

Manual avanzado de Dova y Administrator

Diseño de datos, operación y gobierno de la plataforma.

EDICIÓN 6.0.2 · SEPTIEMBRE 2026

41 capítulos · Procedimientos · Casos de uso · Laboratorios · Referencia

Mapping

Define la transformación.

Session

Vincula entorno y parámetros.

Workflow

Orquesta y supervisa.

Manual en línea y materiales de laboratorio:

docs.cloudata.solutions/dova



Contenido

Primer flujo completo: de la fuente al resultado verificado	4
POC 1: normalización determinista de clientes	6
POC 2: unir TXT y Excel para obtener un resumen regional	8
Seguridad y operación: ejecutar, recuperar y retirar recursos	10
CLI y API: automatización verificable del laboratorio	12
Overview: lectura operativa del workspace	14
Repository: dominios, carpetas y ubicación de objetos	16
Mappings: diseño y publicación de transformaciones	18
Sessions: vincular el diseño con su ejecución	20
Workflows: orquestación, versiones y ejecución	22
Monitor: seguimiento y diagnóstico de ejecuciones	24
Connections: acceso reutilizable a sistemas de datos	26
Notification channels: destinos reutilizables de notificación	28
Workers: capacidad de ejecución y diagnóstico	30
Configuration: runtime, perfiles, scheduler y salud	32
Plans and subscriptions: oferta, pedidos y continuidad del workspace	34
Developer Tools: CLI, distribuciones y API keys	36
Menú de cuenta: workspace, rol y apariencia	38
Profile: preferencias, seguridad y solicitudes de datos	40
Open items: trabajo entre editores abiertos	42
Overview: lectura operativa de Administrator	44
Cloud workers: provisión, operación y retirada segura	46



Users: cuentas, estado y acceso inicial	48
Tenants: workspace, capacidad y ciclo de vida	50
Roles & permissions: diseñar acceso mínimo	52
Subscription & billing: contratos, catálogo y licencias	54
Audit: investigar acciones con contexto	56
Registrations: revisión de solicitudes	58
Developer Tools: CLI, plataformas y API keys	60
Preferences: políticas, autenticación y comunicaciones	62
Operar un catálogo comercial versionado y ofertas por tenant	64
Limpieza ETL y lifecycle de tenants: elegir la operación correcta	66
Plantillas de ejecución, SMTP y políticas de notificación	68
Configurar SSO y completar MFA sin ampliar permisos	70
Instalar workers en Linux, Windows y cloud	72
Migrar referencias y retirar un worker con evidencia	74
Conectores, configuración y autenticación	76
Componentes de entrada y salida de Mapping	81
Transformaciones y reglas de calidad	84
Expresiones, parámetros y variables	88
Permisos y acciones de Dova y Administrator	91

Primer flujo completo: de la fuente al resultado verificado

Esta guía propone un recorrido de laboratorio que une organización, transformación, ejecución y observación. El objetivo es obtener un CSV cuyo contenido pueda comprobarse, dejando claras las versiones y las condiciones de ejecución. Utilice datos sintéticos y destinos separados de cualquier proceso operativo. La [POC de normalización](#) ofrece una entrada y un resultado conocido para completar el recorrido.

Preparar el espacio de trabajo

Entre con una identidad autorizada y compruebe el tenant y el rol activos en el menú de cuenta. Una misma persona puede disponer de contextos diferentes; el nombre del objeto por sí solo no identifica su ámbito. Antes de crear nada, abra [Repository](#) y localice un Domain y una Folder destinados al laboratorio. Si su rol no permite crearlos, solicite que un administrador prepare ese espacio.

Defina nombres reconocibles, por ejemplo un prefijo lab_, y conserve la relación entre Mapping, Session y Workflow. No reutilice identificadores de objetos productivos. Prepare también una carpeta de entrada y otra de salida accesibles para el entorno de ejecución elegido. Una ruta disponible en su portátil no necesariamente existe en el worker remoto. Las credenciales y conexiones deben configurarse mediante [Connections](#), nunca dentro de la muestra o del nombre de archivo.

Diseñar y validar el Mapping

El [Mapping](#) describe las operaciones sobre los datos. Para comenzar, utilice una fuente CSV con encabezado, una transformación sencilla y un destino CSV. Revise delimitador, codificación, nombres de columnas y representación de valores vacíos. Introduzca después la cadena de ocho transformaciones de la primera POC si desea un ejercicio más completo.

Conecte los componentes en el orden necesario. Un filtro colocado antes de una conversión puede producir un comportamiento diferente al mismo filtro después de convertir; el dibujo debe reflejar la regla de negocio. Valide la definición, examine cada advertencia y guarde el borrador. Publique sólo cuando el esquema, las expresiones y los campos de salida sean coherentes. El guardado del borrador y la publicación son pasos diferentes dentro del ciclo de vida.

Asociar la Session

La [Session](#) conecta el Mapping con sus condiciones de ejecución. Seleccione la versión prevista y revise sus bindings de entrada y salida. Para la POC, `source_csv` debe recibir el archivo de clientes y `target_csv` debe producir el CSV normalizado. Conserve los identificadores de los nodos al trasladar la definición; un nombre de archivo correcto no compensa un binding asociado al nodo equivocado.

Revise worker, perfil de ejecución y opciones disponibles. El worker debe ser compatible, estar habilitado y disponer de capacidad. Compruebe por separado su estado operativo y sus slots en [Workers](#). Evite interpretar una VM encendida como garantía de que el agente puede ejecutar el trabajo. Valide y publique la Session después de verificar estas condiciones.

Orquestar y ejecutar manualmente

Cree un [Workflow](#) que invoque la Session publicada. Un primer flujo puede contener una sola tarea. Añada decisiones o temporizadores cuando la ejecución básica tenga un resultado estable. Si utiliza las definiciones de muestra, examine las condiciones complementarias y las variables antes de ejecutarlas: el temporizador forma parte del diseño y no debe confundirse con una programación recurrente.

Valide y publique el Workflow, y lance una ejecución manual en el laboratorio. Registre su identificador. Si la interfaz acepta overrides, conserve qué valores pertenecen a ese intento; no los considere cambios permanentes de la definición. Todavía no habilite una programación: primero necesita comprobar el comportamiento manual, la ubicación del resultado y la evidencia que produce.

Comprobar el resultado y cerrar el ensayo

Abra [Monitor](#) y siga la ejecución hasta un estado terminal. Una solicitud aceptada o una fila en cola no demuestra que exista una salida correcta. Revise tareas, errores y registros disponibles. Después compare el archivo generado con el resultado esperado de la POC. El conteo de filas es una comprobación útil, pero debe acompañarse de valores, esquema y orden; el wrapper de muestras realiza una comparación byte por byte.

Documente tenant, Domain, Folder, versiones, identificador de ejecución, cantidad de filas y ubicación autorizada de la salida. Mantenga esa evidencia sin secretos. Si el resultado difiere, revise primero parsing, bindings y transformaciones; no vuelva a ejecutar repetidamente contra un destino compartido. Corrija el borrador correspondiente, publique la nueva versión y realice un intento identificable.

Cuando el recorrido sea estable, defina si necesita una programación en [Configuration](#), con una zona horaria explícita y una política de recuperación entendida. Limpie las salidas de laboratorio cuando dejen de ser necesarias, sin borrar evidencia requerida por su equipo. Para trasladar el flujo a otro entorno, vuelva a comprobar permisos, rutas, conexiones y capacidad: una POC local valida sus reglas, pero no certifica la infraestructura de destino.

POC 1: normalización determinista de clientes

Esta POC convierte un CSV sintético de 1.000 filas en una salida normalizada de 667 filas. Recorre ocho componentes Transform y comprueba el resultado con un oráculo independiente escrito en Python. El ejercicio permite estudiar limpieza, derivación, conversión, selección de registros y ordenación sin utilizar conexiones externas ni credenciales. Los nombres y valores son datos de prueba repetibles, no registros de personas reales.

Material y alcance

El [paquete de muestras](#) incluye `inputs/customers.csv`, `expected/customer_normalized.csv` y definiciones coherentes de Mapping, Session y Workflow. La entrada contiene `customer_id`, `raw_name`, `location`, `amount` y `status`. Hay espacios repetidos, caracteres acentuados, una ubicación con separador vertical y tres estados alternados: `ACTIVE`, `ENABLED` e `INACTIVE`.

El ejecutor requiere un checkout autorizado de DovaETL y las dependencias Python del proyecto, incluida la versión de `openpyxl` indicada en el paquete. No instala software ni inicia un servidor. Ejecuta el motor de transformación localmente y valida las definiciones relacionadas; no crea objetos en un tenant, no activa una programación y no prueba conectividad hacia workers remotos.

Comprender la cadena

Paso	Operación	Regla que debe conservarse
1	Clean	Recortar espacios, compactarlos y retirar acentos en nombre y ubicación
2	Replace	Reemplazar exactamente <code>ENABLED</code> por <code>ACTIVE</code>
3	Split	Dividir <code>location</code> en <code>city</code> y <code>country</code> , retirando el campo original
4	Concat	Crear <code>customer_key</code> con nombre limpio, # e identificador
5	Derive	Construir <code>segment</code> a partir de país y estado
6	Convert	Convertir <code>amount</code> a <code>amount_number</code> ; fallar ante una conversión inválida
7	Filter	Conservar únicamente el estado <code>ACTIVE</code>
8	Sort	Ordenar por importe numérico descendente

El orden de `Replace` y `Filter` explica por qué también se conservan los registros inicialmente `ENABLED`. `Convert` debe ocurrir antes de `Sort` para ordenar números y no representaciones textuales. La configuración conserva además el campo original `amount`; disponer de ambas representaciones ayuda a examinar el cambio sin ocultar el valor de entrada.

Ejecutar la comprobación local

Descargue y extraiga el paquete. Desde su carpeta principal, indique la ruta de su checkout autorizado y ejecute:

```
python .\run_pocs.py --repo C:\ruta\DovaETL --report .\verification.json
```

Sustituya C:\ruta\DovaETL por su ubicación real; no es una ruta que el programa cree. Consulte también el [README del laboratorio](#). El wrapper ejecuta las dos POC del manual en una misma comprobación. Fuerza DOVAETL_STORAGE_BACKEND=local_json, copia las entradas a un directorio temporal, bloquea conexiones de red y elimina los archivos temporales al salir. La opción --report conserva sólo el informe JSON sintético.

Antes de ejecutar el Mapping, el wrapper verifica los hashes de las muestras frente al manifiesto incluido. También valida Mapping, Session y Workflow y comprueba la referencia de la Session al fingerprint del Mapping. Una modificación accidental del CSV o de una definición debe detectarse antes de aceptar el resultado.

Interpretar el oráculo

El oráculo no llama a operadores DovaETL. Limpia las cadenas con funciones Python, aplica la equivalencia de estados, divide la ubicación, calcula los campos derivados, descarta los registros inactivos y ordena la lista resultante. Después serializa el CSV esperado con el orden de columnas definido por la muestra.

La condición de aceptación es triple: 667 filas de datos, coincidencia exacta entre el oráculo y el CSV de referencia, y coincidencia byte por byte entre el motor y ese oráculo. El informe correcto contiene PASS_BYTE_EXACT. Este texto describe el resultado esperado; si su ejecución devuelve una excepción o un código distinto de cero, la POC no ha pasado en su entorno. Conserve la salida del error antes de cambiar cualquier regla.

Una comparación exacta también detecta diferencias de encabezado, orden, salto de línea o representación numérica. No cambie el resultado esperado para que coincida con una salida que todavía no comprende. Si desea modificar la regla de negocio, escriba primero el nuevo resultado esperado mediante una implementación independiente y conserve la muestra original para reproducir el caso base.

Llevar el ejercicio a la interfaz

Use [Mappings](#) para estudiar la cadena y [Sessions](#) para revisar los bindings. El identificador de fuente es source_csv; el destino es target_csv. Prepare un Domain y una Folder de laboratorio y rutas disponibles para el worker elegido. Los archivos incluidos describen el contrato técnico, pero no sustituyen la selección explícita del entorno.

Después construya o revise el [Workflow](#), ejecute manualmente y consulte [Monitor](#). Compare su salida con expected/customer_normalized.csv. Si la ejecución local pasa y la remota falla, investigue diferencias de versión, lectura de archivos, bindings, permisos y worker. La regla de transformación comprobada no garantiza por sí sola que una ruta del portátil sea accesible desde otro equipo.



POC 2: unir TXT y Excel para obtener un resumen regional

Esta POC combina transacciones sintéticas delimitadas por | con un catálogo de clientes en Excel. Ambos archivos contienen 1.000 filas de datos. La cadena conserva las transacciones aprobadas, relaciona cada cliente con su región y genera un CSV de cuatro filas. El caso muestra cómo revisar un proceso con varias fuentes, claves con representaciones diferentes y una agregación cuyo resultado puede calcularse independientemente.

Entradas y resultado esperado

En el [paquete de muestras](#), `inputs/transactions.txt` contiene `customer_id`, `amount` y `status`. El identificador y el importe se leen desde texto. El estado alterna de manera determinista: cada quinta transacción se marca `REJECTED` y las restantes, `APPROVED`. No se utilizan pagos ni identificadores procedentes de un sistema externo.

`inputs/customers.xlsx` tiene una hoja denominada `customers`, con encabezado y 1.000 filas en las columnas A a C: `customer_id`, `customer_name` y `region`. El identificador se interpreta como entero. Los nombres y regiones contienen variaciones de espacios y acentos para que la limpieza tenga un efecto observable. El archivo esperado, `expected/regional_approved_totals.csv`, contiene únicamente `sales_region` y `approved_total`.

Revisar las ocho transformaciones

La primera rama convierte el importe del TXT a un campo numérico `amount_number`. La segunda limpia nombre y región del Excel: recorta espacios, compacta espacios repetidos, retira acentos y convierte a mayúsculas. Mantener estas ramas separadas permite localizar un problema de parsing antes de atribuirlo al join.

El Join es de tipo `inner`, utiliza `customer_id` en ambos lados y compara claves como enteros. Esta configuración es importante porque una fuente llega desde texto y la otra desde una celda numérica. El lado izquierdo proviene de la conversión del TXT y el derecho de la limpieza del XLSX. La estrategia declarada es `hash`; los valores nulos no se consideran iguales.

Después del Join se filtran las transacciones `APPROVED`. `Aggregate` agrupa por región y suma `amount_number` en `total_amount`. `Sort` ordena ese total de mayor a menor. `Rename` cambia los nombres a `sales_region` y `approved_total`; `Select` conserva exactamente esos dos campos. El destino serializa un CSV UTF-8 con encabezado y salto de línea LF.

Ejecutar sin conexiones externas

Con el checkout autorizado y sus dependencias preparados, descargue y extraiga el paquete. Desde la carpeta extraída, indique la ruta del checkout:

```
python .\run_pocs.py --repo C:\ruta\DovaETL --report .\verification.json
```

El wrapper comprueba también la primera POC. Sustituya la ruta ilustrativa por su checkout real. El [README del laboratorio](#) resume requisitos y archivos. No indique una URL ni una clave: este ejercicio trabaja con bytes de archivos locales. El directorio temporal recibe copias de las muestras y se elimina al terminar; el informe opcional sólo contiene nombres de POC, conteos, hashes y resultados de comparación.

El programa verifica integridad de los archivos, valida las definiciones y resuelve los bindings `source_txt` y `source_xlsx` utilizando sus nombres reales. Comprueba además que los metadatos de lectura reporten 1.000 filas para la hoja `customers`. Un error de nombre de hoja o de selección de filas debe investigarse como un problema de entrada, antes de revisar la agregación.

Aceptación y diagnóstico

El oráculo independiente construye un diccionario de regiones por identificador entero. Recorre las transacciones, descarta las rechazadas, agrupa los importes por región, suma y ordena. No reutiliza el operador Join ni Aggregate del motor. La salida se acepta cuando contiene cuatro filas y coincide byte por byte tanto con el oráculo como con el CSV esperado incluido.

Los importes de esta muestra usan la representación numérica definida por el ejemplo. El ejercicio no establece una política contable universal de redondeo. Para datos financieros reales, acuerde precisión, escala, moneda y momento de redondeo antes de diseñar su regla; no deduzca esas decisiones del formato que una hoja de cálculo muestra visualmente.

Si obtiene menos regiones, revise claves sin correspondencia y la condición de Join. Si los totales parecen duplicados, revise unicidad del catálogo de clientes: varias coincidencias pueden multiplicar filas antes de sumar. Si aparecen regiones separadas por acentos o espacios, examine la rama de limpieza. Si el CSV tiene columnas adicionales, compruebe Rename y Select, además de la configuración del destino.

Trasladar el caso a un Workflow

Revise los nodos en [Mappings](#) y configure ambos bindings en [Sessions](#). No trate una carga de archivo única como sustituto de las dos fuentes. Para ejecutar desde la CLI existe la opción multifuente descrita en la [guía de automatización](#); utilice los identificadores de tarea y nodo de su propia definición.

Publique las dependencias y pruebe manualmente antes de programar el [Workflow](#). Compare los archivos de salida y el estado observado en [Monitor](#). Conserve el identificador de ejecución y las versiones empleadas. El éxito de la POC local demuestra el contrato de transformación sobre estas muestras; la autorización, las rutas remotas y la capacidad del worker requieren su propia comprobación en el laboratorio.

Seguridad y operación: ejecutar, recuperar y retirar recursos

Una operación fiable requiere conocer el contexto autorizado, la versión que se ejecuta y el estado real de sus dependencias. Esta guía organiza esas comprobaciones alrededor de tareas habituales: preparar una ejecución, investigar un fallo, mover referencias y retirar un worker. Las acciones administrativas deben realizarse con un rol apropiado y con evidencia suficiente para distinguir una solicitud aceptada de un cambio confirmado.

Identidad, tenant y permisos

Compruebe el tenant y el rol activos antes de editar o ejecutar. Una cuenta con acceso a varios contextos puede ver objetos distintos al cambiar de selección. Mantenga separados el espacio de laboratorio y el operativo, y utilice nombres que permitan reconocer el propósito de cada objeto. Un nombre similar no demuestra que se trate del mismo identificador ni de la misma versión.

En [Roles and permissions](#), otorgue las capacidades necesarias para cada responsabilidad. La lectura de Monitor, la edición de definiciones y la administración de infraestructura son responsabilidades diferentes. Si una opción no aparece, compruebe permisos y contexto antes de asumir que el producto carece de esa función. No intente resolver un rechazo cambiando rutas o invocando directamente un endpoint administrativo.

Las conexiones y API keys deben permanecer en los mecanismos previstos para ellas. Evite colocar credenciales en variables de ejemplo, archivos de muestra, capturas o notas de incidencias. Una clave personal permite identificar una automatización, pero no convierte a su propietario en administrador. Revise su uso y revóquela cuando deje de ser necesaria mediante [Developer Tools](#).

Preparar una ejecución observable

Antes de ejecutar, revise las versiones publicadas de Mapping, Session y Workflow. Confirme bindings, rutas de salida, worker y capacidad. Si una tarea modifica un destino, determine cómo reconocer una ejecución repetida y qué efecto produciría volver a enviar los mismos datos. La idempotencia de una operación de infraestructura no implica que una transformación escribiendo en un sistema externo sea idempotente.

Registre el identificador del intento y consulte [Monitor](#). Distinga cola, actividad y estado terminal. Conserve los overrides empleados y la evidencia relevante sin copiar valores sensibles. Cuando se genere un archivo, valide su contenido con una regla independiente: un estado satisfactorio indica que el proceso terminó según su contrato técnico, pero la comprobación de calidad de datos sigue siendo necesaria.

Diagnosticar antes de recuperar

Ante un fallo, identifique la tarea, su mensaje y el último estado conocido. Clasifique la causa: definición inválida, entrada no accesible, credencial o permiso, capacidad, dependencia o problema del proveedor. Estas categorías requieren respuestas distintas. Repetir la ejecución no corrige una ruta equivocada ni una autorización insuficiente.

Use las acciones de recuperación que la interfaz permita para ese estado. Cancel, Terminate, Rerun y Recover no deben tratarse como sinónimos. Revise el alcance mostrado y el estado del destino antes de repetir una operación que pudo producir efectos parciales. Si el resultado es ambiguo, conserve el identificador de ejecución y confirme el estado de los recursos afectados antes de iniciar otro intento.

Migrar referencias de un worker

En [Cloud workers](#), diferencie la planificación del runtime y el estado de la VM. Drain evita asignaciones nuevas mientras termina el trabajo activo; deshabilitar la planificación no equivale a detener el proveedor. Revise ejecuciones activas y dependencias antes de retirar capacidad.

Use **Reassign references** para seleccionar origen, destino y referencias, revisar el plan y aplicar una reasignación válida. Los backups de dependencias ayudan a revisar o restaurar referencias conforme a los controles disponibles. Esta operación mueve metadatos de afinidad y dependencias; no copia archivos, discos, bases de datos ni credenciales externas. Valide que el destino pueda acceder a los recursos que necesitan las tareas migradas.

Prepare replacement abre la preparación de un nuevo worker utilizando el flujo existente. El worker anterior permanece hasta que se retire explícitamente. Compruebe compatibilidad, permisos, conectividad y capacidad del reemplazo antes de trasladar trabajo. La creación de infraestructura y la reasignación de referencias son etapas separadas, cada una con su evidencia.

Retirar sin perder el control del recurso

Abra el modal de deprovision y revise recursos cloud, referencias Dova, ejecuciones y bloqueos. **Refresh impact** obtiene el estado actualizado. Si el proveedor figura como **Unavailable**, eso no demuestra que la VM haya desaparecido. Restaure el acceso o investigue el fallo; no trate la falta de respuesta como autorización para limpiar un registro.

Cuando esté permitido, **Stop and deprovision** solicita una confirmación explícita que incluye ambas acciones. El flujo pide la detención, consulta un impacto fresco y sólo solicita eliminación si el servidor la permite. Una respuesta de detención puede indicar que el proveedor todavía está deteniéndose. En ese caso revise de nuevo el impacto más tarde; el flujo no elimina por suposición.

Si el proveedor falla, el registro se conserva para investigación y reintento. Cuando la ausencia de la VM está confirmada, el modal diferencia **Clean up Dova record** y explica que se limpiará únicamente el registro activo. Revise posteriormente inventario y [Audit](#). Finalice la intervención documentando qué se confirmó, qué permanece pendiente y quién debe resolverlo; no describa una solicitud enviada como una retirada completada.



CLI y API: automatización verificable del laboratorio

La CLI permite consultar objetos y ejecuciones, y lanzar acciones autorizadas desde una terminal. Esta guía utiliza comandos y rutas presentes en su implementación actual. No presenta una especificación OpenAPI completa ni presupone que cualquier ruta interna sea un contrato público estable. Antes de automatizar contra otra versión, consulte su ayuda y las instrucciones distribuidas con el artefacto.

Preparación y autenticación

Abra [Developer Tools](#), seleccione la distribución para su plataforma y compruebe versión, estado de firma y SHA-256. Calcule el hash del archivo descargado y compárelo con el publicado. Un nombre de archivo correcto no sustituye esa comprobación. Instale el paquete siguiendo sus instrucciones y utilice `dova --version` para identificar el cliente que ejecutará.

Cree una API key personal para el propósito del laboratorio y ejecute `dova login --url` con la URL autorizada de ese entorno. Por ejemplo:

```
dova login --url https://dova-lab.example.invalid
```

El host `.invalid` es deliberadamente ilustrativo y no ofrece un servicio: sustitúyalo por el host de laboratorio configurado por su organización. La CLI solicita la clave mediante una entrada interactiva. No la escriba en los ejemplos del manual ni la incorpore a una línea de comandos compartida. Compruebe la identidad y el tenant que informa el inicio de sesión antes de continuar.

La CLI conserva configuración de autenticación en el perfil local. Proteja ese perfil y no adjunte su archivo de configuración a incidencias. `dova logout` elimina la sesión local; la revocación de una API key en el servidor es una acción distinta. Si una clave ya no debe utilizarse en ningún equipo, revóquela desde la interfaz y revise las automatizaciones que dependían de ella.

