

Versión 9 Release 1

*IBM InfoSphere Optim
Utilización de Optim Designer*



Versión 9 Release 1

*IBM InfoSphere Optim
Utilización de Optim Designer*



Nota

Antes de utilizar esta información y el producto al que hace referencia, lea la información del apartado “Avisos” en la página 131.

Primera edición

Esta edición se aplica a la versión 2, release 2, modificación 3 de Optim Designer y a todos los releases y modificaciones posteriores mientras no se indique lo contrario en nuevas ediciones.

© Copyright IBM Corporation 1996, 2012.

Contenido

Capítulo 1. Utilización de InfoSphere

Optim Designer 1

Iniciación	1
Perspectiva Optim	1
Utilización de Optim Manager desde Optim Designer.	4
Datos de ejemplo	4
Soporte de base de datos	13
Funciones de accesibilidad	13

Capítulo 2. Gestión de conexiones de

origen de datos 15

Trabajar con un alias de almacén de datos	15
Definición de un alias de almacén de datos.	15
Edición de un alias de almacén de datos.	15
Conexión con un alias de almacén de datos	16
Trabajar con el repositorio de Optim	16
Conexión con el repositorio de Optim	16
Edición de una conexión de repositorio	16
Modificación de una conexión de repositorio	16
Asociación de un directorio Optim con el repositorio.	17
Transformar una solicitud en un servicio	17

Capítulo 3. Gestión de modelos de

datos 19

Trabajar con definiciones de acceso	19
Creación de una definición de acceso.	20
Gestión de tablas en una definición de acceso	20
Gestión de los criterios de selección	21
Gestión de relaciones en una definición de acceso	23
Gestión de variables en una definición de acceso	24
Gestión de listas de apuntar y disparar	25
Trabajar con relaciones de Optim	26
Reglas de compatibilidad para relaciones	28
Creación de una relación de Optim	30
Edición de una relación de Optim	30
Trabajar con claves primarias de Optim	31
Creación de una clave primaria.	32
Edición de una clave primaria	32
Trabajar con correlaciones de tablas	33
Creación de una correlación de tablas	34
Edición de una correlación de tablas	34
Trabajar con correlaciones de columnas	35
Reglas de compatibilidad para correlaciones de columnas	36
Creación de una correlación de columnas	37
Edición de una correlación de columnas.	38

Capítulo 4. Diseño de servicios de

gestión de datos 41

Trabajar con servicios de extracción	41
Creación de un servicio de extracción.	41
Edición de un servicio de extracción	42

Trabajar con servicios de conversión	44
Creación de un servicio de conversión	44
Edición de un servicio de conversión.	45
Trabajar con servicios de inserción.	45
Creación de un servicio de inserción	46
Edición de un servicio de inserción	46
Trabajar con servicios de carga	48
Creación de un servicio de carga	49
Edición de un servicio de carga.	50
Probar un servicio de gestión de datos	51

Capítulo 5. Enmascaramiento de datos 53

Aplicación de una función de enmascaramiento de datos	53
Funciones de enmascaramiento de datos.	53
Funciones de búsqueda	53
Función de reorganización	62
Función TRANS SSN	64
Función TRANS CCN	68
Función TRANS EML	71
Función TRANS COL	75
TRANS NID	77
Función de subserie	88
Función aleatoria	89
Función secuencial	90
Función de identidad o de serie	91
Función de secuencia de Oracle	91
Función de propagación de valor de clave primaria o foránea	92
Expresiones concatenadas	93
Expresiones numéricas	94
Funciones de literal y valor	94
Función de antigüedad	95
Función de moneda	98
Nombre de correo electrónico generado automáticamente	100
Nombre de correo electrónico formateado	101
Función de número aleatorio	101
Función de número secuencial.	101
Utilización de rutinas de salida	101
Escribir rutinas de salida	102
Rutina de salida estándar	103
Salida de formato de origen	105
Salida de formato de destino	107
Escribir procedimientos de correlación de columnas con el script de Lua	110
Funciones de Lua para procedimientos de correlación de columnas.	110
Limitaciones de procedimientos de correlación de columnas.	112
Ejemplo de procedimiento de correlación de columnas: procedimiento genérico	113
Ejemplo de procedimiento de correlación de columnas: búsqueda conmutada	114

Ejemplo de procedimiento de correlación de columnas: enmascaramiento de ID nacional . . .	117
--	-----

Capítulo 6. Gestión de datos 121

Examinar datos.	121
Edición de datos	121
Comparación de datos	122
Definición de una solicitud de comparación . . .	122
Edición de una solicitud de comparación . . .	123
Ejecución de un proceso de comparación . . .	123
Creación de tablas.	123

Capítulo 7. Gestión de datos con Optim Designer 125

Extracción de datos con Optim Designer	125
Creación de una carpeta Explorador de repositorios	125

Creación de una definición de acceso	125
Definición de criterios de selección	126
Creación de un servicio de extracción	127
Enmascaramiento de datos con Optim Designer . . .	127
Creación de una correlación de tablas	128
Edición de una correlación de tablas.	128
Creación de una correlación de columnas . . .	128
Aplicación de una función de enmascaramiento de datos	129
Adición de una correlación de columnas a una correlación de tablas	130
Creación de un servicio de conversión	130

Índice. 135

Capítulo 1. Utilización de InfoSphere Optim Designer

Utilice IBM® InfoSphere Optim Designer para definir modelos de datos, políticas de privacidad de datos, servicios de gestión de datos y almacenes de datos. Puede definir objetos en Optim Designer y compartirlos en el repositorio de Optim. Utilice el Explorador de repositorios para gestionar objetos en el repositorio. Utilice el Explorador de directorios para ver objetos y transformar solicitudes en un directorio Optim.

Servicios de gestión de datos

Utilice un servicio de gestión de datos para extraer, convertir, cargar o insertar datos. Un servicio utiliza definiciones de acceso para definir los datos que se van a extraer. Un servicio también utiliza correlaciones de tablas y correlaciones de columnas para correlacionar datos de origen y destino. También puede transformar las solicitudes de proceso de un directorio Optim en servicios de gestión de datos. Utilice Optim Manager en modalidad incluida para probar los servicios.

Enmascaramiento de datos

Puede enmascarar datos procesados por un servicio de gestión de datos. Utilice el enmascaramiento de datos para transformar datos como, por ejemplo, números de ID nacionales, número de tarjetas de crédito, fechas, valores numéricos e información personal. Utilice una correlación de columnas para especificar una función de enmascaramiento de datos o cree un procedimiento de correlación de columnas con un script LUA.

Definiciones de acceso

Utilice una definición de acceso para definir modelos de datos y subconjuntos de datos en esquemas coincidentes o diferentes. Una definición de acceso identifica criterios de selección y las relaciones utilizadas en un servicio de gestión de datos. Puede definir relaciones fuera de un DBMS y utilizar SQL para definir los criterios de selección.

Optim Manager

Puede abrir Optim Manager desde Optim Designer (modalidad incluida), que le permite probar los servicios de gestión de datos.

Iniciación

Para empezar, debe utilizar el Explorador de repositorios para conectarse a un repositorio de Optim.

Perspectiva Optim

En Optim Designer, la perspectiva Optim proporciona las herramientas necesarias para definir modelos de datos, servicios de gestión de datos y conectarse a un directorio Optim. La primera vez que se abre Optim Designer, la perspectiva Optim es la perspectiva predeterminada.

La perspectiva Optim incluye las siguientes vistas:

Explorador de repositorios

Utilice el Explorador de repositorios para definir alias de almacén de datos, definiciones de acceso, servicios de gestión de datos y otros objetos almacenados en un repositorio de Optim.

Explorador de directorios

Utilice el Explorador de directorios para conectarse a un directorio Optim.

Para abrir la perspectiva Optim, pulse **Ventana > Abrir perspectiva > Otras**. En la ventana Abrir perspectiva, seleccione **Optim**.

Explorador de repositorios

Utilice el Explorador de repositorios para definir alias de almacén de datos, definiciones de acceso, servicios de gestión de datos y otros objetos almacenados en un repositorio de Optim.

Cuando Optim Designer está conectado a un repositorio de Optim, el Explorador de repositorios incluye los siguientes objetos:

Alias de almacén de datos

Un alias de almacén de datos es un objeto definido por el usuario que está asociado con una base de datos. Cuando defina un alias de almacén de datos, proporcione parámetros que Optim utilizará para comunicarse con dicha base de datos. Estos parámetros incluyen el tipo y la versión del sistema de gestión de bases de datos (DBMS) y las propiedades para una conexión de cliente y para una conexión JDBC.

Un nombre de alias de almacén de datos sirve como calificador de alto nivel que le permite acceder a una base de datos específica para realizar las funciones solicitadas. Por ejemplo, en una definición de acceso, debe calificar el nombre de una tabla con un nombre de alias de almacén de datos. El alias de almacén de datos referenciado proporciona los parámetros necesarios para conectar con la base de datos donde reside la tabla.

Claves primarias de Optim

Una clave primaria es la columna o columnas que contienen valores que identifican de forma exclusiva cada fila de una tabla. Una tabla de base de datos debe tener una clave primaria para Optim para insertar, actualizar, restaurar o suprimir datos de una tabla de base de datos. Optim utiliza claves primarias que están definidas en la base de datos. Sin embargo, también puede definir claves primarias de Optim para complementar las claves primarias en la base de datos.

Relaciones de Optim

Optim utiliza las relaciones para determinar los datos que deben recuperarse de las tablas relacionadas y se basa en las relaciones definidas en la base de datos, si está disponible. Sin embargo, también puede definir relaciones de Optim para complementar las relaciones de la base de datos.

Definiciones de acceso

Utilice las definiciones de acceso para especificar las tablas, el cruce de relaciones y los criterios de selección para los datos que desea procesar.

Correlaciones de columnas

Una correlación de columnas proporciona las especificaciones necesarias para encontrar coincidencias con columnas o para excluir columnas del proceso en un servicio de gestión de datos. Los servicios de conversión, inserción y carga deben hacer referencia a una correlación de tablas, que puede hacer referencia a una o más correlaciones de columnas. Puede utilizar una correlación de columnas para definir transformaciones de datos con funciones o procedimientos de correlación de columnas.

Correlaciones de tablas

Utilice una correlación de tablas para definir especificaciones para correlacionar tablas de origen y destino de datos compatibles. Puede correlacionar tablas que tienen diferentes nombres, modificar nombres de tablas, excluir tablas de un proceso o incluir correlaciones de columnas para un mayor control sobre los datos.

Servicios

Utilice un servicio de gestión de datos para extraer, convertir, cargar o insertar datos. Un servicio utiliza definiciones de acceso para definir los datos que se van a extraer. Un servicio también utiliza correlaciones de tablas y correlaciones de columnas para correlacionar datos de origen y destino. Puede enmascarar datos aplicando una política de privacidad en una entidad procesada por un servicio. También puede transformar las solicitudes de proceso de un directorio Optim en servicios de gestión de datos. Utilice Optim Manager en modalidad incluida para probar los servicios.

Carpetas

Una carpeta Explorador de repositorios contiene servicios, definiciones de acceso, correlaciones de columnas y correlaciones de tablas. Utilice las carpetas para organizar estos objetos.

Creación de una carpeta Explorador de repositorios:

Utilice la ventana Nueva carpeta para crear una carpeta en el Explorador de repositorios.

Una carpeta Explorador de repositorios contiene servicios, definiciones de acceso, correlaciones de columnas y correlaciones de tablas. Utilice las carpetas para organizar estos objetos.

Para crear una carpeta Explorador de repositorios.

1. Dependiendo de si existe una carpeta, realice uno de los pasos siguientes:
 - a. Para crear una carpeta inicial, pulse **Pulsar para crear una nueva carpeta** en el Explorador de repositorios.
 - b. Para crear una carpeta adicional, pulse el botón derecho del ratón en la vista Explorador de repositorios y pulse **Nueva > Carpeta**.

Se abre la ventana Nueva carpeta.

2. Especifique un nombre de carpeta y pulse **Aceptar**.











La nueva carpeta se visualiza en el Explorador de repositorios.








Explorador de directorios

Utilice el Explorador de directorios para conectarse a un directorio Optim. Sólo se puede asociar un directorio Optim con el repositorio.

Utilice el Explorador de directorios para transformar solicitudes de proceso en servicios de gestión de datos.

Las siguientes definiciones están disponibles.

Icono	Definición
	definición de acceso
	solicitud de archivado
	columna
	correlación de columna
	solicitud de conversión
	ID creador
	alias DB
	solicitud de supresión
	solicitud de extracción
	solicitud de inserción

Icono	Definición
	solicitud de carga
	clave primaria
	relación
	solicitud de restauración
	tabla
	correlación de tabla
	variable

Utilización de Optim Manager desde Optim Designer

Utilice Optim Manager desde Optim Designer para probar los servicios de gestión de datos y los servicios de interoperatividad Optim. La utilización de Optim Manager desde Optim Designer también es conocida como Optim Manager en modalidad incluida.

Optim Manager es una aplicación web que puede utilizar para probar servicios. Optim Manager se visualiza en el navegador interno proporcionado por Optim Designer o en un navegador externo. Puede seleccionar el navegador que desea que Optim Designer utilice para Optim Manager pulsando **Ventana > Preferencias > General > Navegador web** en Optim Designer.

Abrir Optim Manager desde Optim Designer

Puede abrir Optim Manager probando un servicio. También puede abrir Optim Manager especificando el URL siguiente en un navegador web: `http://localhost:número_de_puerto/console`, donde *número_de_puerto* es el número de puerto asignado a Optim Manager. El número de puerto predeterminado es 60000.

Si el navegador muestra un mensaje acerca de una página no encontrada al abrir Optim Manager, es posible que se esté dando un conflicto de puerto y que deba cambiar el número de puerto de Optim Manager.

También puede abrir Optim Manager desde la barra de herramientas de Optim Designer.

Cambiar el número de puerto de Optim Manager

Para cambiar el número de puerto, debe editar la siguiente propiedad en el archivo `eclipse.ini` que se encuentra en el directorio de instalación de Optim Designer:

```
-Dorg.eclipse.equinox.http.jetty.http.port=número_de_puerto
```

donde *número_de_puerto* es el número de puerto nuevo de Optim Manager. Si Optim Designer está abierto, debe reiniciar la aplicación para aplicar el número de puerto nuevo.

Datos de ejemplo

Optim proporciona datos de ejemplo para enmascarar datos así como orígenes de datos de origen y destino predefinidos.

Tablas y estructura de la base de datos de ejemplo

Como grupo, las tablas de base de datos incluyen información sobre clientes y pedidos, con las instrucciones de envío. Las tablas de ejemplo también incluyen información sobre ventas e inventario. Existen pequeñas diferencias en los tipos de datos, en función del DBMS que utilice para instalar las tablas de ejemplo. En el diagrama siguiente se muestra la estructura básica de la base de datos de ejemplo.

Como grupo, las tablas de base de datos incluyen información sobre clientes y pedidos, con las instrucciones de envío. Las tablas de ejemplo también incluyen información sobre ventas e inventario. Existen pequeñas diferencias en los tipos de datos, en función del DBMS que utilice para instalar las tablas de ejemplo.

También se instala un conjunto adicional de tablas con la base de datos de ejemplo. Las tablas del conjunto adicional tienen los mismos nombres que las tablas del primer conjunto, con el sufijo “2” añadido. Las cuatro tablas de este conjunto adicional son las siguientes:

- OPTIM_CUSTOMERS2
- OPTIM_ORDERS2
- OPTIM_DETAILS2
- OPTIM_ITEMS2

Las tablas en el conjunto adicional no contienen datos. Se utilizan para mostrar los recursos de Optim.

Tabla OPTIM_SALES:

La tabla OPTIM_SALES identifica cada vendedor por nombre, número de ID y gestor.

La tabla OPTIM_SALES tiene las siguientes columnas:

SALESMAN_ID

CHAR; hasta 6 caracteres; no puede contener nulos.

FIRST_NAME

VARCHAR; hasta 15 caracteres; no puede contener nulos.

LAST_NAME

VARCHAR; hasta 15 caracteres; no puede contener nulos.

NATIONALITY

VARCHAR; hasta 30 caracteres

NATIONAL_ID

VARCHAR; hasta 30 caracteres

PHONE_NUMBER

VARCHAR; hasta 20 caracteres; no puede contener un valor nulo.

EMAIL_ADDRESS

VARCHAR; hasta 70 caracteres; no puede contener nulos.

AGE SMALLINT; no puede contener nulos; tiene un valor predeterminado.

SEX CHAR; 1 carácter; no puede contener nulos; tiene un valor predeterminado.

TERRITORY

VARCHAR; hasta 14 caracteres; no puede contener nulos.

MANAGER_ID

VARCHAR; hasta 6 caracteres.

Claves primarias

La columna SALESMAN_ID es la columna de clave primaria.

Relaciones con otras tablas

La tabla OPTIM_SALES es padre de:

- La tabla OPTIM_CUSTOMERS, a través de una clave foránea de la columna SALESMAN_ID.
- La tabla OPTIM_MALE_RATES, a través de una relación controlada por datos de Optim en la columna AGE cuando SEX = 'M'.
- La tabla OPTIM_FEMALE_RATES, a través de una relación controlada por datos de Optim en la columna AGE cuando SEX = 'F'.
- La tabla OPTIM_STATE_LOOKUP, a través de una relación de subserie de Optim utilizando SUBSTR(SALESMAN_ID,1,2).

Tabla OPTIM_CUSTOMERS:

La tabla OPTIM_CUSTOMERS contiene nombres de clientes, números de ID y direcciones.

La tabla OPTIM_CUSTOMERS tiene las siguientes columnas:

CUST_ID

CHAR; hasta 5 caracteres; no puede contener nulos; contiene una restricción de comprobación.

CUSTNAME

CHAR; hasta 20 caracteres; no puede contener nulos.

ADDRESS1

VARCHAR; hasta 100 caracteres; no puede contener nulos.

ADDRESS2

VARCHAR; hasta 100 caracteres; no puede contener nulos.

LOCALITY

VARCHAR; hasta 56 caracteres

CITY VARCHAR; hasta 60 caracteres

STATE

VARCHAR; hasta 30 caracteres

COUNTRY_CODE

CHAR; hasta 2 caracteres

POSTAL_CODE

VARCHAR; hasta 15 caracteres

POSTAL_CODE_PLUS4

CHAR; hasta 4 caracteres; puede contener un valor nulo.

EMAIL_ADDRESS

VARCHAR; hasta 70 caracteres

PHONE_NUMBER

VARCHAR; hasta 20 caracteres

YTD_SALES

DECIMAL; importe en dólares; no puede contener nulos; tiene un valor predeterminado.

SALESMAN_ID

CHAR; hasta 6 caracteres

NATIONALITY
VARCHAR; hasta 30 caracteres

NATIONAL_ID
VARCHAR; hasta 30 caracteres

CREDITCARD_NUMBER
VARCHAR; 19 caracteres

CREDITCARD_TYPE
VARCHAR; hasta 30 caracteres

CREDITCARD_EXP
CHAR; 4 caracteres

CREDITCARD_CVV
VARCHAR; hasta 4 caracteres

DRIVER_LICENSE
VARCHAR; hasta 30 caracteres

CUSTOMER_INFO
XMLTYPE

Claves primarias

La columna CUST_ID es la columna de clave primaria.

Relaciones con otras tablas

La tabla OPTIM_CUSTOMERS es padre de:

- La tabla OPTIM_ORDERS, a través de una clave foránea de la columna CUST_ID.
- La tabla OPTIM_SHIP_TO, a través de una relación de Optim en la columna CUST_ID.

La tabla OPTIM_CUSTOMERS es hijo de:

- La tabla OPTIM_SALES, a través de su clave foránea de la columna SALESMAN_ID.

Tabla OPTIM_ORDERS:

La tabla OPTIM_ORDERS contiene información sobre los pedidos, como por ejemplo el número de pedido, el ID de cliente y el vendedor.

La tabla OPTIM_ORDERS tiene las siguientes columnas:

ORDER_ID
DECIMAL; número de ID de pedido; no puede contener nulos.

CUST_ID
CHAR; número de ID de cliente; hasta 5 caracteres; no puede contener nulos.

ORDER_DATE
TIMESTAMP; fecha de pedido; no puede contener nulos; tiene un valor predeterminado.

ORDER_TIME
TIMESTAMP; hora del día; no puede contener nulos; tiene un valor predeterminado.

FREIGHT_CHARGES
DECIMAL; importe en dólares

ORDER_SALESMAN
CHAR; hasta 6 caracteres

ORDER_POSTED_DATE

TIMESTAMP; no puede contener nulos; tiene un valor predeterminado.

ORDER_SHIP_DATE

CHAR; fecha de envío del pedido; hasta 8 caracteres; no puede contener nulos; tiene un valor predeterminado.

Claves primarias

La columna ORDER_ID es la columna de clave primaria.

Relaciones con otras tablas

La tabla OPTIM_ORDERS es padre de la tabla OPTIM_DETAILS, a través de una clave foránea de la columna ORDER_ID.

La tabla OPTIM_ORDERS es un hijo de la tabla OPTIM_CUSTOMERS, a través de su clave foránea de la columna CUST_ID.

Tabla OPTIM_DETAILS:

La tabla OPTIM_DETAILS contiene información adicional para cada pedido de la tabla OPTIM_ORDERS.

La tabla OPTIM_DETAILS tiene las siguientes columnas:

ORDER_ID

DECIMAL; número de ID de pedido; no puede contener nulos.

ITEM_ID

CHAR; hasta 5 caracteres; número de ID de elemento; no puede contener nulos.

ITEM_QUANTITY

DECIMAL; número de elementos; no puede contener nulos.

DETAIL_UNIT_PRICE

DECIMAL; precio unitario; importe en dólares; no puede contener nulos.

Claves primarias

Las columnas ORDER_ID e ITEM_ID son la clave primaria.

Relaciones con otras tablas

La tabla OPTIM_DETAILS es hijo de:

- La tabla OPTIM_ORDERS, a través de su clave foránea de la columna ORDER_ID.
- La tabla OPTIM_ITEMS, a través de su clave foránea de la columna ITEM_ID.

Tabla OPTIM_ITEMS:

La tabla OPTIM_ITEMS contiene información sobre cada elemento de un pedido, como por ejemplo la descripción, el precio y la cantidad en inventario.

La tabla OPTIM_ITEMS tiene las siguientes columnas:

ITEM_ID

CHAR; hasta 5 caracteres; no puede contener nulos.

ITEM_DESCRIPTION

VARCHAR; hasta 72 caracteres; no puede contener nulos.

CATEGORY

VARCHAR; hasta 14 caracteres; no puede contener nulos.

RATING

CHAR; hasta 4 caracteres; no puede contener nulos.

UNIT_PRICE

DECIMAL; importe en dólares; no puede contener nulos.

ON_HAND_INVENTORY

INTEGER; no puede contener nulos.

Claves primarias

La columna ITEM_ID es la columna de clave primaria.

Relaciones con otras tablas

La tabla OPTIM_ITEMS es padre de la tabla OPTIM_DETAILS, a través de una clave foránea de la columna ITEM_ID.

Tabla OPTIM_SHIP_TO:

La tabla OPTIM_SHIP_TO contiene información de envío de pedidos.

La tabla OPTIM_SHIP_TO tiene las siguientes columnas:

CUST_ID

CHAR; hasta 5 caracteres; no puede contener nulos.

SHIP_ID

DECIMAL; no puede contener nulos.

ADDRESS1

VARCHAR; hasta 100 caracteres

ADDRESS2

VARCHAR; hasta 100 caracteres

LOCALITY

VARCHAR; hasta 56 caracteres

CITY VARCHAR; hasta 30 caracteres

STATE

VARCHAR; hasta 30 caracteres

COUNTRY_CODE

CHAR; abreviatura de 2 caracteres

POSTAL_CODE

VARCHAR; hasta 15 caracteres

POSTAL_CODE_PLUS4

CHAR; 4 caracteres

IN_CARE_OF

VARCHAR; hasta 31 caracteres

SHIPPING_CHANGE_DT

TIMESTAMP; no puede contener nulos; tiene un valor predeterminado.

Claves primarias

La columna SHIP_ID es la columna de clave primaria.

Relaciones con otras tablas

La tabla OPTIM_SHIP_TO es padre de la tabla OPTIM_SHIP_INSTR, a través de una relación de Optim en la columna SHIP_ID.

La tabla OPTIM_SHIP_TO es un hijo de la tabla OPTIM_CUSTOMERS, a través de su relación de Optim en la columna CUST_ID.

Tabla OPTIM_SHIP_INSTR:

La tabla OPTIM_SHIP_INSTR contiene información detallada para el envío de pedidos

La tabla OPTIM_SHIP_INSTR tiene las siguientes columnas:

SHIP_ID

DECIMAL

SHIP_INSTR_ID

INTEGER

ORDER_SHIP_INSTR

VARCHAR; hasta 254 caracteres

SHIP_UPDATED

TIMESTAMP; no puede contener nulos; tiene un valor predeterminado.

Claves primarias

La columna SHIP_INSTR_ID es la columna de clave primaria.

Relaciones con otras tablas

La tabla OPTIM_SHIP_INSTR es un hijo de la tabla OPTIM_SHIP_TO, a través de su relación de Optim en la columna SHIP_ID.

Tabla OPTIM_MALE_RATES:

La tabla OPTIM_MALE_RATES contiene tarifas de seguros en función de la edad.

La tabla OPTIM_MALE_RATES tiene las siguientes columnas:

AGE SMALLINT

RATE_PER_1000

DECIMAL; importe en dólares de la tarifa

Claves primarias

La columna RATE_PER_1000 es la columna de clave primaria.

Relaciones con otras tablas

La tabla OPTIM_MALE_RATES es un hijo de la tabla OPTIM_SALES, a través de su relación controlada por datos de Optim de la columna AGE.

Tabla OPTIM_FEMALE_RATES:

La tabla OPTIM_FEMALE_RATES contiene tarifas de seguros en función de la edad.

La tabla OPTIM_FEMALE_RATES tiene las siguientes columnas:

AGE SMALLINT

RATE_PER_1000

DECIMAL; importe en dólares de la tarifa

Claves primarias

La columna RATE_PER_1000 es la columna de clave primaria.

Relaciones con otras tablas

La tabla OPTIM_FEMALE_RATES es un hijo de la tabla OPTIM_SALES, a través de su relación controlada por datos de Optim de la columna AGE.

Tabla OPTIM_STATE_LOOKUP:

La tabla OPTIM_STATE_LOOKUP contiene códigos de estado y las abreviaturas correspondientes.

La tabla OPTIM_STATE_LOOKUP tiene las siguientes columnas:

DIST_CODE

CHAR; 3 caracteres; no puede contener un valor nulo.

DISTRICT

CHAR; 2 caracteres; no puede contener un valor nulo.

Claves primarias

La tabla OPTIM_STATE_LOOKUP no tiene una clave primaria.

Relaciones con otras tablas

La tabla OPTIM_STATE_LOOKUP es un hijo de la tabla OPTIM_SALES a través de una relación de subserie en la columna DISTRICT utilizando SUBSTR(SALESMAN_ID,1,2).

Tablas de privacidad de datos

Las tablas de privacidad de datos están disponibles para los clientes que tienen una Licencia de privacidad de datos de Optim. Utilice las tablas para enmascarar datos de empresa y personales, como por ejemplo nombres de empleados, nombres de clientes, números de la seguridad social, números de tarjetas de crédito y direcciones de correo electrónico. Utilice las tablas para generar datos transformados que sean a la vez válidos y exclusivos dentro del contexto de la aplicación.

Puede utilizar las tablas de privacidad de datos para:

- Impedir infracciones de privacidad internas desidentificando o enmascarando los datos disponibles para desarrolladores, verificadores de garantía de calidad y otros miembros del personal.
- Mejorar las iniciativas de conformidad con la privacidad sustituyendo los datos del cliente por datos contextualmente precisos pero ficticios.
- Proteger la información confidencial de los clientes y los datos de los empleados en los entornos de desarrollo y prueba de aplicaciones.
- Asegurar resultados de prueba válidos propagando los elementos enmascarados entre las tablas relacionadas para garantizar la integridad referencial de la base de datos.

Cada categoría de datos personales se proporciona en una tabla separada, para los siguientes países (abreviaciones entre paréntesis): Australia (AU), Canadá (CA), Francia (FR), Alemania (DE), Italia (IT), Japón (JP), España (ES), Reino Unido (UK) y Estados Unidos (US). Cada tabla incluye una columna de números secuenciales que se utilizan con las políticas de búsqueda que utilizan valores hash para seleccionar una fila de la tabla de búsqueda.

Cada nombre de tabla se compone de un prefijo de abreviatura de país y de la categoría (*abreviaturapaís_categoría*). Por ejemplo, la tabla de direcciones de Canadá se llama CA_ADDRESSES y la tabla de direcciones de Alemania se llama DE_ADDRESSES.

El esquema incluye las categorías siguientes:

ADDRESSES

Tablas que incluyen columnas para la dirección postal, ciudad, localidad (estado o provincia) y código postal.

FIRSTNAME

Tablas que incluyen una columna con nombres determinados masculinos y femeninos.

FIRSTNAME_F

Tablas que incluyen una columna con nombres determinados femeninos.

FIRSTNAME_M

Tablas que incluyen una columna con nombres determinados femeninos.

LASTNAME

Tablas que incluyen una columna con nombres de familia.

PERSON

Tablas que incluyen columnas para fecha de nacimiento, nombre determinado, apellidos, género, número de teléfono, número de ID nacional, nombre de la empresa y dirección de correo electrónico.

CCN Tablas que incluyen números de tarjeta de crédito para el emisor asociado (MasterCard, VISA, etc.).

DOMAIN_NAMES

Tabla que incluye nombres de dominio para enmascarar direcciones de correo electrónico.

Creación de tablas de datos de ejemplo

Utilice la utilidad de configuración para crear tablas de datos de ejemplo.

Para crear tablas de datos de ejemplo:

1. En Optim Designer, pulse **Programas de utilidad > Configurar** Se abre la utilidad de configuración.
2. Pulse **Tareas > Cargar/descartar datos de ejemplo** Se abre el asistente Cargar/descartar datos de ejemplo.
3. Complete los pasos del asistente.

Debe seleccionar un directorio Optim. El nombre del directorio Optim predeterminado para el repositorio de Optim es PODREPO.

Cuando se le solicite que seleccione un alias de base de datos, seleccione un nombre de alias de almacén de datos.

Creación de tablas de privacidad de datos

Utilice la utilidad de configuración para crear tablas de privacidad de datos.

Para crear tablas de privacidad de datos:

1. En Optim Designer, pulse **Utilidades > Configurar** Se abre la utilidad de configuración.

2. Pulse **Tareas > Cargar/descartar datos de privacidad de datos** Se abre el asistente Cargar/descartar datos de privacidad de datos.
3. Complete los pasos del asistente.
Debe seleccionar un directorio Optim. El nombre del directorio Optim predeterminado para el repositorio de Optim es PODREPO.
Cuando se le solicite que seleccione un alias de base de datos, seleccione un nombre de alias de almacén de datos.

Soporte de base de datos

Optim Designer proporciona soporte para varios sistemas de gestión de base de datos.

Optim Designer soporta las siguientes bases de datos:

- DB2 para z/OS V8.1, V9.1, V10.1
- DB2 for Linux, UNIX, y Windows V8.2, V9.1, V9.5, V9.7
- DB2 para i V5.4
- Informix V10
- Microsoft SQL Server 2005, 2008
- Oracle V10.2, V11, V11.2
- Sybase V12.5, V15, V15.5
- Teradata V2.6, V12, V13
- IBM Informix V11.5, V.11.7

Requisitos previos de DB2

Para permitir Optim para obtener metadatos JDBC completos de una instancia de DB2 z/OS, el valor DESCSTAT en ZPARMS debe ser YES. Además, debe ejecutar el trabajo DSNTIJMS para instalar los procedimientos almacenados que necesita JDBC, enlazar los paquetes necesarios y establecer permisos de seguridad. Además, se necesitan las definiciones del gestor de carga de trabajo (WLM) para garantizar que WLM puede iniciar el espacio de direcciones del procedimiento almacenado cuando lo solicite DB2.

Funciones de accesibilidad

Las funciones de accesibilidad ayudan a las personas con discapacidades físicas, por ejemplo con movilidad limitada o visión limitada, o con otras necesidades especiales, a utilizar correctamente productos de software.

Optim Designer utiliza las funciones de accesibilidad disponibles en el entorno Eclipse.

Las funciones de accesibilidad ayudan a las personas con discapacidades físicas, por ejemplo con movilidad limitada o visión limitada o a aquellas personas que tienen necesidades especiales, a utilizar correctamente productos de software.

Capítulo 2. Gestión de conexiones de origen de datos

Utilice el Explorador de repositorios para crear alias de almacén de datos. Utilice el Explorador de directorios para conectarse a un directorio Optim.

Trabajar con un alias de almacén de datos

Un alias de almacén de datos es un objeto definido por el usuario que está asociado con una base de datos. Cuando defina un alias de almacén de datos, proporcione parámetros que Optim utilizará para comunicarse con dicha base de datos. Estos parámetros incluyen el tipo y la versión del sistema de gestión de bases de datos (DBMS) y las propiedades para una conexión de cliente y para una conexión JDBC.

