no tiene acceso a Production por defecto. Release Manager puede revisar los cuatro entornos y publicar en Test, Stage y Production para promover versiones autorizadas; no obtiene edición ni ejecución de Production. La promoción conserva el preview, las dependencias publicadas exactas y la aprobación explícita de Production. Los roles personalizados permiten asignar las capacidades necesarias por entorno. La identidad del usuario no se duplica por ambiente; verifica el rol activo, la membresía y sus permisos efectivos.

Los editores de Mapping, Session y Workflow de Production permanecen de solo lectura: las versiones aceptadas llegan mediante promoción. Configurar Connections existentes o nuevas de Production es una capacidad separada que exige `etl.connections.manage` y `etl.environments.production.edit`; no habilita edición de esos tres tipos ni modifica Connections legacy de solo lectura.

Trial público: pool Development gestionado

Un Trial público elegible sin una política explícita de Development puede utilizar automáticamente un pool público de Development gestionado por la plataforma. Solo incluye workers públicos compartidos verificados y autorizados para ese Trial; excluye workers privados y workers asignados a otros pools configurados. No guarda una política de Runtime ni concede permisos para administrarla.

Una política explícita de Development, incluso vacía, prevalece sobre esta opción automática. Test, Stage y Production conservan sus políticas. La capacidad depende de disponibilidad, perfil, cuotas y concurrencia; la elegibilidad no garantiza un worker disponible.

Kit GitOps por workspace

El [kit GitOps 1.1.2](#) fija Tenant, Domain y Environment en consultas, reservas de drafts, publicaciones y leases. Usa `source.environment` y `source.domainId` para exportar; `target.environment` y `bindings.domainId` para importar. Los defaults son Development y General; el estado se conserva por

destino. Lee el [ejemplo Finance y la recuperación de estado](#). El autorado directo en Production requiere la promoción gobernada, no el comando de importación.

Antes de detener o retirar capacidad

Pause VM y Stop VM conservan sus comprobaciones de trabajos activos, worker predeterminado y capacidad compartida; revisa las referencias y el impacto antes de utilizarlos. Pause scheduling detiene nuevas asignaciones y permite terminar el trabajo actual. Para retirar definitivamente la VM, utiliza Remove VM and unlink y revisa su confirmación: después de verificar ausencia cloud, Dova limpia las asignaciones y muestra Unassigned sin borrar definiciones ni versiones publicadas. Antes de ejecutar de nuevo, asigna capacidad compatible explícitamente.

Las acciones requieren el permiso del módulo y Runtime del entorno correspondiente, o un permiso global concedido explícitamente. Una VM compartida requiere autorización sobre todos sus entornos.

Promover una versión entre entornos

Revisa dependencias, conexiones del destino, permisos y aprobación antes de aplicar.

Preparar el origen

Una promoción parte de una versión publicada de Mapping, Session o Workflow. Selecciona Tenant, Domain y Environment de origen. Guarda y publica los cambios que quieras incluir; un borrador local no forma parte del paquete revisado.

Abre Promote to Test desde el editor o Configuration → Environment promotions. Selecciona tipo de artefacto, versión publicada y destino. El avance habitual es Development → Test → Production; Stage es opcional cuando está disponible. Crear una copia desde Production hacia Development utiliza Fork to Development, una operación distinta de la promoción hacia adelante.

Revisar el paquete y sus bindings

1. Revisa el preview y la cadena exacta de dependencias publicadas. Un Workflow incorpora las Sessions y Mappings que realmente referencia; una Session incorpora su Mapping.
2. Asigna Connections existentes del destino. Usa identificadores autorizados y compatibles; las credenciales permanecen en Dova Vault.
3. Revisa worker, perfil, parámetros y bindings de entrada/salida aplicables. El worker debe estar permitido en el pool del entorno de destino.
4. Comprueba el plan efectivo, DPU disponibles, concurrencia y límites de procesamiento.
5. Inspecciona diferencias, versiones y procedencia Git antes de continuar.

Si cambias la versión, la configuración o el contexto, vuelve a generar el preview. El servidor verifica de nuevo permisos, revisiones y configuración al aplicar; una revisión obsoleta no autoriza cambios.

Confirmar y verificar

Aplica únicamente el paquete revisado. Production requiere aprobación explícita y comentario, además de los permisos necesarios. La operación crea definiciones separadas en el destino y conserva las versiones de origen. No ejecuta el pipeline ni habilita automáticamente sus calendarios.

En el historial confirma el entorno, la versión, el usuario y el resultado. Abre el destino, revisa Connections y Runtime y realiza una ejecución manual autorizada con datos y destino de prueba. Verifica en Monitor las versiones consumidas, las filas y el archivo final antes de habilitar una programación.

Restaurar una versión

Rollback es una operación manual sobre una versión promovida y su configuración guardada. Revisa la revisión actual del destino y confirma nuevamente; Production conserva su requisito de aprobación. La restauración de definiciones no deshace datos escritos en archivos, bases de datos u otros servicios externos.