Primera comprobación de sólo lectura

Estos comandos utilizan un Domain ilustrativo llamado `Laboratorio`; sustitúyalo por el espacio permitido:

```
dova status --output json
dova repository list --domain Laboratorio --output json
dova mapping list --name sample --limit 50 --output json
dova workflow list --status published --name sample --output json
dova run list --status failed,cancelled --sort createdAt --descending --output csv
```

Compare las consultas con [Repository](#), [Workflows](#) y [Monitor](#). Una lista vacía puede ser el resultado correcto de un filtro. Los nombres no garantizan unicidad: conserve los identificadores devueltos para las operaciones siguientes. El formato JSON facilita procesar campos explícitos; CSV resulta útil para una revisión tabular autorizada.

Rutas verificadas en el cliente

Finalidad	Método y ruta usados por la CLI
Identidad actual	GET /api/session
Domains y Folders	GET /api/etl/domains
Mappings	GET /api/etl/mappings
Sessions	GET /api/etl/session-definitions
Workflows	GET /api/etl/orchestrations
Ejecuciones de Workflow	GET /api/etl/workflow-runs
Lanzar un Workflow	POST /api/etl/orchestrations/{id}/run

El cliente envía autenticación Bearer y espera respuestas JSON. No suponga que todas las colecciones usan la misma clave: Workflows se obtiene como `orchestrations`, mientras que las ejecuciones se obtienen como `runs`. Para integración directa, confirme el esquema correspondiente y gestione rechazos y respuestas inesperadas; no transforme un error de autenticación en una lista vacía aparentemente válida.

Ejecutar una POC publicada

Las [POC local de clientes](#) y [regional](#) funcionan sin red. Para repetir sus reglas mediante un servidor, prepare primero definiciones publicadas, permisos y destinos exclusivos del laboratorio. Esta segunda fase sí comunica con el servicio y puede escribir salidas.

Para un Workflow con una fuente de archivo, la forma admitida es:

```
dova workflow run ID_WORKFLOW --input transform_workflows/inputs/customers.csv --wait --output json
```

En un Workflow con varias fuentes utilice opciones `--source` repetidas, con la forma `TASK_ID:SOURCE_NODE_ID=FILE`. Por ejemplo, reemplace `ID_TAREA` por el identificador real de la tarea que consume la Session:

```
dova workflow run ID_WORKFLOW --source ID_TAREA:source_txt=transform_workflows/inputs/transactions.txt --source ID_TAREA:source_xlsx=transform_workflows/inputs/customers.xlsx --wait --output json
```

No combine `--input` y `--source`. El cliente también admite `--var CLAVE=VALOR` y `--param CLAVE=VALOR`; sus valores deben corresponder al contrato del Workflow. Evite incluir secretos. Registre el identificador del intento, revise su estado final y compare el CSV generado con el oráculo. Que la terminal termine no basta para aceptar la calidad del resultado.

Espera, recuperación y cierre

`--wait` consulta la ejecución hasta un estado terminal. Existen límites configurables mediante `--poll` y `--wait-timeout`; un timeout del cliente no demuestra que el servidor haya cancelado el trabajo. Consulte de nuevo el identificador antes de decidir otra acción. Los comandos de cancelación, terminación, repetición y recuperación tienen efectos distintos y deben usarse conforme al estado y al alcance autorizado.

Conserve una evidencia mínima: versión del cliente, entorno, identificadores, parámetros no sensibles, estado y comparación de salida. Revise los permisos de los informes exportados. Termine la sesión local cuando corresponda y revoque claves temporales que hayan dejado de ser necesarias. Los ejemplos de esta guía describen contratos verificados en código; su ejecución contra un servicio requiere una comprobación separada y no se afirma realizada durante la preparación del manual.

Overview: lectura operativa del workspace

Objetivo y modelo

Overview reúne el estado operativo del workspace: objetos definidos, actividad de ejecución, salud y accesos a otras áreas. Es un punto de orientación, no una reconciliación de los datos de negocio. Un contador de Workflows indica definiciones; un contador de ejecuciones indica intentos de procesamiento. Ninguno demuestra por sí mismo cuántos clientes, facturas o movimientos llegaron al destino.

La vista pertenece al tenant y al rol activos. Antes de interpretar una cifra, compruebe ambos en el [menú de cuenta](#). La información disponible depende de permisos y capacidades habilitadas. Un usuario de lectura puede consultar una sección sin disponer de acciones de creación o ejecución.

Recorrido recomendado

1. Abra **Overview** en el panel izquierdo.
2. Revise **Service health** y la referencia temporal de sincronización antes de comparar métricas.
3. Observe las tarjetas **Mappings**, **Workflows**, **Executions**, **Rows processed** y **Workers online**, cuando estén disponibles.
4. Use **Open Monitor**, **Workflows** o **Mappings** para pasar del resumen al objeto o ejecución que requiere investigación.
5. En las gráficas, seleccione el período que corresponda al incidente. Mantenga el mismo período al comparar actividad y resultados.
6. Si la vista parece desactualizada, actualice mediante el control de recarga y espere su resultado; no interprete una petición pendiente como ausencia de actividad.

Elemento	Interpretación útil	Precaución
Workers online	Capacidad reportada como conectada	Conexión no garantiza compatibilidad con cada Mapping
Execution trend	Distribución temporal de ejecuciones	El período seleccionado cambia la lectura
Execution results	Resultados de intentos terminados	Un reintento puede producir más de un registro
Rows processed	Volumen informado por el motor	No sustituye la conciliación del archivo o tabla de salida

Caso de uso y POC de observabilidad

Prepare un workspace de laboratorio con una [Session](#) publicada y un [Workflow](#) que la utilice. El origen debe contener tres filas sintéticas y el destino debe ser exclusivo del laboratorio. Anote el nombre y la versión de los objetos antes de ejecutar.

Realice una ejecución manual y conserve su identificador. Abra [Monitor](#), espere un estado terminal y compruebe el archivo producido. Después vuelva a Overview y seleccione un período que incluya la prueba. El resultado esperado es una actividad compatible con el registro de Monitor; la cifra concreta depende de las demás ejecuciones presentes y de cuándo se actualice el resumen.

Repita la consulta con un rol de lectura autorizado. La POC se considera correcta si las métricas accesibles siguen siendo consultables y las acciones no autorizadas permanecen restringidas. No es necesario modificar permisos globales para realizar esta comprobación: utilice una cuenta de laboratorio preparada por el administrador.

Validación, errores y recuperación

Una gráfica vacía puede significar que no existen ejecuciones en el período, que el filtro no coincide o que la consulta no se completó. Revise el mensaje visible y contraste con Monitor antes de volver a ejecutar un proceso. Evite generar ejecuciones duplicadas para “hacer aparecer” una cifra.

Si hay trabajadores conectados pero las tareas permanecen pendientes, revise compatibilidad, perfil, exclusiones y capacidad en [Configuration](#) y [Workers](#). Si la salud de plataforma presenta incidencias, registre hora, módulo y mensaje; no copie credenciales ni contenido sensible de los logs.

Relaciones y alcance

Overview resume información; el detalle de versiones pertenece a Mappings, Sessions y Workflows, mientras que el diagnóstico por ejecución pertenece a Monitor. La POC anterior es un procedimiento propuesto, no una prueba ejecutada durante la redacción de este manual.

Repository: dominios, carpetas y ubicación de objetos

Objetivo y modelo

Repository organiza el trabajo dentro del tenant mediante **Domains** y **Folders**. Un Domain delimita una agrupación funcional; una Folder pertenece a un único Domain. Mappings, Sessions, Workflows y Connections conservan una ubicación que ayuda a encontrarlos y a comprender sus relaciones.

La jerarquía no crea un tenant adicional ni elimina los controles de acceso. Compartir un objeto mediante **Shared within tenant** amplía su disponibilidad dentro del tenant según el alcance autorizado; no lo publica para otras organizaciones. El Domain propietario siempre forma parte del alcance del objeto.

Requisitos y navegación

Entre con permisos de consulta del repositorio. Las acciones de creación, edición o eliminación requieren privilegios adicionales; su ausencia en pantalla no implica que la interfaz esté incompleta. Confirme tenant y rol antes de operar sobre una jerarquía.

El explorador permite consultar categorías como **All**, **Mappings**, **Sessions**, **Workflows**, **Connections** y **Shared**. Use la categoría y la ubicación conjuntamente: un objeto puede existir y no aparecer porque se está mirando otra Folder o un filtro distinto.

Crear una estructura de laboratorio

1. Abra **Repository**.
2. Inicie **New domain** y escriba Laboratorio ETL.
3. Añada una descripción que indique que los datos son sintéticos y que la estructura no recibe cargas productivas.
4. Cree una Folder, seleccione el **Parent domain** correspondiente y use Clientes Demo.
5. Abra la nueva ubicación y compruebe que el breadcrumb y la lista corresponden al Domain esperado.
6. Al crear un Mapping, seleccione esa misma ubicación en sus campos **Domain** y **Folder**.

Campo	Contrato visible
Name	Entre 2 y 80 caracteres: letras, números, espacios, guion o guion bajo
Description	Opcional, hasta 1.000 caracteres
Parent domain	Necesario para una Folder; establece su único Domain propietario

Los nombres de objetos ejecutables tienen su propia validación. No aplique automáticamente las reglas de nombre de Domain a un Mapping o una Session.

Estado de una Folder

Antes de editar o ejecutar, revise el estado de la carpeta propietaria:

Estado	Efecto
Active	Edición y ejecución disponibles, sujetas a permisos y demás requisitos.
Frozen	Artefactos de sólo lectura; las ejecuciones publicadas siguen disponibles.
Closed	Artefactos de sólo lectura y ejecución bloqueada.

Congelar no equivale a cerrar. Una carpeta Closed no se vuelve ejecutable por cambiar filtros o abrir el objeto desde Shared. Solicite al responsable autorizado la transición de estado correspondiente; no duplique el artefacto para eludir una restricción de ciclo de vida.

POC: ubicación y compartición controlada

Cree dos Folders dentro del Domain de laboratorio: `Origen` y `Consumo`. Guarde un Mapping de prueba en Origen. Localícelo mediante la categoría Mappings y confirme que su editor muestra la ubicación elegida.

Si su rol permite compartir, active **Shared within tenant** en el editor. Seleccione un alcance autorizado y guarde conforme al ciclo del objeto. Desde una ubicación consumidora permitida, busque el objeto en **Shared**. Resultado esperado: el objeto es localizable dentro del alcance concedido y mantiene su ubicación propietaria; no se duplica ni cambia de tenant por aparecer como compartido.

Para completar la prueba, un usuario sin el permiso necesario debe conservar la restricción de acceso. No altere controles de autorización ni utilice otro tenant real para simular esta comprobación.

Validación y recuperación

Cuando una lista esté vacía, descarte filtros activos y vuelva a la ubicación superior. Ante un error de nombre, corrija el campo señalado sin recrear toda la estructura. Si una eliminación se rechaza por referencias, revise las dependencias y resuelva primero los consumidores; no presuponga una eliminación en cascada.

Documente el identificador y la ubicación del objeto antes de reorganizarlo. Los nombres ayudan a las personas, pero las referencias entre objetos no deben interpretarse como simples rutas de archivo.

Relaciones y límites

Continúe con [Mappings](#), [Sessions](#) y [Workflows](#). La organización del repositorio es distinta de las rutas de archivos resueltas por una Session en el worker. La POC describe resultados esperados; requiere validación en su laboratorio antes de emplearse como evidencia de aceptación.

Mappings: diseño y publicación de transformaciones

Objetivo y modelo

Un **Mapping** define cómo se obtiene, transforma y entrega información. Su canvas contiene componentes, puertos y conexiones que representan el flujo de datos. Una [Session](#) vincula ese diseño con entorno, parámetros y rutas de ejecución; un [Workflow](#) orquesta las Sessions.

Distinga cuatro operaciones: editar un borrador, **Save draft**, **Publish** y **Run**. Guardar conserva cambios de diseño; publicar establece una versión consumible; ejecutar procesa datos. Un guardado correcto no demuestra que el destino haya recibido filas ni que los consumidores existentes utilicen automáticamente la nueva versión.

Requisitos y preparación

Necesita permisos de consulta y, para modificar, autorización de edición del Mapping. Publicar o ejecutar puede exigir privilegios distintos. También necesita una ubicación válida en [Repository](#), conectores configurados cuando correspondan y un worker compatible.

Use nombres que empiecen por una letra y contengan letras, números o guion bajo. El campo de nombre del editor admite de 2 a 79 caracteres. En un laboratorio, utilice `m_ClientesDemo`; evite incluir contraseñas, direcciones privadas o información personal en nombres y descripciones.

Editor y acciones

Área o control	Uso
Components	Buscar y seleccionar componentes disponibles
Configuration	Configurar el componente seleccionado
Domain / Folder	Establecer la ubicación del Mapping
Shared within tenant	Configurar compartición dentro del tenant autorizado
Validate	Revisar problemas antes de guardar o publicar
Save draft / Publish	Conservar cambios y establecer una versión
Auto arrange mapping	Reorganizar posiciones del canvas
Zoom / Fit	Ajustar la visualización sin sustituir la validación

Conecte puertos compatibles y revise cada origen y destino. En archivos planos, el Mapping define formato y esquema; las rutas de ejecución pertenecen a la Session. Esta separación permite reutilizar el mismo diseño sin copiarlo para cada entorno.

Comportamiento de Arrange

La implementación actual ordena componentes de izquierda a derecha a partir de sus dependencias. Requiere al menos dos componentes y un grafo sin ciclos. Si no caben físicamente en el canvas, conserva un diseño legible y permite desplazamiento horizontal. La operación modifica posiciones del borrador; no transforma datos ni publica una versión.

Un ciclo impide el reordenamiento y genera un mensaje de estado. Si el resultado visual no cambia, revise si el diseño ya estaba ordenado, si existe un ciclo o si el Mapping no es editable.

POC reproducible: normalización de clientes

El repositorio incluye una muestra verificada por diseño con 1.000 filas de entrada y ocho componentes **Transform**:

```
clean → replace → split → concat → derive → convert → filter → sort
```

La muestra usa `customers.csv` y define como resultado `customer_normalized.csv`, con 667 filas. Sus definiciones coherentes de Mapping, Session y Workflow están en `samples/transform_workflows/definitions/`.

En una copia local de laboratorio, con la dependencia documentada `openpyxl==3.1.5`, el generador se invoca desde la raíz:

```
python samples/transform_workflows/generate_samples.py
```

El procedimiento regenera artefactos deterministas y un `manifest.json` con conteos, hashes SHA-256 y resultados de validación. El README declara una comparación byte a byte con un oráculo Python independiente. No se ejecutó ese generador durante la redacción: los 667 registros son el resultado esperado documentado, no una nueva medición.

Revise el diseño de ejemplo componente por componente. Anote qué filas descarta Filter, qué tipos produce la conversión y qué orden fija sort. Publique únicamente la versión de laboratorio que haya validado y vinculela a una Session con archivos sintéticos.

Validación, errores y recuperación

Ante una validación fallida, abra la lista de problemas y corrija el componente señalado. Un puerto conectado no garantiza que exista la columna requerida. Compruebe nombres, tipos, nulos y compatibilidad del conector.

Si una ejecución falla por una ruta inexistente, revise primero los bindings de la Session y el acceso del worker. Si falla la autenticación, revise [Connections](#); no copie secretos al canvas. Un error por versión o dependencia requiere revisar qué versión está publicada y cuál conserva el consumidor.

Tras editar un Mapping utilizado por Sessions, confirme el aviso de versión y decida explícitamente cuándo actualizar a los consumidores. Conserve una salida anterior del laboratorio para comparar conteo, columnas y valores; “Run succeeded” no sustituye esa comparación.

Límites y continuidad

El catálogo de componentes depende de las capacidades disponibles. Este capítulo no constituye una referencia exhaustiva de expresiones ni garantiza equivalencia entre todos los conectores. Use [Monitor](#) para verificar la ejecución y [Open items](#) para cambiar entre diseños sin confundir borradores abiertos con versiones publicadas.

Sessions: vincular el diseño con su ejecución

Objetivo y modelo

Una **Session** enlaza un Mapping con un contexto ejecutable: entorno, selección de worker, perfil, recuperación, parámetros y rutas de archivos. No redefine el algoritmo de transformación. El Mapping conserva formato y esquema; la Session resuelve dónde se leen y escriben los archivos en el worker.

Esta separación permite reutilizar un Mapping para pruebas y operación sin mezclar ubicaciones. También exige revisar las versiones: una Session puede referenciar una versión anterior del Mapping aunque ya exista otra publicada.

Requisitos y ciclo de vida

Necesita acceso a la ubicación y al Mapping que desea usar. Las acciones **New session**, **Save draft**, **Publish session**, deshabilitar o eliminar están condicionadas por permisos. Para configurar **File paths**, la interfaz solicita un Mapping publicado. La existencia de un worker en un selector no reemplaza las comprobaciones de elegibilidad.

En el gestor, use búsqueda, filtros de entorno, estado y runtime, además de **Columns**, **Export CSV** y paginación. Las tarjetas de resumen ayudan a filtrar; no cambian el estado de las Sessions. Antes de una acción masiva, confirme la cantidad seleccionada y los nombres incluidos.

Campos por pestaña

Pestaña	Campos y decisiones
General	Name, Mapping, Domain, Folder, compartición, Environment y Status
Runtime	Worker, Override worker, Execution profile, Recovery y opciones de timestamp
File paths	Bindings de origen y destino derivados del Mapping
Parameters	Typed parameter overrides (JSON) y Parameter preview

Environment ofrece Development, Test, Stage y Production. Es una clasificación del contexto; elegir Production no publica por sí solo el objeto ni otorga permisos adicionales.

En **Runtime**, la opción normal es **Use tenant default**. **Override worker** permite elegir un worker explícito cuando se autoriza y resulta compatible. **Execution profile** puede heredar del Workflow. Las políticas de **Recovery** visibles incluyen fallo inmediato, un reintento, hasta tres reintentos y reanudación desde checkpoint; la recuperación efectiva depende del motor, la tarea y el checkpoint disponible.

La región y el proveedor cloud mostrados se derivan del worker y son de sólo lectura. No intente corregir una región editando la Session: revise el worker seleccionado.

Procedimiento de configuración

1. Cree `s_ClientesDemo_Test` mediante **New session**.
2. En General, seleccione el Mapping publicado de la [POC de Mappings](#) y una Folder de laboratorio.
3. Elija Test y revise el aviso de versión del Mapping.

4. En Runtime, conserve el worker del tenant o active un override compatible. Seleccione la política de recuperación que quiere comprobar.
5. En File paths, complete cada binding requerido con rutas accesibles desde el worker. Una ruta del equipo que ejecuta el navegador no es automáticamente una ruta del worker.
6. En Parameters, introduzca únicamente overrides definidos y compatibles con el diseño.
7. Guarde el borrador, revise los mensajes de validación y publique cuando corresponda.

Ejemplo de estructura de un override tipado, sólo si el Mapping define ese parámetro:

```
{
  "batchSize": {
    "type": "integer",
    "value": 5000
  }
}
```

La vista **Parameter preview** ayuda a comprobar nombre, tipo y valor antes de continuar.

POC: mismo diseño, dos contextos

Prepare dos directorios de laboratorio accesibles por el mismo worker. Copie el mismo archivo sintético de entrada en ambos. Cree dos Sessions que usen la misma versión publicada del Mapping: una con Environment Test y otra con Stage. Asigne destinos distintos para evitar sobrescritura.

Ejecute cada Session mediante un Workflow de laboratorio y compare las salidas. Resultado esperado: igualdad de los datos transformados si entradas y parámetros son equivalentes, con archivos de salida independientes. La distinción de entorno debe conservarse en las definiciones; no constituye una migración automática de archivos.

Como prueba negativa, use temporalmente una ruta de origen inexistente en la Session de Test. Observe el error en [Monitor](#), restaure la ruta y valide un nuevo intento. No altere Stage para resolver un fallo exclusivo del laboratorio de Test.

Errores, versiones y recuperación

Un worker inelegible puede indicar capacidad, estado, perfil o exclusiones incompatibles. Revise [Configuration](#) antes de seleccionar otro al azar. Un JSON inválido debe corregirse sin reemplazar los demás campos del borrador.

Si aparece un aviso de versión, registre el Mapping y la versión referenciados. Guardar una actualización puede cambiar el vínculo conforme al aviso mostrado; revise también los Workflows que consumen la Session. No presuponga que publicar un Mapping republique todas las Sessions y todos sus consumidores.

El timestamp de destino puede heredarse, habilitarse o deshabilitarse. La plantilla opcional resuelve %RUN_TIMESTAMP% conservando el directorio de destino; compruebe el nombre resultante en una prueba pequeña antes de cargas repetidas.

Relaciones y límites

La ejecución se sigue en Monitor y la orquestación se define en [Workflows](#). La POC anterior establece un criterio de aceptación reproducible; no informa de una ejecución realizada durante la preparación de esta documentación.

Workflows: orquestación, versiones y ejecución

Objetivo y modelo

Un **Workflow** coordina tareas, sus dependencias y las decisiones que determinan el recorrido de una ejecución. Las Sessions enlazan transformaciones con su contexto; el Workflow establece cómo se combinan. El canvas muestra tareas y conectores, mientras que la configuración de la tarea permite revisar su comportamiento.

Separar Mapping, Session y Workflow facilita reutilización y gobierno, pero introduce referencias versionadas. Una versión publicada del Workflow debe evaluarse junto con las versiones de sus dependencias. Cambiar una definición aguas arriba no equivale a actualizar silenciosamente una ejecución ya publicada.

Requisitos

Necesita permisos de lectura y, para cambios, edición del Workflow. Publicar, ejecutar y administrar programaciones pueden requerir permisos adicionales. Prepare una Session publicada y un worker compatible antes de comprobar la ejecución.

Utilice un Domain y una Folder de laboratorio. En los nombres de objetos ejecutables, siga la validación del editor: letra inicial y caracteres alfanuméricos o guion bajo, dentro del límite mostrado. Evite reutilizar nombres operativos para ensayos.

Recorrido de diseño

1. Abra **Workflows** y cree un Workflow de laboratorio.
2. Revise nombre, Domain, Folder y alcance de compartición.
3. Abra el editor y añada una tarea que consuma una Session preparada.
4. Configure la tarea y sus dependencias. En un diseño con decisiones, compruebe que las ramas tengan condiciones complementarias y que cada destino esté conectado donde corresponde.
5. Use los controles de canvas para seleccionar, mover, reorganizar y ajustar la vista.
6. Valide el diseño, guarde el borrador y publique una versión cuando no queden problemas.
7. Ejecute manualmente antes de añadir una programación periódica.

Acción	Efecto esperado
Save draft	Conserva la definición editable
Publish	Establece una versión para consumo autorizado
Run	Inicia un intento de ejecución
Auto arrange workflow	Reorganiza visualmente tareas; no cambia sus datos de entrada
Fit workflow	Ajusta la vista del canvas

Acción	Efecto esperado
Runtime o worker de la ejecución	Condiciona compatibilidad y capacidad; no sustituye los permisos

Los overrides de una ejecución deben tratarse como valores de ese intento. No los confunda con una edición persistente de la Session o del Workflow. Antes de reutilizar un resultado, registre cuáles se suministraron.

Programación

La sección **Workflow Scheduler** de [Configuration](#) administra programaciones reutilizables. Cree la programación allí y asígnela desde el Workflow. Revise calendario, zona horaria y estado antes de habilitarla. Una programación habilitada expresa intención de ejecución futura; no demuestra que una tarea haya arrancado o terminado.

Pruebe primero el Workflow manualmente. Después configure una frecuencia de laboratorio y compruebe su próxima ejecución en las vistas disponibles. Use una zona horaria explícita: la zona de su [Profile](#) para presentar fechas no debe confundirse con la zona de una programación.