Un nombre de alias de almacén de datos sirve como calificador de alto nivel que le permite acceder a una base de datos específica para realizar las funciones solicitadas. Por ejemplo, en una definición de acceso, debe calificar el nombre de una tabla con un nombre de alias de almacén de datos. El alias de almacén de datos referenciado proporciona los parámetros necesarios para conectar con la base de datos donde reside la tabla.

Un alias de almacén de datos identifica una base de datos específica y sirve como prefijo en los nombres completos de claves primarias, tablas de base de datos y relaciones. Los alias de almacén de datos son elementos esenciales a la hora de gestionar las bases de datos. Las reglas siguientes se aplican a los nombres de alias de almacén de datos:

- Cada base de datos puede tener únicamente un alias de almacén de datos.
- Cada nombre de alias de almacén de datos debe ser exclusivo.
- Los objetos del repositorio no pueden tener el mismo nombre que un alias de almacén de datos.

Definición de un alias de almacén de datos

Utilice el asistente Nuevo alias de almacén de datos para definir un nuevo alias de almacén de datos.

Para definir un alias de almacén de datos, necesita la serie de conexión para una conexión de cliente y el archivo .jar para una conexión JDBC.

Para definir un alias de almacén de datos:

1. En el Explorador de repositorios, pulse con el botón derecho en **Alias de almacén de datos** y pulse **Nuevo alias de almacén de datos**. Se abre el asistente Nuevo alias de almacén de datos.

2. Complete los pasos del asistente.

Debe especificar propiedades para una conexión de cliente y para una conexión JDBC. También debe identificar el juego de caracteres que utiliza la base de datos y cómo gestionar paquetes para acceder a los objetos de Optim.

Edición de un alias de almacén de datos

Utilice el asistente Propiedades de conexión para editar un alias de almacén de datos.

Para editar un alias de almacén de datos:

1. En el Explorador de repositorios, expanda el nodo **Alias de almacén de datos**, pulse con el botón derecho en el alias de almacén de datos y pulse **Abrir**. Se abre el asistente Propiedades de conexión.
2. Complete los pasos del asistente.

Conexión con un alias de almacén de datos

Utilice el Explorador de repositorios para conectarse a un alias de almacén de datos.

Para conectarse a un alias de almacén de datos:

En el Explorador de repositorios, expanda el nodo **Alias de almacén de datos**, pulse con el botón derecho del ratón en el alias de almacén de datos y pulse **Conectar**.

Trabajar con el repositorio de Optim

El repositorio de Optim contiene las definiciones, como por ejemplo alias de almacén de datos, servicios y definiciones de acceso, que se comparten entre los componentes de Optim.

Sólo se puede asociar un directorio Optim con el repositorio.

Conexión con el repositorio de Optim

Utilice la ventana Nueva conexión de repositorio para definir una conexión con el repositorio de Optim.

El repositorio de Optim debe estar ejecutándose para crear una conexión.

Para definir una conexión con el repositorio de Optim:

1. En el Explorador de repositorios, pulse con el botón derecho en **Repositorio de Optim** y pulse **Nueva > Conexión de repositorio**. Se abre la ventana Nueva conexión de repositorio.
2. Especifique la información de conexión y las credenciales de usuario para el repositorio.
3. Pulse **Probar conexión** para verificar la conexión.
4. Pulse **Aceptar**.

Cuando Optim Designer se conecta al repositorio, el Explorador de repositorios muestra el contenido del repositorio.

Edición de una conexión de repositorio

Utilice la ventana Editar conexión de repositorio para definir una conexión con el repositorio de Optim.

Para definir una conexión con el repositorio de Optim:

1. En el Explorador de repositorios, pulse con el botón derecho en **Repositorio de Optim** y pulse **Abrir**. Se abre la ventana Editar conexión de repositorio.
2. Especifique la información de conexión y las credenciales de usuario para el repositorio.
3. Pulse **Probar conexión** para verificar la conexión.
4. Pulse **Aceptar**.

Cuando Optim Designer se conecta al repositorio, el Explorador de repositorios muestra el contenido del repositorio.

Modificación de una conexión de repositorio

Utilice la ventana Conmutar repositorio para conectarse a otro repositorio de Optim.

El repositorio de Optim debe estar en ejecución para cambiar una conexión.

Para cambiar una conexión al repositorio de Optim:

1. En el Explorador de repositorios, pulse con el botón derecho en **Repositorio de Optim** y pulse **Conmutar repositorio**. Se abre la ventana Conmutar repositorio.

2. Seleccione una conexión de repositorio.
3. Pulse **Aceptar**.

Cuando Optim Designer se conecta al repositorio, el Explorador de repositorios muestra el contenido del repositorio.

Asociación de un directorio Optim con el repositorio

Utilice la ventana Asociar directorio para asociar un directorio Optim con el repositorio.

Se debe definir una conexión con el directorio Optim en el registro del sistema de Optim Designer. Utilice el programa de configuración de Optim para definir una conexión en el registro.

Sólo se puede asociar un directorio Optim con el repositorio.

Para asociar un directorio Optim con el repositorio:

1. En el Explorador de directorios, pulse **Pulse aquí para asociar un directorio Optim con la conexión de repositorio**. Se abre la ventana Asociar directorio.
2. Seleccione un directorio Optim.
3. Pulse **Finalizar**.

Cuando Optim Designer se conecta al directorio Optim, el Explorador de directorios muestra el contenido del directorio Optim.

Transformar una solicitud en un servicio

Utilice el asistente Transformar solicitud en servicio Optim para transformar una solicitud de un directorio Optim en un servicio de gestión de datos.

Se debe asociar un directorio Optim con el repositorio.

Para transformar una solicitud de un directorio Optim en un servicio de gestión de datos:

1. En el Explorador de directorios, pulse **Transformar**. Se abrirá el asistente Transformar solicitud en servicio Optim.
2. En la página Alias DB y conexiones de base de datos, revise la información de conexión. Si un alias de almacén de datos no está definido para un alias de base de datos asociado con la solicitud, pulse **Editar conexión** para definir un alias de almacén de datos. Pulse **Siguiente**.
3. Si la transformación es satisfactoria, pulse **Finalizar**. Si la transformación no es satisfactoria, pulse **Siguiente** para revisar la información de error.

Una carpeta con el nombre del directorio Optim que contenía la solicitud se creará en el Explorador de repositorios. La solicitud transformada se listará como servicio dentro de la carpeta.

Capítulo 3. Gestión de modelos de datos

Puede crear modelos de datos que definen datos para extraer y correlacionar datos de destino para los servicios de gestión de datos. También puede crear relaciones y claves primarias de Optim para complementar las relaciones y las claves primarias de la base de datos.

Trabajar con definiciones de acceso

Utilice las definiciones de acceso para especificar las tablas, el cruce de relaciones y los criterios de selección para los datos que desea procesar.

Una definición de acceso incluye los siguientes elementos.

Tablas Una definición de acceso debe hacer referencia como mínimo a una tabla, vista, alias o sinónimo. La tabla, vista, alias o sinónimo desde donde se seleccionan primero las filas se denomina tabla inicial. Puede especificar el nombre de una tabla inicial e incluir fácilmente los nombres de todas las tablas relacionadas con la tabla inicial (hasta un máximo de 24.000 tablas).

Tabla inicial

La tabla inicial es la primera tabla que debe utilizarse al extraer datos. Puede especificar cualquier tabla en la definición de acceso como tabla inicial, excepto una tabla de referencia. Si no especifica explícitamente una tabla inicial, la primera tabla de la lista de tablas es la tabla inicial.

Tabla de referencia

Tablas desde las que se seleccionan todas las filas; a menos que se especifiquen criterios de selección para la tabla de referencia. Especifique cualquier tabla como tabla de referencia, excepto la tabla inicial.

Relaciones

Las relaciones determinan la vía de acceso de cruce para seleccionar datos de las tablas. De forma predeterminada, las relaciones se cruzan de padre a hijo, pero puede controlar la dirección del cruce utilizando los valores de la definición de acceso. Las relaciones entre las tablas referenciadas por la definición de acceso se listan en la ficha de relaciones (hasta un máximo de 24.000 tablas). Puede seleccionar las relaciones que se utilizarán en el proceso y la dirección en la que se cruzarán.

Criterios de selección

Los criterios de selección definen un conjunto específico de datos que deben utilizarse a partir de las tablas en una definición de acceso. Puede especificar valores y operadores de SQL, y utilizar variables de sustitución con los valores predeterminados.

Apuntar y disparar

Utilice una lista de apuntar y disparar para seleccionar filas específicas de una tabla inicial que se incluirán en un servicio.

Variables

Las variables son valores predeterminados definidos por el usuario que se especifican en una definición de acceso. Puede utilizar estas variables de sustitución para especificar criterios de selección de columna o para crear una cláusula WHERE de SQL.

Parámetros adicionales

Utilice parámetros adicionales para extraer filas que se corresponden con un determinado valor de columna en la tabla inicial o para utilizar una velocidad de muestreo especificada (cada enésima fila).

Convenios de denominación

Un nombre de definición de acceso totalmente calificado tiene el formato *identificador.nombre*, donde:
identificador

Calificador asignado a la definición de acceso (1 a 8 caracteres).

nombre Nombre base asignado a la definición de acceso (1 a 12 caracteres).

Un conjunto lógico de convenios de denominación puede identificar el uso para cada definición de acceso y utilizarse para organizar las definiciones con el fin de facilitar el acceso.

Creación de una definición de acceso

Utilice el asistente Nueva definición de acceso para crear una definición de acceso.

Antes de crear una definición de acceso, debe existir un alias de almacén de datos para la base de datos que contiene la tabla inicial.

Para crear una definición de acceso:

1. Expanda una carpeta en el Explorador de repositorios para que contenga la definición de acceso, pulse con el botón derecho del ratón en **Definiciones de acceso** y pulse **Nueva definición de acceso**. Se abre el asistente Nueva definición de acceso.
2. Complete los pasos del asistente.
Debe seleccionar un alias de almacén de datos y una tabla inicial del alias. Puede optar por añadir tablas relacionadas con la tabla inicial.

La nueva definición de acceso se visualiza en el Editor de definiciones de acceso.

Gestión de tablas en una definición de acceso

Utilice la ficha **Tablas** en el Editor de definición de acceso para añadir o eliminar tablas en una definición de acceso. También puede cambiar tablas en las tablas relacionadas o de referencia.

Adición de tablas a una definición de acceso

Utilice el asistente Añadir tablas para añadir tablas de un alias de almacén de datos a una definición de acceso.

Para añadir tablas a una definición de acceso:

1. Expanda una carpeta en el Explorador de repositorios que contiene la definición de acceso, expanda el nodo **Definiciones de acceso** y efectúe una doble pulsación en la definición de acceso. Se abre el Editor de definiciones de acceso.
2. Seleccione la ficha **Tablas**.
3. Pulse **Añadir tablas**. Se abre el asistente Abrir tablas.
4. Complete los pasos del asistente.
Debe seleccionar un alias de almacén de datos que contiene las tablas que desea añadir. Puede añadir tablas de referencia o tablas relacionadas con una tabla seleccionada.
5. Guarde la definición de acceso.

Cambiar tablas a tablas de referencia o relacionadas

Utilice el Editor de definiciones de acceso para cambiar las tablas de una definición de acceso a tablas de referencia o relacionadas.

Para cambiar tablas a tablas de referencia o relacionadas:

1. Expanda la carpeta en el Explorador de repositorios que contiene la definición de acceso, expanda el nodo **Definiciones de acceso** y efectúe una doble pulsación en la definición de acceso. Se abre el Editor de definiciones de acceso.
2. Seleccione la ficha **Tablas**.
3. Seleccione una tabla para cambiar a tabla de referencia o relacionada.
4. Pulse **Cambiar a Referencia** o **Cambiar a Relacionada**. La columna **Tipo** mostrará el nuevo tipo de tabla.
5. Guarde la definición de acceso.

Eliminación de tablas de una definición de acceso

Utilice el Editor de definiciones de acceso para eliminar tablas de una definición de acceso.

Para eliminar tablas de una definición de acceso:

1. Expanda la carpeta en el Explorador de repositorios que contiene la definición de acceso, expanda el nodo **Definiciones de acceso** y efectúe una doble pulsación en la definición de acceso. Se abre el Editor de definiciones de acceso.
2. Seleccione la ficha **Tablas**.
3. Seleccione una tabla que desee eliminar.
4. Pulse **Eliminar tabla**.
5. Guarde la definición de acceso.

Visualización y edición de pasos de cruce

Utilice la ventana Pasos de cruce para ver y editar los pasos de cruce de una definición de acceso.

Puede evaluar las tablas, las relaciones y los pasos realizados para seleccionar datos. Esta evaluación puede asegurar que recuperará el conjunto deseado de datos.

Los pasos de un proceso pueden repetirse cualquier número de veces de acuerdo con las relaciones que seleccione y de los criterios que especifique.

Cualquier tabla puede revisitarse varias veces en pasos sucesivos. Los ciclos también pueden estar implicados. Un ciclo hace que un conjunto de tablas se cruce repetidamente hasta que un paso completo del ciclo no dé como resultado la selección de filas adicionales.

Cuando se cruza una relación para seleccionar filas padre para filas hijo ya seleccionadas, se ignoran los criterios de selección de la tabla padre.

Para ver y editar los pasos de cruce:

1. Expanda una carpeta en el Explorador de repositorios que contiene la definición de acceso, expanda el nodo **Definiciones de acceso** y efectúe una doble pulsación en la definición de acceso. Se abre el Editor de definiciones de acceso.
2. Seleccione la ficha **Tablas**.
3. Pulse **Mostrar pasos**. Se abre la ventana Pasos de cruce.
4. Revise los pasos y edite el orden en el que se han seleccionado las tablas.
5. Pulse **Aceptar**.
6. Guarde la definición de acceso.

Gestión de los criterios de selección

Utilice criterios de selección para centrarse en un conjunto específico de datos relacionados definiendo una cláusula WHERE de SQL y utilizando variables de sustitución con los valores predeterminados.

Los criterios de selección deben estar conformes con la sintaxis SQL e incluir operadores lógicos o relacionales. Los operadores lógicos y la sintaxis varían entre los distintos DBMS. Consulte la documentación apropiada del DBMS para obtener información.

Para seleccionar el conjunto deseado de datos de una tabla, quizás necesite una combinación de operadores lógicos AND y OR.

Si también se utiliza una lista de apuntar y disparar, ésta se incluye de forma lógica con los otros criterios con un operador lógico OR.

Cuando se cruza una relación de hijo a padre, se ignoran los criterios de selección para la tabla padre.

Criterios de fecha

Un operador exclusivo **BEFORE**, le permite seleccionar datos a partir de los valores de una columna DATE. La sintaxis para este operador es:

BEFORE (*nD nW nM nY*)

Utilice los argumentos **D**, **W**, **M** e **Y** en cualquier combinación para indicar el número de días, semanas, meses o años que se restan de la fecha en tiempo de ejecución. Si no se especifica ningún argumento, se utilizará la fecha actual. Las filas con una fecha anterior a la fecha calculada se extraen o se archivan. El multiplicador *n* es un entero y, opcionalmente, puede ir precedido por + o -.

Definición de criterios de selección

Utilice la ventana Editar criterios de selección para definir criterios de selección con una cláusula WHERE de SQL.

Para definir criterios de selección:

1. Expanda una carpeta en el Explorador de repositorios que contiene la definición de acceso, expanda el nodo **Definiciones de acceso** y efectúe una doble pulsación en la definición de acceso. Se abre el Editor de definiciones de acceso.
2. Seleccione la ficha **Tablas**.
3. Seleccione la tabla a la que añadirá los criterios de selección.
4. Pulse **Editar criterios de selección**. Se abre la ventana Editar criterios de selección que muestra una cláusula WHERE para la tabla.
5. Especifique criterios de selección.
Para visualizar una lista de columnas y operadores, pulse con el botón derecho en la cláusula WHERE y pulse **Asistencia de contenido**.
Para añadir una variable, seleccione un **Delimitador de variable** y pulse **Insertar variable**.
Pulse **Comprobar sintaxis** para verificar la sintaxis e identificar errores.
6. Pulse **Aceptar**.
Si la sintaxis no es válida, se abrirá una solicitud para identificar el error. No se pueden guardar los criterios si contienen errores.
7. Guarde la definición de acceso.

Especificación de opciones de agrupación de datos

Utilice la pestaña **Propiedades del grupo de datos** en el Editor de definiciones de acceso para especificar la selección de grupo, que extrae un número de filas en función de los valores de una determinada columna de la tabla inicial. Las filas de cada grupo tienen el mismo valor en la columna seleccionada.

Cuando los criterios se utilizan en combinación con parámetros de selección de grupo, los criterios se aplican primero con parámetros de selección de grupo aplicados al resultado.

Para especificar opciones de agrupación de datos:

1. Expanda una carpeta en el Explorador de repositorios que contiene la definición de acceso, expanda el nodo **Definiciones de acceso** y efectúe una doble pulsación en la definición de acceso. Se abre el Editor de definiciones de acceso.
2. Seleccione la pestaña **Propiedades del grupo de datos**.
3. Seleccione una columna y especifique opciones de agrupación.
4. Guarde la definición de acceso.

Especificación de opciones de muestreo de datos y límite de filas

Utilice la pestaña **Tablas** en el Editor de definiciones de acceso para especificar opciones para extraer una muestra de filas o para limitar el número de filas a extraer.

Si se utiliza una lista de apuntar y disparar para seleccionar filas de la tabla inicial, el servicio de extracción ignora los parámetros de muestreo de datos o de límite de filas para la tabla inicial.

Para especificar opciones para el muestreo de datos y límites de filas:

1. Expanda una carpeta en el Explorador de repositorios que contiene la definición de acceso, expanda el nodo **Definiciones de acceso** y efectúe una doble pulsación en la definición de acceso. Se abre el Editor de definiciones de acceso.
2. Seleccione la ficha **Tablas**.
3. Especifique las siguientes opciones:

Cada enésimo número de filas

Introduzca un valor numérico para especificar un factor de muestreo para una tabla. Por ejemplo, si especifica 5, el proceso extrae cada quinta fila de la tabla, a partir de la quinta fila. Los valores válidos son del 1 al 9999.

Límite de filas

Especifique un valor numérico para limitar el número de filas extraídas de una tabla. Los valores válidos son del 1 al 999999999999.

4. Guarde la definición de acceso.

Gestión de relaciones en una definición de acceso

Utilice la ficha **Relaciones** del Editor de definición de acceso para añadir o eliminar relaciones en una definición de acceso. También puede especificar opciones de acceso a tabla y de cruce de relaciones.

Selección de opciones de cruce de relaciones

Utilice el Editor de definiciones de acceso para gestionar las opciones de cruce de relaciones en una definición de acceso.

Puede especificar las siguientes opciones:

- El número máximo de filas de la tabla hijo que deben seleccionarse para una relación.
- El número máximo de búsquedas de claves realizadas a la vez para una tabla.

Aumentar el límite de búsqueda de claves puede mejorar significativamente el rendimiento. Por ejemplo, si especifica 5 como límite de búsqueda de clave y la clave tiene una sola columna, se buscan 5 valores de clave en una única solicitud al DBMS.

- Las vías de acceso de cruce:
 - Cruce la relación de hijo a padre para seleccionar una fila padre para cada fila hijo y garantizar la integridad relacional. (Opción 1)
 - Cruce la relación para seleccionar filas hijo adicionales para cada fila padre seleccionada como resultado de un cruce de hijo a padre. (Opción 2)

Las opciones 1 y 2 son relevantes cuando la tabla inicial es una tabla hijo o cuando una tabla tiene más de una tabla padre referenciada por la definición de acceso.

La opción 2 sólo es relevante si cruza una relación de hijo a padre. Por ejemplo, si un proceso pasa de hijo a padre (opción 1) y se selecciona una fila padre, la opción 2 hace que el proceso seleccione filas hijo adicionales para dicha fila padre.

Si selecciona la opción 2 para una relación, considere un límite hijo sobre el número de filas hijo a extraer.

Para gestionar las opciones de cruce de relaciones en una definición de acceso:

1. Expanda la carpeta en el Explorador de repositorios que contiene la definición de acceso, expanda el nodo **Definiciones de acceso** y efectúe una doble pulsación en la definición de acceso. Se abre el Editor de definiciones de acceso.
2. Seleccione la ficha **Relaciones** y, a continuación, la ficha **Opciones de cruce**.
3. Especifique las opciones de cruce.
4. Guarde la definición de acceso.

Selección de opciones de acceso a tabla

Utilice el Editor de definiciones de acceso para acceder al método para acceder a la tabla padre o hijo de cada relación.

Puede especificar las siguientes opciones:

Predeterminado

Optim determina el mejor método. Una búsqueda de clave se utiliza cuando un índice de DBMS está disponible, y una exploración se utiliza cuando un índice no está disponible. Sin embargo, si se accede a una parte significativa de la tabla, el valor predeterminado es explorar, aunque exista un índice.

Forzar exploración

Lee todas las filas de una tabla a la vez.

Forzar búsqueda de clave

Localiza filas utilizando una cláusula WHERE para buscar valores de clave primaria o foránea.

Nota: Altere temporalmente el método predeterminado sólo si la información estadística en el informe de proceso indica que el método predeterminado es menos eficaz.

Para gestionar las opciones de acceso a tabla:

1. Expanda la carpeta en el Explorador de repositorios que contiene la definición de acceso, expanda el nodo **Definiciones de acceso** y efectúe una doble pulsación en la definición de acceso. Se abre el Editor de definiciones de acceso.
2. Seleccione la ficha **Relaciones** y, a continuación, la ficha **Acceso a tabla**.
3. Especifique las opciones de acceso a tabla.
4. Guarde la definición de acceso.

Gestión de variables en una definición de acceso

Las variables son valores predeterminados definidos por el usuario que se especifican en una definición de acceso. Puede utilizar estas variables de sustitución para crear una cláusula WHERE de SQL.

Al asignar variables, proporciona valores para las variables cada vez que se procesa la definición de acceso. Como opción, puede proporcionar valores predeterminados para las variables de sustitución. Las variables se guardan con la definición de acceso.

Valores predeterminados

Cuando cree una variable, puede especificar un valor predeterminado opcional que se utilizará cuando no se especifique ningún valor para la variable en tiempo de ejecución.

Los valores predeterminados deben ser del tipo de datos y tamaño apropiados para la columna y deben ajustarse a la sintaxis SQL. Por ejemplo, supongamos que un nombre de variable es **ST** (estado), el delimitador de variables es el signo de dos puntos (:) y la columna requiere datos de tipo carácter.

- Si utiliza la variable con comillas simples en los criterios de selección, debe especificar el valor sin comillas simples:

Criterios de selección	Valor
= 'ST'	CA

- Si utiliza la variable sin comillas simples en los criterios de selección, debe especificar el valor con comillas simples:

Criterios de selección	Valor
= :ST	'CA'

Nota: Los valores predeterminados no se validan hasta el tiempo de ejecución. Si un valor tiene el tamaño o el tipo de datos incorrecto para la columna o no se ajusta a la sintaxis de SQL, se pueden producir errores de proceso.

Serie de solicitud

Debe especificar el texto que solicita el valor de la variable en tiempo de ejecución. Escriba la serie de solicitud exactamente tal como desea que aparezca en el diálogo de solicitud de proceso (hasta 50 caracteres). Esta solicitud se visualiza antes de ejecutar el proceso.

Creación de una variable

Utilice la ventana **Añadir una variable** para crear una variable en una definición de acceso.

Para crear una variable:

1. Expanda la carpeta en el Explorador de repositorios que contiene la definición de acceso, expanda el nodo **Definiciones de acceso** y efectúe una doble pulsación en la definición de acceso. Se abre el Editor de definiciones de acceso.
2. Seleccione la ficha **Variables**.
3. Pulse **Añadir**. Se abre la ventana **Añadir una variable**.
4. Especifique información para la variable. Debe especificar un nombre de variable y texto de solicitud. Pulse **Aceptar**.
5. Guarde la definición de acceso.

Gestión de listas de apuntar y disparar

Utilice una lista de apuntar y disparar para seleccionar filas específicas de una tabla inicial que se incluirán en un servicio.

Una lista de apuntar y disparar está incluida en una definición de acceso. Cuando utilice apuntar y disparar para seleccionar filas de la tabla inicial, las claves primarias de estas filas se almacenan en un archivo de apuntar y disparar. La solicitud de servicio utiliza las claves primarias para identificar las filas que deben procesarse en primer lugar.

Creación de un archivo de lista de apuntar y disparar

Utilice la ventana **Nuevo archivo de apuntar y disparar** para crear un archivo de lista de apuntar y disparar para una tabla en un alias de almacén de datos.

Para crear un archivo de lista de apuntar y disparar:

1. En el Explorador de repositorios, expanda el nodo **Alias de almacén de datos** y el alias de almacén de datos de destino hasta que se visualice la tabla a la que añadirá el archivo de apuntar y disparar.
2. Expanda la tabla, pulse con el botón derecho en **Listas de apuntar y disparar** y pulse **Nuevo archivo de apuntar y disparar**. Se abre la ventana Nuevo archivo de apuntar y disparar.
3. Especifique un nombre de archivo y pulse **Aceptar**.

El archivo de lista de apuntar y disparar se visualiza bajo **Listas de apuntar y disparar**.

Debe utilizar el Editor de apuntar y disparar para seleccionar filas o la lista de apuntar y disparar.

Selección de una lista de apuntar y disparar

Utilice el Editor de definiciones de acceso para seleccionar una lista de apuntar y disparar para una definición de acceso.

Para seleccionar una lista de apuntar y disparar, un archivo de apuntar y disparar debe estar disponible para la tabla inicial.

Para seleccionar una lista de apuntar y disparar:

1. Expanda la carpeta en el Explorador de repositorios que contiene la definición de acceso, expanda el nodo **Definiciones de acceso** y efectúe una doble pulsación en la definición de acceso. Se abre el Editor de definiciones de acceso.
2. Seleccione la ficha **Apuntar y disparar**.
3. Seleccione una lista de apuntar y disparar.
4. Guarde la definición de acceso.

Trabajar con relaciones de Optim

Optim utiliza las relaciones para determinar los datos que deben recuperarse de las tablas relacionadas y se basa en las relaciones definidas en la base de datos, si está disponible. Sin embargo, también puede definir relaciones de Optim para complementar las relaciones de la base de datos.

Con las relaciones de Optim, algunas restricciones de la base de datos se relajan. Por ejemplo:

- No se requieren claves primarias y claves foráneas.
- No hace falta que las columnas correspondientes sean idénticas, pero deben ser compatibles.
- Al menos una columna de un par de columnas correspondientes debe especificarse por nombre de columna. Sin embargo, puede utilizar una expresión para evaluar o definir el valor en la segunda columna. Las expresiones pueden incluir literales de serie, constantes numéricas, NULL, concatenación y subseries.

Las relaciones más flexibles de Optim se denominan relaciones “ampliadas”. Las relaciones ampliadas pueden replicar las relaciones implícitas o gestionadas por aplicación de la base de datos, permitiéndole manipular conjuntos de datos relacionales de la misma manera que en el entorno de producción.

Además, una relación de Optim puede almacenarse en el directorio Optim como:

- Una relación **explícita**, utilizada para un único par de tablas.
- Una relación **genérica**, utilizada para uno o mas pares de tablas que tienen el mismo nombre base, nombres de columna y atributos pero distintos ID de creador.

Las relaciones genéricas son útiles cuando varios conjuntos de tablas sólo se diferencian por el ID de creador. (Por ejemplo, en un entorno de prueba, cada programador puede utilizar una copia separada de las mismas tablas de producción. Cada conjunto de tablas se puede distinguir por el ID de creador).

Mediante la utilización de relaciones genéricas, puede definir un conjunto de relaciones que se aplica a todos los conjuntos de tablas. Además, cuando se añade un conjunto de estas tablas, las relaciones genéricas se aplican automáticamente.

Restricciones








Aunque las reglas para crear relaciones de Optim son más flexibles que las reglas para crear relaciones definidas por la base de datos, existen algunas restricciones:

- Debe hacer referencia como mínimo a una columna para cada tabla de la relación.
- Puede hacer referencia como máximo a 64 columnas para cualquier tabla de la relación.
- No se pueden establecer coincidencias entre un literal o una constante y un literal o una constante.
- No se puede utilizar una columna SQL variant u Objeto grande (LOB).
- La longitud total de todos los valores especificados en la tabla padre o en la tabla hijo no puede superar los 3584 bytes.
- No se puede crear una relación utilizando una columna SQL variant.

En una definición de relación para una base de datos de varios bytes o Unicode:

- No se puede utilizar la función de subserie.
- No se pueden concatenar datos de tipo carácter (CHAR o NCHAR) con binarios (RAW).
- Si se utiliza la semántica de carácter de Oracle para cualquier columna CHAR, todas las columnas CHAR de la relación deben tener la semántica de carácter o un tipo de datos NCHAR.

EJEMPLOS:

Padre	Soportado/No soportado	Hijo	Descripción
CHAR		CHAR	Soportado, la semántica debe coincidir
NCHAR		NCHAR	Soportado, la semántica es irrelevante.
CHAR		NCHAR	No soportado
CHAR		VARCHAR	Soportado, la semántica debe coincidir.
NCHAR		NVARCHAR	Soportado, la semántica es irrelevante.
CHAR NCHAR		NCHAR CHAR	Soportado, si es semántica de carácter; no soportado si es semántica de byte.
CHAR NCHAR		NCHAR NCHAR	Soportado, si es semántica de carácter; no soportado si es semántica de byte.

Reglas de compatibilidad para relaciones

Cuando se define una relación de Optim, los valores correspondientes deben ser compatibles.

Tipo de columna	Es compatible con
Columna de caracteres	<ul style="list-style-type: none">• Columna de caracteres• Columna numérica• Literal de serie• Expresión de caracteres
Columna numérica	<ul style="list-style-type: none">• Columna numérica• Constante numérica• Columna de caracteres
Columna binaria	<ul style="list-style-type: none">• Columna binaria• Literal hexadecimal• Expresión binaria
Columna booleana	<ul style="list-style-type: none">• Columna booleana• Constante booleana (verdadero o falso)
Columna de fecha y hora	Columna de fecha y hora
Columna de fecha	Columna de fecha
Columna de hora	Columna de hora
Columna de intervalo	Columna de intervalo

Nota:

- Durante el proceso, un valor se convierte en el tipo de datos necesario para seleccionar las filas relacionadas. De forma predeterminada, el resultado de convertir un valor numérico a un tipo de datos de carácter se justifica a la derecha con ceros iniciales. Los valores del registro especial permiten cambiar el valor predeterminado por la justificación a la izquierda con espacios iniciales o finales. Además, un emparejamiento de carácter a numérico requiere una escala igual a 0 para la columna numérica.
- Puede utilizar NULL para cualquier columna apta para nulos.
- Las columnas Unicode o de varios bytes deben ser del mismo juego de caracteres.

Tipos de datos

Las clases siguientes de datos y tipos de datos asociados están soportados. Estas clases de datos son importantes para la compatibilidad de los datos cuando se utilizan valores de columnas en las relaciones.

Clase de carácter

DBMS	Tipos de datos
DB2	CHAR, VARCHAR, CLOB
Oracle	CHAR, VARCHAR2, LONG, CLOB, NCLOB, NCHAR, NVARCHAR
Sybase ASE	CHAR, VARCHAR, TXT
SQL Server	CHAR, VARCHAR, TXT
Informix	CHAR, VARCHAR, TXT

Nota: Las columnas de caracteres de un solo byte no son compatibles con las columnas de caracteres Unicode o de varios bytes.

Clase numérica

DBMS

DB2

Oracle

Sybase ASE

SQL Server

Informix

Tipos de datos

INTEGER, SMALLINT, DECIMAL, FLOAT, DOUBLE

NUMBER, FLOAT

TINYINT, INT, SMALLINT, DECIMAL, FLOAT, REAL, MONEY, SMALL MONEY

TINYINT, INT, SMALLINT, DECIMAL, FLOAT, REAL, MONEY, SMALL MONEY

INTEGER, SMALLINT, DECIMAL, FLOAT, REAL, DOUBLE PRECISION, SMALLFLOAT, SERIAL, MONEY, NUMERIC

Clase binaria

DBMS

DB2

Oracle

Sybase ASE

SQL Server

Informix

Tipos de datos

CHAR (para datos de bits), VARCHAR (para datos de bits), BLOB

RAW, LONG RAW

BINARY, VARBINARY, IMAGE

BINARY, VARBINARY, IMAGE

BYTE

Clase booleana

DBMS

Sybase ASE

Tipos de datos

BOOLEAN (TRUE o FALSE)

Fecha y hora

DBMS

DB2

Oracle

Sybase ASE

SQL Server

Informix

Tipos de datos

TIMESTAMP

DATE, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE

DATETIME, SMALL DATE TIME

DATETIME, SMALL DATE TIME

DATE, DATETIME

Clase de fecha

DBMS

DB2

Oracle

Informix

Tipos de datos

DATE

DATE

DATE

Clase de hora

DBMS

DB2

Tipos de datos

TIME

Clase de intervalo

DBMS	Tipos de datos
Oracle	YEAR/MONTH INTERVAL, DAY/SECOND INTERVAL
Informix	YEAR/MONTH INTERVAL, DAY/TIME INTERVAL

Creación de una relación de Optim

Utilice el asistente Nueva relación de Optim para crear una nueva relación de Optim.