Git y DevOps

Puedes registrar repositorio, commit o tag y exportar el manifiesto de una promoción aplicada. La API de promoción de Dova admite preview y apply con autenticación y permisos autorizados. El kit GitOps 1.1.2 exporta el manifiesto mediante `promotion-export`; su flujo de definition-plan/apply separado importa

únicamente en Development, Test o Stage. Conserva el manifiesto y su digest en tu proceso de revisión.

Esta integración no realiza Git push, OAuth del proveedor, webhooks ni despliegues automáticos por sí sola. Un pipeline externo debe configurar sus permisos y llamar deliberadamente a las operaciones disponibles. DovaAI puede explicar un error y sugerir pasos, pero nunca aprueba, aplica, restaura, configura ni ejecuta la promoción.

Consulta [entornos y recuperación](#) y [CLI y API](#).

Preparar Connections del destino

Antes de promover hacia Production, prepara allí Connections propias con custodia y permisos válidos. Un administrador autorizado puede registrarlas o modificar sus metadatos con `etl.connections.manage` y `etl.environments.production.edit`. Esta autorización específica no habilita el editor de Mappings, Sessions o Workflows de Production. Las Connections legacy compartidas o ambiguas siguen de solo lectura.

El preview de promoción y Fork to Development solo ofrece Connections compatibles ya existentes en el destino. Selecciona cada reemplazo explícitamente; no se copian contraseñas ni se inventan referencias de secretos. Domain y Environment de la Connection permanecen ligados a su contexto de propiedad.

Kit GitOps por workspace

El [kit GitOps 1.1.2](#) fija Tenant, Domain y Environment en consultas, reservas de drafts, publicaciones y leases. Usa `source.environment` y `source.domainId` para exportar; `target.environment` y `bindings.domainId` para importar. Los defaults son Development y General; el estado se conserva por destino. Lee el [ejemplo Finance y la recuperación de estado](#). El autorado directo en Production requiere la promoción gobernada, no el comando de importación.

Dova 24, 25 y 26: alcance y cambios

Cambios de Dova 26.0.0, con la base funcional y las referencias históricas de Dova 24 y 25.

Edición actual

Esta edición describe Dova 26.0.0. Comprueba la versión activa de tu instalación antes de seguir un procedimiento. La ruta de esta referencia conserva la continuidad del manual de Dova 24; sus secciones históricas se mantienen debajo.

Cambios de 26.0.0

1. DovaAI permite seleccionar un elemento o un área para revisar una captura y su contexto seguro antes de enviarlos.
2. El lanzador sigue el arrastre con mouse o toque, muestra el espacio de acoplamiento y anima el encaje y desprendimiento de la barra inferior. Se respeta reducir movimiento.
3. Configuration de Mappings organiza campos, ayudas y acciones del componente dentro del panel; Header mapping conserva desplazamiento propio para sus columnas.
4. La barra inferior prioriza Tenant, Domain, Environment y conexión. Operational details reúne Active, Workers, Queue, Plan, Autosave y Updated; DovaAI conserva espacio al acoplarse.
5. Advanced, Validate, Save draft y Publish del editor de Mappings usan íconos con ayuda y nombres accesibles. Se elimina Run: la Session aporta la ejecución y necesita capacidad compatible.
6. El encabezado de Mappings deja de repetir DEV; el entorno permanece visible en la barra inferior.
7. La lista de Mappings elimina los filtros Environment y Runtime y usa el contexto activo.
8. La lista de Mappings alinea Search, los filtros restantes y sus acciones con íconos.
9. Las listas de Sessions y Workflows adoptan la misma barra de acciones. Eliminan solo Environment y conservan Runtime y los demás filtros.

Guardar, publicar, promover y ejecutar siguen siendo acciones distintas sujetas a permisos. Esta documentación describe controles y procedimientos; confirma la versión activa y el resultado de cada operación en tu instalación.

Base funcional de 25.0.0

1. Tenant, Domain y Environment se seleccionan desde el menú del usuario; un indicador compacto conserva el contexto visible y sus colores.
2. Permisos específicos por entorno y Release Manager separan el desarrollo de la promoción a Production. ETL Developer no tiene acceso a Production por defecto.
3. Administrator delimita workers, conexiones cloud, perfiles, operaciones y pools por su entorno autorizado; usuarios y roles mantienen una identidad compartida.
4. El panel de Administrator elimina contadores circulares y utiliza la base visual y de navegación de Dova.
5. La carga por módulo y una caché limitada de lecturas en RAM reducen solicitudes repetidas respetando identidad, rol y contexto.
6. Configuration → Miscellaneous personaliza el color compartido por Tenant+Environment, con vista previa y restauración del predeterminado.
7. ES, EN y PT permiten elegir el idioma de la interfaz desde el menú de usuario.

8. Los controles de VM utilizan las capacidades y el estado real del proveedor y se concilian periódicamente con cambios en la nube.
9. Remove VM and unlink retira capacidad propia y, tras confirmar ausencia cloud, limpia asignaciones y proyecta Unassigned sin borrar versiones ni auditoría.
10. El gráfico de disco identifica espacio libre, capacidad, uso y momento de la muestra, con el estándar visual de Dova.
11. CPU, memoria y disco muestran Not available cuando falta telemetría actual; las muestras históricas permanecen diferenciadas.