POC: normalización con ramas complementarias

La muestra `samples/transform_workflows/` contiene definiciones coherentes de Mapping, Session y Workflow. Cada ejemplo incluye una Session, ramas de decisión complementarias, un temporizador y un destino CSV. Para normalización de clientes, la entrada documentada contiene 1.000 filas y la salida esperada 667.

Regeneración local documentada:

```
python samples/transform_workflows/generate_samples.py
```

Use los artefactos generados como material de laboratorio. Revise primero el Mapping y los bindings de la Session; después inspeccione el Workflow y explique qué rama corresponde a cada resultado. Ejecute con datos sintéticos y conserve el identificador del intento.

El resultado esperado combina dos evidencias: una ruta de tareas consistente con las condiciones configuradas y un archivo que coincide con el oráculo independiente. No basta con que la última tarea muestre éxito si las filas o valores no corresponden. El generador y la ejecución del Workflow deben verificarse en el entorno de laboratorio; no fueron ejecutados durante esta redacción.

Validación y recuperación

Ante tareas sin iniciar, examine dependencias, estado del worker, capacidad y exclusiones. Ante una tarea fallida, abra [Monitor](#) y revise el primer error relevante, no únicamente el estado global.

Si una Session o Mapping cambió de versión, examine los avisos y los vínculos publicados antes de republicar el Workflow. Conserve la versión anterior cuando necesite comparar resultados; no use la edición de un borrador como prueba de que una programación ya consume esos cambios.

La recuperación desde checkpoint depende de lo que haya persistido el motor. No significa necesariamente volver a ejecutar todo, ni garantiza que un destino externo revierta operaciones previas. Antes de reintentar una carga, compruebe si el destino tolera duplicados o si la prueba utiliza archivos de salida independientes.

Relaciones y límites

Consulte [Sessions](#) para bindings y recuperación, [Workers](#) para capacidad y [Monitor](#) para evidencia. Este capítulo no inventaría condiciones de decisión ni expresiones no verificadas: la configuración exacta de cada componente debe revisarse en su panel y documentación específica.

Monitor: seguimiento y diagnóstico de ejecuciones

Objetivo y modelo

Monitor permite seguir intentos de ejecución y consultar su resultado. Su unidad principal es la ejecución, no la definición del Workflow. Dos ejecuciones del mismo Workflow pueden utilizar entradas, parámetros y momentos distintos; deben compararse mediante sus identificadores y el contexto registrado.

Un estado de éxito representa el resultado informado por el motor. La aceptación funcional exige comprobar también la salida: conteo de filas, esquema, valores esperados y ubicación del archivo o tabla. Un error de entrega de notificación tampoco debe confundirse automáticamente con un error de transformación.

Requisitos y lectura de la vista

Necesita permisos para consultar ejecuciones del tenant activo. Acciones como ejecutar, cancelar, recuperar o limpiar historial sólo deben utilizarse cuando estén disponibles para su rol. El alcance de un usuario lector no debe ampliarse para solucionar un problema de visualización.

Revise la vista de ejecuciones y los controles de búsqueda, estado, columnas y período disponibles. En pantallas pequeñas, una tabla ancha puede necesitar desplazamiento horizontal dentro de su contenedor. Ese desplazamiento permite llegar a acciones y filtros sin alterar el ancho de toda la página.

Las pestañas **List**, **Gantt** y **Notifications** separan la lista operativa, la lectura temporal y el historial de entregas. List y Gantt comparten los filtros superiores. **History** permite expresar una ventana en minutos, hours o days; compruebe esa unidad antes de interpretar una ausencia de registros. Notifications tiene su propia tabla y paginación de entregas: una fila allí no representa una ejecución adicional del Workflow.

Dato	Pregunta que ayuda a responder
Identificador de ejecución	¿Estoy revisando exactamente el intento reportado?
Workflow y versión	¿Qué definición se ejecutó?
Estado y duración	¿Sigue activo, terminó o requiere investigación?
Tareas y mensajes	¿Dónde comenzó el problema?
Próxima ejecución programada	¿Se trata de historial real o de intención futura?

Procedimiento de investigación

1. Localice la ejecución por Workflow, estado y momento.
2. Abra su detalle o **Operational log** cuando la acción esté disponible.
3. Busque la primera tarea fallida y relacione el mensaje con su Mapping, Session y worker.
4. Revise entradas y parámetros del intento; no deduzca esos valores únicamente del borrador que está abierto actualmente.

5. Compruebe el destino antes de decidir una recuperación.
6. Registre identificador, hora y mensaje resumido, excluyendo secretos y datos sensibles.

POC: fallo controlado y recuperación

Use el Workflow sintético de [Workflows](#) con un origen pequeño. Realice un primer intento con una ruta de laboratorio inexistente, configurada en la Session de prueba. Resultado esperado: el intento termina con un error identificable y no produce una salida válida.

Restablezca la ruta correcta y realice un nuevo intento conforme a las acciones autorizadas. Compare ambos identificadores. El segundo resultado esperado es una ejecución terminada correctamente y una salida que cumple el conteo previsto. Mantenga archivos de salida separados para distinguir una nueva ejecución de un archivo residual.

Si ensaya recuperación por checkpoint, confirme primero que el intento tiene un checkpoint utilizable y que el destino soporta el comportamiento de reanudación. Un proceso externo parcialmente escrito no se revierte sólo porque el usuario pulse recuperar.

Errores y mantenimiento

Una ejecución pendiente puede esperar un worker elegible, capacidad libre o la satisfacción de dependencias. Consulte [Configuration](#) y [Workers](#) antes de repetir la solicitud. Un worker offline, una VM detenida y un proveedor unavailable son situaciones distintas.

La limpieza de historial terminado es una operación de mantenimiento con permisos y confirmación propios. Consulte el resumen y sus bloqueos; no utilice la limpieza para ocultar una incidencia aún no investigada. Conserve la evidencia necesaria fuera de capturas con datos personales.

Relaciones y límites

Overview ofrece un resumen; Monitor aporta contexto por intento. Las correcciones persistentes se realizan en Mappings, Sessions o Workflows y deben seguir su ciclo de publicación. Esta POC describe un ensayo de aceptación, no evidencia de un ensayo ya ejecutado por el manual.

Connections: acceso reutilizable a sistemas de datos

Objetivo y modelo

Connections centraliza definiciones reutilizables de conexión y su estado. Una conexión combina el tipo de sistema, sus parámetros y un método de autenticación compatible. El catálogo muestra posibilidades del producto; no garantiza que el sistema externo esté accesible, que una licencia lo permita o que el worker disponga de todas las dependencias necesarias.

La definición de conexión y las credenciales custodiadas son conceptos relacionados pero diferentes. Un indicador de credencial configurada no autoriza a revelar el secreto. Tampoco prueba que la credencial siga vigente en el sistema de destino.

Requisitos

Necesita permisos de consulta para ver Connections y permisos de administración para crear, modificar, probar o retirar credenciales según la acción. Prepare un sistema de laboratorio, una cuenta de mínimo privilegio y conectividad desde el worker que ejecutará el Mapping.

El catálogo de autenticación contempla métodos como usuario y contraseña, certificados, identidades administradas, credenciales de cliente OAuth y claves de API. La combinación válida depende del conector seleccionado; no todos los métodos aparecen ni funcionan para todas las familias.

Procedimiento de configuración

1. Abra **Connections** y examine los filtros antes de crear un duplicado.
2. Inicie la creación y seleccione el tipo de conector correspondiente al sistema real.
3. Complete nombre y ubicación, además de host, base, endpoint u otros campos que solicite ese conector.
4. Elija un método de autenticación permitido. Introduzca secretos únicamente en los controles destinados a credenciales.
5. Guarde siguiendo las validaciones del formulario.
6. Utilice la acción de prueba disponible y lea su resultado completo.
7. Consulte metadatos si el conector los ofrece y seleccione objetos que la cuenta pueda leer.
8. Vincule la conexión a un Mapping de laboratorio y ejecute una extracción pequeña.

Evidencia	Qué demuestra	Qué no demuestra
Guardado correcto	La definición pasó sus validaciones	Acceso permanente al sistema externo
Prueba correcta	La comprobación concreta respondió	Todas las consultas o permisos posibles
Metadatos visibles	La cuenta puede consultar ese catálogo	Permiso para escribir en sus objetos
Ejecución correcta	El flujo probado funcionó	Rendimiento de una carga productiva

POC: extracción limitada

Prepare una tabla sintética con tres registros y una cuenta de sólo lectura. Configure una conexión de laboratorio, pruébela y consulte sus metadatos. Cree un Mapping que lea únicamente esa tabla y escriba un archivo de prueba mediante una Session.

Resultado esperado: tres filas en el destino y valores iguales a los datos sintéticos originales, salvo transformaciones expresamente configuradas. Una escritura en la base de origen no forma parte de esta prueba y debe permanecer fuera del alcance de la cuenta.

Para comprobar el manejo de errores, utilice una definición temporal con un endpoint de laboratorio incorrecto. Registre la categoría del error y restaure el endpoint; no cambie las credenciales de una conexión compartida para provocar el fallo.

Diagnóstico y recuperación

Separe errores de red, autenticación, autorización y esquema. Una cuenta válida puede carecer de permiso sobre una tabla. Un nombre de servidor correcto puede ser inaccesible desde el worker aunque responda desde el navegador o el portátil.

Después de una rotación de credenciales, vuelva a probar el flujo mínimo. No publique contraseñas, tokens, referencias internas de secretos ni certificados privados en un ticket. Comparta únicamente el identificador de conexión, la familia del conector y el mensaje no sensible.

Relaciones y límites

Consulte [Mappings](#) para formato y lógica, [Sessions](#) para contexto y [Workers](#) para conectividad efectiva. La matriz exhaustiva de conectores y campos requiere una referencia específica; este capítulo no afirma que todos compartan requisitos ni prestaciones idénticas.

Notification channels: destinos reutilizables de notificación

Objetivo y modelo

Notification channels define destinos reutilizables para eventos de Workflows. Un canal representa dónde enviar una notificación y si está habilitado. Es distinto del contenido de una plantilla, del transporte SMTP y de las preferencias de mensajes dentro de la aplicación.

Mantener esa separación evita diagnosticar en el lugar equivocado: una plantilla puede ser válida y fallar la entrega; un canal puede estar deshabilitado aunque SMTP esté operativo. Asimismo, ver un mensaje de éxito en pantalla no prueba que un destinatario haya recibido un correo.

Requisitos y campos

Necesita autorización para consultar canales y privilegios adicionales para administrarlos o probarlos. Prepare un destino de prueba bajo su control. No utilice listas de clientes reales para una POC ni publique endpoints que contengan credenciales.

Campo o control	Uso
Name	Identifica el destino para quienes configuran Workflows
Type	Determina qué campos requiere el canal
Description	Explica su propósito y responsable
Recipients	Destinatarios del canal de correo
Endpoint	Dirección del canal que utiliza un endpoint
Enable this channel	Permite recibir eventos cuando corresponde

El selector Type ofrece Email, HTTPS Webhook, Slack y Microsoft Teams. Para correo, Recipients admite destinatarios separados por comas y la entrega utiliza la configuración SMTP de plataforma. Los otros tipos solicitan su endpoint específico mediante un campo protegido.

Al editar, una credencial ya configurada no se vuelve necesariamente a mostrar. Un campo vacío de secreto no significa que no exista una credencial custodiada. Revise el indicador de configuración y la ayuda del formulario antes de reemplazarla.

Recorrido de trabajo

1. Abra **Notification channels**.
2. Utilice **Search notification channels** para evitar duplicados.
3. Cree un canal mediante **New notification channel**.
4. Seleccione el tipo y complete los campos exigidos por ese tipo.

5. Añada una descripción clara, por ejemplo "Avisos de la POC Clientes Demo".
6. Guarde y revise el estado que devuelve la aplicación.
7. Asocie el canal desde la configuración del Workflow que lo requiera.
8. Use una prueba autorizada con un destino controlado y compruebe su recepción.

Reload, Filters, Clear filters, Columns y Export CSV actúan sobre la consulta del gestor. Exportar la lista no debería interpretarse como una exportación de secretos.

POC: entrega y deshabilitación

Cree un canal de correo de laboratorio con un buzón propio. Asócielo al Workflow sintético de [Workflows](#), configure el evento correspondiente y realice una ejecución pequeña. Registre el identificador del intento y la hora esperada del aviso.

Resultado esperado: el sistema registra el resultado de entrega y el buzón recibe el mensaje cuando transporte, permisos y configuración son válidos. Confirme que el contenido permite identificar la ejecución sin exponer datos de entrada.

Después deshabilite el canal mediante **Enable this channel** y repita únicamente el ensayo necesario en el laboratorio. El contrato visible indica que los canales deshabilitados permanecen disponibles pero no reciben eventos de Workflow. Vuelva a habilitarlo si ése es el estado previsto para la prueba final.

Errores y recuperación

Ante un fallo, revise tipo de canal, destinatarios, credencial configurada y estado habilitado. Si el transporte es correo, la configuración SMTP corresponde a Administrator; no intente resolverla añadiendo secretos al cuerpo de una plantilla.

Evite reintentos sin control: podrían producir avisos duplicados cuando el proveedor haya aceptado la primera entrega pero la confirmación se retrase. Conserve un identificador de ejecución y un destinatario de laboratorio para distinguir intentos.

Relaciones y límites

La plantilla de email establece el contenido; el canal establece el destino; Monitor permite investigar la ejecución. Consulte [Monitor](#) y [Workflows](#). Los resultados descritos son criterios de una POC pendiente de ejecución, no confirmación de entrega real durante la redacción.

Workers: capacidad de ejecución y diagnóstico

Objetivo y modelo

Un **worker** aporta capacidad para ejecutar cargas compatibles. El inventario permite revisar identidad, estado, recursos y capacidades. No confunda la conexión del agente con la disponibilidad de una VM, ni la disponibilidad de una VM con autorización para ejecutar cualquier Mapping.

Dova distingue la selección del worker, las reglas de elegibilidad y el estado operativo. La administración de infraestructura cloud se realiza en Administrator; la vista Workers de Dova ayuda a entender qué capacidad puede utilizar el workspace.

Requisitos

Necesita permisos de consulta y, para registrar o modificar trabajadores, privilegios específicos. El registro debe realizarse siguiendo el paquete y las instrucciones de la distribución correspondiente. No reutilice tokens de enrollment entre personas ni los incluya en documentación, capturas o repositorios.

Antes de asignar una Session, revise capacidades del worker y compatibilidad del Mapping. Un conector que aparece en el diseñador puede requerir dependencias o acceso de red que el worker aún no tiene.

Interpretar el inventario

Señal	Interpretación
Online	El worker reporta conexión o actividad reciente según su contrato
Offline	No se considera conectado; requiere diagnóstico
Draining	No debe interpretarse como capacidad normal para cargas nuevas
Disabled	Está deshabilitado para el uso previsto
Unhealthy	Requiere revisión de salud y diagnósticos
Free slots / capacity	Describe capacidad reportada, no un SLA de ejecución

En los detalles pueden aparecer host, plataforma, versión, recursos, último heartbeat, perfiles y conectores. Esos datos permiten comprobar compatibilidad, pero no deben copiarse indiscriminadamente a documentación pública.

Procedimiento de evaluación

1. Abra **Workers**. El panel se identifica como **Dova Worker Agent** y ofrece paquetes, **Register worker**, búsqueda, filtros, columnas y paginación. Localice el trabajador previsto para el laboratorio.
2. Revise su estado y abra sus detalles.
3. Compare conectores, tipos de componente y perfiles con el Mapping.

4. Compruebe el contexto de red: el worker debe acceder a los orígenes y destinos.
5. Revise si puede utilizarse como valor predeterminado o como override de una Session.
6. Ejecute una carga mínima y observe el resultado en [Monitor](#).

Si necesita registrar capacidad nueva, utilice la acción de registro autorizada y el paquete vigente. La instalación y los comandos concretos dependen del sistema operativo y de la versión del paquete; la guía operativa del worker debe verificarse antes de ejecutar instrucciones de infraestructura.

POC: elegibilidad antes de capacidad

Prepare una Session de laboratorio con un Mapping de archivos y un worker autorizado. Compruebe que aparece como elegible, asigne rutas que pueda leer y escribir y ejecute tres registros sintéticos.

Resultado esperado: el worker recibe el trabajo y produce la salida verificada. Registre su identificador junto al de la ejecución para demostrar qué capacidad se utilizó.

Como contraste, inspeccione una Session que requiera una capacidad ausente, sin forzar su ejecución. El resultado esperado es una advertencia o restricción de compatibilidad, no una asignación permisiva. Esta prueba valida la selección de capacidad, no el rendimiento máximo del worker.

Recuperación y límites operativos

Ante un worker offline, revise primero su servicio y comunicación con el control plane mediante procedimientos autorizados. No cambie el worker predeterminado de todo el tenant para resolver un problema aislado sin evaluar consumidores.

Unavailable del proveedor no significa que la VM haya desaparecido. Detener una VM, retirar un registro y deprovisionar recursos son operaciones distintas. Las acciones cloud requieren comprobaciones de ownership, referencias y ejecuciones; pertenecen al capítulo de Administrator y no se sustituyen por eliminar filas del inventario.

Relaciones

[Configuration](#) contiene la política de routing, perfiles y exclusiones; [Sessions](#) permite herencia u override. La POC es un procedimiento de laboratorio propuesto y no acredita instalación, conexión ni ejecución realizadas durante esta documentación.

Configuration: runtime, perfiles, scheduler y salud

Objetivo y estructura

Configuration concentra cuatro pestañas: **Dova Integration Runtime**, **Execution profiles**, **Workflow Scheduler** y **Runtime health**. Agrupa controles relacionados, pero cada uno tiene un propósito distinto. El runtime describe identidad y capacidad; el perfil establece condiciones de ejecución; el scheduler organiza cuándo solicitar trabajos; la salud ayuda a diagnosticar el entorno.

La vista conserva controles y contenido desplazable. En escritorio, las pestañas permanecen separadas del área de contenido; en tablas anchas puede existir desplazamiento horizontal interno. Use teclado o toque para alcanzar acciones sin interpretar una columna fuera de pantalla como una función inexistente.

Requisitos y decisiones

Los permisos condicionan consulta y administración. No todos los usuarios pueden registrar workers, guardar políticas, crear perfiles o modificar programaciones. Antes de cambiar una política de tenant, revise las Sessions que heredan ese valor.

Sección	Pregunta principal
Dova Integration Runtime	¿Qué capacidad y routing tiene disponible este tenant?
Execution profiles	¿Qué perfil deben utilizar las ejecuciones?
Workflow Scheduler	¿Qué Workflows se solicitan y con qué calendario?
Runtime health	¿Qué señales ayudan a explicar el estado del runtime?

Dova Integration Runtime

Revise identidad, estado, modo, worker predeterminado y capacidades. **Runtime inventory** incorpora búsqueda, actualización y selección de columnas. El resumen de workers distingue disponibilidad, slots, cola y exclusiones.

En **Tenant routing policy**, examine **Default worker**, **Fallback** y **Save policy**. El código vigente envía una política estricta; no presuponga un fallback automático a cualquier worker. Las exclusiones y la compatibilidad se aplican antes de decidir el destino de ejecución.

Las subpestañas **Workers** y **Exclusion rules** permiten revisar inventario y restricciones. Una exclusión representa una decisión de routing, no una eliminación del worker.

Perfiles y programación

En **Execution profiles**, busque el perfil existente antes de crear otro. Revise sus consumidores y límites; un perfil de ejecución no es una plantilla de aprovisionamiento cloud de Administrator.

En **Workflow Scheduler**, cree una programación reutilizable y valide su nombre, frecuencia, zona horaria y estado. Asígnela después desde el Workflow que deba utilizarla. Pruebe manualmente el Workflow antes de habilitar una frecuencia repetitiva. La próxima fecha programada no es un resultado de ejecución.

POC: política heredada y override

Prepare dos Sessions de laboratorio que utilicen el mismo Mapping publicado. En la primera, conserve **Use tenant default**; en la segunda, active **Override worker** sólo si existe otro worker elegible.

Anote la política actual y compruebe las dos Sessions sin cambiar inicialmente ningún valor global. Resultado esperado: la primera resuelve mediante la política del tenant y la segunda conserva la selección explícita, siempre sometida a elegibilidad.

En un tenant exclusivamente de laboratorio, cambie el worker predeterminado mediante **Save policy** y vuelva a revisar las Sessions. Compruebe qué valor efectivo muestra cada una y ejecute una entrada mínima si dispone de autorización. Restaure la política anterior al terminar. No haga este ensayo sobre un tenant con cargas compartidas activas.

Errores y recuperación

Si no hay workers elegibles, revise capacidades, estado, exclusiones y perfil. Si una programación no genera el resultado esperado, contraste zona horaria y estado con [Monitor](#). No cambie simultáneamente política, perfil y horario: perdería la capacidad de identificar la causa.

Ante un error al guardar, conserve el mensaje, recargue cuando resulte seguro y confirme el valor persistido antes de repetir. Una selección visible en un formulario no prueba que la política del servidor haya cambiado.

Relaciones y límites

Consulte [Workers](#), [Sessions](#) y [Workflows](#). La POC documenta una comprobación controlada, no una ejecución realizada al redactar el manual ni una promesa de capacidad o disponibilidad.

Plans and subscriptions: oferta, pedidos y continuidad del workspace

Objetivo y modelo

La opción **Plans and subscriptions** presenta la oferta disponible para el workspace. El panel se titula **Plans & billing** y reúne catálogo, revisión del pedido e historial de órdenes y pagos. Su propósito es seleccionar capacidad comercial sin confundir esa selección con la activación efectiva de una suscripción.

El catálogo disponible puede depender del tenant y de sus ofertas autorizadas. Un precio mostrado, una orden creada, un pago reportado y un pago confirmado son estados diferentes. No interprete el envío de una referencia de transferencia como confirmación bancaria ni como permiso inmediato para consumir capacidad adicional.

Requisitos y alcance

Utilice el tenant correcto y un rol con acceso al proceso comercial. La visibilidad de planes y métodos de pago depende de la configuración. Los ejemplos de este capítulo deben realizarse en un entorno de prueba sin cobros reales.

El modelo comercial conserva la identidad del workspace durante una actualización autorizada. Aun así, antes de confirmar un cambio, revise condiciones, vigencia y límites mostrados. Esta guía no establece precios ni compromisos comerciales; éstos deben verificarse en la oferta vigente.

Recorrido del pedido

1. Abra la opción y use **Reload** si necesita actualizar la información.
2. Revise el catálogo y seleccione una opción permitida.
3. En **Review your order**, compruebe conceptos, importe y condiciones antes de completar datos.
4. Seleccione **Receipt type**: Boleta o Factura.
5. Introduzca los datos fiscales requeridos y un correo de facturación controlado.
6. Seleccione un **Payment method** disponible.
7. Revise de nuevo el resumen antes de **Create order**.
8. Cuando el flujo lo requiera, use **Transfer or payment reference** y **Report payment**.
9. Consulte **Orders & payments** y confirme el estado que devuelve el sistema.

Campo	Comprobación previa
Document number	Identificador fiscal correspondiente al tipo de comprobante
Legal name	Titular que debe aparecer en el comprobante
Billing email	Buzón autorizado para comunicaciones de facturación
Billing address	Dirección aplicable a la operación
Payment method	Método habilitado para esa oferta

Campo	Comprobación previa
Payment reference	Referencia de una operación; no una contraseña ni datos de tarjeta

Remove retira la selección del carrito según el estado del flujo; no debe interpretarse como cancelación universal de órdenes ya registradas.

POC de revisión sin cobro

En un tenant de laboratorio con catálogo de prueba, seleccione un plan y abra el resumen. Compruebe que el importe, moneda y datos fiscales se corresponden con la oferta sintética. Retire la selección mediante **Remove** y verifique que no quede una orden confirmada por esa simple revisión.