Antes de crear una relación de Optim, debe existir un alias de almacén de datos para las tablas de la relación.

Para crear una relación de Optim:

1. En el Explorador de repositorios, pulse con el botón derecho en **Relaciones de Optim** y pulse **Nueva relación de Optim**. Se abre el asistente Nueva relación de Optim.
2. Complete los pasos del asistente.
Debe seleccionar las tablas padre e hijo en la relación.

La nueva relación de Optim se visualiza en el Editor de relaciones.

Debe utilizar el Editor de relaciones para definir una expresión de columna que identifique las columnas padre e hijo de la relación.

Edición de una relación de Optim

Utilice el Editor de relaciones para definir una expresión de columna que identifique las columnas padre e hijo en una relación de Optim. También puede utilizar el editor para crear una relación genérica y editar columnas.

Creación de una expresión de columna

Utilice la ventana Añadir expresión de columna para crear una expresión de columna que identifica las columnas padre e hijo en una relación de Optim.

Para crear una expresión de columna en una relación de Optim:

1. En el Explorador de repositorios, expanda el nodo **Relaciones de Optim** y efectúe una doble pulsación en la relación que desea editar. Se abre el Editor de relaciones.
2. Pulse **Añadir expresión de columna....** Se abre la ventana Añadir expresión de columna.
3. Pulse **Seleccionar columna padre**. Se abre la ventana Seleccionar una columna padre.
4. Seleccione una columna padre y pulse **Aceptar** para volver a la ventana Añadir expresión de columna.
5. Pulse **Seleccionar columna hijo**. Se abre la ventana Seleccionar una columna hijo.
6. Seleccione una columna hijo y pulse **Aceptar** para volver a la ventana Añadir expresión de columna.
7. Pulse **Aceptar** para volver al editor. Las columnas padre e hijo se listan en el editor.
8. Guarde la relación.

Edición de las columnas de una relación

Utilice las ventanas Seleccionar una columna padre y Seleccionar una columna hijo para editar las columnas de una relación de Optim.

Para editar las columnas de una relación de Optim:

1. En el Explorador de repositorios, expanda el nodo **Relaciones de Optim** y efectúe una doble pulsación en la relación que desea editar. Se abre el Editor de relaciones.
2. Seleccione la fila que contiene una columna que desea editar. Para editar una columna padre:

- a. Pulse **Seleccionar columna padre**. Se abre la ventana Seleccionar una columna padre.
- b. Seleccione una columna padre y pulse **Aceptar** para volver al editor.

Para editar una columna hijo:

- a. Pulse **Seleccionar columna hijo**. Se abre la ventana Seleccionar una columna hijo.
- b. Seleccione una columna hijo y pulse **Aceptar** para volver al editor.

Los elementos seleccionados se listan en el editor.

3. Guarde la relación.

Cambiar el orden de las columnas en una relación

Utilice el Editor de relaciones para cambiar el orden de las columnas en una relación de Optim.

Para cambiar el orden de las columnas en una relación de Optim:

1. En el Explorador de repositorios, expanda el nodo **Relaciones de Optim** y efectúe una doble pulsación en la relación que desea editar. Se abre el Editor de relaciones.
2. Seleccione la fila que contiene las columnas que desea reordenar.
3. Pulse **Subir fila** o **Bajar fila** para cambiar el orden.
4. Guarde la relación.

Creación de una relación genérica

Utilice el Editor de relaciones para crear una relación genérica de Optim.

Algunas bases de datos contienen conjuntos de tablas que son idénticas, excepto por el ID de creador. En lugar de definir una relación para cada conjunto de tablas, puede definir una relación genérica que se aplica a todos los conjuntos de tablas que tienen el mismo nombre base, independientemente del ID de creador. Puede modificar las tablas base en una relación genérica.

Para crear una relación genérica de Optim:

1. En el Explorador de repositorios, expanda el nodo **Relaciones de Optim** y efectúe una doble pulsación en la relación que desea editar. Se abre el Editor de relaciones.
2. Pulse **Genérica**.
3. Guarde la relación.

Trabajar con claves primarias de Optim

Una clave primaria es la columna o columnas que contienen valores que identifican de forma exclusiva cada fila de una tabla. Una tabla de base de datos debe tener una clave primaria para Optim para insertar, actualizar, restaurar o suprimir datos de una tabla de base de datos. Optim utiliza claves primarias que están definidas en la base de datos. Sin embargo, también puede definir claves primarias de Optim para complementar las claves primarias en la base de datos.

Se necesita una clave primaria:

- En cualquier tabla que es visitada más de una vez durante un proceso; por ejemplo, una tabla hijo que tiene dos o más tablas padre referenciadas en la definición de acceso.
- Para habilitar la característica de apuntar y disparar para una tabla inicial.

Nota: Si una clave primaria no está definida y es necesaria para realizar una tarea específica, aparece un mensaje de error.

Tipos de claves primarias de Optim

Puede definir dos tipos de claves primarias:

- Una clave primaria **explícita** se aplica a una sola tabla.

- Una clave primaria **genérica** se aplica a las tablas que tienen el mismo nombre base, los mismos nombres de columna y las mismas especificaciones de atributo, pero tienen diferentes ID de creador.

No hay ninguna diferencia en la función ni en el aspecto entre las claves primarias genéricas y explícitas. Sin embargo, si una tabla tiene claves de ambos tipos, se utiliza la clave primaria explícita.

Convenios de denominación

El nombre totalmente calificado de una clave primaria es el mismo que el nombre totalmente calificado de la tabla de base de datos para la que se ha definido. Este nombre está formado por:
alias.idcreador.nombretabla.

alias Alias que identifica la base de datos donde reside la tabla (1 a 12 caracteres).

idcreador
ID de creador asignado a la tabla (1 a 64 caracteres).

nombretabla
Nombre de la tabla base (1 a 64 caracteres).

Nota:

- La longitud total combinada de las columnas para una clave primaria está limitada a 3584 bytes.

Creación de una clave primaria

Utilice el asistente Clave primaria nueva para crear una clave primaria nueva para una tabla.

Antes de crear una clave primaria, debe existir un alias de almacén de datos para la tabla.

Para crear una clave primaria:

1. En el Explorador de repositorios, pulse con el botón derecho en **Claves primarias de Optim** y pulse **Clave primaria nueva**. Se abre el asistente Clave primaria nueva.
2. Complete los pasos del asistente.
Debe seleccionar una tabla.

La clave primaria nueva se visualiza en el Editor de claves primarias.

Debe utilizar el Editor de claves primarias para seleccionar columnas de clave primaria para la tabla.

Edición de una clave primaria

Utilice el Editor de claves primarias para editar una clave primaria para una tabla. Puede seleccionar columnas de clave y definir una clave primaria genérica.

Selección de columnas de clave primaria

Utilice el Editor de claves primarias para seleccionar columnas de clave primaria para una tabla. También puede establecer el orden de las columnas de clave.

Para seleccionar las columnas de clave primaria para una tabla:

1. En el Explorador de repositorios, expanda el nodo **Claves primarias de Optim** y efectúe una doble pulsación en la relación que desea editar. Se abre el Editor de claves primarias.
2. En la lista **Columnas disponibles**, seleccione las columnas de clave.
3. Pulse >> para mover las columnas a la lista **Columnas de clave**.
Para establecer el orden de las columnas de clave:
 - a. En la lista **Columnas de clave**, seleccione una columna que desee mover.

- b. Pulse **Subir** para mover una columna hacia arriba en el orden, o bien **Bajar** mover una columna hacia abajo en el orden.
4. Guarde la clave primaria.

Creación de una clave primaria genérica

Utilice el Editor de claves primarias para crear una clave primaria genética de Optim.

Algunas bases de datos contienen conjuntos de tablas que son idénticas, excepto por el ID de creador. En lugar de definir una clave primaria explícita para cada tabla, defina una clave primaria genérica para todas las tablas que tienen el mismo nombre base, independientemente del ID de creador.

Para crear una clave primaria genérica de Optim:

1. En el Explorador de repositorios, expanda el nodo **Claves primarias de Optim** y efectúe una doble pulsación en la relación que desea editar. Se abre el Editor de claves primarias.
2. Pulse **Genérica**.
3. Guarde la clave primaria.

Trabajar con correlaciones de tablas

Utilice una correlación de tablas para definir especificaciones para correlacionar tablas de origen y destino de datos compatibles. Puede correlacionar tablas que tienen diferentes nombres, modificar nombres de tablas, excluir tablas de un proceso o incluir correlaciones de columnas para un mayor control sobre los datos.

Utilice una correlación de tablas para:

- Dirigir la colocación de los datos en un servicio de conversión, inserción o carga.
- Excluir una o más tablas de un servicio de conversión, inserción o carga.
- Incluir una correlación de columnas.

Dependiendo del proceso para el que está utilizando una correlación de tablas, los dos conjuntos de tablas se denominan tablas de origen y tablas de destino:

- Las **tablas de origen** son tablas extraídas que contienen datos que se utilizarán en un proceso de conversión, inserción o carga.
- Las **tablas de destino** son las tablas en las que los datos se convertirán, insertarán o cargarán.

Nota: Las tablas coincidentes pueden tener diferentes nombres o ID de creador.

Las correlaciones de tablas se almacenan en el repositorio de Optim o se incluyen en un servicio de gestión de datos.

Como opción, puede utilizar una correlación de columnas para cualquier par de tablas en una correlación de tablas. Las correlaciones de columnas identifican y buscan coincidencias de columnas en un par de tablas. Una correlación de columnas debe utilizarse cuando los nombres o los atributos de columnas son distintos o cuando se necesitan transformaciones de datos.

Convenios de denominación

El nombre totalmente calificado de una correlación de tablas tiene el formato: *identificador.nombre*.

identificador

Identificador asignado a la correlación de tablas (1 a 8 caracteres).

nombre Nombre asignado a la correlación de tablas (1 a 12 caracteres).

Resulta útil utilizar un conjunto lógico de convenios de denominación para identificar el uso de cada uno de ellos y organizar las definiciones con el fin de facilitar el acceso.

Creación de una correlación de tablas

Utilice el asistente Nueva correlación de tablas para crear una nueva correlación de tablas.

Antes de crear una correlación de tablas, debe existir un alias de almacén de datos de archivo para la tabla de origen.

Para crear una correlación de tablas:

1. Expanda una carpeta en el Explorador de repositorios para que contenga la correlación de tablas, pulse con el botón derecho del ratón en **Correlaciones de tablas** y pulse **Nueva correlación de tablas**. Se abre el asistente Nueva correlación de tablas.
2. Complete los pasos del asistente.
Debe seleccionar un alias de almacén de datos de archivo para la tabla de origen.

La nueva correlación de tablas se visualiza en el Editor de correlaciones de tablas.

Debe utilizar el editor para definir los almacenes de datos de destino y los esquemas antes de guardar la correlación de tablas.

Edición de una correlación de tablas

Utilice el Editor de correlaciones de tablas para editar datos de destino y añadir una correlación de columnas.

Definición de datos de destino predeterminados en una correlación de tablas

Utilice el Editor de correlaciones de tablas para definir almacenes de datos de destino predeterminados y esquemas en una correlación de tablas.

Antes de definir datos de destino predeterminados de correlación de tablas, debe existir un alias de almacén de datos para los datos de destino.

Para definir datos de destino predeterminados en una correlación de tablas:

1. Expanda una carpeta en el Explorador de repositorios que contiene la correlación de tablas, expanda el nodo **Correlaciones de tablas** y efectúe una doble pulsación en la correlación de tablas. Se abre el Editor de correlaciones de tablas.
2. Seleccione la ficha **Correlación de tablas**.
3. En el área **Alias del almacén de datos y correlación de esquemas**, seleccione la fila que contiene el almacén de datos de origen.
4. Para definir un alias de almacén de datos de destino, pulse la celda **Alias de almacén de datos de destino** y seleccione el alias de almacén de datos de destino en la lista.
5. Para definir un esquema de destino, pulse la celda **Esquema de destino** y seleccione el esquema de destino en la lista.
6. Guarde la correlación de tablas.

Edición de datos de destino en una correlación de tablas

Utilice el Editor de correlaciones de tablas para editar almacenes de datos, esquemas y tablas de destino para las tablas de origen en una correlación de tablas.

Antes de editar datos de destino de correlación de tablas, debe existir un alias de almacén de datos de destino predeterminado para los datos de destino.

Para editar los datos de destino de una correlación de tablas:

1. Expanda una carpeta en el Explorador de repositorios que contiene la correlación de tablas, expanda el nodo **Correlaciones de tablas** y efectúe una doble pulsación en la correlación de tablas. Se abre el Editor de correlaciones de tablas.
2. Seleccione la ficha **Correlación de tablas**.
3. En el área **Correlación de tablas**, seleccione la fila que contiene la tabla de origen.
4. Para definir un alias de almacén de datos de destino, pulse la celda **Almacén de datos de destino** y especifique el nombre de un alias de almacén de datos de destino.
5. Para definir un esquema de destino, pulse la celda **Esquema de destino** y seleccione el esquema de destino en la lista.
6. Para definir una tabla de destino, pulse la celda **Tabla de destino** y seleccione la tabla de destino en la lista.
7. Guarde la correlación de tablas.

Adición de una correlación de columnas a una correlación de tablas

Utilice el Editor de correlaciones de tablas para añadir una correlación de columnas a un par de tablas correlacionadas.

Para añadir una correlación de columnas a una correlación de tablas:

1. Expanda una carpeta en el Explorador de repositorios que contiene la correlación de tablas, expanda el nodo **Correlaciones de tablas** y efectúe una doble pulsación en la correlación de tablas. Se abre el Editor de correlaciones de tablas.
2. Seleccione la ficha **Correlación de tablas**.
3. En el área **Correlación de tablas**, seleccione la fila que contiene las tablas para la correlación de columnas.
4. Pulse **Añadir correlación de columnas**. Se abrirá la ventana Nueva correlación de columnas con una lista de correlaciones de columnas que contienen las tablas seleccionadas.
5. Seleccione una correlación de columnas o seleccione **Crear una nueva correlación de columnas** para crear una correlación de columnas en función de las tablas seleccionadas. Pulse **Aceptar**.
 - a. Si opta por crear una nueva correlación de columnas, el Editor de correlaciones de columnas se abre y se llena con las columnas de las tablas seleccionadas.
 - b. Edite y guarde la nueva correlación de columnas.
6. Guarde la correlación de tablas.

Trabajar con correlaciones de columnas

Una correlación de columnas proporciona las especificaciones necesarias para encontrar coincidencias con columnas o para excluir columnas del proceso en un servicio de gestión de datos. Los servicios de conversión, inserción y carga deben hacer referencia a una correlación de tablas, que puede hacer referencia a una o más correlaciones de columnas. Puede utilizar una correlación de columnas para definir transformaciones de datos con funciones o procedimientos de correlación de columnas.

Una correlación de columnas debe utilizarse cuando los nombres o los atributos de columnas son distintos, cuando se necesitan transformaciones de datos o cuando se excluyen una o más columnas del proceso. Una correlación de columnas referenciada en un servicio de conversión, inserción o carga puede modificar datos, vencer fechas o convertir divisas. Puede definir transformaciones de datos utilizando funciones o procedimientos de correlación de columnas.

Cuando cree una nueva correlación de columnas, debe elegir un alias de almacén de datos de archivo como origen de las columnas que desea correlacionar. De forma parecida, debe especificar un alias de almacén de datos para los datos de destino.

Las correlaciones de columnas almacenadas en el repositorio de Optim están disponibles para su reutilización o para compartir con otros usuarios. Una correlación de columnas local se almacena como parte de un servicio de gestión de datos y, en cambio, no está disponible para otros servicios. Si la correlación de tablas asociada es local para un servicio, tanto la correlación de tablas como la correlación de columnas sólo están disponibles para el servicio específico.

Convenios de denominación

El nombre totalmente calificado de una correlación de columnas tiene dos partes: *identificador.nombre*.

identificador

Identificador asignado a la correlación de columnas (1 a 8 caracteres).

nombre Nombre asignado a la correlación de columnas (1 a 12 caracteres).

Al crear correlaciones de columnas, resulta útil utilizar un conjunto lógico de convenios de denominación para identificar y organizar definiciones con el fin de facilitar el acceso.

Reglas de compatibilidad para correlaciones de columnas

Las siguientes clases de datos y tipos de datos asociados están soportados. Estas clases de datos son importantes para la compatibilidad de los datos cuando se especifican valores de columnas en las relaciones y las correlaciones de columnas.

Carácter

DBMS	Tipos de datos
DB2	CHAR, VARCHAR, CLOB
Oracle	CHAR, VARCHAR2, LONG, CLOB, NCLOB, NCHAR, NVARCHAR
Sybase ASE	CHAR, VARCHAR, TXT
SQL Server	CHAR, VARCHAR, TXT
Informix	CHAR, VARCHAR, TXT

Nota: Las columnas de caracteres de un solo byte no son compatibles con las columnas de caracteres Unicode o de varios bytes.

Numérico

DBMS	Tipos de datos
DB2	INTEGER, SMALLINT, DECIMAL, FLOAT, DOUBLE
Oracle	NUMBER, FLOAT
Sybase ASE	TINYINT, INT, SMALLINT, DECIMAL, FLOAT, REAL, MONEY, SMALL MONEY
SQL Server	TINYINT, INT, SMALLINT, DECIMAL, FLOAT, REAL, MONEY, SMALL MONEY
Informix	INTEGER, SMALLINT, DECIMAL, FLOAT, REAL, DOUBLE PRECISION, SMALLFLOAT, SERIAL, MONEY, NUMERIC

Binario

DBMS	Tipos de datos
DB2	CHAR (para datos de bits), VARCHAR (para datos de bits), BLOB
Oracle	RAW, LONG RAW
Sybase ASE	BINARY, VARBINARY, IMAGE
SQL Server	BINARY, VARBINARY, IMAGE

DBMS	Tipos de datos
Informix	BYTE

Booleano

DBMS	Tipos de datos
Sybase ASE	BOOLEAN (TRUE o FALSE)

Fecha y hora

DBMS	Tipos de datos
DB2	TIMESTAMP
Oracle	DATE, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE
Sybase ASE	DATETIME, SMALL DATE TIME
SQL Server	DATETIME, SMALL DATE TIME
Informix	DATE, DATETIME

Fecha

DBMS	Tipos de datos
DB2	DATE
Oracle	DATE
Informix	DATE

Hora

DBMS	Tipos de datos
DB2	TIME

Intervalo

DBMS	Tipos de datos
Oracle	YEAR/MONTH INTERVAL, DAY/SECOND INTERVAL
Informix	YEAR/MONTH INTERVAL, DAY/TIME INTERVAL

Creación de una correlación de columnas

Utilice el asistente Nueva correlación de columnas para crear una nueva correlación de columnas.

Antes de crear una correlación de columnas, debe existir un alias de almacén de datos de archivo para los datos de origen y destino.

Para crear una correlación de columnas:

1. Expanda una carpeta en el Explorador de repositorios para que contenga la correlación de columnas, pulse con el botón derecho del ratón en **Correlaciones de columnas** y pulse **Nueva correlación de columnas**. Se abre el asistente Nueva correlación de columnas.
2. Complete los pasos del asistente.
Debe seleccionar un alias de almacén de datos de archivo y una tabla para los datos de origen. También debe seleccionar un alias de almacén de datos y una tabla para los datos de destino.

La nueva correlación de columnas se visualiza en el Editor de correlaciones de columna.

Edición de una correlación de columnas

Utilice el Editor de correlaciones de columnas para editar una correlación de columnas y aplicar funciones de enmascaramiento de datos.

Aplicación de una función de enmascaramiento de datos

Utilice el Editor de correlaciones de columnas para aplicar y editar una función para una columna.

Para aplicar una función de enmascaramiento de datos:

1. Expanda una carpeta en el Explorador de repositorios que contiene la correlación de columnas, expanda el nodo **Correlaciones de columnas** y efectúe una doble pulsación en la correlación de columnas. Se abre el Editor de correlaciones de columnas.
2. Seleccione la columna para la política.
3. Pulse **Aplicar función**. Se abre la ventana Aplicar función.
4. Seleccione la función que desea aplicar. Pulse **Aceptar**. El nombre de la función aparece en la columna asociada y el editor de funciones se abre en el Editor de correlaciones de columnas.
5. Dependiendo de la función, siga uno de estos pasos:
Seleccione la ficha de expresión de función y edite la expresión de función.
Seleccione una pestaña de opciones y seleccione opciones para la función.
6. Guarde la correlación de columnas.

Correlación de una columna de origen

Utilice el Editor de correlaciones de columnas para correlacionar una columna de origen con un destino. Las columnas de origen y destino con el mismo nombre y tipos de datos compatibles se correlacionan automáticamente.

Para correlacionar una columna de origen:

1. Expanda una carpeta en el Explorador de repositorios que contiene la correlación de columnas, expanda el nodo **Correlaciones de columnas** y efectúe una doble pulsación en la correlación de columnas. Se abre el Editor de correlaciones de columnas.
2. Pulse en una columna de origen y seleccione un nombre de columna en la lista.
3. Guarde la correlación de columnas.

Trabajar con procedimientos de correlación de columnas

Una procedimiento de correlación de columnas es un procedimiento que se utiliza para enmascarar o transformar los datos de una columna cuando ejecuta un servicio. Como el nombre indica, debe añadir procedimientos de correlación de columnas a una correlación de columnas. Puede escribir procedimientos de correlación de columnas utilizando el lenguaje de script de Lua.

Crear un procedimiento de correlación de columnas:

Utilice el editor de scripts de Lua para crear un procedimiento de correlación de columnas.

Para crear un procedimiento de correlación de columnas:

1. Expanda una carpeta en el Explorador de repositorios que contiene la correlación de columnas, expanda el nodo **Correlaciones de columnas** y efectúe una doble pulsación en la correlación de columnas. Se abre el Editor de correlaciones de columnas.
2. Seleccione la columna para el procedimiento de correlación de columnas.
3. Pulse **Añadir procedimiento**. Se abre el editor de scripts de Lua.
4. Cree y guarde el procedimiento. La columna asociada con el procedimiento muestra **Procedimiento**.
5. Guarde la correlación de columnas.

Editar un procedimiento de correlación de columnas:

Utilice el editor de scripts de Lua para editar un procedimiento de correlación de columnas.

Para editar un procedimiento de correlación de columnas:

1. Expanda una carpeta en el Explorador de repositorios que contiene la correlación de columnas, expanda el nodo **Correlaciones de columnas** y efectúe una doble pulsación en la correlación de columnas. Se abre el Editor de correlaciones de columnas.
2. Seleccione la columna asociada con el procedimiento de correlación de columnas.
3. Pulse **Editar procedimiento**. Se abre el editor de scripts de Lua.
4. Edite y guarde el procedimiento.
5. Guarde la correlación de columnas.

Entrar valores para parámetros en un procedimiento de correlación de columnas:

Utilice el Editor de correlaciones de columnas para especificar valores para parámetros en un procedimiento de correlación de columnas.

Para especificar valores para parámetros en un procedimiento de correlación de columnas:

1. Expanda una carpeta en el Explorador de repositorios que contiene la correlación de columnas, expanda el nodo **Correlaciones de columnas** y efectúe una doble pulsación en la correlación de columnas. Se abre el Editor de correlaciones de columnas.
2. Seleccione la columna que contiene el procedimiento. El editor **Procedimiento** se abre en el Editor de correlaciones de columnas.
3. En el campo **Parámetros de procedimiento**, especifique valores separados por columnas para los parámetros. Por ejemplo: abc, def.
4. Guarde la correlación de columnas.

Capítulo 4. Diseño de servicios de gestión de datos

Utilice un servicio de gestión de datos para extraer, convertir, cargar o insertar datos. Un servicio utiliza definiciones de acceso para definir los datos que se van a extraer. Un servicio también utiliza correlaciones de tablas y correlaciones de columnas para correlacionar datos de origen y destino. Puede enmascarar datos aplicando una política de privacidad en una entidad procesada por un servicio. También puede transformar las solicitudes de proceso de un directorio Optim en servicios de gestión de datos. Utilice Optim Manager en modalidad incluida para probar los servicios.

Trabajar con servicios de extracción

Utilice un servicio de extracción para copiar un conjunto de filas relacionadas de una o más tablas y guardar las filas en un almacén de datos de archivo.

El servicio de extracción especifica el conjunto de parámetros necesarios para extraer datos y definiciones de objetos de las tablas de origen y el almacén de datos de archivo en el que almacenar la información extraída.

El servicio de extracción siempre extrae definiciones para tablas y columnas. Estas definiciones se utilizan para crear las tablas de destino, si es necesario. También puede elegir extraer otras definiciones de objetos, incluidas las claves primarias, relaciones e índices.

Convenios de denominación

El nombre totalmente calificado de un servicio de extracción consta de: *identificador.nombre*.

identificador

Identificador que sirve como prefijo para el nombre de servicio de extracción (de 1 a 8 caracteres).

nombre Nombre asignado al servicio de extracción (de 1 a 12 caracteres).

Al crear servicios de extracción, resulta útil utilizar un conjunto lógico de convenios de denominación para identificar el uso de cada uno de ellos y organizar las definiciones con el fin de facilitar el acceso.

Creación de un servicio de extracción

Utilice el asistente Nuevo servicio para crear un servicio de extracción.

Un servicio de extracción requiere una definición de acceso. Durante la creación de un servicio, puede seleccionar o crear una definición de acceso.

Para crear un servicio de extracción:

1. Expanda una carpeta en el Explorador de repositorios para que contenga el servicio de extracción, pulse con el botón derecho del ratón en **Servicios** y pulse **Nuevo servicio**. Se abre el asistente Nuevo servicio.
2. Complete los pasos del asistente.

Es preciso que seleccione o cree una definición de acceso. También debe especificar un nombre para el almacén de datos del archivo de destino.

El servicio de extracción nuevo se visualiza en el Editor de servicios de extracción.

Edición de un servicio de extracción

Utilice el Editor de servicios de extracción para especificar opciones de proceso del servicio de extracción, como por ejemplo objetos de datos a extraer, conversión, compresión de archivos y alteraciones temporales de variables.

Especificación de opciones de proceso de extracción

Utilice la pestaña **Propiedades de servicio** del Editor de servicios de extracción para especificar opciones de proceso del servicio de extracción.

Puede especificar las siguientes opciones de proceso del servicio de extracción:

- Gestionar el número de conexiones de base de datos. Aumentar las conexiones de base de datos mejora el rendimiento cuando se procesan grandes cantidades de datos ya que permite a varias hebras extraer filas simultáneamente.
- Limitar el número de filas extraídas.
- Permitir archivos adjuntos.
- Incluir información estadística en el informe del servicio de extracción.

Para especificar opciones de proceso del servicio de extracción:

1. Expanda una carpeta en el Explorador de repositorios que contiene el servicio de extracción, expanda el nodo **Servicios** y efectúe una doble pulsación en el servicio de extracción que desea editar. Se abre el Editor de servicios de extracción.
2. Seleccione la pestaña **Propiedades de servicio**.
3. Edite las opciones de proceso de servicio.
4. Guarde el servicio de extracción.

Especificación de objetos y opciones de agrupación

Utilice la pestaña **Datos y objetos** en el Editor de servicios de extracción para especificar objetos de datos a extraer.

Puede especificar las siguientes opciones:

- Indique si el servicio debe extraer sólo datos, sólo objetos o ambos.
- Seleccione los objetos que desea extraer.

Para especificar objetos de datos a extraer:

1. Expanda una carpeta en el Explorador de repositorios que contiene el servicio de extracción, expanda el nodo **Servicios** y efectúe una doble pulsación en el servicio de extracción que desea editar. Se abre el Editor de servicios de extracción.
2. Seleccione la ficha **Datos y objetos**.
3. Edite las opciones de proceso de servicio.
4. Guarde el servicio de extracción.

Especificación de opciones de conversión

Utilice la ficha **Conversión** en el Editor de servicios de extracción para especificar opciones de conversión. Puede convertir los datos que se extraen en el servicio.

Puede especificar las siguientes opciones:

- Indicar si el servicio debe convertir los datos extraídos.
- Indicar el número máximo de filas descartadas, que no pueden convertirse, que deben permitirse antes de detener el servicio.

Para especificar las opciones de conversión:

1. Expanda una carpeta en el Explorador de repositorios que contiene el servicio de extracción, expanda el nodo **Servicios** y efectúe una doble pulsación en el servicio de extracción que desea editar. Se abre el Editor de servicios de extracción.
2. Seleccione la ficha **Conversión**.
3. Edite las opciones de proceso de servicio.
4. Guarde el servicio de extracción.

Especificación de opciones de compresión de archivos

Utilice la ficha **Opciones de compresión de archivos** en el Editor de servicios de extracción y especifique opciones para comprimir el almacén de datos de archivo o las tablas extraídas.

Puede especificar las siguientes opciones:

- Indicar si el archivo o las tablas específicas están comprimidos.
- Para cada tabla, puede optar por utilizar un umbral de compresión o comprimir tanto como sea posible. El valor umbral es la cantidad mínima de reducción del tamaño que se espera conseguir comprimiendo la tabla. Especifique un valor en el rango de 1 - 99 para establecer un valor de umbral para dicha tabla.

Los siguientes métodos de compresión están disponibles para un almacén de datos de archivo:

Compresión en línea

Los datos se comprimen a medida que se extraen y antes de que se graben en el almacén de datos de archivo. La compresión en línea tiene menos E/S, en comparación con la compresión posterior, pero utiliza recursos de base de datos para la duración del servicio de extracción.

La compresión en línea requiere menos recursos de almacenamiento durante el proceso de extracción cuando se compara con la compresión posterior.

Compresión posterior

Los datos se extraen y se graban en un almacén de datos de archivo sin comprimir. En un segundo paso, Optim lee el almacén de datos de archivo sin comprimir y graba una versión comprimida del almacén de datos de archivo. La ventaja de la compresión posterior es que las conexiones de base de datos se cierran antes con la compresión posterior que con la compresión en línea. Sin embargo, con la compresión posterior aumenta el tiempo transcurrido total porque el almacén de datos de archivo sin comprimir debe cerrarse, leerse y después debe crearse una nueva versión comprimida.

Los sitios que están preocupados por la contención de recursos de la base de datos pueden encontrar útil la compresión posterior ya que reduce el tiempo durante el cual se necesitan recursos de la base de datos. Sin embargo, la compresión posterior aumenta el tiempo transcurrido y los requisitos de almacenamiento para procesar el servicio de extracción. Aunque se necesita un mayor almacenamiento durante la operación de compresión, dicho almacenamiento temporal necesario se liberará cuando haya finalizado la compresión.

Para especificar opciones de compresión de archivos:

1. Expanda una carpeta en el Explorador de repositorios que contiene el servicio de extracción, expanda el nodo **Servicios** y efectúe una doble pulsación en el servicio de extracción que desea editar. Se abre el Editor de servicios de extracción.
2. Seleccione la ficha **Opciones de compresión de archivos**.
3. Edite las opciones de proceso de servicio.
4. Guarde el servicio de extracción.

Especificación de opciones de variables

Utilice la ficha **Variables** en el Editor de servicios de extracción para alterar temporalmente los valores predeterminados de las variables.

Para especificar opciones de variables:

1. Expanda una carpeta en el Explorador de repositorios que contiene el servicio de extracción, expanda el nodo **Servicios** y efectúe una doble pulsación en el servicio de extracción que desea editar. Se abre el Editor de servicios de extracción.
2. Seleccione la ficha **Variables**.
3. Especifique valores para alterar temporalmente los valores predeterminados.
4. Guarde el servicio de extracción.

Cambiar una definición de acceso para un servicio de extracción

Utilice el asistente Cambiar definición de acceso para cambiar una definición de acceso asociada a un servicio de extracción.

Para cambiar una definición de acceso asociada a un servicio de extracción:

1. Expanda una carpeta en el Explorador de repositorios que contiene el servicio de extracción, expanda el nodo **Servicios** y efectúe una doble pulsación en el servicio de extracción que desea editar. Se abre el Editor de servicios de extracción.
2. Pulse **Cambiar**. Se abre el asistente Cambiar definición de acceso.
3. Complete los pasos del asistente.
Seleccione una definición de acceso o elija crear una definición de acceso local.
4. Guarde el servicio de extracción.

Trabajar con servicios de conversión

Utilice un servicio de conversión para transformar los datos de un almacén de datos de archivo. Puede transformar datos para garantizar la privacidad de los datos o para transformar sistemáticamente datos para satisfacer los requisitos de prueba de la aplicación.

Especifique una correlación de tablas para buscar coincidencias entre las tablas del almacén de datos de archivo de origen y las tablas del almacén de datos de archivo de destino o para excluir tablas del servicio de conversión.

Utilice las correlaciones de columnas en la correlación de tablas para especificar qué datos hay que convertir y cómo deben convertirse.