Estas capacidades conservan las comprobaciones de permisos, propiedad, operaciones pendientes y protección del servidor base. La disponibilidad de cada acción depende del proveedor implementado y de la autorización efectiva.

Cambios de 25.0.1

1. Controles compactos de VM que siguen el estado real del proveedor, con Start para Stopped y Resume para Paused según permisos.
2. Icono de DovaAI movable con mouse, tacto y teclado, con posición local por usuario autenticado.
3. Selección de un elemento o un área para compartir contexto y una captura real preparada con redacción.
4. Chat compacto con imágenes pegadas o arrastradas, vista previa y eliminación antes de enviar.
5. Navegación izquierda compartida entre Dova y Administrator y menú de usuario junto al botón, ajustado a la pantalla.
6. Lista Domain del menú sincronizada con las operaciones de creación, actualización y eliminación, conservando los cambios sin guardar.

Correcciones de 25.0.2

1. La navegación HTML de Administrator redirige a Dova a una cuenta autenticada sin permiso administrativo; la autorización de sus API se conserva.
2. La conciliación del periodo de financiación evita el error Subscription funding projection does not match mediante la proyección canónica de Billing, sin cambiar el periodo, el ciclo ni los créditos concedidos.
3. Repository carga el contexto con borradores temporales vencidos y ofrece recuperación ante un fallo de lectura, sin quedar indefinidamente en Loading.

El resumen Tenant, Domain y Environment aparece en la barra inferior de estado operativo de Dova y Administrator. Haz clic en ese resumen para abrir el menú de usuario con los tres selectores. Conserva el distintivo y el color del entorno; no ocupa una franja superior ni duplica los campos.

Consulta los procedimientos de DovaAI, menú de cuenta, Cloud workers y Repository de este manual. Las operaciones de VM, las escrituras y las publicaciones requieren sus propios permisos y comprobación del resultado.

Referencia histórica de Dova 24.0.0

Edición y versión activa

Este manual describe Dova 24.0.0. Comprueba la versión activa de tu instalación antes de seguir un procedimiento. La documentación describe capacidades; no demuestra que una promoción, una ejecución o una publicación de tu instalación haya terminado.

Cambios principales

1. Contexto global Tenant, Domain y Environment, inicialmente Development.
2. Catálogos de Mappings, Sessions, Workflows, Connections y Monitor separados por contexto, con autorización del servidor.
3. Pestañas, filtros y trabajo local conservados por contexto; respuestas anteriores descartadas y cambio bloqueado durante escrituras.
4. Guardar y publicar identifican el entorno; Production requiere capacidades explícitas para modificar sus definiciones.
5. Fork to Development recupera definiciones mediante borradores separados, conexiones compatibles del destino y revisión de dependencias, sin copiar credenciales.
6. Promociones de Mapping, Session o Workflow con dependencias publicadas exactas, preview, bindings del destino, aprobación de Production e historial de restauración manual.
7. Políticas de workers por entorno y capacidades diferenciadas de edición, publicación, ejecución, promoción y administración de Runtime.
8. DovaAI recibe el contexto del entorno y descarta conversaciones anteriores al cambiarlo. Sigue siendo un asistente de diagnóstico de solo lectura.

Actualización de artefactos anteriores

Los artefactos anteriores sin pertenencia inequívoca no se asignan automáticamente a Development. Las referencias publicadas y compartidas se preservan mediante comprobación de pertenencia y cadena exacta. Una definición ambigua puede requerir revisión y Fork to Development. Cambiar el selector nunca renombra el entorno propietario de un artefacto ni actualiza versiones publicadas.

Los roles personalizados antiguos conservan lectura según sus permisos; las nuevas capacidades de modificación por entorno requieren asignación explícita. No uses un entorno personalizado como sustituto implícito de Development o Production.

Límites y responsabilidad

La promoción y el rollback de definiciones no ejecutan pipelines ni revierten datos externos. Las Connections del destino conservan su propia custodia de credenciales. La disponibilidad de un worker, las cuotas del plan y una ejecución satisfactoria se verifican por separado.

DovaAI identifica causas y propone opciones; únicamente el usuario modifica la configuración y confirma operaciones. La procedencia Git y la exportación del manifiesto ayudan a una revisión externa, sin otorgar acceso Git ni iniciar por sí solas un pipeline DevOps.

Consulta [entornos](#) y [promoción](#). La [referencia histórica](#) conserva su ruta y el contenido de las ediciones anteriores.

Kit GitOps por workspace

El [kit GitOps 1.1.2](#) fija Tenant, Domain y Environment en consultas, reservas de drafts, publicaciones y leases. Usa `source.environment` y `source.domainId` para exportar; `target.environment` y `bindings.domainId` para importar. Los defaults son Development y General; el estado se conserva por destino. Lee el [ejemplo Finance y la recuperación de estado](#). El autorado directo en Production requiere la promoción gobernada, no el comando de importación.