Sólo si existe un proveedor sandbox configurado y autorización expresa para pruebas comerciales, complete una orden de laboratorio. Resultado esperado: la orden aparece con su estado correspondiente y no se presenta como pagada únicamente por haber reportado una referencia. Registre el identificador de la orden y la evidencia del proveedor sandbox.

No utilice números de documento de terceros ni haga una transferencia real como prueba técnica.

Errores, recuperación y relaciones

Si el catálogo no carga, conserve el mensaje y vuelva a consultar; no repita envíos de pago sin revisar el historial. Ante una discrepancia de estado, compare el pedido y la confirmación del proveedor mediante el responsable autorizado.

Los límites efectivos del tenant se administran y auditan en Administrator. Los objetos de [Repository](#) y sus ejecuciones se revisan en sus módulos respectivos. Esta POC es un procedimiento propuesto; no confirma compras, pagos ni activaciones realizadas durante la redacción.



Developer Tools: CLI, distribuciones y API keys

Objetivo y modelo

Developer Tools reúne distribuciones de Dova CLI, instrucciones iniciales y gestión de API keys personales. La CLI permite trabajar desde una terminal, pero no evita los permisos, límites ni el aislamiento del tenant. Una clave hereda los permisos de su propietario y puede revocarse.

El panel diferencia plataforma, versión, estado de firma y SHA-256 del artefacto. Una distribución disponible no implica que esté firmada: revise el estado mostrado en lugar de asumir una garantía uniforme para todos los paquetes.

Requisitos y preparación

Necesita acceso a la opción y autorización para las operaciones que ejecutará. Seleccione un equipo de laboratorio compatible con la distribución disponible. No guarde claves en scripts versionados, capturas, tickets o el propio manual.

Antes de instalar, compruebe plataforma y versión. Descargue el artefacto desde la acción **Download** correspondiente y compare su hash con el publicado en el panel. Por ejemplo, en PowerShell puede calcularlo así:

```
Get-FileHash -Algorithm SHA256 -LiteralPath .\dova-cli-descargado.zip
```

El nombre del archivo es ilustrativo; utilice el nombre real descargado. Una diferencia de hash requiere detener la instalación y volver a verificar el origen.

Recorrido de la interfaz

1. Revise **Available platforms**.
2. Seleccione la distribución correcta y compruebe **Version, Signing y SHA-256**.
3. En **API keys**, use **Create API key** sólo para un propósito definido.
4. Capture el secreto únicamente en el mecanismo seguro previsto para su uso; no lo incluya en ejemplos compartidos.
5. Siga **Quick start** para autenticar la CLI.
6. Use **Revoke** cuando la clave deje de ser necesaria y compruebe que no se emplee en otra automatización autorizada.

El inventario de claves muestra información como nombre, prefijo y uso. El prefijo permite identificar una clave sin divulgar su valor completo.

POC de sólo lectura

Tras instalar y configurar la CLI, autentique contra la URL de su laboratorio. La siguiente secuencia usa un dominio reservado como marcador de laboratorio:

```
dova login --url https://dova-lab.example.test
dova repository list --domain "Laboratorio ETL" --output json
dova workflow list --status published --name customer --limit 50
dova run list --status failed,cancelled --sort createdAt --descending --output csv
```

Sustituya URL, Domain y filtros por los valores autorizados del laboratorio. La URL `.example.test` es un marcador y debe reemplazarse por el endpoint real del laboratorio.

Resultado esperado: las consultas devuelven únicamente objetos y ejecuciones accesibles para la identidad autenticada. Una lista vacía puede ser válida si el filtro no coincide. Conserve salida sintética sin secretos para comparar el resultado con [Repository](#) y [Monitor](#).

Ejecución y recuperación: alcance separado

El Quick start también publica comandos de ejecución con archivos, variables y múltiples fuentes, además de recuperación por identificador. Esas operaciones producen efectos y deben ensayarse con destinos exclusivos del laboratorio. Revise primero [Workflows](#), bindings de [Sessions](#) y política de recuperación.

No convierta automáticamente una comprobación de instalación en una ejecución mutante. La POC anterior se limita a lectura.

Errores y mantenimiento

Ante un rechazo, distinga credencial revocada, sesión inválida, permiso insuficiente y filtro incorrecto. Regenerar una clave no resuelve permisos ausentes. Tras una actualización, compruebe la versión instalada y las instrucciones del paquete vigente.

Este capítulo verifica las formas de comando publicadas por la interfaz; no afirma que se hayan descargado paquetes o ejecutado comandos durante su redacción.

Menú de cuenta: workspace, rol y apariencia

Objetivo y modelo

El menú de cuenta muestra la identidad autenticada y el contexto de trabajo. Desde Dova permite revisar tenant y rol, cambiar a un workspace autorizado, ajustar apariencia y abrir funciones personales. Es el primer lugar que debe comprobarse antes de editar o ejecutar algo cuando una persona pertenece a más de un workspace.

El rol activo condiciona las acciones disponibles. Cambiarlo no concede privilegios nuevos: sólo permite utilizar un rol ya autorizado. Del mismo modo, una lista de workspaces no representa un catálogo abierto de tenants.

Estructura del menú

Área	Función
Identidad	Nombre, correo y avatar de la persona
Tenant	Workspace sobre el que opera la sesión
Role	Rol activo y selector cuando corresponde
Workspaces	Membresías autorizadas y workspace actual
Appearance	Mode y Palette
Acciones	Profile, Downloads, API Keys, Documentation y Sign out

En nombres largos, la interfaz puede abreviar el texto visible. Antes de una operación sensible, confirme la identidad y el tenant por los campos disponibles y no sólo por iniciales del avatar.

Dova y Administrator comparten el diseño del menú, pero no todas sus acciones son idénticas. Administrator ofrece **Back to workspace**, mientras que Dova dispone de **Profile**. Las diferencias corresponden al contexto de navegación.

Procedimiento seguro de cambio de contexto

1. Abra el control de cuenta del encabezado.
2. Revise el workspace actual y el rol.
3. Si necesita otro workspace, seleccione una membresía autorizada.
4. Espere a que termine el cambio y confirme nuevamente tenant y rol.
5. Abra el módulo necesario y compruebe que los objetos corresponden al contexto esperado.
6. Si una acción ya no aparece, revise permisos y pertenencia antes de asumir un fallo de interfaz.

Para modificar la apariencia, seleccione **Mode** y **Palette**. Los controles nativos conservan su navegación de teclado. En menús con altura limitada, desplácese dentro del menú para llegar a las acciones inferiores, incluida **Sign out**.

POC: contraste de roles sin cambios de datos

Utilice una cuenta de laboratorio con dos roles previamente autorizados: uno de lectura y otro de edición. Abra [Mappings](#), anote qué acciones están disponibles y cambie al rol lector mediante el menú.

Resultado esperado: la navegación permitida permanece disponible y las acciones que exigen edición quedan restringidas. No intente ejecutar peticiones manuales para saltar esa restricción. Vuelva al rol de edición y confirme el contexto antes de continuar.

Después cambie entre Light y Dark, recorra el menú con teclado y cierre con Escape. El foco debe volver al control que abrió el menú. Esta prueba comprueba acceso y operabilidad, no amplía la autorización de la cuenta.

Validación y recuperación

Si un cambio de contexto falla, conserve el mensaje y compruebe cuál sigue siendo el contexto activo. No dé por realizado el cambio sólo porque pulsó una opción. Si el workspace esperado no figura en la lista, solicite al administrador revisar la membresía.

Sign out termina la sesión mediante el flujo de la aplicación. Cerrar una pestaña del navegador no es un sustituto de esta acción en equipos compartidos. Los cambios de contraseña tienen efectos adicionales descritos en [Profile](#).

Relaciones y límites

Las preferencias personales se administran en Profile; distribuciones y claves se encuentran en [Developer Tools](#); los objetos abiertos se gestionan mediante [Open items](#). La POC es una guía de validación con identidades sintéticas, no una constatación de permisos de usuarios reales.



Profile: preferencias, seguridad y solicitudes de datos

Objetivo y estructura

Profile reúne ajustes personales, información de la cuenta y acciones de privacidad. No es el editor administrativo de usuarios ni un mecanismo para cambiar libremente de tenant o conceder roles.

Abra el menú de cuenta de Dova y seleccione **Profile**. El diálogo muestra identidad y contexto, además de secciones de preferencias, seguridad y datos. En pantallas pequeñas, el cuerpo se desplaza para mantener accesibles los controles y el cierre.

Preferencias personales

Revise nombre, correo, tenant y rol antes de realizar cambios. El campo **Timezone** utiliza una zona horaria, por ejemplo America/Lima, y se guarda mediante **Save preferences**. La ayuda visible indica que se usa para presentar fechas y horas en la experiencia de la cuenta.

No confunda esta preferencia con la zona horaria de una programación de Workflow. Cambiar cómo ve una fecha no debe interpretarse como reprogramación automática de los procesos. Consulte [Configuration](#) para el scheduler.

Use **Change photo** para seleccionar PNG, JPEG o WebP. El código admite un archivo de entrada de hasta 10 MB, recorta el centro a una imagen cuadrada de 256 píxeles y la optimiza antes de enviarla. **Remove photo** retira la foto existente. Espere el mensaje de resultado antes de cerrar; un fallo al procesar la imagen no equivale a un guardado. Utilice una imagen de laboratorio sin información sensible cuando pruebe el flujo.

Seguridad

Campo	Finalidad
Current password	Acreditar la operación de cambio
New password	Establecer la nueva contraseña
Confirm new password	Evitar errores de transcripción
Change password	Enviar la solicitud validada

La interfaz exige entre 12 y 128 caracteres para la contraseña nueva. El servidor puede aplicar comprobaciones adicionales. No utilice la misma contraseña en la documentación y en una cuenta real.

El aviso del diálogo indica que cambiar la contraseña cierra todas las sesiones de navegador y dispositivos confiables. Prepare el método de acceso y los factores de autenticación antes de realizar el cambio. Esta acción no debe emplearse como prueba rutinaria en una cuenta compartida de operación.

Datos y privacidad

Download profile export permite solicitar una exportación del perfil. Trate el archivo como información personal y guárdelo sólo en una ubicación autorizada.

Request profile data deletion y **Request account deletion** son solicitudes distintas. El flujo solicita confirmar escribiendo el correo y enviando **Submit request**. El aviso aclara que las solicitudes se revisan antes de afectar propiedad compartida del tenant o registros de auditoría retenidos.

Por tanto, recibir una confirmación de solicitud no equivale a demostrar un borrado inmediato. No prometa eliminación automática de todos los datos asociados al workspace.

POC reversible de preferencias

En una cuenta de laboratorio, anote la zona horaria actual. Cambie **Timezone** a *America/Lima*, guarde y vuelva a abrir Profile. Resultado esperado: el valor persistido coincide con el seleccionado y las fechas pertinentes se presentan según la preferencia admitida.

Restaurar la zona anterior al terminar. Para comprobar validaciones, introduzca una zona claramente inválida en el laboratorio y observe el mensaje sin insistir con solicitudes repetidas.

Puede abrir una solicitud de eliminación para inspeccionar su confirmación y cancelarla con **Cancel**, sin enviarla. No use **Submit request** como ensayo si no desea realmente iniciar el proceso.

Errores, recuperación y relaciones

Si un guardado falla, conserve el mensaje y confirme el valor persistido antes de repetir. Si la sesión termina después de cambiar la contraseña, vuelva a autenticarse mediante el flujo autorizado; no lo interprete como una pérdida de objetos.

Consulte [Cuenta](#) para contexto y apariencia y [Developer Tools](#) para claves personales. La POC propuesta no fue ejecutada durante esta redacción y no constituye una solicitud real de cambio de contraseña o eliminación.

Open items: trabajo entre editores abiertos

Objetivo y modelo

Open items permite cambiar entre editores de Mapping, Session y Workflow. Cada pestaña identifica el tipo de objeto, su nombre y su estado de edición. La barra organiza el trabajo del navegador; no representa una nueva versión del objeto ni una copia de los datos.

Un objeto abierto puede tener cambios pendientes. Mantenerlo visible en una pestaña no garantiza que esos cambios estén guardados o publicados. Esta distinción es esencial cuando se alterna entre un Mapping, su Session y el Workflow consumidor.

Acceso y controles

La barra aparece en vistas compatibles cuando existen objetos abiertos. En anchuras limitadas, **More** permite acceder a elementos que no caben en la fila. El nombre puede abreviarse, mientras que el identificador de tipo ayuda a distinguir objetos con nombres similares.

Acción del menú contextual	Efecto
Reload	Vuelve a consultar el objeto desde el servidor
Duplicate	Inicia la duplicación cuando el objeto y los permisos lo permiten
Pin tab / Unpin tab	Cambia la fijación de la pestaña
Close	Cierra el elemento elegido mediante su flujo de confirmación
Close other tabs	Cierra otras pestañas no fijadas candidatas
Close tabs to the right	Cierra las pestañas candidatas situadas a la derecha

La disponibilidad depende del tipo, la existencia del objeto, permisos y estado. Una pestaña fijada modifica su tratamiento en operaciones de cierre; no protege el objeto contra cambios administrativos.

Procedimiento recomendado

1. Abra un Mapping desde su gestor.
2. Abra la Session que lo consume y, después, un Workflow relacionado.
3. Cambie entre los tres mediante Open items y confirme el nombre y tipo antes de editar.
4. Guarde o descarte deliberadamente los cambios pendientes.
5. Utilice el menú contextual sólo después de revisar qué pestañas afectará.
6. En móvil, abra **More** para localizar los elementos que no aparecen en la fila.

Para usar **Reload**, compruebe primero si existen cambios locales. Una recarga consulta el servidor y puede requerir resolver el borrador; no es equivalente a refrescar únicamente una cifra del encabezado.

POC: navegación con un borrador pendiente

Prepare tres objetos sintéticos vinculados: `m_ClientesDemo`, `s_ClientesDemo_Test` y un `Workflow` de laboratorio. Abra sus editores. Modifique un campo permitido del Mapping sin publicar y cambie a la Session mediante la barra.

Resultado esperado: la aplicación conserva el estado de edición conforme al flujo de borradores y permite identificar el objeto activo. Regrese al Mapping y compruebe el campo editado antes de decidir si guardarlo.

A continuación intente cerrar el objeto con cambios. Observe la confirmación que ofrece la aplicación y cancele el cierre para verificar que el editor sigue disponible. Finalmente descarte o guarde expresamente los cambios del laboratorio y cierre la pestaña.

Repita en una anchura móvil con suficientes objetos para mostrar **More**. Compruebe que puede localizar el objeto, activarlo y llegar al cierre mediante teclado o toque.

Errores y recuperación

Si **Reload** informa que el objeto ya no existe, no cree inmediatamente otro con el mismo nombre: confirme en el gestor qué ocurrió y quién tiene autorización para recuperarlo. Un error de acción en la pestaña debe investigarse sin asumir que la definición se modificó correctamente.

Los bloqueos de edición y permisos siguen aplicándose aunque el objeto permanezca abierto. Un cambio de rol o contexto puede alterar las acciones disponibles; revise el [menú de cuenta](#).

Relaciones y límites

Open items complementa [Mappings](#), [Sessions](#) y [Workflows](#). No sustituye su publicación ni la verificación en [Monitor](#). La POC describe una comprobación de borradores y navegación; no afirma que cerrar una pestaña elimine el objeto del repositorio.

Overview: lectura operativa de Administrator

Objetivo y permisos

Overview reúne indicadores administrativos para decidir dónde investigar: salud y capacidad, historial de recursos, estado de usuarios, facturas y asignación de uso. No sustituye al Monitor de ejecuciones de Dova ni constituye una conciliación contable. El acceso de navegación corresponde a `admin.overview.view`; las acciones de otros módulos conservan sus propios permisos. Antes de interpretar una cifra, confirme el workspace y el rol activo en el menú de cuenta. Un cambio de tenant cambia el alcance de la información disponible.

Los ejemplos son sintéticos y no representan una instalación real. Interprete los indicadores en el contexto del workspace seleccionado y la versión de Dova que utilice.

Paneles y significado

Panel visible	Pregunta operativa	Precaución
Worker resource history	¿Cómo evolucionan CPU, memoria y disco disponible?	Una muestra no demuestra que una ejecución terminó
Users by status	¿Cuál es la distribución de cuentas por estado?	Cuenta existente no implica acceso habilitado
Invoices by status	¿Qué estados presentan las facturas?	No es evidencia bancaria independiente
Usage allocation	¿Cómo se compara uso y capacidad?	Consultar unidades y tenant
Quick access	¿A qué módulo debo ir?	No concede permisos adicionales

Procedimiento de revisión diaria

1. Abra Administrator y confirme identidad, workspace y rol. Compruebe que corresponde al entorno que pretende revisar.
2. Entre en **Overview** y espere a que terminen los indicadores de carga. Distinga cero de un error o ausencia de muestras.
3. Revise capacidad e historial de recursos. Compare observaciones del mismo worker y periodo antes de concluir que existe una tendencia.
4. Si hay anomalías de usuarios, abra **Users**; si son comerciales, **Subscription & billing**. Para infraestructura utilice **Cloud workers**.
5. Registre periodo, módulo, síntoma y resultado esperado. No copie información personal ni valores sensibles.

Caso de uso y POC sintético

Un tenant de laboratorio tiene dos workers, tres usuarios y una factura ficticia. Durante una carga de prueba aumenta la memoria de un worker. El objetivo es pasar del resumen al detalle sin modificar capacidad por una sola observación. Identifique el worker, abra su historial y contraste el periodo con la ejecución sintética en Dova. El resultado esperado es explicar la observación y su alcance, no concluir automáticamente que falta memoria.

Use nombres como `laboratorio-documentacion` y `worker-demo-a`. Capture sólo datos generados para el ejercicio e indique cuándo no hay suficientes muestras para comparar. No extrapole tiempos de laboratorio a compromisos de rendimiento de producción.

Validaciones, errores y recuperación

Si un panel no carga, conserve el mensaje visible y vuelva a intentarlo cuando sesión y conectividad estén disponibles. No cambie de tenant para eludir una denegación. Si un acceso directo no aparece, revise el rol activo y su permiso específico. Compare resultados después de actualizar; una pestaña antigua puede mostrar valores previos a una acción.

Overview no debe utilizarse para borrar ejecuciones, migrar referencias ni cambiar contratos comerciales. Esas operaciones tienen pantallas y confirmaciones propias. Para dar por resuelta una incidencia, vuelva al módulo que es fuente de información y compruebe el estado final.

Cloud workers: provisión, operación y retirada segura

Objetivo y acceso

Cloud workers administra infraestructura mediante **Workers**, **Provisioning history**, **Connections** y **Profiles**. Leer requiere `admin.cloud_workers.view`; modificar requiere `admin.cloud_workers.manage` y alcance autorizado sobre el tenant. Worker, conexión cloud y perfil son entidades distintas. Crear infraestructura no demuestra todavía que el agente esté registrado y sea elegible para ejecutar.

Connections y Profiles

En **Connections**, el formulario recoge nombre, proveedor, scope, región predeterminada, referencia de identidad y regiones permitidas. Existen campos específicos como tenant de directorio, cuenta de servicio de runtime, subred, imagen aprobada y modo de salida. Complete los campos habilitados para el proveedor elegido. El contrato utiliza identidades federadas: no pegue contraseñas cloud ni claves privadas. Verifique la conexión y siga las instrucciones del proveedor antes de provisionar.

Profiles define nombre, proveedor, vCPU, memoria, disco, política de ubicación y estado. Puede incluir estimación horaria, presupuesto mensual, moneda y fecha de precio. Una estimación no garantiza la factura cloud. Este perfil de infraestructura no es el **Execution profile** de Dova, que expresa política de ejecución.

Workers e historial

Para provisionar seleccione tenant, conexión, perfil, nombre, slots máximos, región y zona cuando corresponda; revise la política de compartición ofrecida. La estimación de precio es opcional. Consulte **Provisioning history** y logs para distinguir solicitud aceptada, recurso creado y agente registrado. No repita una creación porque el worker aún no aparece online.

Drain worker deja de asignar nuevos trabajos mientras termina el trabajo activo. **Disable worker scheduling** modifica planificación; **Stop VM** actúa sobre el proveedor. No son equivalentes. **Start VM**, edición y reemplazo aparecen según estado y permiso.

Migrar antes de retirar

Reassign references permite revisar origen, destino y referencias, generar preview y aplicar un plan válido.

Dependency backups consulta evidencia previa; restaurar o reasignar también requiere validación. El asistente de Sessions ofrece reemplazar worker, distribuir Sessions entre destinos compatibles o deshabilitar Sessions. Esto no copia archivos, bases de datos ni la VM. Revise por separado afinidades, credenciales, reglas, worker predeterminado y ejecuciones pendientes.

Stop and deprovision

1. Abra **Deprovision worker and cloud VM** y lea recursos, referencias, ejecuciones y bloqueos.
2. Si procede detener primero, seleccione esa opción explícita. Escriba la confirmación y marque el reconocimiento del efecto.
3. El asistente solicita stop, obtiene impacto fresco y sólo solicita deprovision cuando el servidor confirma que está permitido. Stop y eliminación usan claves de idempotencia diferentes.

4. Si el proveedor falla o continúa deteniéndose, no se omiten validaciones. Actualice impacto y resuelva el impedimento.
5. Dova elimina su registro activo después de confirmar ausencia del recurso y revisar otra vez las dependencias. Conserva auditoría.

Unavailable no significa absent. No autoriza borrar el registro ni asegura que cesaron los costes. La limpieza de un recurso ausente exige comprobación del proveedor. Running, stopped y agente offline describen aspectos diferentes.

POC, errores y recuperación

En infraestructura de laboratorio autorizada prepare worker-demo-a y un destino compatible. Cree una Session sintética, inspeccione referencias y pruebe migración. Verifique plan y resultado antes de retirar el origen. Simule proveedor indisponible en pruebas locales: debe producir error y conservar registro, nunca eliminación silenciosa.

Jobs activos o en cola, referencias, worker predeterminado, transición de compartición y escaneo incompleto bloquean retirada. No altere almacenamiento manualmente. Tras timeout revise estado e historial antes de reintentar. Cerrar un diálogo no cancela una acción recibida por el proveedor; vuelva a consultar el inventario.

Users: cuentas, estado y acceso inicial

Objetivo y permisos

Users administra cuentas dentro del alcance delegado. La navegación requiere `admin.users.view`; crear y modificar requiere `admin.users.manage`. Ver cuentas no autoriza cambiar roles. Confirme tenant y rol activo antes de abrir **New user**. No conceda capacidades adicionales por comodidad durante un diagnóstico.

Campos y decisiones

Elemento	Decisión administrativa
Name	Nombre identificable, sin contraseñas ni notas sensibles
Email	Dirección válida controlada por el destinatario
Tenant	Workspace dentro del alcance delegado
Estado	Opción correspondiente al ciclo de la cuenta
Grupo/permisos y roles	Capacidades mínimas según opciones permitidas
Método de primer acceso	Mecanismo de activación ofrecido

No todas las opciones se aplican a todas las cuentas. Una identidad externa y una cuenta local pueden tener acciones diferentes. La UI contempla emisión y revocación de activaciones, conversión a acceso local y restauración de SSO en contextos específicos; no son operaciones intercambiables.

Alta controlada

1. Abra **Users** y busque la dirección antes de crear otra cuenta. Evite duplicar una identidad por una invitación pendiente.
2. Seleccione **New user**, complete nombre y email, y elija el tenant correcto.
3. Asigne sólo roles necesarios. Relacione cada permiso con una tarea concreta.
4. Seleccione estado y método de primer acceso disponible. Revise antes de confirmar.
5. Tras guardar, compruebe cuenta y estado en el inventario. Creación correcta no prueba recepción o uso de la activación.

Los enlaces de activación son de un solo uso. Emitir uno nuevo revoca el anterior según el flujo. No copie enlaces a documentación, tickets públicos ni capturas. Ante fallo de entrega revise transporte y operación administrativa antes de emitir repetidamente invitaciones.