Convenios de denominación

Nota: El nombre totalmente calificado de un servicio de conversión consta de las partes siguientes:
identificador.nombre.

identificador

Identificador que sirve como prefijo para el nombre de servicio de conversión (de 1 a 8 caracteres).

nombre Nombre asignado al servicio de conversión (de 1 a 12 caracteres).

Al crear el servicio de conversión, resulta útil utilizar un conjunto lógico de convenios de denominación para identificar el uso de cada uno de ellos y organizarlos con el fin de facilitar el acceso.

Creación de un servicio de conversión

Utilice el asistente Nuevo servicio para crear un servicio de conversión.

Un servicio de conversión requiere una correlación de tablas. Durante la creación de un servicio, puede seleccionar o crear una correlación de tablas.

Para crear un servicio de conversión:

1. Expanda una carpeta en el Explorador de repositorios para que contenga el servicio, pulse con el botón derecho del ratón en **Servicios** y pulse **Nuevo servicio**. Se abre el asistente Nuevo servicio.
2. Complete los pasos del asistente.
Es preciso que seleccione o cree una correlación de tablas. Puede escribir un nombre para el almacén de datos de archivo de destino o bien puede optar por sobrescribir el almacén de datos de archivo de origen.

El nuevo servicio de conversión se visualiza en el Editor de servicios de conversión.

Edición de un servicio de conversión

Utilice el Editor de servicios de conversión para especificar opciones de proceso del servicio de conversión.

Especificación de opciones de proceso de conversión

Utilice la ficha **Opciones de proceso** en el Editor de servicios de conversión para especificar opciones de proceso del servicio de conversión.

Puede especificar las siguientes opciones de proceso del servicio de conversión:

- Establecer un límite de filas descartadas.
- Comprimir el archivo de destino.
- Permitir archivos adjuntos.

Para especificar opciones de proceso del servicio de conversión:

1. Expanda una carpeta en el Explorador de repositorios que contiene el servicio de conversión, expanda el nodo **Servicios** y efectúe una doble pulsación en el servicio de conversión que desea editar. Se abre el Editor de servicios de conversión.
2. Seleccione la ficha **Opciones de proceso**.
3. Edite las opciones de proceso de servicio.
4. Guarde el servicio de conversión.

Cambiar una correlación de tablas para un servicio de conversión

Utilice el asistente Cambiar correlación de tablas para cambiar una correlación de tablas asociada a un servicio de conversión.

Para cambiar una correlación de tablas asociada un servicio de conversión:

1. Expanda una carpeta en el Explorador de repositorios que contiene el servicio de conversión, expanda el nodo **Servicios** y efectúe una doble pulsación en el servicio de conversión que desea editar. Se abre el Editor de servicios de conversión.
2. Pulse **Cambiar**. Se abre el asistente Cambiar correlación de tablas.
3. Complete los pasos del asistente.
Seleccione una correlación de tablas o elija crear una correlación de tablas local.
4. Guarde el servicio de conversión.

Trabajar con servicios de inserción

Utilice un servicio de inserción para insertar los datos almacenados en un almacén de datos de archivo en una base de datos de destino.

Utilice las correlaciones de tablas y las correlaciones de columnas para correlacionar los datos del origen con los del destino. Debe utilizar una correlación de tablas para especificar el destino de las tablas en el almacén de datos de archivo. También puede utilizar las correlaciones de columnas para una o más tablas

de destino. Las correlaciones de columnas le permiten especificar los datos de origen para cada columna y, opcionalmente, transformar los datos antes de insertarlos.

Si las tablas del almacén de datos de archivo no existen en el destino, utilice la utilidad de creación para crearlas.

Convenios de denominación

El nombre totalmente calificado de un servicio de inserción se compone de: *identificador.nombre*.

identificador

Identificador que sirve como prefijo para el nombre de servicio de inserción (de 1 a 8 caracteres).

nombre Nombre asignado al servicio de inserción (1 a 12 caracteres).

Al crear servicios de inserción, resulta útil utilizar un conjunto lógico de convenios de denominación para identificar el uso de cada uno de ellos y organizarlos con el fin de facilitar el acceso.

Creación de un servicio de inserción

Utilice el asistente Nuevo servicio para crear un servicio de inserción.

Un servicio de inserción requiere una correlación de tablas. Durante la creación de un servicio, puede seleccionar o crear una correlación de tablas.

Para crear un servicio de inserción:

1. Expanda una carpeta en el Explorador de repositorios para que contenga el servicio, pulse con el botón derecho del ratón en **Servicios** y pulse **Nuevo servicio**. Se abre el asistente Nuevo servicio.
2. Complete los pasos del asistente.

Es preciso que seleccione o cree una correlación de tablas.

El nuevo servicio de inserción se visualiza en el Editor de servicios de inserción.

Edición de un servicio de inserción

Utilice el Editor de servicios de inserción para especificar opciones de proceso del servicio de inserción. Puede especificar opciones para insertar y suprimir filas y también para manejar desencadenantes y restricciones.

Especificación de opciones de proceso de inserción

Utilice la ficha **Opciones de proceso** del Editor de servicios de inserción para especificar opciones de proceso del servicio de inserción.

Puede especificar las siguientes opciones de proceso:

- Seleccionar el tipo de proceso que desea realizar. Especifique los parámetros para bloquear tablas, establecer una frecuencia de confirmación y establecer un límite de filas descartadas.
- Suprimir filas de todas las tablas o de las tablas especificadas. La supresión es útil para restaurar datos durante las pruebas. Si una fila no puede suprimirse por algún motivo, todas las filas suprimidas hasta la última confirmación se restauran y se detiene el proceso de inserción.
- Inhabilitar desencadenantes de base de datos.
- Inhabilitar restricciones de base de datos.

El proceso de filas incluye las siguientes opciones:

Insertar

Inserta filas nuevas en las tablas.

- Si la clave primaria de una fila en los datos de origen *no coincide* con la clave primaria de una fila en la tabla de destino, se inserta la fila.
- Si la clave primaria de una fila en los datos de origen *coincide* con la clave primaria de una fila en la tabla de destino, la fila se pasa por alto y se marca como descartada.

Mixto Actualiza, inserta o actualiza/inserta de acuerdo con lo que ha seleccionado para cada tabla en la ventana Especificar por tabla. Para utilizar la ventana Especificar por tabla, pulse **Especificar por tabla** y seleccione una opción de proceso para cada tabla.

- Si selecciona **Mixto** y no especifica selecciones en la ventana Especificar por tabla, o establece todas las tablas en la misma selección, la opción de proceso cambia para indicar el proceso utilizado para todas las tablas.

Nota: No puede seleccionar **Mixto** si la opción **Todas las filas se suprimirán** está seleccionada en el área **Opciones de supresión de filas**.

Actualizar

Actualiza filas de las tablas. Las tablas deben tener una clave primaria.

- Si la clave primaria de una fila en los datos de origen *coincide* con la clave primaria de una fila en la tabla de destino, la fila se actualiza.
- Si la clave primaria de una fila en los datos de origen *no coincide* con la clave primaria de una fila en la tabla de destino, la fila se indica como anómala.

Nota: No puede seleccionar **Actualizar** si la opción **Todas las filas se suprimirán** está seleccionada en el área **Opciones de supresión de filas**.

Actualizar/Insertar

Actualiza e inserta filas en las tablas. Las tablas deben tener una clave primaria.

- Si la clave primaria de una fila en los datos de origen *no coincide* con la clave primaria de una fila en la tabla de destino, se inserta la fila.
- Si la clave primaria de una fila en los datos de origen *coincide* con la clave primaria de una fila en la tabla de destino, la fila se actualiza.

Nota: No puede seleccionar **Actualizar/Insertar** si la opción **Todas las filas se suprimirán** está seleccionada en el área **Opciones de supresión de filas**.

Para especificar opciones de proceso de inserción:

1. Expanda una carpeta en el Explorador de repositorios que contiene el servicio de inserción, expanda el nodo **Servicios** y efectúe una doble pulsación en el servicio de inserción que desea editar. Se abre el Editor de servicios de inserción.
2. Seleccione la ficha **Opciones de proceso**.
3. Edite las opciones de proceso de servicio.
4. Guarde el servicio de inserción.

Cambiar una correlación de tablas para un servicio de inserción

Utilice el asistente Cambiar correlación de tablas para cambiar una correlación de tablas asociada a un servicio de inserción.

Para cambiar una correlación de tablas asociada a un servicio de inserción:

1. Expanda una carpeta en el Explorador de repositorios que contiene el servicio de inserción, expanda el nodo **Servicios** y efectúe una doble pulsación en el servicio de inserción que desea editar. Se abre el Editor de servicios de inserción.
2. Pulse **Cambiar**. Se abre el asistente Cambiar correlación de tablas.
3. Complete los pasos del asistente.
 Seleccione una correlación de tablas o elija crear una correlación de tablas local.

4. Guarde el servicio de inserción.

Trabajar con servicios de carga

Utilice un servicio de carga para transformar el contenido de un almacén de datos de archivo en un formato que sea apropiado para un cargador DBMS determinado y, a continuación, si se especifica, inicie el programa de utilidad de carga de base de datos correspondiente.

El servicio de carga genera un archivo de datos en el formato correcto para cada tabla del almacén de datos de archivo y un archivo SQL o un archivo BAT (ejecución por lotes), dependiendo del DBMS, que contiene la sintaxis necesaria para iniciar el cargador de base de datos.

Un servicio de carga contiene los parámetros utilizados para preparar los datos para un cargador DBMS así como las instrucciones necesarias para procesar la carga. Especifique una correlación de tablas en el servicio de carga para correlacionar el destino de los datos que desea cargar. Utilice correlaciones de columnas opcionales en el servicio de carga para transformar los datos antes de cargarlos.

Los datos que deben cargarse debe estar contenidos en un almacén de datos de archivo.

Carga frente a inserción

Optim puede mover datos a una base de datos utilizando un servicio de carga o de inserción. Tenga en cuenta las siguientes consideraciones cuando decida el método que va a utilizar:

- El volumen de datos y la velocidad de utilizar el programa de utilidad de carga de base de datos pueden compensar las ventajas del servicio de inserción.
- Los datos pueden contener ciclos de integridad referencial (RI) que sobrepasen la capacidad del servicio de inserción para insertar satisfactoriamente todos los datos.
- El programa de utilidad de carga de base de datos requiere el control exclusivo de la base de datos e impide el acceso de usuario durante el servicio de carga. La base de datos está disponible para otros usuarios mientras se lleva a cabo el servicio de inserción.
- El programa de utilidad de carga de base de datos inserta datos nuevos o sustituye los datos existentes. El servicio de inserción permite realizar el proceso de actualización/inserción en un solo paso.

Nombres de archivos de salida

Un servicio de carga genera los siguientes tipos de archivos para apoyar el proceso de carga del programa de utilidad de base de datos.

Archivos de datos

Los archivos de datos contienen los datos que desea cargar, preparados en el formato adecuado para el DBMS que está utilizando. Optim genera un archivo de datos para cada tabla del almacén de datos de archivo. Los archivos de datos se denominan igual que el almacén de datos de archivo, pero contienen extensiones de nombre de archivo numeradas secuencialmente. Por ejemplo, un almacén de datos de archivo denominado *demo.xf* que contiene tres tablas generará tres archivos de datos denominados: *demo.001*, *demo.002*, y *demo.003*.

Archivos de mensajes

Los archivos de mensajes contienen información que el programa de utilidad de carga de base de datos genera durante el servicio de carga. Normalmente, hay un archivo de mensajes para todo el servicio de carga. El archivo de mensajes se denomina igual que el almacén de datos de archivo, pero contiene la extensión *.msg*. Por ejemplo, si el almacén de datos de archivo se denomina *demo.xf*, el archivo de mensajes se denomina *demo.msg*.

Archivo SQL

Para DB2, un archivo SQL se genera con una sentencia para cada tabla de destino que contiene la

sintaxis de cargador para ejecutar manualmente el cargador. El archivo SQL se denomina igual que el almacén de datos de archivo, pero tiene la extensión *.sql*.

Archivo BAT

Para Oracle, Sybase ASE, SQL Server e Informix, se genera un archivo BAT que contiene la sintaxis para ejecutar manualmente el cargador para cada tabla. Se genera un archivo BAT para cada alias de almacén de datos especificado en la correlación de tablas. Cada archivo BAT reside en el directorio con el correspondiente almacén de datos de archivo convertido. Si opta por ejecutar manualmente el cargador, el archivo BAT debe editarse (en el Bloc de notas, por ejemplo) para sustituir una serie de ocho signos de interrogación por la información de la contraseña específica (excepto Informix).

Además, se genera un archivo de formato para cada archivo de datos. Un archivo de formato tiene el mismo nombre que el archivo de datos correspondiente, excepto que la extensión de nombre de archivo es diferente. Si hay menos de 500 tablas para cargar, la extensión del nombre de archivo de formato se incrementa numéricamente en 500 respecto a la extensión del nombre de archivo de datos. (Por ejemplo, si hay tres archivos de datos denominados *demo.001*, *demo.002* y *demo.003*, los archivos de formato correspondientes se denominan *demo.501*, *demo.502* y *demo.503*, respectivamente). Si hay más de 500 tablas, se utiliza un algoritmo más complejo para la generación de extensiones de archivo.

Nota: Si el servidor de archivos no permite nombres de archivo largos con más de 8 caracteres y el almacén de datos de archivo tiene un nombre largo, el servicio de carga fallará. La mejor solución es evitar la utilización de nombres de archivo largos para los almacenes de datos de archivo. Si es necesario, puede copiar y renombrar un archivo antes de utilizarlo para un servicio de carga.

Convenios de denominación

El nombre totalmente calificado de un servicio de carga consta de: *identificador.nombre*.

identificador

Identificador que sirve como prefijo para el nombre de servicio (de 1 a 8 caracteres).

nombre Nombre asignado al servicio (1 a 12 caracteres).

Al crear servicios de carga, resulta útil utilizar un conjunto lógico de convenios de denominación para identificar el uso de cada uno de ellos y organizarlos con el fin de facilitar el acceso.

Creación de un servicio de carga

Utilice el asistente Nuevo servicio para crear un servicio de carga.

Un servicio de carga requiere una correlación de tablas. Durante la creación de un servicio, puede seleccionar o crear una correlación de tablas.

Para crear un servicio de carga:

1. Expanda una carpeta en el Explorador de repositorios para que contenga el servicio de extracción, pulse con el botón derecho del ratón en **Servicios** y pulse **Nuevo servicio**. Se abre el asistente Nuevo servicio.
2. Complete los pasos del asistente.
Es preciso que seleccione o cree una correlación de tablas.

El nuevo servicio de carga se visualiza en el Editor de servicios de carga.

Edición de un servicio de carga

Utilice el Editor de servicios de carga para especificar opciones de proceso del servicio de carga. Puede especificar opciones para el proceso de carga y para el cargador DBMS asociado a cada almacén de datos de destino.

Especificación de opciones de proceso de carga

Utilice la ficha **Opciones de proceso** en el Editor de servicios de carga para especificar opciones de proceso del servicio de carga.

Puede especificar las siguientes opciones:

- Seleccionar cómo ejecutar varios cargadores DBMS cuando se utiliza más de un alias de almacén de datos: en paralelo (para ejecutar diferentes cargadores DBMS al mismo tiempo) o en secuencia (para ejecutar diferentes cargadores DBMS uno tras otro).
- Detener un cargador DBMS si se produce un error. Si varios cargadores DBMS se ejecutan en secuencia, el proceso se detiene para el cargador DBMS que tiene un error y todos los cargadores DBMS posteriores.
- Inhabilitar desencadenantes de base de datos.
- Inhabilitar restricciones de base de datos.

Para especificar opciones de proceso del servicio de carga:

1. Expanda una carpeta en el Explorador de repositorios que contiene el servicio de carga, expanda el nodo **Servicios** y efectúe una doble pulsación en el servicio de carga que desea editar. Se abre el Editor de servicios de carga.
2. Seleccione la ficha **Opciones de proceso**.
3. Edite las opciones de proceso de servicio.
4. Guarde el servicio de carga.

Especificación de opciones de cargador DBMS

Utilice la ficha **Opciones de carga** del Editor de servicios de carga y especifique opciones para cada cargador DBMS.

Puede especificar las siguientes opciones:

- Una modalidad de proceso.
- Opciones que son específicas de cada alias de almacén de datos de destino.

En función del DBMS del almacén de datos de archivo de destino seleccionado, están disponibles las siguientes modalidades de proceso:

Insertar

Inserta filas del almacén de datos de archivo de origen en las tablas de destino vacías. Si las tablas de destino contienen datos, el cargador devuelve un error.

Sustituir

Borra y sustituye todas las filas existentes en las tablas de destino por las filas del almacén de datos de archivo de origen. (**Sustituir** podría consumir significativamente más recursos que **Truncar** ya que no se realiza ningún registro).

Añadir

Inserta las filas del almacén de datos de archivo de origen en las tablas de destino. Si los valores de las claves primarias coinciden, las filas duplicadas se descartan o se insertan en la tabla de excepciones (si se ha especificado).

Truncar

Truncar es lo mismo que **Sustituir** pero la base de datos no registra las filas que se suprimen, y **Truncar** requiere que las restricciones RI estén inhabilitadas.

Para especificar opciones de cargador DBMS:

1. Expanda una carpeta en el Explorador de repositorios que contiene el servicio de carga, expanda el nodo **Servicios** y efectúe una doble pulsación en el servicio de carga que desea editar. Se abre el Editor de servicios de carga.
2. Seleccione la ficha **Opciones de carga**.
3. Seleccione el alias de almacén de datos de destino.
4. Edite las opciones de proceso de servicio.
5. Guarde el servicio de carga.

Cambiar una correlación de tablas para un servicio de carga

Utilice el asistente Cambiar correlación de tablas para cambiar una correlación de tablas asociada a un servicio de carga.

Para cambiar una correlación de tablas asociada un servicio de carga:

1. Expanda una carpeta en el Explorador de repositorios que contiene el servicio de carga, expanda el nodo **Servicios** y efectúe una doble pulsación en el servicio de carga que desea editar. Se abre el Editor de servicios de carga.
2. Pulse **Cambiar**. Se abre el asistente Cambiar correlación de tablas.
3. Complete los pasos del asistente.
Seleccione una correlación de tablas o elija crear una correlación de tablas local.
4. Guarde el servicio de carga.

Probar un servicio de gestión de datos

Utilice Optim Manager en modalidad incluida para probar un servicio de gestión de datos.

Para probar un servicio de gestión de datos:

1. Expanda una carpeta en el Explorador de repositorios que contiene el servicio de extracción y expanda el nodo **Servicios**.
2. Pulse el botón derecho del ratón en el servicio y pulse **Ejecutar servicio**. Optim Manager se abre y se muestra la ventana Ejecutar servicio.
3. Pulse **Ejecutar**. Para supervisar el progreso del servicio, utilice la ficha **Supervisión del servicio**.

Capítulo 5. Enmascaramiento de datos

Puede enmascarar datos como, por ejemplo, números de ID nacionales, número de tarjetas de crédito, fechas, valores numéricos e información personal. Utilice una correlación de columnas para especificar una función de enmascaramiento de datos o cree un procedimiento de correlación de columnas con un script LUA. Utilice un servicio de conversión para transformar los datos.

Aplicación de una función de enmascaramiento de datos

Utilice el Editor de correlaciones de columnas para aplicar y editar una función para una columna.

Para aplicar una función de enmascaramiento de datos:

1. Expanda una carpeta en el Explorador de repositorios que contiene la correlación de columnas, expanda el nodo **Correlaciones de columnas** y efectúe una doble pulsación en la correlación de columnas. Se abre el Editor de correlaciones de columnas.
2. Seleccione la columna para la política.
3. Pulse **Aplicar función**. Se abre la ventana Aplicar función.
4. Seleccione la función que desea aplicar. Pulse **Aceptar**. El nombre de la función aparece en la columna asociada y el editor de funciones se abre en el Editor de correlaciones de columnas.
5. Dependiendo de la función, siga uno de estos pasos:
 Seleccione la ficha de expresión de función y edite la expresión de función.
 Seleccione una pestaña de opciones y seleccione opciones para la función.
6. Guarde la correlación de columnas.

Funciones de enmascaramiento de datos

Las funciones de enmascaramiento de datos proporcionan diversos métodos para transformar o enmascarar datos confidenciales.

Funciones de búsqueda

Utilice las funciones de búsqueda para seleccionar valores de una tabla de búsqueda que se utilizan para llenar la tabla de destino. Utilice las funciones de búsqueda y búsqueda hash para seleccionar valores en función del valor de origen. Como alternativa, utilice la función de búsqueda aleatoria para seleccionar valores de una tabla de búsqueda sin tener en cuenta el valor de origen.

Función de búsqueda

La función de búsqueda obtiene el valor para una columna de destino de una tabla de búsqueda, según el valor de una columna de origen. Hay dos formatos de la función de búsqueda: una sola columna y varias columnas.

El formato de una sola columna inserta un valor en una columna de destino única. El formato de varias columnas inserta valores de diversas columnas de la tabla de búsqueda en las columnas de destino correspondientes.

Puede especificar la función de búsqueda de varias columnas para cualquier columna de origen que se sustituirá por un valor de la tabla de búsqueda, pero debe editar la correlación de columnas para eliminar los nombres de las columnas de origen restantes que también se sustituirán.

El parámetro *ignore* le permite ignorar la tabla de búsqueda y utilizar un valor de origen cuando una fila de una columna de origen especificada contiene un valor especificado (NULL, SPACES (para columnas CHAR) o VARCHAR de longitud cero).

Puede utilizar el parámetro *preserve* para ignorar la tabla de búsqueda y utilizar un valor de origen cuando una fila de una columna de origen especificada contiene un valor especificado (NULL, SPACES (para columnas CHAR) o VARCHAR de longitud cero). *preserve* también se puede utilizar para ignorar la tabla de búsqueda si una columna de origen no contiene un valor.

Si no se encuentra una coincidencia en la tabla de búsqueda, se informa de un error de conversión.

La sintaxis es:

```
LOOKUP ( [colbúsquedaorigen, | SRCSEARCH=(columnorigen1,...,columnorigenn)]
        [dest=(col1, coln) , ]
        nombretablalq ( {ColBúsquedaTablaBúsqueda
        | LKPSEARCH=(ColBúsquedaTablaBúsqueda1,...,ColBúsquedaTablaBúsquedan)},
        {value | values=(col1, coln) }
        [,cache | ,nocache ] )
        [,ignore=(nombrecol ( spaces, null, zero_len ), )
        | PRESERVE=( [ NOT_FOUND, ] nombrecol (spaces, null, zero_len),... ) ] )
```

colbúsquedaorigen

Para la búsqueda de una sola columna, nombre de la columna de la tabla de origen que contiene el valor de búsqueda (opcional). Si no se especifica, se utiliza el nombre de la columna de destino.

SRCSEARCH=

Para la búsqueda de varias columnas, nombres de las columnas de la tabla de origen que contienen los valores de búsqueda. Los nombres de columna deben ir entre paréntesis y deben separarse con comas.

dest= Nombres de las columnas de la tabla de destino en las que se insertan los valores de la tabla de búsqueda. (Se requiere para la búsqueda de varias columnas).

col1, coln

Nombres de las columnas de la tabla de destino. El orden de los nombres de columna debe corresponderse con las columnas de la tabla de búsqueda en el parámetro *values=*.

lktablename

Nombre de la tabla de búsqueda. Puede especificar el nombre de la tabla de búsqueda como *aliasbd.idcreador.nombretabla*, *idcreador.nombretabla* o *nombretabla*. Si no califica totalmente el nombre de tabla, se utilizan los calificadores para la tabla de destino.

ColBúsquedaTablaBúsqueda

Para la búsqueda de una sola columna, nombre de la columna de la tabla de búsqueda que contiene un valor que debe compararse con el valor de búsqueda de la columna de origen.

value Nombre de la columna en la tabla de búsqueda que contiene el valor de búsqueda convertido que debe insertarse en el destino. (Se requiere para la búsqueda de una sola columna).

values=

Nombres de las columnas de la tabla de búsqueda que contienen valores que deben insertarse en el destino. (Se requiere para la búsqueda de varias columnas).

col1, coln

Nombres de las columnas de la tabla de búsqueda. El orden de los nombres de columna deben corresponderse con las columnas de la tabla de destino en el parámetro *dest=*.

cache | nocache

Especifique *cache* (valor predeterminado) para mantener una tabla de valores de búsqueda encontrados en memoria o *nocache* para descartar los valores encontrados. La utilización de *cache* es más rápida cuando se recupera un valor muchas veces, pero requiere memoria adicional.

ignore=

Lista de columnas de origen con valores que se insertan en el destino en lugar del valor de búsqueda cuando la columna tiene una fila con un valor declarado (NULL, SPACES, ZERO o VARCHAR de longitud cero).

col El nombre de la columna de origen.

Para la búsqueda de una sola columna, especifique únicamente un nombre de columna.

Para la búsqueda de varias columnas, el orden de los nombres de columna debe corresponderse con las columnas de la tabla de destino en el parámetro *dest=*. El número de columnas debe ser igual al de las columnas en el parámetro *dest=* y al menos una columna debe incluir valores. Para no especificar valores para una columna, no especifique un valor. Por ejemplo, *coln()*.

null Ignore la tabla de búsqueda si la fila de la columna de origen tiene un valor null.

SPACES

Ignore la tabla de búsqueda si la fila de la columna de origen tiene un valor SPACES. Sólo para columnas CHAR.

ZERO_LEN

Ignore la tabla de búsqueda si la fila de la columna de origen tiene un valor VARCHAR de longitud cero.

preserve=

Lista de columnas de origen con valores que se insertan en el destino en lugar del valor de búsqueda cuando la columna de origen contiene un valor declarado (NOT_FOUND, NULL, SPACES o VARCHAR de longitud cero).

NOT_FOUND

Ignore la tabla de búsqueda si no se encuentra ninguna coincidencia para la fila de la columna de origen.

Nota:

preserve= e *ignore=* se excluyen mutuamente. *ignore=* dejará de utilizarse en un release futuro.

Los operandos *col*, *null*, *spaces* y *zero_len* tienen el mismo efecto cuando se utilizan con *preserve=* o *ignore=*.

Ejemplo de una sola columna

Utilice la función de búsqueda para convertir el valor de origen de una tabla de búsqueda en un valor correspondiente en otra tabla.

Por ejemplo, supongamos que la columna de origen, STATE, contiene abreviaturas de estados (por ejemplo, NJ) y la columna de destino debe contener el nombre completo del estado (en este ejemplo, **Nueva Jersey**). Una tabla de búsqueda denominada STATE_LOOKUP contiene una columna (CODE) para las abreviaturas o códigos de estado y una columna (NAME) para los nombres correspondientes.

Si desea obtener el valor para la columna de destino utilizando la tabla STATE_LOOKUP, especifique:

LOOKUP(STATE,STATE_LOOKUP(CODE,NAME))

La función de búsqueda busca un valor en la columna CODE de la tabla STATE_LOOKUP que coincida con el valor (NJ) en la columna STATE de la tabla de origen. Cuando se encuentra una coincidencia, la función inserta el valor correspondiente de la columna NAME (**New Jersey**) en la columna de destino.

Ejemplo de varias columnas

Utilice la función de búsqueda para insertar valores de las columnas de una fila de la tabla de búsqueda en las columnas de una fila de la tabla de destino, en función de un valor de una columna de origen.

Por ejemplo, en función de una columna de origen (SOC_SEC) que contiene números de seguridad social, puede sustituir los valores de las columnas de destino (FIRST_NAME y LAST_NAME) por los nombres de pila y apellidos de una tabla de búsqueda. Una tabla denominada NAME_LOOKUP contiene una columna (SSN) con los números de seguridad social de la tabla de origen así como las columnas (FIRST_MASK y LAST_MASK) para enmascarar los nombres correspondientes en el destino.

Para sustituir los nombres de la tabla de destino en función de un número de seguridad social, especifique:

```
LOOKUP(SOC_SEC,DEST=(FIRST_NAME, LAST_NAME),  
NAME_LOOKUP(SSN,VALUES=(FIRST_MASK, LAST_MASK)))
```

La función de búsqueda busca un valor en la columna SSN de la tabla NAME_LOOKUP que coincide con el valor de columna SOC_SEC de la tabla de origen. Cuando se encuentra una coincidencia, la función inserta los valores correspondientes de las columnas FIRST_MASK y LAST_MASK de la tabla de búsqueda en las columnas de destino correspondientes.

Ejemplo de ignorar

Utilice la sentencia siguiente para ampliar el ejemplo de una sola columna, en el que desea utilizar los valores *NULL* y *SPACES* de origen en lugar de los valores de la tabla de búsqueda:

```
LOOKUP(STATE,STATE_LOOKUP(CODE,NAME),  
IGNORE=(STATE(NULL,SPACES)))
```

Ejemplo de NoCache

Utilice la sentencia siguiente para ampliar el ejemplo de una sola columna, en el que no desea mantener una tabla de valores de búsqueda encontrados en la memoria.

```
LOOKUP(STATE,STATE_LOOKUP(CODE,NAME),NOCACHE)
```

Función de búsqueda hash

La función de búsqueda hash obtiene el valor para una columna de destino de una tabla de búsqueda, según un valor hash derivado de una columna de origen. La función de búsqueda hash le permite enmascarar datos de forma coherente cuando se utilizan las mismas tablas de origen y de búsqueda en cualquier entorno.

La columna de origen a la que se haya aplicado el hash no necesita ser una columna que se sustituirá por los valores de la tabla de búsqueda.

La función de búsqueda hash es sensible a las mayúsculas y minúsculas. Por ejemplo, el método hash considera los valores de origen John y JOHN como valores distintos. Puede utilizar el parámetro TRIM para convertir el valor de origen a mayúsculas antes de que se aplique hash.

Hay dos formatos de la función de búsqueda hash: una sola columna y varias columnas. El formato de una sola columna inserta un valor en una columna de destino única. El formato de varias columnas inserta valores de diversas columnas de la tabla de búsqueda en las columnas de destino correspondientes, en función de un único valor hash de una columna de origen.

Puede especificar la función de búsqueda hash de varias columnas para cualquier columna de origen que se sustituirá por los valores de la tabla de búsqueda, pero debe editar la correlación de columnas para eliminar los nombres de las columnas de origen restantes que también se sustituirán.

La tabla de búsqueda debe incluir una columna clave que contiene valores de número secuenciales sin espacios, y las columnas restantes contienen valores de sustitución. La columna clave debe ser un tipo de datos numérico. La tabla de búsqueda suele estar indexada. La función ejecuta hash en una columna de origen para derivar números secuenciales a partir de 1 hasta el valor máximo en la columna de clave de la tabla de búsqueda. El valor hash de la tabla de origen se compara con los números secuenciales de la tabla de búsqueda y los valores de la tabla de búsqueda correspondiente se insertan en el destino.

Si la columna de origen utilizada para derivar el valor hash contiene ciertos valores (NULL, espacios (para columnas CHAR), VARCHAR de longitud cero), el valor no es hash y se utilizan los siguientes valores reservados como claves de la tabla de búsqueda:

Valor de origen	Clave de la tabla de búsqueda
NULL	-1
espacios (CHAR o VARCHAR)	-2
VARCHAR de longitud cero	-3

La tabla de búsqueda debe incluir una fila para cada uno de estos números que le permita insertar un valor de búsqueda para cada uno de estos valores de origen. Si se encuentra uno de estos valores de origen y un número correspondiente no está en la tabla de búsqueda, se informa de un error de conversión.

El parámetro *ignore* le permite ignorar la tabla de búsqueda y utilizar un valor de origen cuando una fila de una columna de origen especificada contiene un valor especificado (NULL, SPACES (para columnas CHAR) o VARCHAR de longitud cero).

Puede utilizar el parámetro *preserve* para ignorar la tabla de búsqueda y utilizar un valor de origen cuando una fila de una columna de origen especificada contiene un valor especificado (NULL, SPACES (para columnas CHAR) o VARCHAR de longitud cero). *preserve* también se puede utilizar para ignorar la tabla de búsqueda si una columna de origen no contiene un valor.

El parámetro *trim* le permite especificar caracteres que se recortarán del valor de origen antes de que se aplique hash. Por ejemplo, si opta por recortar las comas de un valor de origen, el método hash considerará que los valores Smith, John y Smith John son el mismo valor. También puede utilizar este parámetro para convertir el valor de origen a mayúsculas antes de que se aplique hash.

Si el valor de origen se convierte a mayúsculas, los caracteres de recorte también se convierten a mayúsculas.

Puede utilizar el parámetro *seed* para variar el cálculo realizado por el algoritmo de hash. El valor hash de la columna de origen y el valor *seed* se emparejan con un número secuencial de la tabla de búsqueda para obtener el valor de sustitución para la columna de destino.