Caso de uso y POC sintético

Prepare en un entorno aislado `analista.demo@example.test`, vinculada a `laboratorio-documentacion`. El objetivo es permitir lectura necesaria sin administración de `workers`. Cree previamente el rol adecuado, asigne la cuenta y verifique opciones con una sesión de prueba autorizada. Incluya tanto una lectura permitida como una acción administrativa no disponible o rechazada.

`example.test` es ilustrativo: no sirve como destinatario real para comprobar entrega. Si prueba correo, utilice un buzón de laboratorio controlado y no publique su configuración. La evidencia debe mostrar el resultado del acceso, nunca secretos del primer inicio.

Errores y recuperación

Corrija campos incompletos antes de enviar; email tiene formato validado y tenant es obligatorio. Un error de autorización exige revisar el rol activo, no elevar automáticamente permisos de la cuenta objetivo. Si su estado cambió en paralelo, recargue y compare antes de editar.

Para una identidad SSO diagnostique proveedor y política primero. **Convert to local access** cambia el mecanismo de inicio; no reenvía simplemente un email. **Restore SSO access** elimina el bloqueo de activación local en casos soportados conservando tenants y roles. Registre motivo y verifique estado después. Cerrar el diálogo descarta interacción local, pero no deshace una petición ya confirmada.

Una revisión periódica debe comprobar que las capacidades siguen respondiendo al trabajo de la persona. Al retirar acceso, use las acciones autorizadas y conserve trazabilidad. No comparta una cuenta administrativa para evitar crear identidades individuales: impediría atribuir correctamente operaciones.

Tenants: workspace, capacidad y ciclo de vida

Objetivo y alcance

Un tenant delimita el workspace del cliente y sus recursos. **Tenants** administra esa frontera; no es una carpeta de Repository ni un rol. Leer requiere `admin.tenants.view` y modificar `admin.tenants.manage`, sujeto al alcance del administrador. Confirme nombre e identidad del tenant, contexto activo y finalidad del cambio.

Los límites expresan capacidad concedida; el uso representa consumo observado. Aumentar capacidad no crea infraestructura ni transfiere automáticamente datos. Las pantallas comerciales y el lifecycle pueden añadir condiciones que deben respetarse.

Campos principales

Campo	Interpretación
Name	Nombre del workspace
Type	Tipo entre opciones disponibles
Parent	Relación jerárquica permitida
Status	Estado administrativo admitido
Plan	Plan disponible en el contexto
Seats	Capacidad de puestos
Monthly jobs	Límite de trabajos mensuales
Storage MB	Capacidad expresada en MB

Las capacidades son numéricas y obligatorias. No interprete vacío como ilimitado ni copie cifras de otro tenant sin verificar unidad y política. Interfaz y servidor determinan valores y transiciones admisibles.

Procedimiento de cambio

1. Localice el tenant y revise estado y uso antes de editar.
2. Identifique la necesidad: presentación, capacidad o lifecycle. Evite agrupar cambios comerciales y operativos sin revisión.
3. Modifique campos autorizados y compruebe unidades. Si cambia jerarquía, confirme que el destino pertenece al alcance permitido.
4. Guarde y revise el inventario. Reabra el detalle para comprobar valores persistidos.
5. Documente motivo y resultado sin exponer datos del cliente.

Suscripción y trial pueden tener fechas y condiciones propias. Para ampliar un trial use el flujo de extensión disponible, con motivo y modalidad, en lugar de recrear el workspace. Mantener identidad permite continuidad de recursos y evita confundir una extensión con alta nueva.

POC sintético

Use laboratorio-documentacion sin datos reales. Asigne capacidad de ejemplo acorde al entorno y registre unidades. Compruebe que un operador delegado sólo puede ver o editar workspaces autorizados. El resultado esperado es una modificación trazable dentro del mismo tenant, no una copia a otro.

Para demostrar trial utilice fixture o sandbox. Registre identidad antes y después de una extensión autorizada: debe mantenerse. No use el ejercicio para renovar un trial real ni saltar condiciones comerciales. El capítulo Subscription & billing explica revisiones contractuales por separado.

Validaciones, errores y recuperación

Si guardar falla, conserve el mensaje y revise campos obligatorios, unidades y alcance. Ante conflicto recargue antes de repetir: otra acción pudo cambiar estado o capacidad. No altere archivos del almacenamiento para resolver una inconsistencia visual.

Data Cleanup es una operación distinta en Preferences. No presuponga que cambiar estado del tenant borra datos, ni que borrar un objeto libera inmediatamente toda cuota. El inventario y el contrato de cada recurso determinan el resultado. Revise compartición e historial retenido antes de una baja.

Cuando el objetivo sea diagnosticar uso, empiece por lectura e identifique qué tipo de objeto consume la capacidad. Reducir un límite sin conocer el uso puede impedir operaciones futuras. Para recuperar una configuración equivocada compare con el valor previo autorizado y aplique una corrección administrativa trazable; no reconstruya tenant ni identidades.

Roles & permissions: diseñar acceso mínimo

Objetivo y permisos

Roles & permissions separa **Roles**, conjuntos de capacidades, de **Permissions**, catálogo consultable. La navegación usa `admin.roles.view`; mutaciones requieren la autorización administrativa correspondiente. La autoridad efectiva depende además del rol activo y del tenant permitido. Una membresía secundaria potente no equivale a utilizar ese rol como contexto activo.

Un rol debe responder a un trabajo reconocible. No lo diseñe marcando todas las casillas de un administrador existente. La ausencia de un botón no reemplaza la comprobación del servidor; mostrarlo tampoco concede acceso si la sesión carece de permiso.

Campos

Campo	Regla visible
Key	Obligatorio; letra minúscula inicial, después minúsculas, números o guion bajo; 3-64 caracteres
Label	Nombre obligatorio, hasta 80 caracteres
Scope	Alcance ofrecido; formulario estático muestra Tenant
Tenant	Workspace obligatorio
Description	Explicación opcional, hasta 500 caracteres
Permissions	Selección desde catálogo disponible

La clave no es una contraseña. Use una convención estable como `operador_laboratorio` y una etiqueta descriptiva. La descripción debe explicar tareas, sin datos personales ni excepciones secretas.

Diseño y revisión

1. Liste tareas necesarias: consultar, crear, modificar o administrar.
2. Consulte **Permissions** y distinga lectura de gestión; no infiera que una contiene la otra.
3. Cree el rol en **Roles**, elija tenant y asigne el conjunto mínimo.
4. Guarde y revise. Asigne el rol a una cuenta sintética autorizada desde Users.
5. Compruebe el camino permitido y el rechazado. Verifique rol activo y cierre la sesión de prueba al terminar.

Modificar un rol usado por varias personas es un cambio compartido. Una sola pantalla no prueba todo el alcance. Mantenga una matriz de tarea, permiso y resultado esperado que permita revisar cambios posteriores.

POC sintético

El laboratorio necesita consultar cuentas y auditoría sin crear workers. Prepare una cuenta ficticia y un rol con capacidades de lectura disponibles para esas tareas. Compruebe Users y Audit dentro del alcance esperado y ausencia de gestión de Cloud workers. Si falta una lectura, añada únicamente el permiso identificado y repita.

La evidencia debe mostrar nombres sintéticos y resultados de autorización, nunca sesiones ni tokens. No use una cuenta de plataforma para representar menor privilegio: ocultaría errores. Repita la comprobación tras cambiar contexto de workspace cuando la cuenta de prueba tenga más de una membresía autorizada.

Errores, recuperación y límites

Ajuste una clave inválida antes de enviar. Si tenant no aparece, confirme alcance delegado; no introduzca identificadores manuales para sortear el selector. Ante rechazo revise la sesión y rol activo del administrador que realiza la acción.

Eliminar un rol tiene confirmación propia. Revise asignaciones y efecto sobre usuarios: no asuma que puede eliminar cualquier rol ni que se sustituirá automáticamente. Un error del servidor no se resuelve quitando verificaciones. Para recuperar acceso tras un cambio incorrecto utilice una vía administrativa autorizada y documente la corrección, sin compartir credenciales.

Separe permisos técnicos de responsabilidades organizativas. La aplicación valida capacidades, pero el operador debe justificar el motivo del cambio. Revise periódicamente roles que ya no correspondan al trabajo actual y retire capacidades mediante el proceso aprobado.

Subscription & billing: contratos, catálogo y licencias

Objetivo y autorización

El módulo tiene cinco pestañas: **Subscription**, **Licenses**, **Commerce**, **Catalog** y **Offers**. Leer requiere `billing.view`. Gestionar requiere `billing.manage`; las mutaciones del catálogo global exigen además el rol activo Platform Administrator. Un usuario de facturación tenant no puede asignarse un precio negociado ni modificar términos globales.

Diferencie suscripción, factura, licencia y orden comercial. Cambiar un registro no demuestra automáticamente el resultado de los demás. Las pruebas de este manual son de sandbox, sin cobros ni activaciones reales.

Subscription y facturación

El editor de suscripción expone plan y estado. El perfil de facturación incluye nombre legal, email, identificación fiscal, país y dirección. La factura permite número, importe, moneda, vencimiento, estado y descripción; el cambio de estado tiene diálogo propio.

Revise unidades y moneda antes de confirmar. La dirección de facturación no es la identidad de acceso. Verifique la factura seleccionada y vuelva al inventario tras modificarla. No marque una factura pagada para resolver un problema de acceso: su estado debe corresponder al proceso administrativo autorizado.

Licenses

Emitir recoge identificador manual, factura, activaciones máximas y vencimiento. Activar utiliza clave e identificador de activación; revocar requiere confirmación. No copie claves reales a capturas o PDFs. Revise vínculo con factura y vigencia antes de emitir. Revocar una licencia no debe presentarse como devolver dinero: ese efecto no se deduce del formulario.

Commerce, Catalog y Offers

Entidad	Contrato
Product	Identidad global y presentación
Plan	Identidad estable
PlanVersion	Revisión con estados draft, published y retired
Price	Vinculado a versión; términos económicos se congelan al usarse
TenantOffer	Oferta tenant referida a versión y precio globales
Order / Subscription	Instantánea contractual de términos aplicables

Edita términos en draft y publique tras revisión. Para cambiar condiciones usadas cree una revisión/precio nuevos, sin reescribir historial. Las ofertas no duplican catálogo por tenant. Incluyen vigencia, moneda y descuento o importe negociado; superposiciones incompatibles pueden rechazarse. Las fechas incluyen zona horaria. El catálogo Commerce

admite PEN y USD. El formulario administrativo Invoice ofrece USD, PEN y EUR; esa opción adicional de factura no amplía las monedas del catálogo Commerce.

Procedimiento y POC sintético

En sandbox prepare producto de demostración, plan estable y revisión draft. Revise términos, publique y cree precio correspondiente. Asigne una oferta a un tenant sintético autorizado y compruebe su catálogo efectivo. Registre la versión utilizada. Cree después una revisión nueva: el ejercicio debe demostrar que una orden anterior conserva su instantánea.

El checkout muestra precio, cantidad, método de pago, tipo de cliente, tipo/número de documento, nombre legal, email, país y dirección. No envíe tokens ficticios a un proveedor real. Documento con fixtures o sandbox autorizado; deténgase antes de un cobro real.

Validaciones y recuperación

El modelo almacena importes en unidades monetarias menores enteras; tasas y descuentos utilizan puntos básicos. No copie directamente un decimal visible a una API sin respetar el contrato. La versión técnica del almacenamiento no es una revisión comercial ni la release de aplicación.

Conflictos de versión o transición devuelven 409: recargue, compare y reintente la decisión correcta. Un 403 exige revisar rol activo y alcance. No existe API para eliminar historial comercial. Si una orden antigua carece de términos verificables requiere revisión comercial; no complete automáticamente con valores actuales.

Para investigar discrepancias compare factura, suscripción y orden en su contexto. No repita checkout sólo porque la UI tarda en actualizar: una operación del proveedor puede haber ocurrido. El estado definitivo debe verificarse mediante el flujo comercial autorizado.

Audit: investigar acciones con contexto

Objetivo y acceso

Audit muestra actividad administrativa para reconstruir qué acción se solicitó y cuál fue su resultado. Navegar requiere `admin.audit.view`; la información queda limitada al alcance permitido. Auditoría no es el log de ejecución de un Mapping ni una copia de datos transformados. Para investigar una ejecución utilice Monitor en Dova y relacione sólo identificadores permitidos.

Confirme tenant y rol activo antes de buscar. La ausencia de un registro visible no prueba que nunca ocurrió una acción: puede haber una restricción de alcance, un filtro o una diferencia entre el registro consultado y la operación investigada.

Leer la evidencia

La tabla presenta registros mediante el componente común de datos. Use controles visibles de búsqueda, ordenación, columnas y paginación según estén habilitados. Examine tipo de acción, objeto, contexto temporal y resultado expuestos. No complete campos ausentes mediante suposiciones.

Situación	Interpretación
Solicitud registrada	Se recibió una intención; no prueba éxito final
Resultado completado	El flujo registró finalización según su contrato
Fallo registrado	Revisar motivo y estado actual antes de reintentar
Eventos cercanos	Pueden ser etapas o reintentos, no necesariamente efectos duplicados

En acciones cloud se distingue solicitud de resultado. La operación del proveedor y la limpieza del registro Dova tienen un orden propio. Conserve la secuencia al elaborar conclusiones.

Procedimiento de investigación

1. Formule la pregunta: por ejemplo, si terminó una modificación de rol o falló una retirada.
2. Acote tenant y periodo autorizados. No mezcle casos independientes.
3. Localice el evento y lea su detalle. Busque resultado correspondiente cuando haya varias etapas.
4. Consulte el módulo del objeto para confirmar estado actual. Auditoría histórica y estado actual responden preguntas distintas.
5. Documente referencias mínimas: clase de evento, momento y resultado. Evite datos personales, secretos y payloads de otros tenants.

Si debe compartir evidencia, prepare un resumen con valores sintéticos o debidamente minimizados. Conserve el vínculo interno autorizado al caso original fuera del documento público. La explicación debe distinguir lo observado de lo inferido.

POC sintético

En un entorno local aislado cambie la etiqueta de un rol de laboratorio y compruebe el resultado en **Roles**. Busque la actividad en **Audit** usando los controles ofrecidos. Repita una acción rechazada por permiso insuficiente mediante fixture autorizado y compare la evidencia disponible.

Para cloud, los tests de API ofrecen una expectativa precisa: fallo del proveedor conserva worker y no registra éxito de eliminación. Explique ese resultado sin provocar fallos en una VM real. Use nombres como `rol_revision_demo`. La prueba debe concluir con estado final verificado, no sólo con una captura del mensaje de confirmación.

Errores, recuperación y límites

Si la tabla no carga revise mensaje y sesión. Un 403 no se resuelve cambiando tenant manualmente. Si no encuentra evento, revise filtros, periodo y módulo. No efectúe un segundo cambio sólo para comprobar que aparece en auditoría.

Este capítulo no promete retención indefinida, firma criptográfica del historial, exportación forense ni campos ausentes en UI. Esas garantías necesitan contrato específico. Para investigación formal preserve evidencia mediante el procedimiento autorizado; una captura aislada no sustituye ese proceso.

Cuando dos eventos parezcan contradictorios, confirme su orden y alcance. Una solicitud fallida seguida de reintento exitoso no implica pérdida de auditoría. Evite usar únicamente el último mensaje visible para describir toda la operación.

Registrations: revisión de solicitudes

Objetivo y permisos

Registrations revisa solicitudes de workspace y registra aprobación o rechazo. Ver requiere `admin.registrations.view`; decidir requiere `admin.registrations.manage` y alcance aplicable. Decidir no equivale a provisionar VM, crear automáticamente tenant ni emitir licencia. La UI indica que provisión de tenant y usuario permanece separada.

Revise identidad del solicitante presentada en la fila: nombre, email y organización permiten confirmar la solicitud correcta. No copie información personal a ejemplos publicados ni use solicitudes reales para probar el flujo.

Completar la decisión

Desde el menú seleccione **Approve** o **Reject**. El diálogo muestra solicitante, motivo y casilla explícita de confirmación. El cierre y Cancel permiten salir sin enviar una nueva decisión.

Elemento	Regla
Solicitante	Verificar antes de confirmar
Reason al rechazar	Obligatorio, de 1 a 1.000 caracteres
Reason al aprobar	Opcional, máximo 1.000 caracteres
Confirmación	Obligatoria para enviar
Guardado en curso	Evita solicitudes duplicadas simultáneas

El motivo debe ser mínimo y útil, sin contraseñas, documentos completos ni detalles sensibles innecesarios. Aprobar una solicitud equivocada y corregirla después no sustituye revisar identidad antes del envío.

Procedimiento

1. Localice una solicitud pendiente y abra la revisión.
2. Lea identidad completa, especialmente con nombres y organizaciones largos.
3. Elija resultado adecuado. Para rechazo escriba un motivo claro del requisito incumplido.
4. Marque confirmación y envíe una vez. Espere sin abrir otra revisión de la misma solicitud.
5. Compruebe estado actualizado y continúe procesos separados de alta/provisión cuando correspondan.

El cliente incluye revisión esperada del registro e identificador de intento. Reintentar idéntica decisión conserva identificador; cambiar contenido genera un intento distinto. No edite estos valores ni construya peticiones para saltar concurrencia.

POC sintético

Use una solicitud local de Persona de Demostración, `solicitante@example.test`, organización Laboratorio de Integración. Abra Reject e intente enviar sin motivo: debe validarse. Complete motivo sintético, deje confirmación sin marcar y compruebe que no se envía. Finalmente confirme en el fixture y observe resultado.

Simule 422: el diálogo conserva contexto para corregir/reintentar y devuelve foco a la acción cuando procede. Simule 409: bloquea envío y guía al cierre y actualización, sin insistir sobre revisión anterior. Pruebe Tab y Escape con un nombre largo a 320px; la identidad y confirmación deben seguir legibles. Ningún paso del POC debe enviar decisiones reales.

Errores y recuperación

Ante validación corrija el dato señalado sin perder de vista el solicitante. Con 409 cierre, actualice Registrations y revise estado reciente antes de decidir. Otro administrador puede haber actuado mientras el diálogo estaba abierto.

Si falla conectividad después de enviar, compruebe estado antes de iniciar otra decisión. Cerrar no garantiza cancelación de una petición recibida. No recree la solicitud para ocultar conflicto. Mantenga el motivo de revisión trazable y separe el resultado de cualquier alta posterior.

Si la acción no está disponible, confirme estado actual y permisos. La ausencia puede ser una protección del ciclo de vida, no un fallo visual. No use una cuenta más privilegiada sin comprobar primero por qué no procede la revisión.

Developer Tools: CLI, plataformas y API keys

Objetivo y acceso

Developer Tools reúne Dova CLI, instrucciones iniciales y API keys. Navegar en Administrator requiere `admin.access`; cada operación de claves/API conserva sus validaciones. Descargar una herramienta no concede acceso a un tenant ni autoriza comandos administrativos.

La vista comparte componente con Dova. Seleccionar plataforma cambia distribución recomendada e instrucciones relacionadas. Revise el sistema de destino, no el del navegador desde el que consulta la guía.

Secciones

Sección	Uso
Descarga principal	Elegir plataforma y distribución publicada
Available platforms	Comparar disponibilidad y metadatos
Quick start	Seguir instrucciones de la versión publicada
API keys	Revisar o crear credenciales según autorización
Release notes	Identificar cambios de distribución

No todas las plataformas están disponibles en cada release. Firma, checksum y disponibilidad son propiedades distintas. Una distribución marcada sin firma no debe describirse como firmada. Compare integridad con el hash del artefacto seleccionado.

Instalación controlada

1. Abra Developer Tools y espere catálogo y sesión.
2. Seleccione plataforma y compruebe coherencia entre enlace principal y Quick start.
3. Descargue y compare integridad con checksum publicado.
4. Lea instrucciones de esa distribución; no reutilice comandos antiguos sin contrastarlos.
5. Configure acceso mediante el mecanismo documentado y alcance mínimo. Empiece por una lectura permitida.

No se inventan subcomandos en este capítulo. Copie comandos desde Quick start de la release verificada o referencia CLI. En automatizaciones documente versión y variables requeridas, manteniendo secretos fuera del repositorio.

POC sintético

En un fixture publique dos entradas ficticias Windows/Linux con enlaces locales y hashes sintéticos. Seleccione Linux y verifique enlace e instrucciones; vuelva a Windows y repita. Esto valida navegación, no instala binarios ni crea API keys reales.

Para un laboratorio funcional posterior utilice tenant aislado y credencial limitada. Registre sólo resultado, versión y recurso sintético. Si muestra configuración use `<API_KEY_DE_LABORATORIO>` como marcador textual, nunca como valor que enviar a producción. Los datos de ejemplo deben permitir reconocer el alcance sin revelar información de una cuenta real.

Un ejercicio útil consiste en separar tres comprobaciones: descarga disponible, archivo íntegro y operación autenticada autorizada. Aprobar una de ellas no implica aprobar las demás. Mantenga evidencias independientes para que el diagnóstico no atribuya a la herramienta un error de permisos o de red.

Errores, recuperación y límites

Si no aparece una distribución revise catálogo y error de carga. No adivine URL ni nombres de archivo. Si difiere el hash descarte la descarga y obtenga el artefacto correcto mediante soporte; no continúe ignorándolo.

Un 401 o 403 requiere revisar credencial, tenant y permisos. No genere una clave más potente como primera solución. Las claves son sensibles incluso en capturas temporales; custodia y retirada deben seguir el flujo soportado y la política organizativa.

La página dispone contenido en flujo normal; algunas tablas pueden necesitar desplazamiento horizontal. En PDF conserve comandos completos y copiables, sin cortar argumentos largos ni sustituir guiones por caracteres tipográficos. Verifique rutas relativas y enlaces de descarga al integrar este capítulo en el portal, porque el manual puede vivir bajo `/dova/` y la aplicación bajo otro origen.

Preferences: políticas, autenticación y comunicaciones

Objetivo y alcance

Preferences reúne **Notifications**, **Authentication**, **Limits**, **Email delivery** y **Data Cleanup**. Navegación vinculada a `etl.configuration.view`; cada guardado mantiene la autorización de su API. Hay tenant settings y platform settings: lea el alcance antes de editar. Un cambio global puede afectar a más personas que el workspace visible.

Notifications y plantilla de ejecución

Configure feedback y canales según necesidad: posición, duración y cantidad visible; in-app, email, webhook, Slack y Teams; eventos de ejecución, worker, conexión, límites y seguridad. También existen severidades, roles destinatarios, umbrales, deduplicación, escalado, digest y horario silencioso con zona horaria.

Delivery history distingue enviados, fallidos, suprimidos o reconocidos. Generar evento no garantiza recepción externa. Endpoints webhook son sensibles: no mostrarlos en capturas.

Execution email template abre el editor de contenido. Use variables ofrecidas; no invente placeholders ni confunda SMTP con cuerpo. Revise diseño y preview antes de guardar. El preview seguro no debe ejecutar scripts ni cargar contenido activo externo; imágenes remotas pueden aparecer como marcadores. Compruebe después en un buzón de laboratorio, porque cada cliente de correo puede renderizar diferente.

Authentication

La pantalla contiene Google, Microsoft, Okta, Auth0, GitHub y GitLab. Habilitar proveedor requiere guardar la configuración autorizada y después usar Test provider sobre los valores persistidos, no sólo marcar una casilla. Esa prueba de disponibilidad o discovery no sustituye un inicio de sesión completo con la cuenta de laboratorio. Client ID y secreto son distintos. No publique secretos ni URLs privadas.

Google self-registration, cuando se habilita, permite el onboarding de identidades verificadas según la política de workspace individual y trial administrado; no concede administración. Conserve una vía administrativa autorizada antes de cambiar autenticación y pruebe en laboratorio. La sesión ya abierta no demuestra que una configuración nueva permita iniciar sesión.