La sintaxis es:

```

HASH_LOOKUP( [columnorigen,] [trim=([caract1caract2 ] [\u]),]
dest=(col1, coln), nombretablabq (búsqueda,
{ valor | values=(col1, coln) } ) [ ,cache | ,nocache ]
[,ignore=(col (spaces, null, zero_len), )
| PRESERVE=( [ NOT_FOUND, ] nombrecol (spaces, null, zero_len), ) ] [,seed=n])

```

columnorigen

Nombre de la columna de la tabla de origen desde donde se derivan los valores hash (opcional). Si no se especifica, se utiliza el nombre de la columna de destino.

trim= Lista de caracteres que deben recortarse del valor de origen antes de que también se aplique hash como una opción para convertir el valor de origen a mayúsculas antes de que se aplique hash. Si el valor de origen resultante es NULL o se han recortado todos los espacios después de los caracteres, no se ejecutará hash en el valor de origen y se le asignará el valor reservado correspondiente (-1 ó -2).

caract1caract2...

Caracteres que se recortarán del valor de origen antes de que se aplique hash. La lista es sensible a las mayúsculas y minúsculas. Puede especificar un espacio o una coma como carácter. Después de la aparición inicial de un carácter, las apariciones adicionales en la lista se ignoran.

Para especificar una barra invertida “\” o un paréntesis derecho “)”, debe preceder el carácter con un carácter de escape de barra invertida. Por ejemplo, para especificar un paréntesis derecho, especifique: *trim=(\)*.

Sólo puede utilizar el carácter de escape con una barra invertida, un paréntesis derecho, o como parte del indicador de mayúsculas.

\u Indica que el valor de origen debe convertirse a mayúsculas antes de que se aplique hash. Los caracteres que deben recortarse también se convierten a mayúsculas.

dest= Nombres de las columnas de la tabla de destino en las que se insertan los valores de la tabla de búsqueda. (Se requiere para la búsqueda de varias columnas).

col1,coln

Nombres de las columnas de la tabla de destino. El orden de los nombres de columna debe corresponderse con las columnas de la tabla de búsqueda en el parámetro *values=*.

nombretablabq

Nombre de la tabla de búsqueda. Puede especificar el nombre de la tabla de búsqueda como **aliasbd.idcreador.nombretabla**, **idcreador.nombretabla** o **nombretabla**. Si no califica totalmente el nombre de tabla, se utilizan los calificadores para la tabla de destino.

búsqueda

Nombre de la columna de la tabla de búsqueda que contiene valores secuenciales que deben compararse con los valores hash de la columna de origen.

valor Nombre de la columna en la tabla de búsqueda que contiene el valor de búsqueda convertido que debe insertarse en el destino. (Se requiere para la búsqueda de una sola columna).

values=

Nombres de las columnas en la tabla de búsqueda que contienen valores que deben insertarse en el destino. (Se requiere para la búsqueda de varias columnas).

col1,coln

Nombres de las columnas de la tabla de búsqueda. El orden de los nombres de columna deben corresponderse con las columnas de la tabla de destino en el parámetro *dest=*.

cache | *nocache*

Especifique *cache* (valor predeterminado) para mantener una tabla de valores de búsqueda encontrados en memoria o *nocache* para descartar los valores encontrados. La utilización de *cache* es más rápida cuando se recupera un valor muchas veces, pero requiere memoria adicional.

ignore=

Lista de columnas de origen con valores que se insertan en el destino en lugar del valor de búsqueda cuando la columna tiene una fila con un valor declarado (NULL, SPACES, ZERO o VARCHAR de longitud cero).

col El nombre de la columna de origen.

Para la búsqueda de una sola columna, especifique únicamente un nombre de columna.

Para la búsqueda de varias columnas, el orden de los nombres de columna debe corresponderse con las columnas de la tabla de destino en el parámetro *dest=*. El número de columnas debe ser igual al de las columnas en el parámetro *dest=* y al menos una columna debe incluir valores. Para no especificar valores para una columna, no especifique un valor. Por ejemplo, *coln()*.

NULL Ignore la tabla de búsqueda si la fila de la columna de origen tiene un valor NULL.

SPACES

Ignore la tabla de búsqueda si la fila de la columna de origen tiene un valor SPACES. Sólo para columnas CHAR.

ZERO_LEN

Ignore la tabla de búsqueda si la fila de la columna de origen tiene un valor VARCHAR de longitud cero.

preserve=

Lista de columnas de origen con valores que se insertan en el destino en lugar del valor de búsqueda cuando la columna contiene un valor declarado (NOT_FOUND, NULL, SPACES o VARCHAR de longitud cero).

NOT_FOUND

Ignore la tabla de búsqueda si no se encuentra ninguna coincidencia para la fila de la columna de origen.

Nota:

preserve= e *ignore=* se excluyen mutuamente. *ignore=* dejará de utilizarse en un release futuro.

Los operandos *col*, *null*, *spaces* y *zero_len* tienen el mismo efecto cuando se utilizan con *preserve=* o *ignore=*.

seed= Utilice *seed=* para variar el cálculo del algoritmo hash. Pueden utilizarse valores del 1 al 2.000.000.000. Si utiliza un valor de 0, se ignora el parámetro *seed=*.

Ejemplo de una sola columna

Utilice la función de búsqueda hash para insertar valores de una columna de una tabla de búsqueda en una columna de la tabla de destino, en función de un valor hash de una columna de origen.

Por ejemplo, supongamos que la columna de origen, FIRST_NAME, contiene los nombres de pila y que la columna de destino incluirá los nombres de pila de sustitución de la tabla de búsqueda. Una tabla de búsqueda, NAME_LOOKUP, contiene una columna (FIRST) con los nombres de pila y una columna (SEQ) que contiene valores secuenciales.

Para obtener valores para la columna de destino utilizando la tabla NAME_LOOKUP, especifique:

```
HASH_LOOKUP(FIRST_NAME,NAME_LOOKUP(SEQ, FIRST))
```

La función de búsqueda hash busca coincidencias entre los valores hash de la columna de origen con los valores de la columna SEQ de la tabla NAME_LOOKUP. Cuando se encuentra una coincidencia, la función inserta el valor correspondiente de la columna FIRST en la columna de destino.

Ejemplo de varias columnas

Utilice la función de búsqueda hash para insertar valores de las columnas de una fila de la tabla de búsqueda en las columnas de una fila de la tabla de destino, en función de un valor hash de una columna de origen.

Por ejemplo, en función de los valores hash de una columna de origen (FIRST_NAME) que contiene nombres de pila, puede sustituir los valores de las columnas de destino (FIRST y LAST) por los nombres de pila y apellidos de una tabla de búsqueda. Una tabla de búsqueda denominada NAME_LOOKUP contiene una columna (SEQ) con valores secuenciales así como columnas (FIRST_MASK y LAST_MASK) para enmascarar valores en el destino.

Para sustituir los nombres de la tabla de destino en función de los valores hash de una columna de origen, especifique:

```
HASH_LOOKUP(FIRST_NAME,DEST=(FIRST, LAST), NAME_LOOKUP(SEQ,VALUES=(FIRST_MASK, LAST_MASK)))
```

La función de búsqueda hash busca coincidencias entre los valores hash de la columna FIRST_NAME de origen con los valores de la columna SEQ de la tabla NAME_LOOKUP. Cuando se encuentra una coincidencia, la función inserta los valores correspondientes de las columnas FIRST_MASK y LAST_MASK de la tabla de búsqueda en las columnas de destino correspondientes.

Ejemplo de ignorar

Utilice la sentencia siguiente para ampliar el ejemplo de una sola columna, en el que desea utilizar los valores NULL y SPACES de origen en lugar de los valores de la tabla de búsqueda:

```
HASH_LOOKUP(FIRST_NAME,NAME_LOOKUP(SEQ, FIRST),IGNORE=(FIRST_NAME(NULL,SPACES)))
```

Ejemplo de NoCache

Utilice la sentencia siguiente para ampliar el ejemplo de una sola columna, en el que no desea mantener una tabla de valores de búsqueda encontrados en la memoria.

```
HASH_LOOKUP(FIRST_NAME,NAME_LOOKUP(SEQ, FIRST),NOCACHE)
```

Ejemplo de recorte

Utilice la siguiente sentencia para ampliar el ejemplo de una sola columna, en el que desea recortar espacios y comas del valor de origen así como convertir el valor de origen a mayúsculas antes de ejecutarse hash en él:

```
HASH_LOOKUP(FIRST_NAME, TRIM( ,\u),NAME_LOOKUP(SEQ,FIRST))
```

Función de búsqueda aleatoria

La función de búsqueda aleatoria selecciona un valor al azar de una tabla de búsqueda especificada para insertarlo en una columna de destino. La función genera un número aleatorio entre 1 y el límite o número de filas en la tabla de búsqueda para utilizar como subíndice en la tabla. El valor o valores de columna de la fila correspondiente al subíndice se insertan en la columna de destino.

Hay dos formatos de la función de búsqueda aleatoria: una sola columna y varias columnas. El formato de una sola columna inserta un valor en una columna de destino única. El formato de varias columnas inserta valores de diversas columnas de la tabla de búsqueda en las columnas de destino correspondientes.

Puede especificar la función de búsqueda aleatoria de varias columnas para cualquier columna de origen que se sustituirá por un valor de la tabla de búsqueda, pero debe editar la correlación de columnas para eliminar los nombres de las columnas de origen restantes que también se sustituirán.

El parámetro *ignore* le permite ignorar la tabla de búsqueda y utilizar un valor de origen cuando una fila de una columna de origen especificada contiene un valor especificado (NULL, SPACES (para columnas CHAR) o VARCHAR de longitud cero).

Puede utilizar el parámetro *preserve* para ignorar la tabla de búsqueda y utilizar un valor de origen cuando una fila de una columna de origen especificada contiene un valor especificado (NULL, SPACES (para columnas CHAR) o VARCHAR de longitud cero).

La sintaxis es:

```
RAND_LOOKUP(lktablename, { columnname | dest=(col1,coln) ,values=(col1,coln) }  
[,limit] [,ignore=(col(spaces, null, zero_len), ) | PRESERVE=( nombrecol (spaces, null, zero_len), ) ] )
```

lktablename

Nombre de la tabla de búsqueda. Puede especificar el nombre de la tabla de búsqueda como *aliasbd.idcreador.nombretabla*, *idcreador.nombretabla* o *nombretabla*. Si el nombre de tabla no está totalmente calificado, se utilizan calificadores de tabla de destino.

nombrecol

Nombre de la columna en la tabla de búsqueda que contiene los valores que se seleccionarán aleatoriamente para la inserción en el destino. (Se requiere para la búsqueda de una sola columna).

dest= Nombres de las columnas de la tabla de destino en las que se insertan los valores de la tabla de búsqueda. (Se requiere para la búsqueda de varias columnas).

col1,coln

Nombres de las columnas de la tabla de destino. El orden de los nombres de columna debe corresponderse con las columnas de la tabla de búsqueda en el parámetro *values=*.

values=

Nombres de las columnas en la tabla de búsqueda que contienen valores que deben insertarse en el destino. (Se requiere para la búsqueda de varias columnas).

col1,coln

Nombres de las columnas de la tabla de búsqueda. El orden de los nombres de columna deben corresponderse con las columnas de la tabla de destino en el parámetro *dest=*.

limit Límite opcional sobre el número de filas de la tabla de búsqueda que se utiliza para seleccionar valores de columna. Especifique un entero hasta un valor máximo de 2.000.000.000. Si no se especifica ningún límite, se utilizan todas las filas.

Nota: En la memoria se genera una tabla de valores de columna. El tamaño de esta tabla puede estar limitado por los recursos del sistema.

ignore=

Lista de columnas de origen con valores que se insertan en el destino en lugar del valor de búsqueda cuando la columna tiene una fila con un valor declarado (NULL, SPACES o VARCHAR de longitud cero).

col El nombre de la columna de origen.

Para la búsqueda de una sola columna, especifique únicamente un nombre de columna.

Para la búsqueda de varias columnas, el orden de los nombres de columna debe corresponderse con las columnas de la tabla de destino en el parámetro *dest=*. El número

de columnas debe ser igual al de las columnas en el parámetro *dest=* y al menos una columna debe incluir valores. Para no especificar valores para una columna, no especifique un valor. Por ejemplo, *coln()*.

null Ignore la tabla de búsqueda si la fila de la columna de origen tiene un valor NULL.

spaces Ignore la tabla de búsqueda si la fila de la columna de origen tiene un valor SPACES. Sólo para columnas CHAR.

zero_len Ignore la tabla de búsqueda si la fila de la columna de origen tiene un valor VARCHAR de longitud cero.

preserve=

Lista de columnas de origen con valores que se insertan en el destino en lugar del valor de búsqueda cuando la columna contiene un valor declarado (NOT_FOUND, NULL, SPACES o VARCHAR de longitud cero).

NOT_FOUND

Ignore la tabla de búsqueda si no se encuentra ninguna coincidencia para la fila de la columna de origen.

Nota:

preserve= e *ignore=* se excluyen mutuamente. *ignore=* dejará de utilizarse en un release futuro.

Los operandos *col*, *null*, *spaces* y *zero_len* tienen el mismo efecto cuando se utilizan con *preserve=* o *ignore=*.

Ejemplo de una sola columna

Para seleccionar un valor al azar de la columna STATE en las 50 primeras filas de una tabla denominada STATE_LOOKUP e insertarlo en la columna de destino, especifique:

```
RAND_LOOKUP(STATE_LOOKUP,STATE,50)
```

Ejemplo de varias columnas

Para seleccionar valores de las columnas CITY, STATE y ZIPCODE en una fila aleatoria de una tabla denominada STATE_LOOKUP e insertarlos en las columnas de destino correspondientes, especifique:

```
RAND_LOOKUP(STATE_LOOKUP,  
DEST=(CITY,STATE,ZIPCODE),  
VALUES=(CITY,STATE,ZIP))
```

Ejemplo de ignorar

Utilice la sentencia siguiente para ampliar el ejemplo de una sola columna, donde la columna de origen se denomina STATES y desea utilizar los valores NULL y SPACES de origen en lugar de los valores de la tabla de búsqueda:

```
RAND_LOOKUP(STATE_LOOKUP,STATE,50, IGNORE=(STATES(NULL,SPACES)))
```

Función de reorganización

La función de reorganización sustituye un valor de origen por otro valor de la columna que se insertará en una columna de destino. La fila de origen y la fila que contiene el valor de sustitución nunca serán las mismas pero, en función de los datos, los valores de origen y de sustitución pueden ser idénticos.

Puede indicar el número de veces que la función recuperará de nuevo un valor de sustitución hasta que se encuentre un valor que no coincida con el valor de origen (a “reintento”), o bien puede permitir que un valor de sustitución coincida con el origen. Cada función de reorganización funciona de manera independiente de otras funciones de reorganización utilizadas en una correlación de columnas.

Hay dos formatos de la función de reorganización: una sola columna y varias columnas. El formato de una sola columna inserta un valor de sustitución en una columna de destino única. El formato de varias columnas inserta valores de sustitución de diversas columnas de una fila en las columnas de destino correspondientes. Una columna no se puede incluir en más de una función de reorganización de una correlación de columnas. Si se utiliza la característica de reintentos con una reorganización de varias columnas, la función recuperará de nuevo otra fila de sustitución si algún valor de la columna de la fila de origen coincide con el valor de una columna de fila de sustitución correspondiente. (El formato de varias columnas no se puede utilizar en una función de propagación).

Para crear una función de reorganización de varias columnas, especifique la función para una columna de origen que se sustituirá por valores reorganizados y edite la correlación de columnas para eliminar los nombres de otras columnas de origen con valores que también se sustituirán.

El parámetro *ignore* impide que la función sustituya una fila de origen o utilice una fila de sustitución si una de ellas contiene un valor especificado (NULL, SPACES (para columnas CHAR) o VARCHAR de longitud cero). Si no se permiten reintentos, el parámetro *ignore* no se aplicará a la fila de sustitución.

La sintaxis es:

```
SHUFFLE [ ( dest=(col1,coln) ) ] |
[ ( dest=(col1,coln) , retry[=number] ) ] |
[ ( dest=(col1,coln) [ , retry[=number] ] , ignore=( col1 ( [spaces] | [spaces,null]
| [spaces,null,zero_len] | [null] | [null,zero_len] | [zero_len] ) , coln (...) ) ) ] |
[ ( retry[=number] ) ] |
[ ( retry[=number] , ignore=( col ( [spaces] | [spaces,null] | [spaces,null,zero_len]
| [null] | [null,zero_len] | [zero_len] ) ) ) ] |
[ ( ignore=( col ( [spaces] | [spaces,null] | [spaces,null,zero_len]
| [null] | [null,zero_len] | [zero_len] ) ) ) ]
```

donde:

dest= Nombres de las columnas de la tabla de destino en las que se insertan los valores de sustitución. (Se requiere para la reorganización de varias columnas).

col1, coln, ...

Nombres de las columnas de la tabla de destino.

retry Número de veces que hay que recuperar de nuevo un valor de sustitución para buscar un valor que no coincida con la fila de origen. Especifique cero para permitir que un valor de sustitución coincida con el origen.

Nota: La utilización de un valor de reintentos alto con columnas que contienen muchos valores duplicados aumentará el tiempo de proceso. Para estas columnas, puede ser mejor utilizar un valor de reintentos de cero.

=number

Especifique un valor dentro del rango de 0-1000. Especifique 0 para permitir que un valor de sustitución coincida con el origen.

ignore=

Lista de columnas para las que la función no sustituirá un valor de origen o no utilizará un valor de sustitución si uno de ellos es un valor especificado (NULL, SPACES (para columnas CHAR) o VARCHAR de longitud cero). Si se ignora un valor de sustitución, la función recuperará de nuevo otro valor de sustitución. Si no se permiten reintentos, el parámetro *ignore=* no se aplicará a los valores de sustitución.

- col* El nombre de la columna de origen.
- Para la reorganización de una sola columna, especifique únicamente un nombre de columna.
- Para la reorganización de varias columnas, el orden de los nombres de columna debe corresponderse con las columnas de la tabla de destino en el parámetro *dest=*. El número de columnas debe ser igual al de las columnas en el parámetro *dest=* y al menos una columna debe incluir valores. Para no especificar valores para una columna, no especifique un valor. Por ejemplo, *coln()*.
- null* No sustituya el valor de origen ni utilice un valor de sustitución si uno de ellos es un valor NULL.
- spaces* No sustituya el valor de origen ni utilice un valor de sustitución si uno de ellos es un valor SPACES. Sólo para columnas CHAR.
- zero_len* No sustituya el valor de origen ni utilice un valor de sustitución si uno de ellos es un valor VARCHAR de longitud cero.

Ejemplo de valor predeterminado de una sola columna

El ejemplo siguiente inserta valores reorganizados en una sola columna.

```
SHUFFLE
```

Ejemplo de reintentos de una sola columna

El ejemplo siguiente inserta valores reorganizados en una sola columna y recupera de nuevo un valor de sustitución que no coincide con el origen hasta 12 veces.

```
SHUFFLE(RETRY=12)
```

Ejemplo de varias columnas

El ejemplo siguiente inserta valores reorganizados en las columnas STATE y ZIP y recupera de nuevo un valor de sustitución que no coincide con el origen hasta 12 veces.

```
SHUFFLE(DEST=(STATE, ZIP), RETRY=12)
```

Ejemplo de ignorar

El ejemplo siguiente inserta valores reorganizados en las columnas STATE y ZIP y recupera de nuevo un valor de sustitución que no coincide con el origen hasta 12 veces. El ejemplo tampoco sustituye un valor de origen ni utiliza un valor de sustitución para la columna STATE si una fila de origen o sustitución contiene un valor NULL o SPACES, pero no ignora ninguna fila de origen o sustitución para la columna ZIP.

```
SHUFFLE(DEST=(STATE, ZIP), RETRY=12,  
IGNORE=(STATE(NULL, SPACES), ZIP()))
```

Función TRANS SSN

Utilice la función TRANS SSN para generar un número de seguridad social (SSN) de Estados Unidos que sea exclusivo y válido. De forma predeterminada, TRANS SSN genera algorítmicamente un destino alterado de forma coherente de acuerdo con el SSN de origen. TRANS SSN también puede generar un SSN aleatorio cuando los datos de origen no tienen un valor SSN o cuando no hay necesidad de transformar el SSN de origen de un modo coherente.

Un SSN está formado por 3 subcampos. Los primeros 3 dígitos (área) representa un área generalmente determinada por el estado en el que se emite el SSN. Los siguientes 2 dígitos (grupo) definen un número de grupo correspondiente al número de área. Los últimos 4 dígitos (serie) son un número de serie secuencial. Independientemente del tipo de proceso, predeterminado o aleatorio, TRANS SSN generará un SSN con un número de grupo adecuado en el número de área.

El método de proceso predeterminado genera un SSN que incluye el número de área de origen así como los números de serie y de grupo alterados en función del SSN de origen.

El método de proceso aleatorio genera un SSN que puede incluir el número de área de origen y utiliza un número de grupo que la Administración de la Seguridad Social ha emitido recientemente para el número de área de destino. Los números de serie empiezan por 001 y se incrementan de 1 en 1 para cada SSN adicional generado para el número de área. Cuando el número de serie es mayor que 9999, el número de serie se restablecerá en 0001 y se utilizará el número de grupo que precede al número emitido más recientemente para el número de área.

La sintaxis de TRANS SSN es:

TRANS SSN [('[=distintivos] [columnorigen [preserve=invalid]')]

distintivos

Puede especificar uno o más distintivos de opción de proceso que no son sensibles a mayúsculas y minúsculas.

- n* Genera un SSN aleatorio que no se basa en un valor de origen.
- m* Utiliza el grupo máximo de todos los valores de área de SSN, incluidos los valores comprendidos entre el 773 y el 899, y excluyendo los números de área no válidos.
- r* Genera un SSN con un número de área aleatorio correspondiente al mismo estado que el SSN de entrada.
- v* Valida el número de grupo de origen comparándolo con los números que utiliza la Administración de la Seguridad Social.
- El SSN de destino debe incluir guiones para separar los campos (por ejemplo, 123-45-6789). Requiere una columna de destino de tipo de carácter con una longitud mínima de 11 caracteres.

columnorigen

El nombre de la columna de origen. Si no se especifica un nombre de columna de origen, se utilizará el nombre de la columna de destino. Si un nombre de columna de origen no se especifica y el nombre de la columna de destino no coincide con un nombre de columna en la tabla de origen, se producirá un error durante el proceso.

preserve=invalid

Si la columna de origen contiene un SSN no válido, no lo sustituya por un valor generado. El valor de la columna de origen se utilizará en la columna de destino.

Tipos de datos permitidos

Se permiten los siguientes tipos de datos de origen y destino:

CHAR

La longitud de los datos de la columna debe ser entre 9 y 256 caracteres.

DECIMAL

La precisión de la columna debe ser de 9 - 20 y la escala 0.

INTEGER

Sin restricciones.

VARCHAR

La longitud de los datos de la columna debe ser entre 9 y 254 caracteres.

Si una columna de origen o destino no cumple estas restricciones, se producirá un mensaje de error durante el proceso.

Reglas del proceso de destino

Las reglas siguientes se aplican al valor SSN de destino, de acuerdo con el valor o tipo de datos de destino:

CHAR

Si el valor de origen es 0, espacios o un VARCHAR de longitud cero, el valor de destino se establecerá en espacios.

Si un valor de origen tiene 11 caracteres o más e incluye guiones (-), o si se ha especificado el distintivo '-', el valor de destino incluirá guiones si la longitud de la columna de destino es de 11 caracteres o más.

DECIMAL, INTEGER

Si el valor de origen es 0, espacios, o un VARCHAR de longitud cero, el valor de destino será 0.

VARCHAR

Si el valor de origen es 0, espacios, o un VARCHAR de longitud cero, la longitud de destino será 0.

Si un valor de origen tiene 11 caracteres o más e incluye guiones (-), o si se ha especificado el distintivo '-', el valor de destino incluirá guiones si la longitud de la columna de destino es de 11 caracteres o más.

NULL Si el valor de origen es NULL, el valor de destino será NULL.

Filas omitidas

Las condiciones siguientes pueden hacer que una fila de origen se omita y no se grabe en el destino.

- El valor de origen es NULL y la columna de destino no permite un valor NULL.
- La columna de origen es CHAR o VARCHAR, y el valor de origen es inferior a 9 caracteres, contiene un carácter no numérico (que no sean guiones entre los 3 subcampos) o es demasiado grande.
- La Administración de la Seguridad Social no ha utilizado el número de área de origen.
- La Administración de la Seguridad Social no ha utilizado el número de grupo de origen con el número de área (sólo si se ha especificado el distintivo 'v').
- El número de serie de origen es 0000, o el SSN es un valor reservado no emitido por la Administración de la Seguridad Social (por ejemplo, 078-05-1120).
- El valor de origen no se puede convertir en un formato al que TRANS SSN dé soporte.

Mensajes de error

Pueden emitirse los siguientes mensajes de error:

SSN01

Parm on Col ccccc ("ppp") is invalid

Explicación

La columna contiene una función TRANS con un distintivo de opción de proceso que no es válido.

Acción del usuario

Asegúrese de que la función TRANS en la columna especificada utiliza un distintivo de opción de proceso válido (n, r, v,-).

SSN02

Col ccccc not on source

Explicación

La columna que se ha especificado como parámetro *columorigen* o el nombre de la columna de destino (si se ha omitido el parámetro *columorigen*) no se ha encontrado en la tabla de origen.

Acción del usuario

Compruebe la tabla de origen y resuelva las discrepancias o las columnas que faltan.

SSN03

Source Col ccccc-aaa invalid

Explicación

El formato de la columna de origen no está soportado porque el atributo indicado no es válido.

Acción del usuario

Compruebe la columna de origen y asegúrese de que los valores para tipo, longitud, precisión y escala son adecuados.

SSN04

Dest Col ccccc-aaa invalid

Explicación

El formato de la columna de destino no está soportado porque el atributo indicado no es válido.

Acción del usuario

Compruebe la columna de destino y asegúrese de que los valores para tipo, longitud, precisión y escala son adecuados.

SSN05

Get col ccccc data-rc=nnn

Explicación

Se ha producido un error interno inesperado al obtener el valor desde la columna de origen.

Acción del usuario

Compruebe los valores de las columnas de origen y destino y asegúrese de que los valores para tipo, longitud, precisión y escala son adecuados. Si el problema persiste, póngase en contacto con el Centro de soporte de software de IBM.

SSN08

Put col ccccc data-rc=nnn

Explicación

Se ha producido un error interno inesperado al establecer el valor en la columna de destino.

Acción del usuario

Compruebe los valores de las columnas de origen y destino y asegúrese de que los valores para tipo, longitud, precisión y escala son adecuados. Si el problema persiste, póngase en contacto con el Centro de soporte de software de IBM.

Si se producen otros errores, póngase en contacto con el soporte técnico.

Ejemplo 1

El ejemplo siguiente utiliza un nombre de columna de origen que coincide con la columna de destino y genera un SSN aleatorio que no se basa en el valor de origen:

TRANS SSN ('=n')

Ejemplo 2

El ejemplo siguiente utiliza un nombre de columna de origen (NATIONAL_ID) que difiere de la columna de destino y genera un SSN utilizando el método de proceso predeterminado e incluye guiones:

TRANS SSN ('=- NATIONAL_ID')

Función TRANS CCN

Utilice la función TRANS CCN para generar un número de tarjeta de crédito (CCN) exclusivo y válido. De forma predeterminada, TRANS CCN genera algorítmicamente un CCN alterado de forma coherente de acuerdo con el CCN de origen. TRANS CCN también puede generar un valor aleatorio cuando los datos de origen no tienen un valor CCN o cuando no hay necesidad de transformar el CCN de origen de un modo coherente.

Un CCN, según se define en ISO 7812, consiste en un identificador de emisor de 6 dígitos, seguido de un número de cuenta de longitud variable y un único dígito de control como número final. El dígito de control comprueba la validez del CCN y se genera pasando los números de identificador de emisor y de cuenta por el algoritmo Luhn. La longitud máxima de un CCN es de 19 dígitos.

El método de proceso predeterminado genera un CCN incluyendo los 4 primeros dígitos del identificador de emisor del CCN de origen y modificando los 2 dígitos restantes del número de identificador de emisor y el número de cuenta basándose en el CCN de origen. También se asigna un dígito de comprobación válido.

El método de proceso aleatorio genera un CCN que puede incluir los 4 primeros dígitos del número de identificador de emisor de origen o un número de identificador de emisor asignado a American Express, Discover, MasterCard o VISA. También se asigna un dígito de comprobación válido. Si se incluyen los cuatro primeros dígitos de un número de identificador de emisor de origen, el primer número de cuenta basado en dichos dígitos empezará por 1 y, para cada CCN adicional que utiliza esos dígitos, el número de cuenta se incrementará en 1.

La sintaxis de TRANS CCN es:

TRANS CCN [('=distintivo'] [columnorigen] [preserve=invalid])]

distintivo

Especifique un distintivo de opción para generar un CCN aleatorio.

- n* Genere un CCN aleatorio que no se base en un valor de origen e incluya un número de identificador de emisor asignado a American Express, Discover, MasterCard o VISA.
- r* Genere un CCN aleatorio que incluya los 4 primeros dígitos del número de identificador de emisor de origen.

columnorigen

El nombre de la columna de origen. Si no se especifica un nombre de columna de origen, se utiliza el nombre de la columna de destino.

Si un nombre de columna de origen no se especifica y el nombre de la columna de destino no coincide con un nombre de columna en la tabla de origen, se producirá un error durante el proceso.

preserve=invalid

Si la columna de origen contiene un CCN no válido, no lo sustituya por un valor generado. El valor de la columna de origen se utilizará en la columna de destino.

Tipos de datos permitidos

Se permiten los siguientes tipos de datos de origen y destino:

CHAR

La longitud de la columna debe tener entre 13 y 256 caracteres.

VARCHAR

La longitud de la columna debe tener entre 13 y 254 caracteres.

DECIMAL

La precisión de la columna debe ser de 13 a 254 y la escala 0.

Si una columna de origen o destino no cumple estas restricciones, se producirá un mensaje de error.

Reglas del proceso de destino

Las reglas siguientes se aplican al valor CCN de destino, de acuerdo con el valor o tipo de datos de destino:

CHAR

Si el valor de origen son espacios o un VARCHAR de longitud cero, el valor de destino se establecerá en espacios.

VARCHAR

Si el valor de origen son espacios o un VARCHAR de longitud cero, la longitud de destino será 0.

DECIMAL

Si el valor de origen es 0, el valor de destino será 0.

NULL Si el valor de origen es NULL, el valor de destino será NULL.

Filas omitidas

Las condiciones siguientes pueden hacer que una fila de origen se omita y no se grabe en el destino.

- El valor de origen es NULL y la columna de destino no permite un valor NULL.
- El valor de origen es inferior a 13 caracteres, contiene un carácter no numérico, es demasiado grande o tiene un dígito de comprobación incorrecto.
- La longitud del valor de origen no es válida para el emisor de la tarjeta de crédito.
- El valor de origen no se puede convertir en un formato al que TRANS CCN dé soporte.

Mensajes de error

Pueden emitirse los siguientes mensajes de error:

CCN01

Parm on Col ccccc ("ppp") is invalid

Explicación

La columna indicada contiene una función TRANS con un distintivo de opción de proceso que no es válido.

Acción del usuario

Asegúrese de que la función TRANS en la columna especificada utiliza un distintivo de opción de proceso válido (n, r, 6).

CCN02

Col ccccc not on source

Explicación

La columna que se ha especificado como parámetro *columorigen* o el nombre de la columna de destino (si se ha omitido el parámetro *columorigen*) no se ha encontrado en la tabla de origen.

Acción del usuario

Compruebe la tabla de origen y resuelva las discrepancias o las columnas que faltan.

CCN03

Source Col ccccc-aaa invalid

Explicación

El formato de la columna de origen no está soportado porque el atributo indicado no es válido.

Acción del usuario

Compruebe la columna de origen y asegúrese de que los valores para tipo, longitud, precisión y escala son adecuados.

CCN04

Dest Col ccccc-aaa invalid

Explicación

El formato de la columna de destino no está soportado porque el atributo indicado no es válido.

Acción del usuario

Compruebe la columna de destino y asegúrese de que los valores para tipo, longitud, precisión y escala son adecuados.

CCN05

Get col ccccc data-rc=nnn

Explicación

Se ha producido un error interno inesperado al obtener el valor desde la columna de origen.

Acción del usuario

Compruebe los valores de las columnas de origen y destino y asegúrese de que los valores para tipo, longitud, precisión y escala son adecuados. Si el problema persiste, póngase en contacto con el Centro de soporte de software de IBM.

CCN08

Put col ccccc data-rc=nnn

Explicación

Se ha producido un error interno inesperado al establecer el valor en la columna de destino.

Acción del usuario

Compruebe los valores de las columnas de origen y destino y asegúrese de que los valores para tipo, longitud, precisión y escala son adecuados. Si el problema persiste, póngase en contacto con el Centro de soporte de software de IBM.

Si se producen otros errores, póngase en contacto con el soporte técnico.

Ejemplo 1

El ejemplo siguiente utiliza un nombre de columna de origen (CREDITCARD) que difiere de la columna de destino y genera un CCN aleatorio que no se basa en el valor de origen:

```
TRANS CCN ('=n CREDITCARD')
```

Ejemplo 2

El ejemplo siguiente utiliza un nombre de columna de origen (CREDITCARD) que difiere de la columna de destino y genera un CCN utilizando el método de proceso predeterminado:

```
TRANS CCN ('CREDITCARD')
```

Función TRANS EML

Utilice la función TRANS EML para generar una dirección de correo electrónico. Una dirección de correo electrónico consta de dos partes, un nombre de usuario seguido de un nombre de dominio, separados por una arroba ('@'). Por ejemplo, usuario@dominio.com.

TRANS EML genera una dirección de correo electrónico con un nombre de usuario basado en los datos de destino o en un literal concatenado con un número secuencial. El nombre de dominio puede basarse en una dirección de correo electrónico de los datos de origen, un literal o seleccionarse aleatoriamente de una lista de proveedores de servicio de correo electrónico de gran tamaño. La dirección de correo electrónico también se puede convertir a mayúsculas o minúsculas.

TRANS EML puede generar un nombre de usuario basado en los valores de una o dos columnas de la tabla de destino (que normalmente contienen el nombre de un usuario). Las opciones de proceso le permiten utilizar únicamente el primer carácter del valor de la primera columna (por ejemplo, la letra inicial de un nombre de pila) y separar los valores de ambas columnas utilizando un punto o un carácter de subrayado.

Si el nombre de usuario se basa en un único valor de la columna de destino o un literal, el nombre se concatenará con un número de secuencia. Si un nombre de usuario se basa en los valores de dos columnas de la tabla de destino y no se utiliza un punto o un carácter de subrayado como separadores, los valores se concatenarán. Si no se proporciona un parámetro para el nombre de usuario, el nombre se formará mediante el literal "correo electrónico" concatenado con un número secuencial. Los números secuenciales para los nombres de usuario son sufijos que comienzan por 1 y se incrementan de 1 en 1.

La sintaxis de TRANS EML es:

```
TRANS EML [( '[=distintivos] , [{columnorigen | "dominio" | , }  
[{nombre1col[nombre2col] | "pfxusuario"}] ] [preserve=invalid]' )]
```

distintivos

Puede especificar uno o más distintivos de opción de proceso que no son sensibles a mayúsculas y minúsculas.

- n* Genera un nombre de dominio aleatorio a partir de una lista de proveedores de servicio de correo electrónico de gran tamaño.
- .* Separa los valores *nombre1col* y *nombre2col* con un punto.
- _* Separa los valores *nombre1col* y *nombre2col* con un carácter de subrayado.
- i* Utiliza sólo el primer carácter del valor *nombre1col*.
- l* Convierte la dirección de correo electrónico a minúsculas.
- u* Convierte la dirección de correo electrónico a mayúsculas.

columnorigen

El nombre de la columna de origen con direcciones de correo electrónico que se utiliza para proporcionar el nombre de dominio.

Si no se ha definido el distintivo 'n' ni el parámetro *domain*, se utiliza el nombre de dominio de la columna de origen. (Si no se ha definido *columnorigen*, el nombre de la columna de origen se basa en el nombre de la columna de destino).

Si un nombre de columna de origen no se especifica y el nombre de la columna de destino no coincide con un nombre de columna en la tabla de origen, se producirá un error durante el proceso.

dominio

Un literal, de hasta 31 caracteres, que forma el nombre de dominio.

,

Se necesita una coma si no se ha definido un parámetro *columorigen* ni *dominio* y se ha definido un literal o uno o varios nombres de columna para el nombre de dominio.

nombre1col

Un nombre de columna de tabla de destino con los valores utilizados para formar la primera (o única) parte del nombre de usuario.

nombre2col

Un nombre de columna de tabla de destino con los valores utilizados para formar la segunda parte del nombre de usuario.

pxusuario

Un literal, de hasta 31 caracteres, que se concatena con un número secuencial para formar el nombre de usuario.

preserve=invalid

Si la columna de origen contiene una dirección de correo electrónico no válida, no la sustituya por un valor generado. El valor de la columna de origen se utilizará en la columna de destino.

Tipos de datos permitidos

Se permiten los siguientes tipos de datos de origen y destino:

CHAR

La longitud de la columna debe tener entre 3 y 4096 caracteres.

VARCHAR

La longitud de la columna debe tener entre 3 y 4094 caracteres.

Si una columna de origen o destino no cumple estas restricciones, se emitirá un mensaje de error.

Reglas del proceso de destino

Las reglas siguientes se aplican al valor de correo electrónico de destino, de acuerdo con el valor o tipo de datos de destino:

CHAR

Si el valor de origen son espacios o un VARCHAR de longitud cero, el valor de destino se establecerá en espacios.

VARCHAR

Si el valor de origen son espacios o un VARCHAR de longitud cero, la longitud de destino será 0.

NULL Si el valor de origen es NULL, el valor de destino será NULL.

Filas omitidas

Las condiciones siguientes pueden hacer que una fila de origen se omita y no se grabe en el destino.

- El valor de origen es NULL y la columna de destino no permite un valor NULL.
- El valor de origen es un VARCHAR que tiene menos de 3 caracteres.
- El valor de correo electrónico de origen no contiene una '@'.
- El valor de origen no se puede convertir en un formato al que TRANS EML dé soporte.

Mensajes de error

Pueden emitirse los siguientes mensajes de error:

EML01

Parm on Col ccccc ("ppp") is invalid

Explicación

La columna indicada contiene una función TRANS con un distintivo de opción de proceso que no es válido.

Acción del usuario

Asegúrese de que la función TRANS en la columna especificada utiliza un distintivo de opción de proceso válido (n, ., _, i, l, u).

EML02

Col ccccc not on source

Explicación

La columna que se ha especificado como parámetro *columorigen* o el nombre de la columna de destino (si se ha omitido el parámetro *columorigen*) no se ha encontrado en la tabla de origen.

Acción del usuario

Compruebe la tabla de origen y resuelva las discrepancias o las columnas que faltan.

EML03

Source Col ccccc-aaa invalid

Explicación

El formato de la columna de origen no está soportado porque el atributo indicado no es válido.

Acción del usuario

Compruebe la columna de origen y asegúrese de que los valores para tipo, longitud, precisión y escala son adecuados.

EML04

Dest Col ccccc-aaa invalid

Explicación

El formato de la columna de destino no está soportado porque el atributo indicado no es válido.

Acción del usuario

Compruebe la columna de destino y asegúrese de que los valores para tipo, longitud, precisión y escala son adecuados.

EML05

Get col ccccc data-rc=nnn

Explicación

Se ha producido un error interno inesperado al obtener el valor desde la columna de origen.

Acción del usuario

Compruebe los valores de las columnas de origen y destino y asegúrese de que los valores para tipo, longitud, precisión y escala son adecuados. Si el problema persiste, póngase en contacto con el Centro de soporte de software de IBM.

EML08

Put col ccccc data-rc=nnn

Explicación

Se ha producido un error interno inesperado al establecer el valor en la columna de destino.

Acción del usuario

Compruebe los valores de las columnas de origen y destino y asegúrese de que los valores para tipo, longitud, precisión y escala son adecuados. Si el problema persiste, póngase en contacto con el Centro de soporte de software de IBM.

EML09

Domain literal sssss too long

Explicación

La serie especificada como literal de nombre de dominio (domain) sobrepasada el límite máximo de 31 caracteres.

Acción del usuario

Especifique un nombre de dominio formado por 31 caracteres o menos.

EML10

User literal sssss too long

Explicación

La serie especificada como literal de nombre de usuario (*pfxusuario*) sobrepasa el límite máximo de 31 caracteres.

Acción del usuario

Especifique un nombre de usuario formado por 31 caracteres o menos.

EML11

Name1 Col ccccc not on dest

Explicación

Para realizar la función TRANS indicada, debe especificarse una columna *nombre1col* en la tabla de destino.

Acción del usuario

Verifique que la columna *nombre1col* especificada coincide con la columna *nombre1col* indicada en la función TRANS.

El nombre de la columna *nombre1col* no se ha encontrado en la tabla de destino.

EML12

Name1 Col ccccc-aaa invalid

Explicación

Para realizar la función TRANS indicada, la columna *nombre1col* debe estar en un formato válido.

Acción del usuario

Verifique que la longitud, tipo y disponibilidad de la columna *nombre1col* son adecuados.

EML13

Name2 Col ccccc not on dest

Explicación

Para realizar la función TRANS indicada, debe especificarse una columna *nombre2col* en la tabla de destino.

Acción del usuario

Verifique que la columna *nombre2col* especificada coincide con la columna *nombre2col* indicada en la función TRANS.

El nombre de la columna *nombre2col* no se ha encontrado en la tabla de destino.

EML14

Name2 Col ccccc-aaa invalid

Explicación

Para realizar la función TRANS indicada, la columna *nombre2col* debe estar en un formato válido.

Acción del usuario

Verifique que la longitud, tipo y disponibilidad de la columna *nombre2col* son adecuados.

Si se producen otros errores, póngase en contacto con el soporte técnico.

Ejemplo 1

El ejemplo siguiente utiliza un literal (optim.com) para formar el nombre de dominio y dos columnas de la tabla de destino (NAME_FIRST y NAME_LAST) para formar un nombre de usuario que incluya un carácter de subrayado:

```
TRANS EML ('=_ "optim.com" NAME_FIRST NAME_LAST')
```

Ejemplo 2

El ejemplo siguiente utiliza un nombre de dominio de la columna de origen y un literal (OptimUser) para formar un nombre de usuario que tendrá como sufijo un número secuencial.

```
TRANS EML ('', "OptimUser"')
```

Función TRANS COL

La función TRANS COL puede enmascarar datos que no tienen ningún formato inherente o un formato que no es ampliamente conocido. TRANS COL mantiene el formato y tipo de carácter de los datos de origen en el destino.

Si los datos de origen son caracteres alfabéticos en mayúsculas, TRANS COL genera caracteres alfabéticos en mayúsculas en el destino. Esta función enmascara los caracteres alfabéticos y numéricos, pero otros caracteres en los datos de origen se copian en el destino sin ser modificados. Puede utilizar TRANS COL para enmascarar tipos de datos CHAR, VARCHAR y numéricos no flotantes. Puede generar valores exclusivos, un valor diferente para cada aparición del mismo origen y puede generar valores con una longitud diferente del origen.

La sintaxis de TRANS COL es:

```
TRANS COL ( '{ unique | hash }[ source=nombrecol ]  
[ copy=( start,len [, "lit" ] )... ]  
[ seed= {"lit" | var (variable) | RANDOM} ]  
[ length=n | max ] [ preserve=( [ null ] [ spaces ] [ zero_len ] ) ]  
[ TRIM=(caract1 [ caractn,...] [\u] [\r] ) ] [num]' )
```

unique Genera un valor de destino exclusivo. La longitud del valor de destino será la misma que la longitud del valor de origen.

hash Genera un valor de destino aplicando hash al valor de origen. Cuando se utiliza hash, diferentes valores de origen pueden generar los mismos valores de destino cada vez que se ejecuta el proceso.

Nota:

Para el mismo valor de origen, es posible obtener el mismo valor de destino cuando se utiliza el

parámetro exclusivo o el parámetro hash. Utilice hash con el parámetro seed para generar diferentes valores de destino cada vez que se ejecuta el proceso.

source=nombrecol

Utilice este parámetro para especificar el nombre de la columna de origen si la columna de destino es distinta del nombre de la columna de origen. El valor que especifique se convertirá a mayúsculas; para impedir que el valor se convierta a mayúsculas, escriba el valor entre comillas.

copy= Uno o más pares de subseries que deben copiarse en el destino sin que se enmascaren. Si proporciona una serie literal, se sustituyen los caracteres de origen en las posiciones especificadas. El parámetro *copy=* sólo es válido para una columna de tipo de datos de carácter.

seed= Valor utilizado para modificar el comportamiento de los algoritmos de enmascaramiento. Especifique una serie literal, una referencia a una variable de entorno o RANDOM.

"lit" Para especificar una serie literal, escriba la serie entre comillas.

var (variable)

Especifique una variable de entorno entre paréntesis. El nombre de la variable y su valor no pueden incluir comillas.

RANDOM

Genera un valor semilla aleatorio a partir de la fecha y hora actuales del sistema.

length={n | max }

Genera un valor de destino con una longitud distinta de la del valor de origen. Utilice *length=max* para generar un valor de destino que rellenará completamente la columna. Si se especifica una longitud menor que el valor de origen, el valor de origen se truncará cuando se escriba en el destino. El valor que seleccione para *n* no puede sobrepasar la longitud definida de la columna de destino. El parámetro *length=* sólo es válido con *hash=*.

preserve=

Lista uno o más valores de origen que no deben sustituirse en el destino.

null Si la columna de origen tiene un valor nulo, no sustituya el valor en el destino.

spaces Si la columna de origen tiene un valor de espacios, no sustituya el valor en el destino. Sólo para columnas CHAR.

zero_len

Si la columna de origen tiene un valor VARCHAR de longitud cero, no sustituya el valor en el destino.

TRIM=(caract1 [caractn,...])

El carácter o los caracteres de la columna de origen especificada no se enmascaran y no se graban en el destino. Por ejemplo, si especifica *TRIM=(x,y,z,1,2,3)*, si alguno de los caracteres x, y, z, 1, 2, 3 aparece en algún lugar de la columna de origen, no se enmascararán ni se grabarán en el destino.

[u] Utilice este parámetro para convertir el carácter o los caracteres a mayúsculas antes de su enmascaramiento. Si un carácter no tiene representación en mayúsculas, se deja igual. Por ejemplo, especificar *TRIM=(x,y \u)* no enmascara los caracteres x e y si aparecen en cualquier lugar de la columna de origen y cambia cualquier otro carácter de la columna de origen a mayúsculas antes de enmascararlos.

[r] Utilice este operando para eliminar espacios de cola. Por ejemplo, *TRIM=(x,y \u \r)* no enmascara los caracteres x e y si aparecen en cualquier lugar de la columna de origen, cambia cualquier otro carácter de la columna de origen a mayúsculas antes de enmascararlos y elimina los espacios de cola antes del enmascaramiento.

num Utilice este parámetro para transformar los enteros de una columna de tipo de datos de carácter para que sean idénticos a los de una columna de tipo de datos numéricos. El parámetro *num* sólo es válido en valores numéricos en una columna de tipo de datos de carácter. Cuando se utiliza de

esta manera, la integridad de las claves foráneas se mantiene en diferentes tipos de datos. Si utiliza este parámetro, no especifique *copy=* o *length=*.

Ejemplos:

- TRANS COL UNIQUE para el valor de origen **CDE-7834**, podría producir un valor de destino de: **ZWQ-4598**
- TRANS COL UNIQUE SEED=RANDOM para el valor de origen **CDE-7834-2008** podría producir un valor de destino de: **SWX-3162-8451**
- TRANS COL UNIQUE para el valor de origen **Smith, John** podría producir un valor de destino de: **Fnxwq, Lrzp**
- TRANS COL UNIQUE para el valor de origen **SMITH JOHN** podría producir un valor de destino de: **FNXWQ LRZP**
- TRANS COL UNIQUE (COPY=((1,3)(10,4))) para el valor de origen **CDE-7834-2008** podría producir un valor de destino de: **CDE-4032-2008**
- TRANS COL HASH (LENGTH=13) para el valor de origen **CDE-7834** podría producir un valor de destino de: **ZWQ-4598RN7A0**
- TRANS COL UNIQUE PRESERVE=(spaces) para el valor de origen **XYZ 477 6835** podría producir un valor de destino de: **LMN 623 0972**
- TRANS COL HASH TRIM=(e \u) para el valor de origen **InfoSphere** podría producir un valor de destino de: **RBIWACRL**

TRANS NID

Utilice la función TRANS NID para enmascarar números de ID nacionales.

Puede enmascarar números de ID nacionales con un método repetible que conserva parte del valor de origen o un método aleatorio que no conserva ninguna parte del valor de origen.

También puede especificar el tipo de separadores que se utilizan en los valores de salida (guiones, puntos, espacios, o sin separadores).

La función TRANS NID utiliza la siguiente sintaxis:

```
[TRANS] NID ('SWI=código_país', [FMT=(formato_salida)], [MTD={mask|random}],  
[SRC=nombre_columna], [VAL={Y|N}], [PRE=INV]')
```

SWITCH o SWI

Un valor de dos caracteres que indica el tipo de ID nacional que se debe enmascarar. Este parámetro es obligatorio. Sólo se permite un valor de conmutación.

Los siguientes valores de dos caracteres son válidos:

Canadá: número de seguridad social (SIN) canadiense
CA

Francia: número del Instituto Nacional Francés para estudios económicos y estadísticos (INSEE)
FR

Italia: número de código fiscal (CF) italiano
IT

España: número de identificación fiscal (NIF) español
ES

Reino Unido: número de seguro nacional (NINO) del Reino Unido
UK

Estados Unidos de América: número de seguridad social (SSN) de Estados Unidos US

FMT o FORMAT

Determina el formato de salida y qué partes del valor de origen deben enmascarse. La sintaxis de este parámetro está determinada por el ID nacional que se está enmascarando:

- Número de seguridad social (SIN) canadiense
- Número del Instituto Nacional Francés para estudios económicos y estadísticos (INSEE)
- Número de código fiscal (CF) italiano
- Número de identificación fiscal (NIF) español/Número de identificación de extranjeros (NIE)
- Número de seguro nacional (NINO) del Reino Unido
- Número de seguridad social (SSN) de Estados Unidos

Si se omite este parámetro y *MTD=mask*, se utiliza el formato de origen y se enmascararán los campos predeterminados.

Si se omite este parámetro y *MTD=random*, los valores de salida no incluirán separadores.

Si la columna de destino no es lo suficientemente grande para contener un formato de salida que utiliza separadores, los separadores no se incluirán.

MTD o METHOD

Determina el método de enmascaramiento que se utilizará: repetible o aleatorio. Si se omite este parámetro, se realiza el enmascaramiento repetible (*MTD=mask*).

mask Los valores de origen se enmascaran de forma repetible. Los valores de salida se basan en los valores de origen. Valor predeterminado.

random

Los valores de salida se generan mediante un algoritmo de enmascaramiento aleatorio. Los valores de salida no se basan en los valores de origen.

Nota:

- *MTD=random* no es compatible con los siguientes parámetros: *VAL=Y* y *PRE=INV*.
- Si *MTD=random* y el parámetro *FMT* especifica que parte del valor de origen debe copiarse en el valor de salida, el valor de salida no incluirá valores de origen; sin embargo, los separadores especificados en el parámetro *FMT* se incluirán en el valor de salida.

SRC o SOURCE

El nombre de la columna que contiene los valores de origen. Utilice este parámetro sólo si los nombres de las columnas de origen y destino no coinciden.

VAL o VALIDATE

Determina si la validación específica del país se realiza en los valores de origen. Si se omite este parámetro, la validación no se realiza (*VAL=N*).

Y Valida los valores de origen.

Nota: La opción no se puede utilizar si *MTD=random*.

N No valida los valores de origen. Valor predeterminado.

PRE o PRESERVE

Determina si los valores de origen no válidos se copian en la columna de destino. Si se omite este parámetro, los valores de origen no válidos no se copian en la columna de destino y las filas que contienen estos valores se omiten. Este parámetro sólo incluye una opción, *INV* o *INVALID*.

INV o *INVALID*

Copia valores de origen no válidos en la columna de destino.

Nota: La opción no se puede utilizar si *MTD=random*.

Ejemplos

La siguiente sintaxis enmascarará los números de seguridad (SSN) de Estados Unidos con un método repetible. La función copiará los tres primeros dígitos del SSN de origen e incluirá un separador de guión para el valor de salida. La validación se realizará en el valor de salida.

```
NID('SWITCH=US, FMT=(US=3C-2X-4X), MTD=MASK, VAL=Y')
```

La siguiente sintaxis enmascarará los números del Instituto Nacional Francés para estudios económicos y estadísticos (INSEE) con un método aleatorio. La función utilizará el formato de salida predeterminado, que no incluirá separadores.

```
NID('SWITCH=FR, MTD=RANDOM')
```

La siguiente sintaxis enmascarará los números de identificación fiscal (NIF) españoles con un método repetible. La función conservará los valores de origen no válidos y utilizará el formato de salida predeterminado.

```
NID('SWITCH=ES, MTD=MASK, PRE=INV')
```

Enmascaramiento del número de seguridad social canadiense

Puede utilizar la función TRANS NID para enmascarar números de seguridad social (SIN) canadiense.

Un SIN es un número de nueve dígitos que está formado por un número de código de región de un dígito seguido de un número de serie de ocho dígitos. Los tres primeros dígitos se denominan la cabecera. El último dígito del número de serie es un dígito de comprobación.

La función TRANS NID genera un SIN enmascarado con un dígito de comprobación que se calcula a partir de los ocho dígitos enmascarados precedentes del valor de salida.

Formatos de salida (FMT=)

Los siguientes formatos de salida están disponibles para un SIN.

C indica los valores que se copian. X indica valores que se enmascaran. Por ejemplo, 3C4X indica que los tres primeros caracteres se copian y que los cuatro caracteres siguientes se enmascaran.

Campos que se deben enmascarar	Formato sin separador	Formato con separador de guión	Formato con separador de espacio	Formato con separador de punto
Número de serie sin dígitos de cabecera (valor predeterminado <i>MTD=mask</i>)	CA=3C6X	CA=3C-3X-3X	CA=3C 3X 3X	CA=3C.3X.3X
Número de serie y dígitos de cabecera	CA=9X	CA=3X-3X-3X	CA=3X 3X 3X	CA=3X.3X.3X

Tipos de datos

Los tipos de datos siguientes están permitidos para las columnas de origen y destino. Se indican las restricciones para cada tipo de datos.

CHAR y NCHAR

La longitud de la columna debe tener al menos 9 caracteres.

DECIMAL

La precisión de la columna debe estar entre 9 y 20 y la escala debe ser 0.

INTEGER

Sin restricciones.

VARCHAR y NVARCHAR

La longitud de la columna debe tener al menos 9 caracteres.

Validación (VAL=Y)

Si se utiliza el parámetro (VAL=Y), se omitirá una fila de origen cuando se cumpla una de las siguientes condiciones:

- El primer dígito es ocho.
- Hay tres ceros consecutivos en las posiciones 1-3, 4-6 ó 7-9.
- El dígito de comprobación no es válido.

Proceso especial

Se realizarán las siguientes comprobaciones durante el proceso:

- Si el valor de origen es 0 (tipo de datos INTEGER o DECIMAL), espacios o bien un VARCHAR o NVARCHAR de longitud cero, y
 - la columna de destino es un tipo de datos INTEGER o DECIMAL, el valor de salida será 0.
 - la columna de destino es un tipo de datos CHAR o NCHAR, el valor de salida serán espacios.
 - la columna de destino es un tipo de datos VARCHAR o NVARCHAR, la longitud del destino será 0.
- Si la longitud de la columna de destino es 11 caracteres o más, y se han especificado separadores para el valor de salida, los separadores se incluirán si la columna de destino es un tipo de datos CHAR, NCHAR, VARCHAR o NVARCHAR.
- Si el valor de origen es NULL, el valor de salida será NULL.

Filas omitidas

Una fila de origen se omitirá y no se grabará en la tabla de destino si se produce lo siguiente:

Valor nulo

El valor de origen es NULL pero la columna de destino no permite un valor NULL.

SIN no válido

- La columna de origen es un CHAR, NCHAR, VARCHAR o NVARCHAR pero el valor de origen es inferior a 9 caracteres (sin incluir los separadores).
- La columna de origen es un CHAR, NCHAR, VARCHAR o NVARCHAR pero el valor de origen es superior a 9 caracteres (sin incluir los separadores).
- El valor de origen incluye un carácter no numérico.

Enmascaramiento de número del Instituto Nacional Francés para estudios económicos y estadísticos

Puede utilizar la función TRANS NID para enmascarar números del Instituto Nacional Francés para estudios económicos y estadísticos (INSEE).

Un número del INSEE es un número de 15 dígitos con el siguiente formato: SYMMDDCCCCOOCK.

S Información de sexo y nacionalidad.

YY Dos últimos dígitos del año de nacimiento.

MM Mes de nacimiento.

DD Departamento de origen.

CCC Municipio de origen.

OOO Número de orden.

KK Clave de control o dígitos de comprobación.

La función TRANS NID genera un INSEE enmascarado según las reglas siguientes:

- Si el campo de departamento se enmascara, el campo de municipio también se enmascara con un valor compatible.
- El campo de orden siempre está enmascarado.
- El campo de dígito de comprobación se calcula a partir de los 13 dígitos enmascarados precedentes del valor de salida.

Formatos de salida (FMT=)

Los siguientes formatos de salida están disponibles para un INSEE.

Todos los formatos enmascaran los campos de orden y dígito de comprobación. Si el campo de departamento se enmascara, el campo de municipio también se enmascarará con un valor compatible.

C indica los valores que se copian. X indica valores que se enmascaran. Por ejemplo, 3C4X indica que los tres primeros caracteres se copian y que los cuatro caracteres siguientes se enmascaran.

Campos a enmascarar (además de Orden y Dígito de comprobación)	Formato sin separador	Formato con separador de guión	Formato con separador de espacio
Sexo, Año, Mes, Municipio (valor predeterminado <i>MTD=mask</i>)	FR=5X2C8X	FR=5X2C6X-2X	FR=5X2C6X 2X
Sexo	FR=1X9C5X	FR=1X9C3X-2X	FR=1X9C3X 2X
Sexo, Año	FR=3X7C5X	FR=3X7C3X-2X	FR=3X7C3X 2X
Sexo, Mes	FR=1X2C2X5C5X	FR=1X2C2X5C3X-2X	FR=1X2C2X5C3X 2X
Sexo, Municipio	FR=1X6C8X	FR=1X6C6X-2X	FR=1X6C6X 2X
Sexo, Departamento	FR=1X4C8X	FR=1X4C6X-2X	FR=1X4C6X 2X
Sexo, Año, Mes	FR=5X5C5X	FR=5X5C3X-2X	FR=5X5C3X 2X
Sexo, Año, Municipio	FR=3X4C8X	FR=3X4C6X-2X	FR=3X4C6X 2X
Sexo, Año, Departamento, Municipio	FR=3X2C10X	FR=3X2C8X-2X	FR=3X2C8X 2X
Sexo, Mes, Municipio	FR=1X2C2X2C8X	FR=1X2C2X2C6X-2X	FR=1X2C2X2C6X 2X
Sexo, Mes, Departamento, Municipio	FR=1X2C12X	FR=1X2C10X-2X	FR=1X2C10X 2X
Sexo, Año, Mes, Departamento, Municipio	FR=15X	FR=13X-2X 13X	FR=13X 2X
Año	FR=1C2X7C5X	FR=1C2X7C3X-2X	FR=1C2X7C3X 2X
Año, Mes	FR=1C4X5C5X	FR=1C4X5C3X-2X	FR=1C4X5C3X 2X
Año, Municipio	FR=1C2X4C8X	FR=1C2X4C6X-2X	FR=1C2X4C6X 2X
Año, Departamento	FR=1C2X2C10X	FR=1C2X2C8X-2X	FR=1C2X2C8X 2X
Año, Mes, Municipio	FR=1C4X2C8X	FR=1C4X2C6X-2X	FR=1C4X2C6X 2X
Año, Mes, Departamento	FR=1C14X	FR=1C12X-2X	FR=1C12X 2X
Mes	FR=3C2X5C5X	FR=3C2X5C3X-2X	FR=3C2X5C3X 2X
Mes, Municipio	FR=3C2X2C8X	FR=3C2X2C6X-2X	FR=3C2X2C6X 2X

Campos a enmascarar (además de Orden y Dígito de comprobación)	Formato sin separador	Formato con separador de guión	Formato con separador de espacio
Mes, Departamento	FR=3C12X	FR=3C10X-2X	FR=3C10X 2X
Municipio	FR=7C8X	FR=7C6X-2X	FR=7C6X 2X
Departamento	FR=5C10X	FR=5C8X-2X	FR=5C8X 2X

Tipos de datos

Los tipos de datos siguientes están permitidos para las columnas de origen y destino. Se indican las restricciones para cada tipo de datos.

CHAR y NCHAR

La longitud de la columna debe tener al menos 15 caracteres.

VARCHAR y NVARCHAR

La longitud de la columna debe tener al menos 15 caracteres.

Validación (VAL=Y)

Si se utiliza el parámetro (VAL=Y), se omitirá una fila de origen cuando se cumpla una de las siguientes condiciones:

- El valor del campo de municipio de origen no es válido.
- El campo de dígito de comprobación de origen no es válido.

Proceso especial

Se realizarán las siguientes comprobaciones durante el proceso:

- Si el valor de origen son espacios o bien un VARCHAR o NVARCHAR de longitud cero, y
 - la columna de destino es un tipo de datos CHAR o NCHAR, el valor de salida serán espacios.
 - la columna de destino es un tipo de datos VARCHAR o NVARCHAR, la longitud del destino será 0.
- Si el valor de origen es NULL, el valor de salida será NULL.
- Si la longitud de una columna de destino son 16 caracteres o más y se ha especificado un separador para el valor de salida, se incluirá el separador.

Filas omitidas

Una fila de origen se omitirá y no se grabará en la tabla de destino si el valor de origen incluye los siguientes errores:

Valor nulo

El valor de origen es NULL pero la columna de destino no permite un valor NULL.

INSEE no válido

- El tamaño del valor INSEE de entrada es mayor o menor que el válido.
- El valor INSEE de entrada contiene separadores no válidos o separadores en la posición incorrecta.
- El campo de sexo no es uno de los siguientes valores: 1, 2, 7 u 8.

Enmascaramiento de un número de código fiscal italiano

Puede utilizar la función TRANS NID para enmascarar códigos fiscales (CF) italianos.

Un CF es un valor alfanumérico de 16 caracteres con el siguiente formato: FFF-NNN-YYMDD-RRRRC.

FFF Apellido codificado.
NNN Nombre de pila codificado.
YY Año de nacimiento.
M Mes de nacimiento.
DD Fecha de nacimiento.
RRRR Código de región.
C Carácter de control.

La función TRANS NID genera un CF enmascarado según las reglas siguientes:

- Cualquier consonante que aparece en los campos de nombre de pila o apellido se enmascara como consonante y cualquier vocal se enmascara como vocal. Si una X aparece después de una vocal, deberá copiarse en el valor de salida.
- El campo de carácter de control se calcula a partir de los 15 dígitos enmascarados precedentes del valor de salida.

Formatos de salida (FMT=)

Los siguientes formatos de salida están disponibles para un CF.

C indica los valores que se copian. X indica los valores que se enmascaran. Por ejemplo, 3C4X indica que los tres primeros caracteres se copian y que los cuatro caracteres siguientes se enmascaran.

Campos que se deben enmascarar	Formato sin separador	Formato con separador de guión	Formato con separador de espacio
Fecha de nacimiento, Región (valor predeterminado <i>MTD=mask</i>)	IT=6C10X	IT=3C-3C-5X-5X	IT=3C 3C 5X 5X
Apellido, Nombre de pila, Región	IT=6X5C5X	IT=3X-3X-5C-5X	IT=3X 3X 5C 5X
Apellido, Nombre de pila, Fecha de nacimiento	IT=11X4C1X	IT=3X-3X-5X-4C1X	IT=3X 3X 5X 4C1X
Apellido, Nombre de pila	IT=6X9C1X	IT=3X-3X-5C-4C1X	IT=3X 3X 5C 4C1X
Fecha de nacimiento	IT=6C5X4C1X	IT=3C-3C-5X-4C1X	IT=3C 3C 5X 4C1X
Región	IT=11C5X	IT=3C-3C-5C-5X	IT=3C 3C 5C 5X
Apellido, Nombre de pila, Fecha de nacimiento, Región	IT=16X	IT=3X-3X-5X-5X	IT=3X 3X 5X 5X

Tipos de datos

Los tipos de datos siguientes están permitidos para las columnas de origen y destino. Se indican las restricciones para cada tipo de datos.

CHAR y NCHAR

La longitud de la columna debe tener al menos 16 caracteres.

VARCHAR y NVARCHAR

La longitud de la columna debe tener al menos 16 caracteres.

Validación (VAL=Y)

Si se utiliza el parámetro VAL=Y, se omitirá una fila de origen si contiene un carácter de control no válido.

Proceso especial

Se realizarán las siguientes comprobaciones durante el proceso:

- Si el valor de origen son espacios o bien un VARCHAR o NVARCHAR de longitud cero, y
 - la columna de destino es un tipo de datos CHAR o NCHAR, el valor de salida serán espacios.
 - la columna de destino es un tipo de datos VARCHAR o NVARCHAR, la longitud del destino será 0.
- Si el valor de origen es NULL, el valor de salida será NULL.
- Si la longitud de una columna de destino es de 19 caracteres o más, y se han especificado separadores para el valor de salida, se incluirán los separadores.

Filas omitidas

Una fila de origen se omitirá y no se grabará en la tabla de destino si se produce lo siguiente:

Valor nulo

El valor de origen es NULL pero la columna de destino no permite un valor NULL.

CF no válido

La columna de origen tiene un valor de origen no válido o bien el valor de origen es demasiado grande.

El valor de origen tiene menos de 16 caracteres.

Enmascaramiento del número de identificación fiscal español y del número de identificación de extranjeros

Puede utilizar la función TRANS NID para enmascarar números de identificación fiscal (NIF) y números de identificación de extranjeros (NIE) de España.