Limits

Configuration scope separa el alcance de edición. Categorías: **Workers and capacity**, **Executions and objects**, **Retention**, **API and budget**, **Security and access**. Cada fila muestra descripción, valor y unidad. Distinga herencia de política de modificaciones locales. **Save changes** comienza deshabilitado sin cambios.

Modifique por un motivo concreto, revise unidades y guarde; reabrir confirma persistencia. No ponga un valor extremo para ocultar una incidencia. Límites comerciales, técnicos y de seguridad pueden actuar simultáneamente.

Email delivery

Transporte SMTP de plataforma para activaciones y notificaciones: habilitación, host, puerto, seguridad, usuario, contraseña, nombre/email remitente, reply-to y destinatario de prueba. Configure transporte autorizado y pruebe con buzón controlado. Guardar no confirma entrega; revise prueba e historial. No pruebe con destinatarios reales de clientes ni publique credenciales en el manual.

Data Cleanup

Empiece por inventario ETL del tenant. Revise categorías, bloqueos y alcance; confirme con texto y reconocimiento. No utilice limpieza para resolver un fallo visual ni retirar una VM. Respete ejecuciones activas y retención. Conserve evidencia autorizada y compruebe inventario final después del éxito. Una operación de limpieza no equivale a eliminar la identidad del tenant.

POC, errores y recuperación

En un tenant sintético configure feedback visible, evento de ejecución fallida y horario silencioso de laboratorio. Provoque el evento mediante fixture local y observe si el resultado corresponde a la política. Edite plantilla con texto ficticio, revise preview y restaure baseline. Pruebe SMTP en sandbox separado, sin confundir ese resultado con validación visual.

Ante fallo conserve mensaje y corrija campo. Tras conflicto recargue baseline antes de repetir. Si cambia de pestaña con edición pendiente atienda descarte. Una conexión correcta no valida destinatarios, contenido y permisos de todas las comunicaciones. No desactive MFA ni controles de tenant para facilitar recuperación de acceso.

Operar un catálogo comercial versionado y ofertas por tenant

Objetivo y autoridad

El catálogo global define productos, planes, revisiones y precios una sola vez. Una oferta de tenant referencia esas entidades; no duplica el catálogo. Esta separación permite cambiar la oferta futura sin reescribir contratos anteriores. Las lecturas requieren `billing.view`. Las escrituras administrativas requieren `billing.manage` y el rol activo Platform Administrator; una membresía secundaria inactiva no basta.

Este capítulo describe el modelo implementado y sus contratos. Antes de usarlo en una instalación, verifique que el catálogo y su migración se publicaron. Un documento local o un formulario visible no prueba que una migración de datos haya terminado.

Identidades y campos relevantes

Entidad	Campos/estado que debe controlar
Product	Identidad, presentación, disponibilidad y versión de concurrencia
Plan	Identidad estable y relación con draft/publicación
PlanVersion	Revisión comercial, version, draft/published/retired
Price	Versión vinculada, términos económicos, moneda y disponibilidad
TenantOffer	Tenant, versión, precio, moneda, vigencia, estado, descuento o importe negociado
Order/Subscription	Snapshot contractual de precio, impuestos, límites y derechos

revision comercial y version de concurrencia no significan lo mismo. Tampoco equivalen a `CATALOG_VERSION`, que identifica evolución técnica de almacenamiento, ni a la release de Dova.

Los importes se representan en unidades monetarias menores enteras y porcentajes en puntos básicos. Como ejemplo didáctico, 12.300 unidades menores de PEN representan S/123,00; 1.000 puntos básicos representan 10 %. Verifique siempre la semántica del campo concreto y no copie importes de pantalla sin conversión correcta.

Publicar nuevas condiciones

1. Revise **Catalog** y confirme el producto y plan estables.
2. Cree o abra una versión draft autorizada. Compare términos con la revisión anterior y documente qué cambiará.
3. Edite únicamente draft. Revise límites, derechos y datos comerciales antes de publicar.
4. Publique usando la versión de concurrencia actual.
5. Cree un Price vinculado a la publicación correcta. Revise moneda y condiciones antes de ofrecerlo.

Una revisión publicada no se corrige sobrescribiendo términos históricos. Para cambios comerciales cree otra revisión. Un precio utilizado congela condiciones y vínculo; sustitúyalo por uno nuevo cuando necesite cambiarlos. Retirar disponibilidad no elimina el historial de uso.

Asignar una oferta

En **Offers**, seleccione el tenant autorizado y referencias correctas de versión/precio. Revise vigencia con zona horaria, moneda y descuento o importe negociado. El inicio de vigencia es inclusivo y el fin exclusivo. Ofertas activas solapadas para la misma combinación tenant/producto/moneda/intervalo pueden rechazarse.

Compruebe el catálogo efectivo del tenant después de guardar. No use el catálogo global como evidencia de que una oferta particular está vigente. La visibilidad y elegibilidad también dependen del contexto de acceso. Un usuario de facturación del tenant no puede negociarse a sí mismo un precio mediante parámetros enviados desde el navegador.

Contratos y cambios concurrentes

Al crear una orden o suscripción se conserva una instantánea contractual. Activar después no debe volver a leer términos actuales para sustituir lo acordado. Si cambia el catálogo entre creación y activación, compare la instantánea original, no sólo la pantalla vigente.

Los endpoints de administración utilizan `version` para concurrencia. Un 409 exige recargar y revisar qué cambió; no cambie el número manualmente para hacer pasar una edición obsoleta. Un 400 indica petición inválida y un 403 alcance no autorizado. No se proporciona una API de eliminación de historial comercial.

POC sintético

Use un producto de demostración y un tenant aislado. Publique revisión A y precio ficticio en PEN; asigne oferta con una ventana temporal inequívoca. Genere una orden simulada con esa revisión. Publique revisión B con términos diferentes y compruebe que la orden A conserva los originales. Pruebe además una oferta solapada y una edición con versión antigua: ambas deben rechazarse según el contrato.

La prueba no cobra dinero ni usa tarjetas reales. No invente tokens para un proveedor productivo. Separe validación del modelo, renderizado de UI y autorización de cualquier pago real.

Migración y recuperación

Instalaciones antiguas requieren migración explícita; no se ejecuta automáticamente al iniciar. La herramienta local de migración no autoriza mutar producción. Una orden legacy con snapshot incompleto necesita revisión comercial; no rellene historia con defaults actuales. Ante resultado incierto de proveedor conserve identidad e intención y use reconciliación autorizada, evitando repetir efectos financieros.

Limpieza ETL y lifecycle de tenants: elegir la operación correcta

Objetivo y alcance

Eliminar objetos ETL, extender un trial, suspender acceso y cancelar una suscripción son operaciones distintas.

Preferences → **Data Cleanup** elimina datos ETL de un tenant seleccionado preservando su identidad y configuración de acceso. No borra una VM cloud ni sustituye un procedimiento de eliminación sujeto a retención.

La ruta administrativa aplica permisos de tenant y exige además `etl.configuration.manage` para la limpieza. La autoridad se valida en el servidor, no sólo en el selector. Para extender trial se exige rol activo Platform Administrator con `billing.manage`. Antes de actuar confirme tenant, finalidad y mecanismo apropiado.

Qué elimina y qué preserva Data Cleanup

La pantalla identifica Mappings, Sessions, Workflows, Connections, credenciales, runs, Repository y preferencias ETL del tenant como categorías de eliminación. Indica que preserva usuarios, tenant, roles, billing y auditoría global administrativa. Revise siempre el inventario actual; esta lista explica el alcance, no garantiza que cada categoría contenga registros.

El selector **Tenant**, **Refresh inventory** y **Clean tenant ETL data** forman el recorrido. El botón destructivo permanece bloqueado hasta contar con condiciones válidas. El diálogo exige la confirmación indicada y reconocimiento del efecto. No realice el procedimiento como prueba en un workspace con datos reales.

Procedimiento de limpieza

1. Confirme autorización y necesidad de eliminar datos ETL, no sólo historial de una ejecución.
2. Seleccione tenant y pulse **Refresh inventory**. Revise categorías, conteos y cualquier bloqueo.
3. Obtenga y verifique el respaldo autorizado que corresponda a su operación; el formulario no debe describirse como backup automático de todo el tenant.
4. Abra **Clean tenant ETL data**, compare identidad y frase de confirmación, y acepte sólo el alcance revisado.
5. Envíe una vez y espere el resultado. El cliente utiliza la revisión esperada del inventario.
6. Actualice inventario y compruebe categorías eliminadas y entidades preservadas. Registre la conclusión sin exportar información sensible.

El servidor serializa la operación con mutaciones de Mappings, orquestación, jobs y conexiones. Una revisión obsoleta o conflicto exige actualizar; no sustituya `expectedRevision` manualmente. Una respuesta fallida no autoriza borrar archivos directamente.

Restricción de workspaces individuales administrados

Los tenants individuales con suscripción administrada no admiten esta limpieza genérica. El servidor exige un flujo separado que considere retención. No cambie tipo de tenant ni desactive la gestión de suscripción para habilitar el botón. La restricción evita confundir borrado ETL masivo con derechos y obligaciones del lifecycle individual.

Estados y continuidad

Estado efectivo	Comportamiento relevante
trial	Operaciones contratadas durante vigencia
trial_expired	Acceso seguro de lectura, exportación y billing
active	Operaciones de la suscripción vigente
past_due	Lectura segura y billing; puede reactivarse tras pago confirmado
suspended/cancelled	Restricciones comerciales; nunca saltar suspensión administrativa o de seguridad

El modo de lectura no permite crear/editar/borrar negocio, ejecutar, despachar scheduler, enrollment/claims nuevos ni crear o rotar API keys. Hay excepciones de contención autorizadas, como cerrar sesión o terminar una ejecución ya activa, que no autorizan trabajo nuevo. No interprete lectura disponible como permiso para reconstruir el tenant.

Extender trial sin recrear

En **Subscription & billing** → **Subscription** → **Extend trial**, el valor predeterminado es siete días adicionales. Puede elegirse un número de 1-365 días o una fecha/hora personalizada. El diálogo muestra zona local del navegador y almacena instantes UTC. Para trial expirado los días parten de aceptación; para trial vigente, de su vencimiento.

La fecha personalizada debe superar ahora y vencimiento actual, dentro del máximo permitido. El motivo es obligatorio, hasta 500 caracteres. Se conservan tenant, suscripción, inicio original, objetos e historial; no se reinician cuotas ni se conceden nuevos derechos. Trials pagados, cancelados, suspendidos administrativamente o inconsistentes no son elegibles. Un legacy sin propiedad verificada requiere revisión separada.

POC y recuperación

En un fixture aislado genere un tenant de laboratorio y objetos ficticios. Compruebe inventario, rechazo con revisión obsoleta y preservación de usuarios tras limpieza autorizada. En otro fixture pruebe extensión de trial y verifique que identidad y cuotas no cambian. No mezcle ambos ejercicios: un tenant individual administrado debe rechazar limpieza genérica.

Tras error de red revise estado antes de repetir. La extensión idéntica reintentada no debe sumar dos veces. Ante conflicto recargue metadatos y decida otra vez. La activación por pago conserva tenant y objetos; no recree workspace como solución a una incidencia comercial.

Plantillas de ejecución, SMTP y políticas de notificación

Separar las tres capas

El contenido de un correo, su transporte SMTP y la política que decide enviarlo son configuraciones diferentes.

Execution email template define asunto y cuerpos; **Email delivery** configura transporte de plataforma; **Notifications** selecciona eventos, canales y condiciones. Un preview correcto no prueba entrega y una conexión SMTP correcta no demuestra que un evento cumpla la política.

Las rutas de plantillas y preferencias conservan permisos `etl.notifications.view` o `etl.notifications.manage` según lectura o modificación. Guardar y probar transporte SMTP requiere Platform Administrator. Verifique también tenant y, para plantillas con alcance de carpeta, el contexto aplicable. No use una cuenta de plataforma para simular permisos de un operador limitado.

Campos y eventos de plantilla

El contrato acepta `enabled`, `events`, `subject`, `preheader`, `htmlBody`, `textBody` e `inheritTenant`. Los eventos de ejecución permitidos son `succeeded`, `failed` y `cancelled`; debe elegirse al menos uno. El cuerpo de texto no es opcional: permite una alternativa legible al HTML.

Campo/límite	Contrato revisado
Subject	Obligatorio, hasta 160 caracteres, sin saltos de línea
Preheader	Hasta 180 caracteres, sin saltos de línea
Text body	Obligatorio, hasta 20.000 caracteres
HTML	Sanitizado y acotado a 1.500.000 bytes
Imágenes	Hasta 5 en total. Sólo las imágenes embebidas data: tienen límite de 256 KiB por imagen y 1 MiB total; para las remotas se valida HTTPS, sin afirmar un tamaño de descarga medido

No interprete estos máximos como recomendación de diseño. Un correo corto con enlaces pertinentes es más útil que incorporar logs completos o información sensible en el cuerpo.

Variables válidas

Las variables se escriben como `%NOMBRE%`. El catálogo incluye:

Grupo	Variables
Identidad	TENANT_NAME, TENANT_ID, WORKFLOW_NAME, WORKFLOW_ID, RUN_ID
Estado y tiempo	STATUS, STARTED_AT, COMPLETED_AT, DURATION
Métricas	INPUT_ROWS, OUTPUT_ROWS, FILTERED_ROWS, REJECTED_ROWS, SESSION_COUNT

Grupo	Variables
Ejecución	WORKER_NAMES, FILE_NAMES, LOCATIONS, FAILURE_REASON, EXECUTION_CONTEXT, TRIGGER, ENVIRONMENT
Presentación/enlace	DOVA_LOGO_URL, CLOUDATA_LOGO_URL, EXECUTION_URL

Ejemplo de contenido sintético, no petición a producción:

Asunto: Resultado %STATUS% de %WORKFLOW_NAME%
Ejecución: %RUN_ID%
Filas de salida: %OUTPUT_ROWS%
Detalle autorizado: %EXECUTION_URL%

Variables desconocidas se rechazan. No utilice sintaxis inventada ni trate el cuerpo como código ejecutable. El sanitizador permite un subconjunto de HTML y estilos, bloquea construcciones activas y controla enlaces/imágenes. El editor representa imágenes remotas de forma segura; su ausencia en preview no autoriza relajar CSP.

Procedimiento de edición

1. Abra **Preferences** → **Notifications** → **Execution email template**.
2. Revise alcance e herencia antes de editar. No sobrescriba una plantilla compartida pensando que sólo cambia un correo.
3. Seleccione eventos, escriba asunto y preheader e inserte variables del catálogo.
4. Mantenga equivalencia útil entre HTML y texto. Compruebe contraste, títulos, enlaces y lectura estrecha en preview.
5. Guarde y vuelva a cargar para verificar persistencia. Pruebe después el correo en un buzón de laboratorio autorizado.

Transporte SMTP

En **Email delivery** revise enabled, host, port, security, username, password, from name, from email y reply-to. El destinatario de prueba debe estar bajo su control. No use emails de clientes para verificar configuración. La contraseña se proporciona por el flujo seguro disponible, nunca en capturas o ejemplos públicos.

Un test fallido puede deberse a conexión, autenticación, seguridad o política del servidor remoto. Conserve el mensaje seguro; no cambie simultáneamente todas las opciones. Una prueba de mensaje debe comprobar recepción y legibilidad, además de la aceptación inicial del transporte.

Notificaciones y POC

Configure canales, eventos, severidades, roles destinatarios, umbrales, deduplicación, digest y horario silencioso con zona horaria. Observe **Delivery history**: sent, failed, suppressed y acknowledged no son equivalentes. Un evento suprimido por política no debe diagnosticarse automáticamente como fallo SMTP.

En fixture local genere una ejecución ficticia fallida, renderice ambas versiones del cuerpo y pruebe una variable desconocida. En sandbox separado envíe a un buzón controlado y compruebe asunto, texto, enlace y ausencia de datos sensibles. Documente cada resultado por separado. Ningún ejemplo debe incluir claves SMTP, endpoints secretos, tokens ni datos de ejecuciones reales.

Configurar SSO y completar MFA sin ampliar permisos

Objetivo y autoridad

Preferences → **Authentication** configura proveedores de inicio de sesión. El servidor exige Platform Administrator para guardar, sustituir secretos y probar proveedores. Acceder a Preferences por `etl.configuration.view` no basta para modificar autenticación global. La respuesta de configuración indica si el usuario puede gestionar; no dependa exclusivamente de que aparezca un botón.

SSO identifica a una persona; RBAC determina qué puede hacer. MFA añade verificación del acceso conforme al flujo de cuenta. No suponga que habilitar un proveedor crea usuarios, asigna roles o elimina la verificación multifactor. Prepare una vía administrativa autorizada de recuperación antes de cambiar una integración y pruebe con identidades de laboratorio.

Proveedores y campos

Proveedor	Configuración visible relevante
Google	Habilitación, Client ID, sustitución de client secret y callback autorizado
Microsoft	Habilitación, Client ID, tenant del proveedor, secreto y callback
Okta	Client ID, Issuer URL HTTPS, secreto y callback
Auth0	Client ID, Issuer URL HTTPS, secreto y callback
GitHub	Client ID, secreto y callback; identidad con email primario verificado
GitLab	Client ID, Issuer URL, secreto y callback; GitLab.com o self-managed

La interfaz muestra **Authorized callback URL** como sólo lectura y permite copiarlo. Registre esa URL exacta en la aplicación del proveedor, sin adivinar rutas ni intercambiar dominios. Para secretos existe **Replace client secret**: dejarlo vacío conserva el actual conforme al formulario. Un valor de configuración visible no autoriza publicar el secreto ni su ubicación de custodia.

Procedimiento de configuración

1. Confirme rol activo de plataforma y el proveedor que desea integrar.
2. Prepare una aplicación autorizada en el proveedor según el procedimiento de su organización. No utilice una aplicación ajena o personal como sustituto temporal.
3. Copie el callback mostrado por Dova y configure la integración correspondiente en el proveedor.
4. Introduzca Client ID, issuer o tenant cuando proceda y reemplace el secreto sólo si es necesario.
5. Guarde los ajustes y el reemplazo de secreto mediante el flujo del proveedor, y confirme su persistencia. Después use **Test provider**: la prueba utiliza la configuración guardada, no los campos aún editados.

6. Realice un inicio nuevo con la cuenta de laboratorio. Verifique identidad, workspace y rol final, además del resultado del proveedor.

Test provider comprueba disponibilidad o discovery según el proveedor; no valida por sí solo el inicio de sesión completo ni la validez integral del client secret. Un resultado correcto tampoco prueba que toda cuenta externa esté autorizada. GitHub requiere email primario verificado que coincida con un usuario Dova activo según el texto de la UI. Revise cada política; no generalice el autoregistro de Google a los seis proveedores.

Autoregistro Google y lifecycle

Google self-registration es una política global independiente. El contrato de onboarding individual utiliza identidad verificada para crear el workspace privado individual y trial administrado cuando aplica; no concede acceso administrativo. No describa el texto de ayuda histórico sobre tenant predeterminado como garantía de destino. La identidad y el alcance finales deben verificarse con el comportamiento del onboarding vigente.

Habilitar esta opción no convierte cuentas existentes ni renueva trials automáticamente. El modelo preserva identidades pagadas y aplica condiciones comerciales al tenant creado. Una denegación por lifecycle no debe corregirse asignando un rol de plataforma.

MFA y recuperación

El flujo de autenticación ofrece **Verification or recovery code**. Para configuración inicial muestra QR y clave de setup; registre el factor en su autenticador de forma privada y envíe el código requerido. Los códigos de recuperación son de un solo uso y deben custodiarse en un gestor de contraseñas. Nunca capturarlos para el manual, soporte público o un POC compartido.

Trust this device es una elección explícita del usuario, no un mecanismo para desactivar MFA globalmente. Use **Use another account** si seleccionó la identidad equivocada. Si no dispone del factor, utilice recuperación autorizada; no modifique almacenamiento ni políticas para saltar la verificación.

POC, errores y diagnóstico

En laboratorio pruebe un usuario existente autorizado, otro no autorizado y un intento con código incorrecto. Compruebe que sólo el caso permitido entra al workspace esperado. Registre resultados sin códigos, cookies ni tokens. Pruebe nuevamente tras cerrar sesión: la sesión administrativa abierta no valida el nuevo inicio.

Ante fallo revise callback exacto, issuer/tenant, habilitación y correspondencia de identidad. Si el proveedor responde pero Dova rechaza acceso, examine cuenta, membresía y rol. No repita sustituciones de secreto sin identificar la causa. Mantenga separados un error de conexión del proveedor, MFA y una denegación de autorización.

Instalar workers en Linux, Windows y cloud

Elegir el recorrido correcto

Hay dos recorridos distintos: registrar capacidad en un servidor existente o provisionar una VM mediante **Administrator** → **Cloud workers**. El primero instala Dova Worker Agent en infraestructura ya disponible; el segundo utiliza conexión cloud y perfil. Ninguno asigna automáticamente todas las Sessions existentes. Antes de empezar documente tenant, sistema operativo, capacidad autorizada y responsable de la infraestructura.

Esta guía describe contratos revisados en fuente. Las versiones de paquetes citadas son las que documenta el README local; confirme el catálogo publicado antes de descargar. Los comandos de instalación actúan sobre el servidor donde se ejecutan: no son comandos para probar directamente en producción. Realice el POC en un servidor de laboratorio autorizado.

Linux: requisitos e instalación

Requisitos documentados: Python 3.10 o posterior, venv, systemd, util-linux con flock, acceso de administrador y HTTPS saliente al servicio Dova y al registro de paquetes o mirror autorizado. La capacidad debe cumplir los requisitos configurados. El instalador comprueba requisitos; no instala silenciosamente otro Python ni cambia el sistema operativo.

El paquete documentado es Linux 1.3.5. Después de obtenerlo y verificarlo:

```
# Laboratorio Linux; directorio del paquete extraído.
sudo sh execution-worker/install.sh --check
sudo sh execution-worker/install.sh
```

En Dova, **Register worker** genera el enrollment para el sistema elegido. Defina nombre y concurrencia permitida, copie el comando generado y entregue el token separado mediante el prompt oculto. No lo añada a argumentos, historial ni documentación. La configuración generada tiene permisos restringidos.

```
# Sólo después de enrollment correcto en el servidor de laboratorio.
sudo systemctl enable --now dovaetl-execution-worker
sudo systemctl status dovaetl-execution-worker
```

La concurrencia representa de 1 a 64 unidades de slot. Un trabajo normalmente utiliza una unidad; slotWeight puede reservar varias. Cuatro slots admiten cuatro trabajos de peso uno o dos de peso dos, no cuatro trabajos independientemente del peso.

Windows Server

Se requiere Windows Server de 64 bits soportado, PowerShell 5.1 o posterior y Python 3.10+ de 64 bits instalado para toda la máquina, accesible por la cuenta de servicio. Un alias de Windows Store o Python instalado sólo en el perfil del usuario no es adecuado.

El README documenta paquete Windows 1.3.3. Extraiga el paquete, abra PowerShell elevado en su directorio y seleccione el ejecutable real instalado:

```
# Ejemplo de ubicación; sustituir por el Python de la máquina de laboratorio.
.\execution-worker\install-windows.ps1 -Python 'C:\Program Files\Python312\python.exe'
```

No deshabilite globalmente la política de ejecución para instalar. Genere enrollment seleccionando **Windows Server** y use el prompt oculto. Tras inscripción correcta:

```
Set-Service -Name DovaWorkerAgent -StartupType Automatic
Start-Service -Name DovaWorkerAgent
Get-Service -Name DovaWorkerAgent
```

El servicio usa una cuenta virtual dedicada, no una sesión interactiva. Configuración, logs rotados y scratch se separan de releases inmutables. Los Job Objects controlan recursos y descendientes; esos límites no deben describirse como sandbox para ejecutar código arbitrario no confiable.

Cloud: connection, profile y provisioning

En **Connections** prepare identidad federada, scope y regiones autorizadas; verifique las instrucciones específicas del proveedor. En **Profiles** configure vCPU, memoria, disco y ubicación. Luego, en **Workers**, seleccione tenant, conexión, perfil, nombre, slots, región y zona cuando corresponda. Revise compartición y estimación opcional antes de crear.

Observe **Provisioning history** hasta distinguir VM creada de agente registrado. Un timeout del navegador no demuestra que el proveedor canceló la creación. Consulte la operación antes de repetirla; duplicar solicitudes puede generar infraestructura adicional.

Verificación y POC

Use worker-documentacion y una transformación CSV pequeña con datos ficticios. Verifique versión instalada, Online, heartbeat reciente y recursos. Seleccione el worker en la Session o política de ejecución, ejecute el ejemplo y consulte Monitor. Registre resultado esperado, filas obtenidas y ubicación autorizada del archivo.

Si no está online revise servicio, conectividad y logs sin copiar secretos. Si está online pero no es elegible, revise capacidad, capacidades requeridas, política y tenant. Enrollment no configura permisos de bases de datos ni transfiere archivos. Para actualizar use el instalador y sus mecanismos de preservación de identidad/rollback documentados; no copie una configuración sensible entre máquinas.

Migrar referencias y retirar un worker con evidencia

Alcance

El asistente de Dova 6.0.2 conserva validaciones de tenant, permisos, propiedad, actividad y dependencias. No permite forzar una eliminación cuando se desconoce el estado de la VM.

El objetivo es trasladar referencias de ejecución a capacidad compatible y retirar el origen cuando corresponda. No se migran automáticamente archivos, bases de datos, secretos ni el sistema operativo. Antes de comenzar confirme `admin.cloud_workers.manage`, tenant de origen, destino autorizado y ventana operativa.

Inventariar y preparar

Desde el worker abra **Reassign references** o **Dependency backups**. Revise referencias actuales y el alcance antes de elegir destino. El modal de deprovision también presenta recursos cloud, referencias Dova y ejecuciones. No confunda un contador vacío con escaneo incompleto: un inventario incompleto bloquea acciones.

Hallazgo	Preparación necesaria
Worker predeterminado	Seleccionar otro destino predeterminado autorizado
Jobs en cola o activos	Resolver actividad según su contrato antes de retirar
Sessions y Workflows	Evaluar compatibilidad y referencias publicadas
Conexiones o perfiles de credenciales	Revisar custodia y compatibilidad del destino
Compartición/transición	Completar retiro o transición conforme al contrato
Provider unavailable	Restaurar acceso y obtener observación válida

Use **Drain worker** cuando necesite dejar de asignar trabajo nuevo y esperar el actual. Deshabilitar planificación y detener VM son operaciones distintas; no corte infraestructura para sustituir una espera de trabajos activos.

Preview y apply

1. Seleccione origen, destino y referencias en el flujo disponible.
2. Genere preview. Lea cambios, advertencias y bloqueos, no sólo el número de referencias.
3. Si cambia destino, selección o configuración, vuelva a generar preview. El plan anterior deja de representar la intención.
4. Aplique únicamente un plan permitido y compruebe el resultado y backup generado.
5. Actualice impacto del origen antes de continuar con retirada.

El servidor exige `planToken` del preview y `requestId` estable para apply. Recalcula el plan dentro de bloqueos y transacción; si cambiaron referencias o destino, rechaza y exige otro preview. El identificador de petición no puede reutilizarse con otra intención. Estos valores los gestiona la interfaz: no deben modificarse para superar un conflicto.

Antes de escribir cambios se guarda un snapshot de las proyecciones actuales. Las definiciones versionadas conservan historial; el resultado incluye número de cambios y referencia del backup. Un backup de dependencias no es una imagen de VM ni un respaldo de todos los datos.

Resolución de Sessions

El asistente ofrece **Replace worker**, **Apply distribution** y **Disable Sessions** según estrategia. Reemplazar exige un destino compatible con todas las Sessions afectadas. Distribuir requiere asignar cada Session exactamente una vez a un destino compatible. Deshabilitar cambia el comportamiento de esas Sessions y sus Workflows: revise el efecto antes de elegirlo como salida rápida.

Si existen ejecuciones o cambia el conjunto durante la operación, el servidor puede bloquearla. No borre referencias de almacenamiento manualmente. Tras restauración o reasignación desde un backup, vuelva a validar compatibilidad y publicación; el backup no garantiza que el destino actual sea idéntico al anterior.

Stop, impacto fresco y deprovision

Cuando la VM está running y procede detenerla, la opción **Stop and deprovision** pide confirmación explícita. Envía stop con su propia clave, obtiene impacto fresco y sólo continúa si `canDeprovision` es verdadero. Delete utiliza otra clave. Si stop falla, no envía delete; si sigue stopping o aparece una dependencia, solicita resolver y actualizar.

El registro Dova se retira después de confirmación de ausencia del proveedor y comprobación final de referencias. Un fallo preserva el registro para diagnóstico/reintento. **Clean up Dova record** corresponde a ausencia confirmada; `unavailable` nunca equivale a `absent`. Cerrar y reabrir obliga a consultar estado; cerrar no revoca un efecto ya recibido por el proveedor.

POC y recuperación

En fixture local cree una Session sintética en `worker-origen-demo` y un destino compatible. Verifique preview, apply, backup y desaparición de la referencia original. Simule cambio concurrente y compruebe rechazo del plan obsoleto. Simule error de proveedor y confirme conservación del registro sin éxito de eliminación en auditoría.

En una incidencia real, preserve mensajes seguros e historial, actualice estado y reintente mediante la UI. No anuncie ahorro cloud hasta confirmar retirada del recurso y revisar el proveedor.

Conectores, configuración y autenticación

Alcance y criterio de compatibilidad

Una Connection reúne dirección, metadatos, método de autenticación y custodia de credenciales. Un componente de Mapping selecciona esa conexión y define la operación sobre un recurso. El Worker ejecuta la operación con sus capacidades y permisos efectivos. Son tres contratos distintos: crear una conexión no garantiza que un proveedor admita todas las modalidades de lectura, escritura o recuperación.

Use esta matriz para preparar los datos del formulario [Connections](#). Los identificadores entre comillas de código permiten reconocer configuraciones e interpretar errores. Un asterisco indica un campo marcado como obligatorio en el catálogo del formulario; otros campos pueden resultar obligatorios por autenticación, protocolo o validación del proveedor. Los puertos son valores iniciales, no instrucciones para abrir redes indiscriminadamente. El endpoint debe corresponder al entorno autorizado de la organización.

Matriz del catálogo de conexiones

Conector	Endpoint / puerto inicial	Metadatos específicos	Métodos declarados
PostgreSQL	Host / 5432	database, schema	username_password, client_certificate, aws_iam, azure_workload_identity, gcp_iam, kerberos
MySQL	Host / 3306	database *	username_password, client_certificate
MariaDB	Host / 3306	database *	username_password, client_certificate
Microsoft SQL Server	Host / 1433	database *, schema	username_password
Oracle Database	Host / 1521	database *, schema	username_password
Snowflake	Account host / 443	account, database *, schema, warehouse *, role	snowflake_key_pair, username_password, bearer_token
Google BigQuery	HTTPS API endpoint / 443	database, schema	service_account_json, oauth_client_credentials, workload_identity, attached_service_account
Databricks	Workspace host / 443	httpPath *, catalog *, schema	personal_access_token, oauth_client_credentials
Amazon S3	HTTPS S3 endpoint / 443	bucket *, region	access_key, aws_iam
Azure Blob Storage	Storage account URL / 443	container *, directoryId	oauth_client_credentials, sas_token, account_key, azure_managed_identity, azure_workload_identity
Google Cloud Storage	HTTPS API endpoint / 443	bucket *, project *	attached_service_account, workload_identity, service_account_json
Salesforce	Salesforce HTTPS URL / 443	apiVersion *	oauth_client_credentials, jwt_bearer
Dynamics 365	Dataverse HTTPS URL / 443	directoryId *	oauth_client_credentials, azure_managed_identity, azure_workload_identity
Odoo	Odoo HTTPS URL / 443	database *	api_key
REST API	Base HTTPS URL / 443	-	api_key, bearer_token, oauth_client_credentials, client_certificate, workload_identity, no_auth

Conector	Endpoint / puerto inicial	Metadatos específicos	Métodos declarados
SAP HANA	HANA host / 39015	database *, schema	username_password
SAP S/4HANA / ECC	SAP HTTPS URL / 443	-	bearer_token, basic_auth
Oracle NetSuite	NetSuite account URL / 443	account *	bearer_token
MongoDB	MongoDB host / 27017	database *, authSource, mongoScheme *	username_password
FTP / SFTP	File server host / 22	protocol *, rootPath *	username_password, key_pair
Teradata	Teradata host / 1025	database *, schema	username_password, ldap
Elasticsearch / OpenSearch	Cluster HTTPS URL / 443	-	api_key, basic_auth, bearer_token
Microsoft SharePoint	SharePoint site URL / 443	directoryId *, siteId *	oauth_client_credentials, azure_managed_identity, azure_workload_identity
ServiceNow	ServiceNow instance URL / 443	-	bearer_token, basic_auth
HubSpot	HubSpot API URL / 443	-	bearer_token

La matriz recoge todas las entradas declaradas por el formulario. La oferta efectiva de la instalación se obtiene del catálogo disponible y puede ser menor. Tampoco afirma que una identidad adjunta funcione fuera de su nube o que un controlador esté instalado en cualquier Worker. PostgreSQL figura en la interfaz; no debe interpretarse la lista del registro genérico de conectores como el único mecanismo de integración de PostgreSQL.

Interpretación de los métodos

Identificador	Información que prepara el administrador
username_password, basic_auth, ldap	Usuario y contraseña protegida; el usuario permanece como metadato.
client_certificate	Paquete de certificado y clave privada, con frase de paso cuando corresponda.
kerberos	Principal de servicio y entorno Kerberos disponible en ejecución.
aws_iam	Identidad IAM del runtime; en bases de datos se solicita usuario o rol. En S3 se usa la identidad adjunta del Worker.
azure_managed_identity, azure_workload_identity	Identidad de aplicación o administrada y permisos del recurso destino.
gcp_iam	Principal IAM para autenticación de base de datos.
snowflake_key_pair	Usuario Snowflake y clave privada protegida; el formulario recomienda PKCS#8 cifrada.
oauth_client_credentials	Client ID, secreto y audiencia o scope cuando la integración lo pide.
workload_identity	Identidad federada o rol y audiencia OIDC.
attached_service_account	Cuenta de servicio adjunta al entorno de ejecución.
service_account_json	Documento de cuenta de servicio bajo custodia protegida.
access_key	ID de acceso como metadato y clave secreta bajo custodia.
sas_token, account_key	SAS delimitado al recurso o clave de cuenta de almacenamiento Azure.
api_key	Nombre de cabecera o parámetro y valor secreto. Odoos no solicita ese nombre: la clave identifica al usuario.
bearer_token, personal_access_token	Token con permisos y vigencia adecuados. Snowflake solicita además su usuario.
jwt_bearer	Consumer key de Salesforce, usuario como audiencia y clave de firma protegida.
key_pair	Usuario y clave privada SSH; sólo corresponde al protocolo SFTP.
no_auth	Recurso que permite acceso sin credenciales; no concede permisos al destino.

El catálogo general de autenticación también define `oracle_wallet`, `oci_resource_principal`, `web_identity` e `instance_profile`. No están asignados como opciones independientes a las entradas de conexión de esta matriz; su existencia no implica que puedan seleccionarse para cualquier conector. Para Amazon S3, `aws_iam` contempla la identidad de runtime ya adjunta.

Variantes que cambian la preparación

En BigQuery, Database significa Project ID y Schema significa Dataset. En Oracle, Database se presenta como Service name. Databricks requiere HTTP path y Catalog. Snowflake solicita Database y Warehouse; Role y Schema completan el contexto. No intercambie un nombre de base, una ruta HTTP y un identificador de proyecto aunque todos sean textos.

Salesforce parte de API version v65.0; exige la URL My Domain y la configuración de la aplicación e integración correspondientes. Dynamics 365 solicita Directory (tenant) ID y deriva el scope del entorno. Odoos identifica explícitamente su API JSON-2 de versión 19 o posterior. SharePoint añade Site ID o ruta relativa del sitio. Estas diferencias pertenecen al contrato del conector, no a una opción genérica de SQL.

La entrada FTP / SFTP permite escoger protocolo. SFTP usa inicialmente puerto 22; FTP y FTPS, 21. La clave SSH desaparece como opción para protocolos distintos de SFTP. La casilla TLS se ajusta al protocolo: FTP no equivale a una conexión cifrada. Conserve la política de transporte exigida por su organización.

Custodia y preparación segura

Dova Worker Agent conserva el secreto en el entorno del cliente. Customer vault guarda una referencia al secreto de un almacén externo. Passwordless usa una identidad de ejecución sin almacenar un secreto reutilizable. Dova Vault es una modalidad de excepción con entrada cifrada que no vuelve a mostrar el valor. Managed reference identifica referencias administradas existentes.

Los proveedores de referencias reconocidos por el formulario incluyen AWS Secrets Manager, Azure Key Vault, Google Secret Manager, HashiCorp Vault, CyberArk Conjur y Kubernetes Secret. Cada uno valida su propia forma de referencia. Una URI bien formada no demuestra que el Worker pueda resolverla: revise identidad, red y autorización del almacén. Nunca sustituya una referencia por el valor del secreto en nombres, descripciones o archivos de evidencia.

El selector de autenticación depende de la custodia. Passwordless filtra los métodos marcados sin secreto; la modalidad protegida ofrece los métodos pertinentes. Si una combinación carece de métodos válidos, cambie la custodia o el conector conforme al diseño, sin forzar valores ocultos del formulario.

POC de conectividad y resultados esperados

Prepare un recurso aislado con dos registros sintéticos y una identidad limitada a lectura. Cree la conexión en el tenant de pruebas, complete endpoint y metadatos, seleccione la custodia y guarde. Ejecute la comprobación de conexión desde la interfaz. Registre conector, Worker, hora y resultado; omita material secreto. Después seleccione esa Connection desde el componente fuente correspondiente y lea exclusivamente el recurso de prueba.

El resultado esperado tiene dos niveles: la comprobación confirma las condiciones que el conector verifica; la lectura confirma acceso al objeto y proyección de sus dos registros. No considere aprobada una escritura porque la prueba de conexión haya pasado. Para evaluar un destino, utilice otro recurso descartable, pruebe el modo de escritura elegido y compare claves, cantidad y contenido. Esta secuencia es una práctica reproducible, no un resultado de conectividad ya obtenido en su entorno.

Diagnóstico y límites

Ante rechazo de autenticación, compruebe método, identidad, vigencia y permisos. Ante objeto inexistente, confirme catálogo, base, esquema o ruta. Ante timeout, revise primero el Worker y su acceso de red. Ante una opción de escritura ausente, consulte las capacidades efectivas del conector: el componente genérico no garantiza todos los modos para cada proveedor. No convierta un timeout en evidencia de credenciales incorrectas ni copie respuestas que puedan incluir datos del origen.

Continúe con [componentes de Mapping](#), [transformaciones](#) y [permisos por acción](#).

Componentes de entrada y salida de Mapping

Del diseño a la ejecución

Un Mapping transforma registros mediante un grafo de componentes. Las fuentes producen registros; las transformaciones proyectan, enriquecen o validan; los destinos escriben resultados. Una Session asocia el diseño a parámetros, archivos y ejecución. Un Workflow coordina Sessions y decisiones. Esta separación permite cambiar un archivo de entrada o un Worker sin convertir cada variación operativa en otro diseño.

Los componentes incorporados para archivos se distinguen de los que requieren un Worker compatible. Database, API y Object storage se declaran con `runtime compatible_worker`; seleccionar el nodo no instala automáticamente controladores, concede permisos ni crea una conexión. Consulte [conectores y autenticación](#) antes de configurar estos componentes.

Esta referencia enumera todas las fuentes y destinos del catálogo de Mapping. Los nombres técnicos ayudan a reconocer configuraciones; la interfaz muestra las etiquetas de la primera columna. Las opciones enumeradas son las del componente genérico. La conexión y el Worker pueden imponer un subconjunto más limitado.

Catálogo de fuentes

Componente / identificador	Campos y opciones declarados
Flat File input / <code>source_csv</code>	<code>format (csv, tsv, txt); delimiter; hasHeader; headerRowNumber; columns; headerMappings; encoding (utf-8, utf-8-bom)</code>
JSON input / <code>source_json</code>	<code>format (json, jsonl); recordPath; fieldMappings; encoding (utf-8, utf-8-bom)</code>
Excel input / <code>source_excel</code>	<code>workbookFormat (auto, xlsx, xlsm, xls); sheetMode (single, list, pattern, regex, all); sheetName; sheetNames; sheetPattern; sheetNameRegex; includeHiddenSheets; hasHeader; headerRowNumber; dataStartRowNumber; startColumn; endColumn; columns; headerMappings; skipEmptyRows; includeSheetName; sheetNameField; passwordProtected</code>
Database input / <code>source_database</code>	<code>connectionId; sourceMode (table, query); schema; objectType (table, view); objectName; sqlQuery; selectedColumns; fetchSize; queryTimeoutSeconds</code>
API input / <code>source_api</code>	<code>connectionId; method (GET); resourcePath; responsePath; fieldMappings; paginationMode (none, page, offset, cursor); pageSize; retries; timeoutSeconds</code>
Object storage input / <code>source_object</code>	<code>connectionId; resource; format (csv, jsonl); selectedColumns; delimiter</code>

Archivos delimitados y JSON

Flat File input interpreta CSV, TSV y texto delimitado. Configure el delimitador real y si la primera fila contiene nombres. Header row number es una posición física que comienza en uno; no confunda una cabecera desplazada con líneas de datos que deben conservarse. Si el archivo no trae cabecera, proporcione Field names. Header mapping relaciona posiciones o encabezados con campos lógicos y tipos. Encoding distingue UTF-8 y UTF-8 with BOM.

JSON input acepta un array JSON o JSON Lines. Record path permite ubicar el array dentro de un documento; Field mapping proyecta rutas anidadas a campos lógicos. Un JSON válido puede tener una estructura incompatible con la colección esperada. Prepare primero un documento pequeño cuyo número de registros conozca y confirme que la proyección obtiene los campos correctos.

Excel

Excel input contempla detección automática, XLSX, XLSM y XLS. Read sheets permite una hoja, lista, patrón, expresión regular o todas. Configure explícitamente hojas ocultas cuando deban incluirse. Header row number, Data starts at row, Start column y End column delimitan la región. Skip empty rows controla filas vacías; Include sheet name permite conservar procedencia en el campo configurado, inicialmente `_sheet_name`.

La opción Password protected indica que la contraseña se aporta en ejecución; no debe almacenarse dentro del Mapping. Leer un libro XLSM no constituye una autorización para ejecutar macros. Antes de incluir todas las hojas, revise que compartan estructura. Una hoja de notas o totales puede producir registros ajenos al conjunto esperado.

Fuentes remotas

Database input ofrece Table or view y SQL query de lectura. Seleccione conexión, esquema y objeto, o configure la consulta correspondiente. Fetch size y Query timeout delimitan aspectos operativos; no reemplazan filtros de negocio. Selected fields permite una proyección explícita. Diseñe consultas de lectura para un recurso autorizado y no utilice el nombre del campo SQL como promesa de soporte universal de dialectos.

API input declara GET, Resource path, Record path y Field mapping. La paginación puede ser None, Page number, Offset o Cursor. Page size, Retries y Timeout deben ser compatibles con el servicio. Una primera página correcta no demuestra que se haya leído la colección completa: el caso de prueba debe superar deliberadamente el tamaño de página.

Object storage input selecciona Connection y Object key. Sus formatos declarados son CSV y JSON Lines. Selected fields y CSV delimiter completan la lectura. La clave del objeto no es una ruta local del navegador. Compruebe el prefijo y el recurso final para evitar leer otro archivo con nombre semejante.

Catálogo de destinos

Componente / identificador	Campos y opciones declarados
Flat File output / <code>target_csv</code>	<code>format (csv, txt); fileName; delimiter; includeHeader; textQualifier (double, single, none); quoteMode (minimal, all); escapeCharacter; lineEnding (lf, crlf); encoding (utf-8, utf-8-bom); nullValue</code>
JSON output / <code>target_json</code>	<code>format (json, jsonl); fileName; pretty; includeNulls; lineEnding (lf, crlf); encoding (utf-8, utf-8-bom)</code>
Database output / <code>target_database</code>	<code>connectionId; schema; objectName; writeMode (append, replace, upsert); batchSize; commitIntervalRows; keyColumns; createIfMissing; queryTimeoutSeconds</code>
API output / <code>target_api</code>	<code>connectionId; method (POST, PUT, PATCH); resourcePath; writeMode (append, upsert); keyColumns; uniqueKeysEnforced; batchSize; successCodes; retries; timeoutSeconds</code>
Object storage output / <code>target_object</code>	<code>connectionId; resource; format (csv, jsonl); writeMode (replace); delimiter; idempotencyKey; ifAbsent; expectedEtag; expectedGeneration</code>

Representación y escritura

Flat File output define nombre, delimitador, cabecera, calificador, política de comillas, escape, fin de línea, codificación y representación de nulos. Estos parámetros forman parte del contrato de intercambio. Si el consumidor requiere CRLF o BOM, declare esa expectativa antes de comparar archivos. JSON output distingue array y JSON Lines, permite formato legible y decide si conserva campos nulos.

Database output declara Append, Replace y Upsert; las claves, creación de tabla, lotes, intervalo de commit y timeout completan la configuración. Confirme que la capacidad efectiva del conector admite el modo. Replace debe probarse sólo sobre una tabla de ensayo. Upsert requiere una definición correcta de claves; no es un mecanismo genérico para reparar duplicados históricos.

API output declara POST, PUT y PATCH, modos Append o Upsert, claves y una confirmación de unicidad impuesta por el proveedor. Success status codes parte de 200, 201, 202 y 204. Un código 202 puede representar aceptación asíncrona del servicio externo: no lo convierta por documentación en confirmación de procesamiento final. Los reintentos deben evaluarse junto con la semántica de idempotencia del destino.

Object storage output usa Conditional replace, exige Idempotency key y ofrece Create only when absent, Expected ETag y Expected generation. Esas condiciones protegen la intención de escritura frente a versiones del objeto. No retire una precondition sólo para convertir un conflicto en éxito: recupere el estado esperado y decida si la nueva escritura sigue siendo válida.

POC de contrato de archivo

Cree un CSV sintético con cabecera `id,name` y dos filas: `1,Ana` y `2,Luis`. Diseñe Flat File input → Flat File output, establezca delimitador coma y cabecera en ambos extremos, y guarde una Session con ese archivo de entrada. Ejecute en un entorno de pruebas autorizado. El resultado esperado conserva dos registros y los dos campos; puede diferir en comillas o fin de línea si cambia la configuración de salida.

Repita con un JSON Lines que contenga dos objetos equivalentes. Utilice JSON input y JSON output, seleccione JSON Lines y compruebe dos líneas de datos. Esta segunda variante separa errores de transporte de errores de representación. Los pasos describen una comprobación reproducible; el administrador debe registrar su resultado real antes de aceptar el contrato.

Validación, fallos y recuperación

Revise primero estructura del grafo, conexiones de nodos y campos obligatorios. Después compruebe nombres lógicos, tipos y parámetros. Si falla la lectura, conserve una muestra sintética equivalente y reduzca el problema a una fuente y un destino. Si falla la escritura, identifique si hubo efectos parciales antes de reintentar; no deduzca ausencia de cambios sólo porque el Workflow terminó con error.

Use [Monitor](#) para correlacionar ejecución y tareas, y [Sessions](#) para verificar archivos, perfil y Worker. Una ejecución pequeña valida forma y acceso, no capacidad para un volumen arbitrario. Amplíe gradualmente el conjunto y conserve un conteo esperado independiente.