Un NIF es un valor de ocho caracteres con el formato siguiente NNNNNNN-A, donde los siete primeros caracteres son un número de serie y el carácter final es un sufijo alfabético. El sufijo es un dígito de comprobación.

Los extranjeros residentes en España utilizan un número de identificación de extranjeros (NIE), que es un valor de nueve caracteres que utiliza el mismo formato que un NIF, pero va precedido por una X. Un NIE utiliza el siguiente formato: X-NNNNNNN-A.

La función TRANS NID genera un NIF o NIE enmascarado con un dígito de comprobación que se calcula a partir de los 7 dígitos enmascarados precedentes del valor de salida.

Formatos de salida (FMT=)

Los siguientes formatos de salida están disponibles para un NIF o NIE.

Para cada formato, todos los caracteres están enmascarados. Los números NIF y NIE utilizan las mismas opciones de formato. Un valor de origen NIE siempre incluirá un prefijo X en el valor de salida.

Campos que se deben enmascarar	Formato sin separador	Formato con separador de guión	Formato con separador de espacio
Serie, Sufijo (valor predeterminado <i>MTD=mask</i>)	ES=8X	ES=7X-1X	ES=7X 1X

Tipos de datos

Los tipos de datos siguientes están permitidos para las columnas de origen y destino. Se indican las restricciones para cada tipo de datos.

CHAR y NCHAR

La longitud de la columna debe tener al menos 8 caracteres.

VARCHAR y NVARCHAR

La longitud de la columna debe tener al menos 8 caracteres.

Validación (VAL=Y)

Si se utiliza el parámetro VAL=Y, se omitirá una fila de origen si contiene un sufijo no válido.

Proceso especial

Se realizarán las siguientes comprobaciones durante el proceso:

- Si el valor de origen son espacios o bien un VARCHAR o NVARCHAR de longitud cero, y
 - la columna de destino es un tipo de datos CHAR o NCHAR, el valor de salida serán espacios.
 - la columna de destino es un tipo de datos VARCHAR o NVARCHAR, la longitud del destino será 0.
- Si el valor de origen es NULL, el valor de salida será NULL.
- Si la longitud de una columna de destino son 11 caracteres o más y se ha especificado un separador para el valor de salida, se incluirá el separador.

Filas omitidas

Una fila de origen se omitirá y no se grabará en la tabla de destino si se produce lo siguiente:

Longitud no válida

El valor de entrada no contiene separadores y tiene menos de 8 caracteres (NIF) o tiene menos de 9 caracteres (NIE).

El valor de entrada contiene separadores y tiene menos de 9 caracteres (NIF) o tiene menos de 11 caracteres (NIE).

Patrón no válido

La longitud del valor de origen es válida, pero el valor no coincide con el patrón de un NIF o NIE.

Separadores

Se han encontrado diferentes separadores en las posiciones 2 y 10 de un valor de origen de NIE.

El valor de origen contiene un separador no válido.

Valor nulo

El valor de origen es NULL pero la columna de destino no permite un valor NULL.

Enmascaramiento del número de seguro nacional del Reino Unido

Puede utilizar la función TRANS NID para enmascarar números de seguro nacional (NINO) del Reino Unido.

Un NINO consta de tres partes: dos letras (el prefijo), seis dígitos (el número) y una letra opcional (el sufijo).

Formatos de salida (FMT=)

Los siguientes formatos de salida están disponibles para un NINO.

Un NINO se puede enmascarar sin un separador o con un separador en un formato de tres o cinco partes.

C indica los valores que se copian. X indica los valores que se enmascaran. Por ejemplo, 3C4X indica que los tres primeros caracteres se copian y que los cuatro caracteres siguientes se enmascaran.

Para crear un NINO sin un separador, utilice los parámetros siguientes:

Campos que se deben enmascarar	Formato sin separador
Prefijo, Número	UK=8X1C
Número (valor predeterminado <i>MTD=mask</i>)	UK=2C6X1C
Prefijo, Número, Sufijo (sólo <i>MTD=random</i>)	UK=9X

Para crear un NINO con un formato de tres o cinco partes, utilice los siguientes parámetros:

Campos que se deben enmascarar	Formato con separador de guión	Formato con separador de espacio	Formato con separador de punto
Prefijo, Número (tres partes)	UK=2X-6X-1C	UK=2X 6X 1C	UK=2X.6X.1C
Prefijo, Número (cinco partes)	UK=2X-2X-2X-2X-1C	UK=2X 2X 2X 2X 1C	UK=2X.2X.2X.2X.1C
Número (tres partes)	UK=2C-6X-1C	UK=2C 6X 1C	UK=2C.6X.1C
Número (cinco partes)	UK=2C-2X-2X-2X-1C	UK=2C 2X 2X 2X 1C	UK=2C.2X.2X.2X.1C
Prefijo, Número, Sufijo (tres partes) (sólo <i>MTD=random</i>)	UK=2X-6X-1X	UK=2X 6X 1X	UK=2X.6X.1X
Prefijo, Número, Sufijo (cinco partes) (sólo <i>MTD=random</i>)	UK=2X-2X-2X-2X-1X	UK=2X 2X 2X 2X 1X	UK=2X.2X.2X.2X.1X

Tipos de datos

Los tipos de datos siguientes están permitidos para las columnas de origen y destino. Se indican las restricciones para cada tipo de datos.

CHAR y NCHAR

La longitud de la columna debe tener al menos 9 caracteres.

VARCHAR y NVARCHAR

La longitud de la columna debe tener al menos 9 caracteres.

Validación (VAL=Y)

El parámetro VAL=Y no es válido para el enmascaramiento de NINO y se pasará por alto.

Proceso especial

Se realizarán las siguientes comprobaciones durante el proceso:

- Si el valor de origen son espacios o bien un VARCHAR o NVARCHAR de longitud cero, y
 - la columna de destino es un tipo de datos CHAR o NCHAR, el valor de salida serán espacios.
 - la columna de destino es un tipo de datos VARCHAR o NVARCHAR, la longitud del destino será 0.
- Si el valor de origen es NULL, el valor de salida será NULL.
- Si la columna de destino no puede acomodar un separador, el valor de salida no incluirá un separador.

Filas omitidas

Una fila de origen se omitirá y no se grabará en la tabla de destino si se produce lo siguiente:

Valor nulo

El valor de origen es NULL pero la columna de destino no permite un valor NULL.

NINO no válido

- El tamaño del valor de origen es menor o mayor que el tamaño de un NINO válido.
- El valor de origen incluye separadores en las posiciones incorrectas.
- El valor de origen incluye un separador no válido.
- El valor de origen incluye un prefijo no válido.
- El valor de origen incluye un sufijo distinto de A, B, C o D.
- El valor de origen incluye un campo de número que no está entre 000001 y 999999.

Enmascaramiento del número de seguridad social de Estados Unidos

Puede utilizar la función TRANS NID para enmascarar números de seguridad social (SSN) de Estados Unidos.

Un SSN está formado por 3 subcampos con el formato siguiente: AAAGGSSSS.

AAA Número de área. El área normalmente está determinada por el estado en el que se emite el SSN.

GG Número de grupo. Se asigna un número de grupo en función del número de área.

SSSS Número de serie.

La función TRANS NID genera un SSN enmascarado según las reglas siguientes:

- Se genera un número de grupo que sea adecuado para el número de área. El grupo será el grupo más reciente que ha utilizado la Administración de la Seguridad Social para el área.
- Los números de serie empiezan por 001 y se incrementan de 1 en 1 para cada SSN adicional generado para el número de área. Cuando el número de serie es mayor que 9999, el número de serie se restablecerá en 0001 y se utilizará el número de grupo que precede al número emitido más recientemente para el número de área.
- Cuando $MTD=mask$, el valor de salida incluirá un número de área correspondiente al mismo estado que el número de área de origen.

Formatos de salida (FMT=)

Los siguientes formatos de salida están disponibles para un SSN.

C indica los valores que se copian. X indica valores que se enmascaran. Por ejemplo, 3C4X indica que los tres primeros caracteres se copian y que los cuatro caracteres siguientes se enmascaran.

Campos que se deben enmascarar	Formato sin separador	Formato con separador de guión	Formato con separador de espacio	Formato con separador de punto
Grupo, número de serie (valor predeterminado $MTD=mask$)	US=3C6X	US=3C-2X-4X	US=3C 2X 4X	US=3C.2X.4X
Número de área, grupo o serie	US=9X	US=3X-2X-4X	US=3X 2X 4X	US=3X.2X.4X

Tipos de datos

Los tipos de datos siguientes están permitidos para las columnas de origen y destino. Se indican las restricciones para cada tipo de datos.

CHAR y NCHAR

La longitud de la columna debe tener al menos 9 caracteres.

DECIMAL

La precisión de la columna debe estar entre 9 y 20 y la escala debe ser 0.

INTEGER

Sin restricciones.

VARCHAR y NVARCHAR

La longitud de la columna debe tener al menos 9 caracteres.

Validación (VAL=Y)

Si se utiliza el parámetro (VAL=Y), se omitirá una fila de origen cuando se cumpla una de las siguientes condiciones:

- El número de área de origen sobrepasa el valor máximo.
- La Administración de la Seguridad Social no ha utilizado el número de área de origen.
- El número de grupo de origen no se ha utilizado para el número de área de origen.

Proceso especial

Se realizarán las siguientes comprobaciones durante el proceso:

- Si el valor de origen es 0 (tipo de datos INTEGER o DECIMAL), espacios o bien un VARCHAR o NVARCHAR de longitud cero, y
 - la columna de destino es un tipo de datos INTEGER o DECIMAL, el valor de salida será 0.
 - la columna de destino es un tipo de datos CHAR o NCHAR, el valor de salida serán espacios.
 - la columna de destino es un tipo de datos VARCHAR o NVARCHAR, la longitud del destino será 0.
- Si la longitud de una columna de destino es de 11 caracteres o más, y se han especificado separadores para el valor de salida, se incluirán los separadores.
- Si el valor de origen es NULL, el valor de salida será NULL.

Filas omitidas

Una fila de origen se omitirá y no se grabará en la tabla de destino si se produce lo siguiente:

Valor nulo

El valor de origen es NULL pero la columna de destino no permite un valor NULL.

SSN no válido

- La columna de origen es un CHAR, NCHAR, VARCHAR o NVARCHAR, pero el valor de origen tiene más o menos de 9 caracteres (sin incluir los separadores).
- El valor de origen incluye un carácter no numérico.
- El valor de origen es 0 o un valor reservado como 078-05-1120 y 457-55-5462.

Función de subserie

La función de subserie devuelve una subserie del contenido de la columna nombrada.

SUBSTR(*columnname*, *start*, [*length*])

columnname

Nombre de una columna binaria o de caracteres.

start La posición del primer carácter en la serie.

length Número de caracteres que deben utilizarse.

- Si el entorno local utiliza una coma como separador decimal, debe dejar un espacio después de cada coma que separa los parámetros numéricos (por ejemplo, después de la coma entre *start* y *length*).
- *start* y *length* son enteros mayores o iguales a 1.
- La suma de *start* y *length* no puede sobrepasar la longitud total de los datos más 1.
- Los valores *column-name* y *start* son obligatorios. Si sólo especifica un entero, se utilizará como valor *start*. La subserie empieza en *start* e incluye el resto del valor de la columna.

Ejemplo

Si la columna PHONE_NUMBER se define como CHAR(10), puede utilizar la función de subserie para correlacionar el código de área. Para obtener una subserie de las tres primeras posiciones del número de teléfono (código de área) para la columna de destino, especifique:

```
SUBSTR(PHONE_NUMBER, 1, 3)
```

Función aleatoria

La función aleatoria devuelve un número seleccionado aleatoriamente en el rango indicado por los valores bajo y alto.

`RAND(low, high)`

low Valor aleatorio más bajo posible.

high Valor aleatorio más alto posible.

- Utilice la función aleatoria con datos de tipo carácter o numéricos.
- Si el entorno local utiliza una coma como separador decimal, debe dejar un espacio después de la coma.
- *low* y *high* son enteros dentro del rango de -2.147.483.648 a 2.147.483.647.
- *low* y *high* también están limitados por el tipo y la longitud de los datos de la columna de destino.
- *low* debe ser menor que *high*.
- Cuando se utiliza la función aleatoria en una expresión concatenada, se devuelve una serie de longitud variable.

Ejemplo

Puede utilizar la función aleatoria para enmascarar o cambiar los datos de ventas de una base de datos de prueba. Supongamos que la columna YTD_SALES está definida como DECIMAL(7,2). El número máximo de dígitos que preceden al decimal es 5; el rango posible para esta columna es de -99999 a 99999. Para crear datos de prueba dentro de un rango de 1000 (bajo) a 89999 (alto), especifique:

```
RAND(1000, 89999)
```

En este ejemplo, la función devuelve valores de ventas aleatorios dentro del rango que ha especificado entre 1000,00 y 89999,99.

Función secuencial

La función secuencial devuelve un número que se incrementa secuencialmente. La sintaxis es:

`SEQ(start, step)`

start Valor inicial.

step Valor incremental.

- Utilice la función secuencial con datos de tipo carácter y numéricos.
- Si el entorno local utiliza una coma como separador decimal, debe dejar un espacio después de la coma.
- *start* y *step* son enteros dentro del rango de **-2.147.483.648** y **2.147.483.647**.
- *start* y *step* también están limitados por el tipo y la longitud de los datos de la columna de destino.
- Si el valor calculado sobrepasa la longitud de la columna de destino, la función se restablece automáticamente en el valor *start*.
- Cuando se utiliza la función secuencial en una expresión concatenada, se devuelve una serie de longitud variable.

Ejemplo 1

Puede utilizar la función secuencial para cambiar los datos del cliente para una base de datos de prueba. Supongamos que la columna CUST_ID está definida como CHAR(5). Para incrementar de 50 en 50 empezando por 1, especifique:

`SEQ(1, 50)`

En este ejemplo, la función devuelve valores CUST_ID que empiezan en '00001' y se incrementan de 50 en 50 para generar '00051', '00101', etc. Cuando el resultado sobrepasa '99951', la función se restablece en el valor *start* de 1.

Ejemplo 2

Puede utilizar la función secuencial en una correlación de columnas para enmascarar los datos de ventas de una base de datos de prueba. Supongamos que la columna YTD_SALES está definida como DECIMAL(7,2). Para incrementar de 100 en 100 empezando por 1000, especifique:

`SEQ(1000, 100)`

En este ejemplo, la función devuelve valores YTD_SALES que empiezan en 1000 y se incrementan de 100 en 100 para generar 1100, 1200, etc. Cuando el resultado sobrepasa 99999, la función se restablece en el valor *start* de 1000.

Ejemplo 3

Supongamos que la columna SALESMAN_ID está definida como CHAR(6). Para insertar valores que empiecen por 'NJ,' seguidos por un número que empieza en 50 y se incrementa de 10 en 10, utilice la función en una expresión concatenada:

`'NJ' || SEQ(50, 10)`

En este ejemplo, la función devuelve valores SALESMAN_ID que empiezan en 'NJ50 ' y se incrementan de 10 en 10 para generar 'NJ60 ', 'NJ70 ', etc. Cuando el resultado sobrepasa 'NJ9990', la función se restablece en el valor *start*.

Función de identidad o de serie

Las funciones de identidad o de serie indican al DBMS que proporcione un valor secuencial (entero) para una columna de destino. La sintaxis de estas funciones es la siguiente:

IDENTITY()

SERIAL ()

- Utilice la función de identidad para las columnas de identidad en bases de datos DB2, Sybase ASE y SQL Server.
- Utilice la función de serie para columnas de serie en bases de datos Informix.
- Las dos funciones son válidas para el proceso de inserción (actualizar/insertar) y el proceso de carga, pero no son válidas para el proceso de conversión.
- Si las filas se actualizan en un proceso de inserción (actualizar/insertar), la columna de destino que señala la función de identidad o la función de serie conserva el valor original. Además, si la columna de destino forma parte de la clave primaria, el valor de la columna permanece invariable cuando se actualiza la fila.
- Puede utilizar la función de identidad o la función de serie con la función de propagación para el proceso de inserción; sin embargo, no puede propagar las columnas de identidad o de serie en un proceso de carga.

Función de secuencia de Oracle

La función de secuencia de Oracle asigna un valor a la columna de destino utilizando una secuencia de Oracle.

schema.seqname.NEXTVAL [INCL_UPD]

schema Calificador para el nombre de secuencia de Oracle.

seqname

Nombre de la secuencia de Oracle que asigna valores secuenciales.

NEXTVAL

Palabra clave que inserta el siguiente valor de Oracle en la columna de destino.

INCL_UPD

Palabra clave opcional que actualiza un valor de secuencia asignado a una columna cuando se actualizan las filas durante un proceso de inserción. Si no se especifica (valor predeterminado), el valor de la columna permanece invariable cuando se actualiza la fila.

- Puede utilizar la función de secuencia de Oracle para asignar valores secuenciales exclusivos para filas que se insertarán en una base de datos de Oracle.
- La función de secuencia de Oracle es válida cuando se utiliza en una correlación de columnas para el proceso de inserción o de carga, pero no es válida para el proceso de conversión.
- Si las filas se actualizan en un proceso de inserción (actualización/inserción) y la columna de destino forma parte de la clave primaria, el valor de la columna permanece invariable cuando se actualiza la fila en la tabla de destino. Para utilizar la secuencia de Oracle al realizar una actualización/inserción, incluya *INCL_UPD* con la función.
- Durante un proceso de carga, el proceso utiliza la función de secuencia de Oracle para asignar un valor nuevo a cada fila de destino. La carga llama al DBMS para obtener estos valores. Si decide no ejecutar el cargador de Oracle, estos valores de secuencia no se utilizarán nunca.

Ejemplo 1

Para asignar un valor secuencial para aumentar los números de cliente, donde el nombre de la secuencia de Oracle es **schema.numeven**, especifique:

schema.numeven.NEXTVAL

Ejemplo 2

Para expandir el primer ejemplo y actualizar un valor de secuencia existente, especifique:

schema.numeven.NEXTVAL(INCL_UPD)

Función de propagación de valor de clave primaria o foránea

La función de propagación asigna un valor a una columna de clave primaria o de clave foránea y propaga este valor a todas las tablas relacionadas.

La sintaxis es:

```
PROP( { valor [, nombrecolumna | ] EXIT nombresalida |  
PROC { LOCAL | identificador.nombre } } )
```

valor Valor que se debe asignar a la columna. Especifique cualquier valor de origen de correlación de columnas válido (por ejemplo, un nombre de columna, literal de serie, expresión o función). El valor debe ser apropiado para la columna.

nombrecolumna

Nombre de la columna de origen que contiene el valor que es el asunto de la función. El valor resultante se inserta en la columna de destino de la tabla correlacionada y la columna de destino apropiada en las tablas relacionadas participantes.

El nombre de columna *sólo* es necesario si no hay ninguna columna de origen que coincida con la columna de destino tanto en el nombre como en el tipo de datos. Si no se especifica, se utiliza el nombre de la columna de destino.

nombresalida

Un nombre de salida de la correlación de columnas.

identificador.nombre

Un nombre de procedimiento de la correlación de columnas.

- Si el entorno local utiliza una coma como separador decimal, debe dejar un espacio después de cada coma que separe los parámetros numéricos.
- La función de propagación es válida en una correlación de columnas para el proceso de inserción (pero no actualización o actualización/inserción), carga o conversión.
- Cuando se utiliza la función de propagación, al menos una tabla relacionada debe incluirse en el proceso. Puede utilizar la función de propagación varias veces para el mismo proceso.
- Puede utilizar la función de propagación para una columna de clave primaria o su correspondiente columna de clave foránea, pero no para ambas.
- Si varias columnas definen una relación, puede utilizar la función de propagación para una o más de esas columnas. Sin embargo, en una relación ampliada de Optim, sólo puede especificar la función de propagación en las relaciones de columna a columna.
- Puede utilizar la función de identidad o la función de serie dentro de la función de propagación para el proceso de inserción; sin embargo, no puede propagar la función de identidad en un proceso de carga.
- Los parámetros especificados en la función de propagación no se validan hasta el tiempo de ejecución. Si hay conflictos, el proceso no se ejecutará.
- La inserción puede tener ciclos de propagación. Sin embargo, el proceso de carga e inserción podría no producir ciclos de propagación. Los ciclos se detectan cuando el proceso se valida en tiempo de ejecución. Si una solicitud de carga o conversión genera un ciclo de propagación, el proceso no se ejecuta.

- Optim recuerda los valores de origen y los valores asignados a las columnas de destino correspondientes. Por lo tanto, puede propagar a las columnas de destino donde el origen es una expresión. Cuando la expresión evaluada coincide con un valor de origen, Optim asigna el valor de destino correspondiente. Cuando la expresión evaluada no coincide con ningún valor de origen, se produce un error de conversión.

Antes de ejecutar un proceso de inserción, carga o conversión, puede revisar la correlación de columnas para verificar cómo se utiliza la función de propagación en el proceso.

Ejemplo 1

Puede generar un número aleatorio, asignarlo a la columna de destino predeterminada y propagar el número en las columnas de destino de las tablas relacionadas. Para generar un valor entre 10000 y 99999, insertarlo en la columna de destino correlacionada y propagarlo a las columnas de destino de las tablas relacionadas, especifique:

```
PROP(RAND(10000, 99999))
```

Ejemplo 2

Puede realizar la misma función que en el Ejemplo 1 cuando los nombres de las columnas de origen y destino no coinciden. Para incluir el nombre de la columna de origen (CUST_NUMBER) en la función de propagación, especifique:

```
PROP(RAND(10000, 99999), CUST_NUMBER)
```

Ejemplo 3

Puede utilizar la secuencia de Oracle para generar el valor para la columna de destino y propagar dicho valor en las columnas de destino de las tablas relacionadas. Para propagar la secuencia de Oracle denominada **schema.numeven**, especifique:

```
PROP(schema.numeven.NEXTVAL)
```

Expresiones concatenadas

La concatenación permite combinar valores de columna o combinar un valor de columna con otro valor, utilizando un operador de concatenación (CONCAT, ||, o +). Una expresión concatenada puede incluir valores de caracteres o valores binarios, pero no ambos:

Valores de caracteres

Los valores de caracteres concatenados pueden ser columnas de caracteres, literales de serie, subseries de valores en columnas de caracteres, la función secuencial o la función aleatoria.

Valores binarios

Los valores de caracteres concatenados pueden ser columnas binarias, literales hexadecimales, subseries de columnas binarias, la función secuencial o la función aleatoria.

- Una expresión concatenada no puede incluir un literal de serie de longitud cero (' '), un registro especial ni la función de antigüedad.

Ejemplo

Supongamos que la tabla CUSTOMERS almacena una dirección en dos columnas: ADDRESS1 y ADDRESS2. La tabla SHIP_TO almacena una dirección en una columna: ADDRESS. Puede utilizar una expresión concatenada para combinar información de dirección de dos columnas de una tabla con una columna de otra.

Para combinar la dirección, especifique uno de los ejemplos siguientes:

Tabla CUSTOMERS

ADDRESS1 || ADDRESS2

ADDRESS1 CONCAT ADDRESS2

ADDRESS1 + ADDRESS2

Tabla SHIP_TO

ADDRESS

ADDRESS

ADDRESS

Expresiones numéricas

Utilice una expresión numérica para especificar un valor en la columna de origen siempre que los tipos de datos para las columnas de origen y de destino correspondientes sean compatibles.

Una expresión numérica consta de la siguiente sintaxis:

operando1 operador operando2

Cada operando debe ser una columna numérica o una constante numérica. El operador especifica si hay que sumar (+), restar (-), dividir (/) o multiplicar (*).

Ejemplo 1

Para aumentar en un 10 por ciento el valor de una columna UNIT_PRICE definida como DECIMAL(5,2), especifique el siguiente ejemplo:

1.1 * UNIT_PRICE

Ejemplo 2

Para dividir por la mitad el valor de una columna ON_HAND_INVENTORY definida como INTEGER, especifique el siguiente ejemplo:

ON_HAND_INVENTORY / 2.

Funciones de literal y valor

Utilice las funciones de literal y valor para especificar literales como una serie o hexadecimal y valores como NULL o un registro especial.

Función	Descripción
Nombre de columna	Un nombre de columna explícito (los nombres de columna son sensibles a mayúsculas y minúsculas)
NULL	NULL. La columna de destino debe admitir nulos.
Constante numérica	Una constante numérica. El valor de constante debe caber en la columna de destino tal como está definida por su tipo de datos, precisión y escala.
Constante booleana	Una constante booleana (TRUE o FALSE).

Función	Descripción
Registro especial	<p>Un registro especial:</p> <ul style="list-style-type: none"> • CURRENT DATE • CURRENT_DATE • CURRENT TIME • CURRENT_TIME • CURRENT TIMESTAMP • CURRENT_TIMESTAMP • CURRENT SQLID • CURRENT_SQLID • CURDATE() • CURTIME() • GETDATE() • GETTIME() • SYSDATE() • NOW() • WORKSTATION_ID • USER
Literal de serie	<p>Un literal de serie, entre comillas simples. La columna de destino debe contener datos de tipo carácter.</p> <p>Ejemplo: 'CA' o '90210'.</p>
Literal hexadecimal	<p>Un literal hexadecimal.</p> <p>Ejemplo: X'1234567890ABCDEF' o 0X1234567890ABCDEF</p>
Literal de fecha/hora	<p>Un literal de fecha/hora entre comillas simples. Separe la fecha y la hora con un espacio. Para formatear la fecha/hora con una fracción decimal, coloque un punto después de la hora, seguido por la fracción. El formato de fecha está determinado por los valores de Opciones regionales en el Panel de control del sistema.</p> <p>Nota: Para columnas de indicación de fecha y hora de Oracle con huso horario, debe especificar el sufijo de huso horario en último lugar.</p>

Función de antigüedad

Utilice la función de antigüedad para definir la antigüedad de los valores de una columna de origen. La columna de origen puede contener datos de tipo carácter, numéricos, de fecha o de fecha y hora. Una columna CHAR o VARCHAR tiene una longitud máxima de 256 bytes.

El formato de la función de antigüedad es el siguiente:

AGE(parameters)

- Defina la función de antigüedad para incluir uno o más parámetros de antigüedad.
- Utilice comas o espacios para separar los parámetros de la función de antigüedad.
- Los parámetros se pueden especificar en cualquier orden.

Parámetro	Formato	Valores válidos
Nombre de columna: Especifique el nombre de la columna de origen si es distinto de la columna de destino.	<i>SC=nombre-columna</i> <i>SRCCOL=nombre-columna</i>	Nombre de columna

Parámetro	Formato	Valores válidos
Valor predeterminado: Defina la antigüedad de las fechas en función del valor de ajuste de fecha especificado en una solicitud de proceso.	DEF	Utiliza el valor de ajuste de fecha especificado en la solicitud de proceso.
Ninguno: No defina la antigüedad del valor.	NONE	No se debe definir la antigüedad del valor independientemente de las especificaciones en la solicitud del proceso.
Incremental – La antigüedad incremental se basa en una unidad de tiempo conocida. Optim da soporte a la antigüedad de fecha en unidades individuales (por ejemplo 20 años) o en varias unidades (por ejemplo, 2 años, 3 meses, 2 días).	+ o -] nY [+ o -] nM [+ o -] nW [+ o -] nD (El signo más [+] es opcional.)	nY -2500 a +1581 nM -30000 a +30000 nW -30000 a +30000 nD -99999 a +99999
Año específico: Defina la antigüedad de las fechas en función de un año específico de cuatro dígitos en el formato deseado	nnnnY	1582 - 3999
Múltiplo/regla: Defina la antigüedad de las fechas en función del número de veces que se debe aplicar una regla de negocio. Si define la función de antigüedad utilizando Múltiplo/regla, también debe incluir el parámetro RULE.	nnnnnR	1 - 30000

Antigüedad semántica

La antigüedad semántica se basa en un conjunto de reglas que se definen para gestionar fechas que se producen en vacaciones, fines de semana, etc. Puede utilizar la antigüedad semántica para ajustar fechas para que se produzcan en días laborables válidos.

Calendario –

Nombre del calendario que define las fechas especiales a las que se aplican las reglas. Si utiliza CALENDAR, también debe especificar un valor RULE.

CA=nombre-calendario

CALENDAR=nombre-calendario

Regla –

Nombre de la regla que define el ajuste para fechas especiales. Si se especifica DEF, se utiliza la regla predeterminada especificada en la solicitud de proceso.

RU=nombre-regla

RULE=nombre-regla

RU=DEF

RULE=DEF

Giro de siglo –

Determina el siglo para años de dos dígitos. Especifique un valor de 00 a 99.

PI=nn

PIVOT=nn

- Si especifica **AGE(RU=DEF)**, se utiliza el valor RULE especificado en una solicitud de proceso. Debe especificar valores para cualquier otro parámetro de función de antigüedad.

- Si utiliza RULE y no especifica un valor CALENDAR, la función de antigüedad utiliza el calendario predeterminado que especifique en una solicitud de proceso.
- Si no incluye CALENDAR, RULE y PIVOT donde sea necesario en la función de antigüedad, se aplicarán los valores predeterminados que especifique en una solicitud de proceso.
- Para especificar el siglo correcto para un año de dos dígitos, debe incluir el valor PIVOT en la función de antigüedad.
- Si especifica un valor PIVOT, todos los años de dos dígitos iguales o mayores que el valor PIVOT se colocan en el siglo XX (19xx). Todos los años de dos dígitos inferiores al valor PIVOT se colocan en el siglo XXI (20xx). El valor PIVOT predeterminado es 65.

Formatos de fecha

El formato de fecha de origen y el formato de fecha de destino deben contener un único formato de fecha válido que debe ser menor o igual que la longitud de la columna de destino. La serie de formato debe estar delimitada por comillas simples.

Formato de fecha de origen –

Aplica la serie de formato de la columna de origen al carácter de antigüedad y a las columnas numéricas.

SF='serie-formato'

SRCFMT='serie-formato'

Si la columna de origen es de tipo carácter o numérica, debe utilizar SRCFMT o una rutina de salida de origen (SRCEXIT) para describir el contenido de la columna. Estos parámetros se excluyen mutuamente.

Formato de fecha de destino –

Aplica la serie de formato de la columna de destino al carácter de antigüedad y a las columnas numéricas.

DF='serie-formato'

DSTFMT='serie-formato'

Si la columna de destino es de tipo carácter o numérico, puede especificar DSTFMT o una rutina de salida de destino (DSTEXIT). Si no especifica un formato para el destino, la función de antigüedad de fecha utiliza SRCFMT de forma predeterminada. La columna de destino para una función AGE no puede ser binaria.

Utilice las series de caracteres siguientes para especificar los componentes del formato de fecha:

Año	Mes	Día	Hora	Fracciones/Segundos
YYYY	MONTH	DDD	HH	FFFFFF
CCYY	MMM	DD	MI	FFFFF
YY	MM	D	SS	FFFF
	M			FFF
				FF
				F

- Si especifica un signo de interrogación (?) en una serie de formato, la función de antigüedad correlaciona el valor de carácter tal como está. (Utilice el signo de interrogación para incluir barras inclinadas, guiones, puntos, etc., en el formato de fecha).
- Si especifica un asterisco (*) en una serie de formato, la función de antigüedad correlaciona los caracteres restantes de la columna de origen con la columna de destino. (Utilice el asterisco cuando el valor de columna es una fecha concatenada con caracteres adicionales).

Nota: Puede utilizar el programa de utilidad de calendario para definir un separador predeterminado y un año de salida predeterminado. Estos valores predeterminados se aplican cuando los formatos de origen y destino requieren separadores o un año específico.

Ejemplo 1

Para definir la antigüedad de una columna de fecha en 2 años, 6 meses, 40 semanas y 15 días y, a continuación, aplicar una regla, formatee la función de antigüedad del modo siguiente:

```
AGE(+2Y,+6M,+40W,+15D,RU=NEXTPAYDAY)
```

Ejemplo 2

Para definir la antigüedad solamente de la parte del año de una columna de fecha en el año 2020, y aplicar una regla, formatee la función de antigüedad del modo siguiente:

```
AGE(2020Y,RU=NEXTWORKDAY)
```

Ejemplo 3

Para definir la antigüedad de una columna de fecha utilizando MULTIPLE/RULE para incrementar en cinco apariciones de una regla denominada NEXTSTRTQTR, utilizando un calendario denominado PSAPRULE, formatee la función de antigüedad del siguiente modo:

```
AGE(CA=PSAPRULE,RU=NEXTSTRTQTR,5R)
```

Ejemplo 4

Para definir la antigüedad de los datos de una columna de tipo carácter o numérico de acuerdo con los siguientes parámetros.

- Una columna de origen con nombre.
- El formato de origen, utilizando los dos primeros caracteres para los dos últimos dígitos del año y los 3 dígitos restantes como día del calendario juliano.
- Un giro de siglo para determinar el siglo correcto porque el origen se formatea con un año de dos dígitos. El giro de siglo de este ejemplo es 42. Todos los años de dos dígitos superiores o iguales a 42 se colocan en el siglo XX (19xx). Todos los años de dos dígitos inferiores a 42 se colocan en el siglo XXI (20xx).
- Definir la antigüedad de una fecha en 5 años.