Transformaciones y reglas de calidad

Objetivo y contrato de los registros

Las transformaciones integradas trabajan sobre los registros que llegan desde sus padres. Una operación puede cambiar campos, valores, cantidad de filas u orden. Antes de conectar un destino, declare cuáles de esas propiedades deben conservarse. La calidad no se limita a que el Mapping sea válido: un grafo correcto puede producir un resultado de negocio incorrecto si las claves o el tipo de unión no representan la relación real.

El catálogo siguiente contiene todos los componentes de categorías Transform y Quality. Sus identificadores son estables dentro de la configuración; las etiquetas son las mostradas al usuario. Los campos enumerados describen el contrato del componente, no funciones de un lenguaje de programación abierto. Para fuentes y destinos consulte [componentes de Mapping](#).

Catálogo completo

Componente / identificador	Categoría	Configuración declarada
Row filter / filter	Transform	field; operator (equals, not_equals, contains, starts_with, ends_with, gt, gte, lt, lte, not_empty); value
Field selector / select	Transform	fields
Field mapper / rename	Transform	mapping
Calculated field / derive	Transform	field; template
Convert type / convert	Transform	field; targetType (string, integer, number, boolean, date, datetime); outputField; onError (fail, null, keep, default); defaultValue
Clean text / clean	Transform	fields; operations
Replace values / replace	Transform	field; find; replacement; matchMode (contains, exact, starts_with, ends_with); caseSensitive
Split field / split	Transform	field; delimiter; outputFields; keepSource; trimParts
Concatenate fields / concat	Transform	fields; outputField; separator; skipEmpty
Join / join	Transform	joinType (inner, left, right, full); leftInputId; rightInputId; leftKey; rightKey; conditions; keyType (auto, string, integer, number, boolean, date, datetime); strategy (auto, hash, sorted); caseSensitive; trimStrings; nullsEqual; rightPrefix
Default value / defaults	Quality	field; value; when (null_or_empty, null, empty)
Limit rows / limit	Transform	count; offset
Deduplicate / dedupe	Quality	fields
Row sort / sort	Transform	field; direction (asc, desc)
Group aggregate / aggregate	Transform	groupBy; metric (count, sum, avg); field; outputField
Quality check / validate	Quality	field; maxNullPercent
Required fields / quality_required	Quality	fields; maxInvalidPercent
Type check / quality_type	Quality	field; expectedType (string, integer, number, boolean, date, datetime); allowEmpty; maxInvalidPercent
Pattern check / quality_pattern	Quality	field; pattern; allowEmpty; maxInvalidPercent

Componente / identificador	Categoría	Configuración declarada
Range check / quality_range	Quality	field; minimum; maximum; inclusive; allowEmpty; maxInvalidPercent
Allowed values / quality_domain	Quality	field; allowedValues; caseSensitive; allowEmpty; maxInvalidPercent

Proyección y enriquecimiento

Field selector conserva la lista de campos elegida. Field mapper relaciona nombres de origen con nombres de destino; úselo para establecer un esquema de salida reconocible por el consumidor. Revise colisiones antes de renombrar: dos conceptos distintos no deben terminar representados por un mismo nombre lógico.

Calculated field crea un campo a partir de una plantilla. `${customer}` toma el valor de una columna y `${param.region}` toma un parámetro. La sustitución produce texto; no interpreta una fórmula aritmética ni ejecuta JavaScript o SQL. Consulte [expresiones y variables](#) para distinguir esta plantilla de una condición de Workflow.

Convert type admite String, Integer, Number, Boolean, Date y Date and time. Output field vacío reemplaza el campo de origen; un nombre distinto permite conservar el valor inicial. On conversion error decide entre fallar, asignar nulo, conservar el original o usar un valor predeterminado. Conservar valores originales puede dejar una columna heterogénea: valide la salida antes de enviarla a una tabla tipada.

Limpieza de texto

Clean text aplica una lista ordenada de operaciones: trim, collapse_whitespace, lower, upper y remove_accents. Por defecto recorta extremos y colapsa espacios. Aplique una política coherente a ambos lados de una clave de unión. No elimine acentos de un nombre que deba conservarse legalmente sólo para facilitar una comparación; cree un campo normalizado separado cuando sea necesario.

Replace values distingue coincidencia por contenido, valor exacto, prefijo y sufijo; Case sensitive controla el tratamiento de mayúsculas. Split field divide por un delimitador y asigna Output fields, con opciones para conservar el origen y recortar partes. Concatenate fields define separador, campo de salida y omisión de valores vacíos. Ninguno de estos componentes debe confundirse con un analizador de expresiones regulares arbitrarias.

Cantidad, orden y agrupación

Row filter acepta Equals, Does not equal, Contains, Starts with, Ends with, comparadores mayor/menor con o sin igualdad, e Is not empty. Su lista de operadores no coincide con la de decisiones de Workflow. Tipifique números antes de comparar cuando el origen sea textual y valide los límites con registros exactamente iguales al umbral.

Limit rows combina Maximum rows y Rows to skip. Sirve para una ventana controlada, pero no convierte una muestra en una validación de toda la población. Row sort ordena por Field y Direction. Group aggregate agrupa por Group fields y calcula Count, Sum o Average, con Value field y Output field según corresponda. No documente medianas, percentiles ni múltiples métricas implícitas: no figuran en este catálogo.

Deduplicate configura Match fields. Es una regla de identidad, por lo que debe probarse con registros iguales en la clave y distintos en otros atributos. Confirme qué registro resulta antes de usarla para reconciliación de datos; no presuponga una política de “última actualización” que no haya modelado explícitamente.

Uniones

Join declara Inner, Left, Right y Full, padres izquierdo y derecho, claves y condiciones adicionales AND. Key type puede ser automático o un tipo explícito. Strategy distingue Automatic hash, Hash y Sorted merge. Case sensitive, Trim strings y Match nulls modifican la equivalencia de claves. Right collision prefix, inicialmente `right_`, permite distinguir nombres repetidos.

Compruebe unicidad y cardinalidad antes de unir. Dos coincidencias a la izquierda y tres a la derecha pueden multiplicar registros; ese resultado no es necesariamente un defecto del motor. Prepare casos sin coincidencia, con coincidencia única, con duplicados y con nulos. Un Left join debería preservar la intención de mantener registros izquierdos, pero la cantidad exacta depende de cuántas coincidencias produzca cada clave.

Reglas de calidad

Default value completa un campo cuando es nulo, vacío o cualquiera de ambos, según Apply when. Quality check controla porcentaje máximo de nulos. Required fields valida completitud de una lista. Type check valida el tipo declarado y puede permitir vacíos. Pattern check aplica un patrón. Range check define mínimo, máximo e inclusión de límites. Allowed values limita el dominio y ofrece sensibilidad a mayúsculas.

Los controles de porcentaje constituyen umbrales de aceptación; no describen una salida automática de cuarentena. No presente una regla como separación de filas válidas e inválidas si el diseño no la implementa. Establezca cero cuando ninguna infracción sea admisible y pruebe el comportamiento con una fila inválida conocida.

POC sintética de limpieza y control

Prepare un CSV con `id,name,amount`: dos registros válidos y un tercero con `amount` vacío. Conecte Flat File input → Clean text sobre `name` → Convert type sobre `amount` → Required fields sobre `id` y `amount` → Flat File output. Use `trim` y `upper`; configure conversión con Set null y calidad con Maximum invalid percent igual a cero.

La expectativa es un fallo de calidad debido al tercer registro, sin considerar esa ejecución una carga aceptada. Corrija el dato sintético de entrada, repita y verifique tres registros, nombres normalizados e importes numéricos compatibles con la salida. Si usa un destino remoto, haga esta prueba en un recurso descartable y compruebe efectos parciales antes de reintentar.

Como ampliación, prepare otra fuente con dos claves y una descripción, añada Join y compruebe manualmente el resultado esperado. Registre filas antes y después, claves sin coincidencia y duplicados. Este procedimiento es un ejercicio de aceptación para ejecutar en su entorno; no supone que un destino real ya haya sido validado.

Expresiones, parámetros y variables

Tres mecanismos distintos

Dova separa las plantillas de campos de Mapping, los parámetros tipados de ejecución y las condiciones JSON de Workflow. Cada mecanismo tiene sintaxis y alcance propios. Una plantilla `${field}` no es una condición de decisión; un valor booleano JSON no es la cadena `"true"`; un parámetro secreto es una referencia protegida, no una contraseña para interpolar en un archivo.

Use parámetros para representar variaciones de entorno y de tarea que no cambian la estructura del proceso. Use variables de Workflow cuando la definición de orquestación las contemple. Use condiciones para seleccionar ramas a partir de valores o estados disponibles. La guía de [Sessions](#) describe dónde se editan estos valores, y [Workflows](#) explica su relación con las tareas.

Definición de parámetros

Cada definición admite `name`, `type`, `description`, `required`, `sensitive`, `overridable`, `scopes`, `default`, `enum`, `minimum`, `maximum` y `maxLength`. El contrato admite hasta 100 definiciones, con nombres únicos. Un nombre comienza por letra, tiene hasta 64 caracteres y puede incluir letras, números, guion bajo, punto y guion. Los prefijos `sys.` y `dova.` están reservados.

Tipo	Valor aceptado y consideración
string	Cadena; longitud predeterminada máxima 4000 si no se define otra.
integer	Entero JSON; un booleano no se acepta como entero.
number	Número finito; no booleano, infinito ni NaN.
boolean	Booleano JSON <code>true</code> o <code>false</code> , no texto equivalente.
datetime	Texto ISO con zona horaria; se normaliza a UTC y precisión de segundos.
json	Valor JSON acotado a 32768 bytes por el normalizador.
secret	Referencia de secreto validada, no valor de secreto en claro.

Description está limitada a 300 caracteres. Max length admite valores entre 1 y 32768. Minimum no puede superar Maximum. Enum debe contener entre 1 y 50 candidatos, se normaliza según el tipo y no está disponible para JSON o Secret. Un parámetro marcado Sensitive debe usar el tipo Secret: no se convierte una cadena ordinaria en un secreto seguro mediante una etiqueta.

Las referencias Secret reconocen los esquemas `gcp-secret`, `aws-secretsmanager`, `azure-keyvault`, `vault` y `worker-secret`. Esta sintaxis pertenece a parámetros de orquestación; no debe mezclarse mecánicamente con las formas específicas que usa Customer vault en Connections. Resolver una referencia requiere la infraestructura y autorización correspondientes.

Capas y precedencia

Orden de aplicación	Capa	Uso conceptual
1	Default de la definición	Valor base cuando no existe una sustitución permitida.
2	global	Configuración general de la ejecución.
3	environment	Variación del entorno seleccionado.
4	worker	Variación asociada al entorno de ejecución.
5	task	Valor específico de la tarea.

Una capa posterior reemplaza el valor anterior dentro de las reglas de validación. Scopes limita las capas aceptadas y Overridable participa en el control de sustitución. Required exige que exista un valor después de resolver defaults y capas. El resultado conserva la procedencia del valor, lo que permite diagnosticar por qué no se aplicó el valor que esperaba el diseñador.

Por ejemplo, defina REGION como String con default NORTE. Una capa Environment que establezca SUR debe prevalecer sobre ese default; una capa Task permitida que establezca CENTRO debe prevalecer sobre ambas. No elimine restricciones de scope para resolver un error sin comprobar quién tiene autoridad sobre esa configuración.

Plantillas de Calculated field

La plantilla reconoce `${nombre}` para campos del registro y `${param.nombre}` para parámetros. La sustitución produce texto y limita el resultado a 4000 caracteres. Un identificador reconocido pero ausente se sustituye por cadena vacía. El nombre de campo se busca como clave del registro; escribir un punto no garantiza navegación por un objeto anidado.

Con una fila que contiene `name = Ana` y un parámetro `region = SUR`, la plantilla `Cliente ${name} / ${param.region}` produce `Cliente Ana / SUR`. La expresión `${amount} * 2` produce una representación textual con `* 2`; no calcula una multiplicación. Use los componentes de transformación adecuados y no introduzca código esperando evaluación dinámica.

El helper de valor parametrizado reconoce una referencia completa `${param.nombre}`. No implica que todos los campos de todos los conectores acepten interpolación parcial. Confirme la configuración del componente concreto antes de trasladar una plantilla de un campo a otro.

Condiciones JSON de Workflow

Una condición simple contiene `ref`, `op` y, salvo dos operadores, `value`. Las referencias admitidas son `parameters.nombre`, `variables.nombre`, `tasks.identificador.status` y `tasks.identificador.outputs.nombre`. No admiten acceso arbitrario a propiedades del servidor o código ejecutable.

Operador	Significado
eq, ne	Igualdad o desigualdad con el valor JSON esperado.
gt, gte, lt, lte	Comparación ordenada; tipos incompatibles producen falso.
in, not_in	Pertenencia o ausencia en una lista proporcionada como Value.
contains	Contención en una cadena, lista o diccionario del contexto.

Operador	Significado
exists	La referencia existe; no requiere Value.
truthy	Evalúa el valor lógico del dato; no requiere Value.

Si una referencia no existe, las comparaciones ordinarias producen falso. Exists permite distinguir ausencia de un valor existente pero vacío. In y Not in requieren que el valor esperado sea una lista. Los números deben seguir siendo números JSON; la comparación no promete conversión automática desde texto.

Los grupos all y any combinan entre 1 y 20 expresiones; not niega una expresión. Se rechazan profundidades superiores a cinco y árboles que superen 40 nodos. El valor de comparación tiene un límite de 4096 bytes. Estos límites favorecen condiciones revisables; divida decisiones demasiado complejas en tareas claras.

```
{
  "all": [
    {"ref": "parameters.RUN_MODE", "op": "eq", "value": "full"},
    {"ref": "tasks.load.status", "op": "eq", "value": "succeeded"}
  ]
}
```

La referencia load debe corresponder al identificador real de una tarea. Las salidas sólo pueden consultarse cuando el contexto las contiene. El ejemplo no declara una salida universal de conteos ni presupone nombres de campos de un conector.

Decisiones y estados

Los tipos de tarea son Start, Session, Decision, Timer y End. Los enlaces contemplan Success, Failure, Always y Expression. Una decisión requiere ramas True y False complementarias: una expresión y su negación directa. La interfaz construye la rama falsa como not de la verdadera; no escriba dos condiciones independientes que puedan resultar ambas verdaderas o ambas falsas.

Los estados terminales reconocidos incluyen Succeeded, Failed, Skipped, Cancelled y Terminated. Ready, Queued, Running, Waiting y Retry wait son estados activos. La preparación de una tarea considera sus padres terminales y la política de unión. Always no significa que una tarea pueda ignorar todos los requisitos de la orquestación.

POC y diagnóstico

Prepare un Workflow sintético con una Session que procese dos filas, una Decision y dos ramas de finalización. Defina RUN_MODE como String con valores permitidos full y sample. Use una condición de igualdad sobre parameters.RUN_MODE, ejecute primero con full y después con sample, y compruebe que cambia la rama activa manteniendo la complementaria.

Después pruebe un valor no incluido en Enum y un booleano donde se exige String. El resultado esperado es rechazo de validación, no conversión silenciosa. Estas pruebas deben realizarse en un entorno autorizado y registrar el resultado observado. Si una rama no se activa, revise tipo JSON, nombre técnico, capa efectiva, estado del padre y existencia de la referencia antes de modificar la lógica.



Permisos y acciones de Dova y Administrator

Cómo interpretar una autorización

La autorización depende del usuario autenticado, el tenant activo, el rol activo, sus permisos efectivos y las restricciones del recurso. Mostrar un módulo es una decisión de navegación; ejecutar una acción exige su comprobación correspondiente. Un usuario con varios roles asignados no debe suponer que puede utilizar simultáneamente todos sus privilegios sin seleccionar el contexto adecuado.

La suscripción y la elegibilidad de recursos también intervienen. Un permiso de ejecución no garantiza un Worker disponible, capacidad de plan o acceso a una conexión. Una respuesta de sólo lectura por suscripción inactiva no se resuelve añadiendo permisos indiscriminadamente. Esta referencia describe los permisos del catálogo y la correspondencia de rutas y acciones; las comprobaciones adicionales del recurso siguen siendo obligatorias.

Matriz de Dova

Superficie o acción	Consulta	Modificación u operación
Overview	<code>etl.overview.view</code>	No existe <code>etl.overview.manage</code> en el catálogo.
Domains and folders	<code>etl.domains.view</code>	<code>etl.domains.manage</code>
Mappings	<code>etl.mappings.view</code>	<code>etl.mappings.manage</code>
Sessions	<code>etl.sessions.view</code>	<code>etl.sessions.manage</code>
Workflows	<code>etl.workflows.view</code>	<code>etl.workflows.manage</code>
Ejecutar Workflow	Consulta del diseño según contexto	<code>etl.workflows.run</code>
Cancelar o terminar Workflow run	<code>etl.monitor.view</code>	<code>etl.workflows.terminate</code>
Rerun o Recover de Workflow run	<code>etl.monitor.view</code>	<code>etl.workflows.run</code>
Monitor	<code>etl.monitor.view</code>	<code>etl.monitor.manage</code> para gestión distinta de las operaciones específicas anteriores.
Connections, perfiles y proveedores de credenciales, catálogo	<code>etl.connections.view</code>	<code>etl.connections.manage</code>
Workers, Agents, Runtime y Execution profiles	<code>etl.runtime.view</code>	<code>etl.runtime.manage</code>
Workflow Scheduler	<code>etl.scheduler.view</code>	<code>etl.scheduler.manage</code>
Políticas, preferencias y configuración general	<code>etl.configuration.view</code>	<code>etl.configuration.manage</code>
Notification channels y notificaciones	<code>etl.notifications.view</code>	<code>etl.notifications.manage</code>
Plans and subscriptions	<code>billing.view</code>	Las operaciones administrativas requieren <code>billing.manage</code> y las restricciones de alcance aplicables.

Configuration puede aparecer si existe cualquiera de los permisos de consulta de Configuration, Runtime o Scheduler. Sus pestañas aplican reglas específicas. Esta alternativa no concede los otros permisos: un usuario autorizado para ver programación no recibe por ello capacidad de modificar políticas de ejecución.

La consulta de dominios también puede servir a los selectores de Mappings, Sessions, Workflows y Connections con sus respectivos permisos de lectura. No confunda esa consulta auxiliar con permiso para crear carpetas. Del mismo modo, la consulta de bloqueos de edición admite permisos de lectura de los tipos de artefacto; las mutaciones de bloqueos requieren permisos de gestión y las comprobaciones del recurso.

Matriz de Administrator

Superficie	Consulta	Gestión
Acceso a Administrator	<code>admin.access</code>	No sustituye los permisos de cada sección.
Overview e historiales de recursos	<code>admin.overview.view</code>	Los cambios de recursos usan sus permisos propios.
Users y presencia	<code>admin.users.view</code>	<code>admin.users.manage</code>
Tenants y limpieza de recursos del tenant	<code>admin.tenants.view</code>	<code>admin.tenants.manage</code>
Roles y catálogo de permisos	<code>admin.roles.view</code>	<code>admin.roles.manage</code>
Audit	<code>admin.audit.view</code>	No hay permiso de modificación de auditoría en el catálogo.
Cloud workers y dependencias	<code>admin.cloud_workers.view</code>	<code>admin.cloud_workers.manage</code>
Tenant registrations	<code>admin.registrations.view</code>	<code>admin.registrations.manage</code>
Billing, Plans y Commerce	<code>billing.view</code>	<code>billing.manage</code>
Plantilla de correo de ejecución y ajustes de notificación	<code>etl.notifications.view</code>	<code>etl.notifications.manage</code>

Los dos permisos de Registrations están declarados exclusivos de plataforma. Además, la gestión del catálogo comercial global y de ofertas de tenant comprueba simultáneamente rol activo de plataforma y `billing.manage`. Un rol de facturación de tenant no debe interpretarse como autorización para cambiar productos globales.

Roles incorporados

Rol	Composición base y límites relevantes
Platform Administrator	Comodín de plataforma; sujeto a controles de sesión y acciones.
Tenant Owner	Todos los permisos del catálogo salvo los dos exclusivos de Registrations.
Tenant Administrator	Administración de tenant, usuarios, roles, auditoría y cloud workers, consulta de billing y permisos ETL. No incluye <code>billing.manage</code> en su conjunto base.
Billing Administrator	Acceso y Overview administrativos, <code>billing.view</code> y <code>billing.manage</code> .
Security Administrator	Consulta administrativa de identidad y auditoría; consulta de Configuration, Connections y Runtime; consulta y gestión de notificaciones.
ETL Developer	Permisos ETL salvo <code>etl.monitor.manage</code> y <code>etl.workflows.terminate</code> . Incluye ejecución de Workflows.
ETL Operator	Permisos ETL de consulta, ejecución y terminación de Workflows y gestión de Monitor.
ETL Viewer	Permisos ETL que terminan en <code>.view</code> .
Public Trial	Subconjunto explícito de diseño y ejecución; no equivale a todos los permisos ETL.
Public	Consulta de billing y Overview ETL.
ETL Administrator (legacy)	Alias base de Tenant Administrator.

Estos son conjuntos incorporados, no una afirmación de que toda organización conserve exactamente esas asignaciones. Roles personalizados y asignaciones efectivas deben revisarse en Roles & permissions. Tampoco utilice la compatibilidad de roles del navegador como fuente de autoridad: el servidor obtiene y verifica los permisos de la sesión.

Public Trial incluye consulta y gestión de Domains, Mappings, Sessions, Workflows y Connections; ejecución de Workflows; consulta de Monitor; consulta y gestión de Runtime; y consulta de Billing y Overview. Su lista no incluye gestión de Scheduler, gestión de Monitor ni terminación de Workflows. La vigencia del acceso y el aislamiento del tenant siguen siendo condiciones independientes.

Del nombre de módulo a la operación

La interfaz llama Workflows a la orquestación de Sessions. Sus rutas de orquestación usan permisos `etl.workflows.*`. Existen rutas históricas con nombre workflows que corresponden al diseño de Mapping y usan `etl.mappings.*`, salvo la acción específica de ejecución. Para integrar o auditar, compruebe el contrato de la acción; deducir permisos únicamente de una palabra de la URL puede producir una conclusión incorrecta.

Las alternativas aceptadas por una ruta se evalúan como "cualquiera de estos permisos", cuando así está declarado. Después pueden existir validaciones más precisas sobre artefacto, tenant, suscripción o estado. Una matriz de permisos no debe utilizarse para omitir esas validaciones ni como promesa de una API pública estable.

POC de mínimo privilegio

En un tenant de laboratorio, prepare un Mapping, una Session y un Workflow con dos filas sintéticas. Asigne a una cuenta un rol de consulta equivalente a ETL Viewer. Compruebe que puede abrir los módulos permitidos y consultar el diseño, pero no guardar cambios ni iniciar una ejecución. Realice la comprobación con la sesión de esa cuenta, sin reutilizar credenciales administrativas.

Cambie al rol de operador autorizado y verifique que puede ejecutar el Workflow y consultar Monitor, sin adquirir gestión del diseño por esa sola asignación. Después use un rol de desarrollo para editar y validar el diseño; compruebe separadamente sus restricciones de terminación y gestión de Monitor. Registre cada acción, contexto y resultado sin copiar tokens ni datos personales.

El resultado esperado es una separación reproducible entre consultar, diseñar y operar. La prueba debe ejecutarse en el entorno de la organización; esta guía no afirma que las asignaciones de una cuenta concreta ya hayan sido verificadas. Si una acción falla, revise primero tenant y rol activos, después permiso efectivo, suscripción y estado del recurso. No responda a cualquier rechazo asignando el comodín de plataforma.

Relación con navegación y auditoría

Use [cuenta](#) para comprender el cambio de contexto y [Repository](#) para distinguir estructura y artefactos. Después de cambios sensibles, revise la auditoría disponible y confirme que el resultado pertenece al tenant correcto. Un botón oculto mejora la navegación, pero el control de acceso efectivo permanece en el servidor.