Formatee la función de antigüedad del siguiente modo:

```
AGE(5Y,SC=ORDER_DATE,SF='YYDDD',PI=42)
```

Función de moneda

Utilice la función de moneda para convertir un valor de moneda en una columna de origen de una moneda a otra. La columna de origen debe definirse como numérica, pero no de coma flotante. Están disponibles dos métodos de conversión:

Conversión directa

Proporcione parámetros de conversión basados en los valores definidos en una definición de moneda. Utilice la función de moneda para convertir un valor monetario de una columna (sustituyendo el valor anterior) o bien, definiendo columnas de origen y destino distintas, conserve el valor original y el valor convertido. Puede definir explícitamente los tipos de moneda de origen y de destino o puede identificar una columna de referencia para indicar el tipo de moneda.

La primera preferencia de cálculo es utilizar el tipo de cambio de la moneda de origen en la moneda de destino. La segunda preferencia de conversión del cálculo es utilizar el tipo de cambio para la moneda de destino en el origen.

Triangulación

Convierta el valor de una columna desde la moneda de origen al euro dólar y, a continuación, el euro dólar a la moneda de destino. Ambos tipos de cambio deben proporcionarse en la tabla de tipos de cambio: una entre el euro dólar y la moneda de origen, y la otra entre el euro dólar y la moneda de destino. La expresión de la especificación es TRIANG o TR.

El formato de la función de moneda es el siguiente:

```
CURRENCY( {ST=code | SS=(column-name,Types Table number)}
{DT=code | DS=(column-name,Types Table number)}
[SC=column-name] [TR] [CU=Currency Definition name]
[TD=transaction-date-column-name] [DF='format']
[NS=scale] )
```

La función de moneda debe incluir al menos una combinación del tipo de moneda de origen (ST) o especificación de origen (SS) y el tipo de moneda de destino (DT) o especificación de destino (DS). El resto de los parámetros son opcionales.

Los tipos de moneda de origen y de destino pueden especificarse de una de las dos maneras siguientes:

1. Utilice las palabras clave ST/DT para permitir la especificación explícita de la moneda utilizando el código de moneda ISO 4217 de tres caracteres.
2. Utilice las palabras clave SS/DS para permitir la especificación indirecta del tipo de moneda donde un valor en una columna nombrada en la fila se utiliza como clave. La clave se correlaciona con un tipo de moneda tal como se define en la tabla de tipo de definición de moneda especificada.

Si especifica una fecha de transacción (TD) y la columna de fecha de transacción no utiliza el formato DATE, también debe especificar un formato de fecha (DF). Si una fecha de transacción especificada está fuera del rango de fechas especificado en la tabla de tipos de definición de moneda, se utiliza el rango de fechas más próximo para los cálculos de conversión.

Nota: Si faltan datos necesarios (como por ejemplo tipos de moneda, tipos de cambio), se producirán errores de conversión en tiempo de ejecución.

La tabla siguiente describe el formato válido y valores permitidos para los parámetros de función de moneda. Los parámetros se pueden especificar en cualquier orden. Utilice comas o espacios para separar los parámetros de la función de moneda.

Parámetro	Formato
Columna de origen	SC=nombre-columna SRCCOL=nombre de columna
Tipo de moneda de origen	ST=código SRCTYP=código donde código = Código de moneda ISO 4217
Tipo de moneda de destino	DT=código DSTTYP=código donde código = Código de moneda ISO 4217

Parámetro	Formato
Especificación de origen	SS=nombre de columna, número de tabla de tipos SRCSPC=nombre de columna, número de tabla de tipos donde nombre de columna, número de tabla de tipos = columna y número de tabla de tipos (definidos en la definición de moneda) que se deben utilizar para especificar el tipo de moneda de origen
Especificación de destino	DS=nombre de columna, número de tabla de tipos DSTSPC=nombre de columna, número de tabla de tipos donde nombre de columna, número de tabla de tipos = columna y número de tabla de tipos (definidos en la definición de moneda) que se deben utilizar para especificar el tipo de moneda de destino
Triangulación (fuerza la conversión a través del euro dólar)	TR TRIANG
Definición de moneda	CU=nombre de definición de moneda CURTBL=nombre de definición de moneda donde nombre de definición de moneda = definición de moneda que contiene los parámetros de conversión apropiados.
Fecha de transacción	TD=nombre de columna TRNDAT=nombre de columna donde nombre de columna = nombre de la columna Fecha de transacción para identificar la fecha de conversión
Formato de fecha	DF='formato' DATFMT='formato' donde formato = formato de la columna de fecha de transacción, si no es tipo de fecha.
Escala numérica	NS=escala NUMSCL=escala donde escala = escala que debe aplicarse a las columnas de destino numéricas de Oracle con una escala no definida.

Ejemplo 1

Para convertir de marcos finlandeses a euro dólares, formatee la función de moneda del siguiente modo:

```
CURRENCY(ST=FIM DT=EUR)
```

Si se debe conservar el valor original, utilice la función de moneda para proporcionar un valor para una columna diferente en la tabla de destino.

Ejemplo 2

Para convertir de marcos finlandeses a euro dólares, y crear una nueva columna para conservar el valor de origen original (en marcos finlandeses) en una columna con la etiqueta ITEM_COST, formatee la función de moneda del siguiente modo:

```
CURRENCY(ST=FIM DT=EUR SC=ITEM_COST)
```

Nombre de correo electrónico generado automáticamente

La función de nombre de correo electrónico generado automáticamente genera una dirección de correo electrónico con un nombre de usuario basado en un literal concatenado con un número secuencial. Los

números secuenciales son sufijos que comienzan por 1 y se incrementan de 1 en 1. La función utiliza el nombre de dominio de una dirección de correo electrónico en una columna de origen especificada.

Nombre de correo electrónico formateado

La función de nombre de correo electrónico formateado genera una dirección de correo electrónico con un nombre de usuario basado en los valores obtenidos de uno o dos atributos. La política utiliza el nombre de dominio de una dirección de correo electrónico de una columna de origen especificada.

Función de número aleatorio

La función de número aleatorio genera números seleccionados aleatoriamente en el rango indicado por los valores bajo y alto.

La función de número aleatorio se puede utilizar para sustituir datos numéricos o caracteres. Los valores inferior y superior deben ser enteros dentro del rango -2,147,483,648 a 2,147,483,647. El valor bajo debe ser menor que el valor alto.

Función de número secuencial

La función de número secuencial genera números que se incrementan secuencialmente.

La función de número secuencial se puede utilizar para sustituir datos numéricos o de caracteres. Es preciso que entre un valor inicial y un valor por el cual se incrementan los números. Los valores inicial e incremental deben ser enteros dentro del rango -2,147,483,648 a 2,147,483,647.

El valor generado está limitado por el tipo de datos y la longitud de la columna de destino. Si el valor generado sobrepasa la longitud de la columna de destino, la función se restablece automáticamente en el valor inicial.

Utilización de rutinas de salida

Cuando se crea un servicio para convertir, crear, insertar, cargar o restaurar datos, puede especificar una correlación de tablas que incluya una o más correlaciones de columnas para derivar los valores adecuados para las columnas de destino.

Existen diversas maneras de especificar un valor de columna en una correlación de columnas. Una forma consiste en especificar una rutina de salida como origen; la salida establece valores que de otro modo no se podrían definir para las columnas de destino. Otra forma consiste en utilizar rutinas de salida para excluir filas del proceso. Puede utilizar tres tipos de rutinas de salida:

Salida estándar

La rutina de salida estándar se llama para derivar el valor para una columna de destino en una correlación de columnas. Este tipo de rutina de salida es útil cuando se desea realizar transformaciones de datos que están más allá del ámbito de las correlaciones de columnas. Por ejemplo, una salida puede cambiar un número de departamento de empleado para las filas seleccionadas de acuerdo con un algoritmo complejo, o bien seleccionar filas específicas que se deben procesar y descartar todas las demás.

Una salida estándar puede obtener un segmento de subserie de una columna LOB de origen. Para insertar un valor LOB nuevo en una columna LOB de destino, una salida puede crear un archivo y pasar el nombre de archivo de nuevo al procesador de correlaciones de columnas.

Salida de formato de origen

Se llama a la salida de formato de origen para formatear la columna de origen en una función de antigüedad que de otra forma no estaría soportada en una correlación de columnas. Esta rutina de salida

examina la fecha de origen en una columna de caracteres o enteros y la convierte en un formato de fecha utilizable como entrada a la función de antigüedad.

Salida de formato de destino

Se llama a la salida de formato de destino para formatear la columna de destino en una función de antigüedad que de otra forma no estaría soportada en una correlación de columnas. Esta rutina de salida convierte una fecha en uno de los cuatro formatos de destino distintos. El tipo de datos de la columna de destino determina el formato.

Salida en una correlación de columnas

Para utilizar una rutina de salida en una correlación de columnas, debe especificar una de las opciones siguientes en la columna de origen apropiada:

Salida estándar

EXIT *dllname*[(*parm*[[,*parm*]...)]

Salida de formato de origen

AGE(SRCEXIT=*dllname*)

Salida de formato de destino

AGE(DSTEXIT=*dllname*)

El proceso llama a la rutina de salida de la correlación de columnas, una vez para cada fila de datos procesada y pasa una llamada de terminación después de que se haya procesado la última fila. Los parámetros opcionales especificados con una rutina de salida estándar se pasan a la salida y deben ser literales de serie (entre comillas simples) o literales numéricos (límite 8).

Escribir rutinas de salida

Puede escribir rutinas de salida en cualquier lenguaje de programación; sin embargo, las llamadas a subrutinas deben ajustarse a los convenios que se utilizan en el lenguaje de programación C.

Archivos de cabecera

Para definir los parámetros, estructuras y códigos de retorno utilizados en la rutina de salida, una rutina de salida debe incluir dos archivos de cabecera de programa C, PSTEXIT.H y, en función del formato de caracteres de los metadatos (nombres de tabla, nombres de columna, etc.), ya sea PSTCMXIT.H o PSTCMWXT.H:

PSTEXIT.H

Especifica los tipos de datos, códigos de retorno y estructuras para tipos de datos definidos por Optim.

PSTCMXIT.H

Para metadatos en formato de un solo byte (por ejemplo, ASCII).

Proporciona los prototipos para la rutina de devolución de llamada de correlación de columnas y especifica las *definiciones* y *definiciones de estructura* para los parámetros de salida de correlación de columnas.

PSTCMWXT.H

Para metadatos en formato UTF-16 (WCHAR).

Proporciona los prototipos para las rutinas de devolución de llamada de correlación de columnas y especifica las *definiciones* y *definiciones de estructura* para los parámetros de salida de correlación de columnas.

Estos archivos de cabecera se encuentran en el mismo directorio con los archivos de aplicación de IBM InfoSphere Optim.

Nota: Si un parámetro tiene un formato de un solo byte y un formato UTF-16, este capítulo proporciona el formato UTF-16 entre paréntesis.

Utilización de DLL

Debe compilar y enlazar cada rutina de salida como una DLL distinta. Optim carga cada DLL dinámicamente en tiempo de ejecución. Una DLL sólo puede contener una aparición de un determinado tipo de salida de correlación de columnas. Sin embargo, puede incluir una de cada tipo de rutina de salida de correlación de columnas en la misma DLL.

Debe utilizar el nombre de la DLL en la correlación de columnas. Especifique el nombre de la DLL tal como se ha definido para la plataforma (es decir, no utilice el nombre genérico). Escriba el nombre de la DLL y exporte las funciones reales que implementan la rutina de salida:

Salida estándar

PSTColMapExit
(PSTColMapWExit)

Salida de formato de origen

PSTColMapAgeSrcExit
(PSTColMapAgeSrcWExit)

Salida de formato de destino

PSTColMapAgeDstExit
(PSTColMapAgeDstWExit)

Requisitos

Cada rutina de salida de correlación de columnas debe satisfacer los siguientes requisitos:

- La salida debe generar un valor apropiado para la columna de destino y no debe cambiar otras áreas de almacenamiento.
- Optim utiliza la clava primaria para crear áreas de trabajo internas. Si la rutina de salida asigna un valor a una columna que forma parte de la clave primaria, y dicha rutina se llama varias veces para la misma fila de origen, asegúrese de que la rutina de salida genera el mismo valor de salida cada vez.
- Una salida puede llamarse frecuentemente; evite una sobrecarga innecesaria. Por ejemplo, la salida debe inicializar el proceso en la primera llamada y guardar la información para las llamadas posteriores en el área de trabajo.

Rutina de salida estándar

Cuando escriba una salida estándar, especifique la función de salida, PSTColMapExit (PSTColMapWExit), y los parámetros siguientes:

un solo byte

```
PSTColMapExit  
(PST_STRUCT_CM_EXIT_PARM * pInputParms,  
 PST_STRUCT_CM_EXIT_COL_LIST * pSrcColList,  
 PST_STRUCT_CM_EXIT_COL_LIST * pDstColList)
```

UTF-16

```
PSTColMapWExit  
(PST_STRUCT_CM_WEXIT_PARM * pInputParms,  
 PST_STRUCT_CM_WEXIT_COL_LIST * pSrcColList,  
 PST_STRUCT_CM_WEXIT_COL_LIST * pDstColList)
```

Parámetros

Cuando un proceso llame a una rutina de salida estándar, el proceso pasa los siguientes parámetros en cada llamada:

pInputParms

Puntero a PST_STRUCT_CM_EXIT_PARM (PST_STRUCT_CM_WEXIT_PARM). Esta estructura contiene información acerca de las tablas de origen y destino, la naturaleza de la llamada actual, el número de parámetros opcionales especificados con la salida, así como punteros a las funciones de devolución de llamada, las áreas de trabajo y los parámetros opcionales.

El primer campo de esta estructura es el campo **FuncCode** identificado mediante PST_CM_FUNC_TRANSFORM (PST_CMW_FUNC_TRANSFORM) o PST_CM_FUNC_TERMINATE (PST_CMW_FUNC_TERMINATE).

El campo *NumParms* contiene el número de parámetros opcionales especificados con la salida de correlación de columnas (0 a 8).

El campo *pParm* contiene una matriz de punteros a cada parámetro opcional especificado con la salida de correlación de columnas.

pScrColList

Puntero a PST_STRUCT_CM_EXIT_COL_LIST (PST_STRUCT_CM_WEXIT_COL_LIST). Esta estructura describe las columnas de origen.

pDstColList

Puntero a PST_STRUCT_CM_EXIT_COL_LIST (PST_STRUCT_CM_WEXIT_COL_LIST). Esta estructura describe las columnas de destino.

Funciones de devolución de llamada

Cada vez que un proceso Optim llama a una rutina de salida estándar, el proceso pasa las direcciones de las siguientes funciones de devolución de llamada:

pPSTGetColValue()

Recupera datos para todas las columnas de origen y la mayoría de las columnas de destino en la fila de datos actual. En general, la rutina de salida llama a esta función sólo una vez para recuperar los datos para una columna de origen. Sin embargo, la rutina de salida puede llamar a esta función varias veces para recuperar los datos para diferentes columnas.

pPSTPutColValue()

Especifica datos para la columna de destino en la fila de datos actual. La rutina de salida determina el valor para la columna de destino y devuelve el valor a Optim. Debe llamarse a esta función a no ser que la fila se rechace o el proceso termine anormalmente.

Proceso

El proceso típico para la rutina de salida estándar se resume en los pasos siguientes:

1. En cada llamada de Optim, la rutina de salida comprueba una primera llamada. En la primera llamada, la salida realiza las tareas de inicialización y el proceso normal (paso 2). En las llamadas siguientes, sólo se realiza el proceso normal (paso 2).
2. Optim no pasa datos de columna a la rutina de salida estándar; sin embargo, la rutina de salida puede realizar llamadas a la función de devolución de llamada **pPSTGetColValue()** para obtener datos para las columnas de origen. Estos valores son necesarios para determinar el valor de la columna de destino.
3. Después de que la rutina de salida genera el valor de destino, la salida llama a la función devolución de llamada **pPSTPutColValue()** para almacenar el valor en la columna de destino o pasa un código de retorno apropiado indicando al proceso que ignore la fila de datos o que termine anormalmente.

- Una vez que se ha procesado la última fila de datos, Optim pasa una llamada de terminación a la rutina de salida, identificada mediante un valor de PST_CM_FUNC_TERMINATE (PST_CMW_FUNC_TERMINATE) en el campo FuncCode field. Esta llamada solicita a la rutina de salida que libere cualquier almacenamiento asignado dinámicamente. Cuando las tareas finales se hayan completado, la rutina de salida pasa un código de retorno a Optim.

Códigos de retorno

Los siguientes códigos de retorno se aplican a las rutinas de salida estándar:

PST_CM_EXIT_SUCCESS
(PST_CMW_EXIT_SUCCESS)

Se asigna un valor a la columna de destino o ésta se transforma satisfactoriamente.

PST_CM_EXIT_REJECT_ROW
(PST_CMW_EXIT_REJECT_ROW)

No se puede asignar un valor a la columna de destino ni tampoco se puede transformar. Descarte la fila.

PST_CM_EXIT_ABORT_PROCESS
(PST_CMW_EXIT_ABORT_PROCESS)

Error muy grave. Termine el proceso. Para devolver un mensaje de error, coloque el mensaje en el área de trabajo y establezca el espacio no utilizado en blancos o NULL.

Salida de formato de origen

Cuando escriba una rutina de salida de formato de origen para Optim, especifique la función de salida, PSTColMapAgeSrcExit (PSTColMapAgeSrcWExit), y los parámetros siguientes:

un solo byte

```
PSTColMapAgeSrcExit  
(PST_STRUCT_CM_AGE_SRCFMT_PARM * pInputParms,  
 PST_STRUCT_CM_EXIT_COL_LIST * pSrcColList,  
 PST_STRUCT_CM_EXIT_COL_LIST * pDstColList)
```

UTF-16

```
PSTColMapAgeSrcWExit  
(PST_STRUCT_CM_AGE_SRCFMT_WPARAM *  
 pInputParms,  
 PST_STRUCT_CM_WEXIT_COL_LIST * pSrcColList,  
 PST_STRUCT_CM_WEXIT_COL_LIST * pDstColList)
```

Parámetros

Cuando un proceso de Optim llama a una rutina de salida de formato de origen, el proceso pasa los parámetros siguientes:

pInputParms

Puntero a PST_STRUCT_CM_AGE_SRCFMT_PARM (PST_STRUCT_CM_AGE_SRCFMT_WPARAM). Esta estructura contiene información sobre las tablas de origen y destino, la naturaleza de la llamada actual, y los punteros a una función de devolución de llamada y las áreas de trabajo.

El primer campo de esta estructura es el campo **FuncCode** identificado mediante PST_CM_SRCFMT_TRANSFORM (PST_CMW_SRCFMT_TRANSFORM) o PST_CM_SRCFMT_TERMINATE (PST_CMW_SRCFMT_TERMINATE).

pSrcColList

Puntero a PST_STRUCT_CM_EXIT_COL_LIST (PST_STRUCT_CM_WEXIT_COL_LIST). Esta estructura describe las columnas de origen.

pDstColList

Puntero a PST_STRUCT_CM_EXIT_COL_LIST (PST_STRUCT_CM_WEXIT_COL_LIST). Esta estructura describe las columnas de destino.

Función de devolución de llamada

Cada vez que un proceso de Optim llama a una rutina de salida de formato de origen, el proceso pasa la dirección de la siguiente función de devolución de llamada:

pPSTGetColValue()

Recupera datos para todas las columnas de origen y la mayoría de las columnas de destino en la fila de datos actual. En general, no es necesario que la rutina de salida llame a esta función porque los datos para la columna de origen se proporcionan en el primer parámetro. Sin embargo, la rutina de salida puede llamar a esta función para recuperar los datos para diferentes columnas.

Proceso

El proceso típico para la rutina de salida de formato de origen se resume en los pasos siguientes:

1. En cada llamada de Optim, la rutina de salida comprueba una primera llamada. En la primera llamada, la salida realiza las tareas de inicialización y el proceso normal (paso 2). En las llamadas siguientes, sólo se realiza el proceso normal (paso 2).
2. La salida recibe el valor de la columna de origen como se especifica en la función de antigüedad definida en la correlación de columnas. Optim no aplica la función de antigüedad antes de llamar a la rutina de salida y almacena el valor sin formato en uno de los campos de la unión **InputValue** tal como indica el campo **ValueType** en el archivo de cabecera.
3. Si la salida necesita examinar otras columnas para calcular el valor para la columna de destino, la salida debe llamar a la función de devolución de llamada **pPSTGetColValue()** para obtener el valor para dichas columnas.
4. Una vez que se ha generado el valor de destino, la rutina de salida debe formatear el valor y colocarlo en el campo **OutputTimeStamp** o en el campo **OutputSybDateTime**. Optim valida este valor y aplica la función de antigüedad. La salida debe devolver un código de retorno apropiado que indique el campo donde se guardan los datos o que instruya al proceso que se salte la fila de datos o que termine anormalmente.
5. Una vez que se ha procesado la última fila de datos, Optim pasa una llamada de terminación a la rutina de salida, identificada mediante un valor de PST_CM_SRCFMT_TERMINATE (PST_CMW_SRCFMT_TERMINATE) en el campo **FuncCode**. Esta llamada solicita a la rutina de salida que libere cualquier almacenamiento asignado dinámicamente. Cuando las tareas finales se hayan completado, la rutina de salida pasa un código de retorno a Optim.

Modalidades de terminación anormal

Existen varias maneras en que la rutina de salida de formato de origen puede terminar anormalmente el proceso:

- Procesar filas con fechas omitidas o fechas no válidas. Si selecciona una de estas opciones en una solicitud de proceso, y las columnas de origen y de destino tienen los mismos atributos, la columna de origen se copia sin cambios a la columna de destino. Si no selecciona ninguna de las dos opciones, la fila se rechaza.
- Rechazar la fila. Rechace la fila independientemente de las opciones de proceso para las fechas omitidas o no válidas, en función de las especificaciones de la rutina de salida.
- Terminar anormalmente todo el proceso, en función de las especificaciones de la rutina de salida.

La rutina de salida devuelve un código de retorno que indica el formato de fecha o la modalidad de terminación anormal que se debe utilizar.

Durante un proceso, la rutina de salida puede interrogar las columnas de la fila de entrada y algunas de las columnas de la fila de destino. Sin embargo, la rutina de salida no puede interrogar una columna de destino que incluye una rutina de salida y está definida en la correlación de columnas después de la columna de destino actual. Todas las otras columnas de destino están disponibles.

Códigos de retorno

Los siguientes códigos de retorno se aplican a las salidas de formato de origen:

```
PST_CM_SRCFMT_USE_TIMESTAMP  
(PST_CMW_SRCFMT_USE_TIMESTAMP)
```

La columna de destino tiene asignado un valor en el campo **OutputTimeStamp** del primer parámetro que se pasa a la salida.

```
PST_CM_SRCFMT_USE_SYB_DATETIME  
(PST_CMW_SRCFMT_USE_SYB_DATETIME)
```

La columna de destino tiene asignado un valor en el campo **OutputSybDateTime** del primer parámetro que se pasa a la salida.

```
PST_CM_SRCFMT_SKIP  
(PST_CMW_SRCFMT_SKIP)
```

No se aplica la antigüedad. Si no selecciona la opción para **Procesar filas con fechas omitidas**, la fila se rechaza. De lo contrario, los datos se copiarán en el origen, siempre que el origen y el destino sean compatibles. Si no son compatibles, la fila se rechaza.

```
PST_CM_SRCFMT_COL_INVALID  
(PST_CMW_SRCFMT_COL_INVALID)
```

No se aplica la antigüedad. Si no selecciona la opción para **Procesar filas con fechas no válidas**, la fila se rechaza. De lo contrario, los datos se copiarán en el origen, siempre que el origen y el destino sean compatibles. Si no son compatibles, la fila se rechaza.

```
PST_CM_SRCFMT_REJECT_ROW  
(PST_CMW_SRCFMT_REJECT_ROW)
```

La columna de origen no puede tener asignado un valor. Rechace (descarte) la fila.

```
PST_CM_SRCFMT_ABORT_PROCESS  
(PST_CMW_SRCFMT_ABORT_PROCESS)
```

Error muy grave. Termine. Para devolver un mensaje de error, coloque el mensaje en el área de trabajo y establezca el espacio no utilizado en blancos o NULL.

Salida de formato de destino

Cuando escriba una rutina de salida de formato de destino para Optim, especifique la función de salida, PSTColMapAgeDstExit (PSTColMapAgeDstWExit), y los parámetros siguientes:

un solo byte

```
PSTColMapAgeDstExit  
(PST_STRUCT_CM_AGE_DSTFMT_PARM * pInputParms,  
 PST_STRUCT_CM_EXIT_COL_LIST * pSrcColList,  
 PST_STRUCT_CM_EXIT_COL_LIST * pDstColList)
```

UTF-16

```
PSTColMapAgeDstWExit  
(PST_STRUCT_CM_AGE_DSTFMT_WPARAM * pInputParms,  
 PST_STRUCT_CM_WEXIT_COL_LIST * pSrcColList,  
 PST_STRUCT_CM_WEXIT_COL_LIST * pDstColList)
```

Parámetros

Cuando un proceso de Optim llama a una rutina de salida de formato de destino, el proceso pasa los parámetros siguientes:

pInputParms

Puntero a PST_STRUCT_CM_AGE_DSTFMT_PARM (PST_STRUCT_CM_AGE_DSTFMT_WPARAM). Esta estructura contiene información sobre las tablas de origen y destino, la naturaleza de la llamada actual, y los punteros a una función de devolución de llamada y las áreas de trabajo.

El primer campo de esta estructura es el campo **FuncCode**, identificado mediante:

PST_CM_DSTFMT_TO_CHAR (PST_CMW_DSTFMT_TO_WCHAR),
PST_CM_DSTFMT_TO_INTEGER (PST_CMW_DSTFMT_TO_INTEGER),
PST_CM_DSTFMT_TO_TIMESTAMP (PST_CMW_DSTFMT_TO_TIMESTAMP),
PST_CM_DSTFMT_TO_SYB_DATETIME (PST_CMW_DSTFMT_TO_SYB_DATETIME), or
PST_CM_DSTFMT_TERMINATE (PST_CMW_DSTFMT_TERMINATE).

pSrcColList

Puntero a PST_STRUCT_CM_EXIT_COL_LIST (PST_STRUCT_CM_WEXIT_COL_LIST). Esta estructura describe las columnas de origen.

pDstColList

Puntero a PST_STRUCT_CM_EXIT_COL_LIST (PST_STRUCT_CM_WEXIT_COL_LIST). Esta estructura describe las columnas de destino.

Función de devolución de llamada

Cada vez que un proceso de Optim llama a una rutina de salida de formato de destino, el proceso pasa la dirección de la siguiente función de devolución de llamada:

pPSTGetColValue()

Recupera datos para todas las columnas de origen y la mayoría de las columnas de destino en la fila de datos actual. En general, no es necesario que la rutina de salida llame a esta función porque los datos para la columna de origen antigua se proporcionan en el primer parámetro. Sin embargo, la rutina de salida puede llamar a esta función para recuperar los datos para diferentes columnas.

Formatos

La fecha de entrada está a la vez en un formato PST_C_TIMESTAMP y en un formato PST_C_SYB_DATETIME. La salida se dirige a transformar esa fecha en uno de los siguientes formatos, en función del tipo de datos de la columna de destino.

PST_C_CHAR_SZ

Columnas de destino CHAR y VARCHAR.

PST_C_INTEGER_CHAR_SZ

Columnas de destino NUMERIC.

PST_C_TIMESTAMP

Columnas DATE/TIME de DB2 y Oracle.

PST_C_SYB_DATETIME

Columnas DATETIME de Sybase ASE

Proceso

El proceso típico para la rutina de salida de formato de destino se resume en los pasos siguientes:

1. En cada llamada de Optim, la rutina de salida comprueba una primera llamada. En la primera llamada, la salida realiza las tareas de inicialización y el proceso normal (paso 2). En las llamadas siguientes, sólo se realiza el proceso normal (paso 2).
2. La salida recibe el valor de la columna de origen como se especifica en la función de antigüedad definida en la correlación de columnas. Optim aplica la función de antigüedad antes de llamar a la salida de la correlación de columnas y almacena el valor antiguo de los campos **InputTimeStamp** y **InputSybaDateTime** en el archivo de cabecera.
3. Si la salida necesita examinar otras columnas para calcular el valor para la columna de destino, la salida debe llamar a la función de devolución de llamada **pPSTGetColValue()** para obtener el valor para dichas columnas.
4. Una vez que se ha generado el valor de destino, la rutina de salida debe formatear el valor y colocarlo en uno de los campos de la unión **OutputValue**. El campo **FuncCode** indica el campo de la unión **OutputValue** donde debe colocarse el valor. La salida debe devolver un código apropiado que indique el campo donde se guardan los datos o que instruya al proceso que se salte la fila o que termine anormalmente.
5. Una vez que se ha procesado la última fila de datos, Optim pasa una llamada de terminación a la rutina de salida, identificada mediante un valor de **PST_CM_DSTFMT_TERMINATE** (**PST_CMW_DSTFMT_TERMINATE**) en el campo **FuncCode**. Esta llamada solicita a la rutina de salida que libere cualquier almacenamiento asignado dinámicamente. Cuando las tareas finales se hayan completado, la rutina de salida pasa un código de retorno a Optim.

Modalidades de terminación anormal

Existen varias maneras en que la rutina de salida de formato de destino puede terminar anormalmente el proceso:

- Procesar filas con fechas omitidas o fechas no válidas. Si selecciona una de estas opciones en una solicitud de proceso, y las columnas de origen y de destino tienen los mismos atributos, la columna de origen se copia sin cambios a la columna de destino. Si no selecciona ninguna de las dos opciones, la fila se rechaza.
- Rechazar la fila. Rechace la fila independientemente de las opciones de proceso para las fechas omitidas o no válidas, en función de las especificaciones de la rutina de salida.
- Terminar anormalmente todo el proceso, en función de las especificaciones de la rutina de salida.

La rutina de salida devuelve un código de retorno que indica si la conversión ha sido satisfactoria o la modalidad de terminación anormal que se debe utilizar.

Durante un proceso, la rutina de salida puede interrogar las columnas de la fila de entrada y algunas de las columnas de la fila de destino. Sin embargo, la rutina de salida no puede interrogar una columna de destino que incluye una rutina de salida y está definida en la correlación de columnas después de la columna de destino actual. Todas las otras columnas de destino están disponibles.

Códigos de retorno

Los siguientes códigos de retorno se aplican a las salidas de formato de destino:

PST_CM_DSTFMT_SUCCESS
(**PST_CMW_DSTFMT_SUCCESS**)

La columna de destino tiene asignado un valor, tal como se especifica en el campo **FuncCode** del primer parámetro que se pasa a la salida.

PST_CM_DSTFMT_SKIP
(**PST_CMW_DSTFMT_SKIP**)

Si no selecciona la opción para **Procesar filas con fechas omitidas**, la fila se rechaza. De lo contrario, los datos de origen se copiarán en el destino, siempre que el origen y el destino sean compatibles. Si no son compatibles, la fila se rechaza.

PST_CM_DSTFMT_COL_INVALID
(PST_CMW_DSTFMT_COL_INVALID)

Si no selecciona la opción para **Procesar filas con fechas no válidas**, la fila se rechaza. De lo contrario, los datos se copiarán en el origen, siempre que el origen y el destino sean compatibles. Si no son compatibles, la fila se rechaza.

PST_CM_DSTFMT_REJECT_ROW
(PST_CMW_DSTFMT_REJECT_ROW)

La columna de destino no puede tener asignado un valor. Rechace (descarte) la fila.

PST_CM_DSTFMT_ABORT_PROCESS
(PST_CMW_DSTFMT_ABORT_PROCESS)

Error muy grave. Termine. Para devolver un mensaje de error, coloque el mensaje en el área de trabajo y establezca el espacio no utilizado en blancos o NULL.

Escribir procedimientos de correlación de columnas con el script de Lua

Una procedimiento de correlación de columnas es un procedimiento que se utiliza para enmascarar o transformar los datos de una columna cuando ejecuta un servicio. Como el nombre indica, debe añadir procedimientos de correlación de columnas a una correlación de columnas. Puede escribir procedimientos de correlación de columnas utilizando el lenguaje de script de Lua.

Para obtener más información sobre el lenguaje de programación de Lua, consulte el sitio web de Lua en <http://www.lua.org>

Funciones de Lua para procedimientos de correlación de columnas

Los procedimientos de correlación de columnas soportan la mayoría de funciones Lua estándar. Los procedimientos de correlación de columnas también soportan funciones específicas de Optim.

Nombres de función estándar para funciones definidas por el usuario

Utilice los nombres de función siguientes en los procedimientos de correlación de columnas de Lua. Cada una de estas funciones se llama automáticamente en el punto indicado. También puede crear funciones con otros nombres de función y llamar a estas funciones desde las funciones estándar.

Nombre	Descripción	Necesaria
cm_load()	Esta función se llama antes de procesar cualesquiera tablas.	No
cm_unload()	Esta función se llama después de procesar todas las tablas.	No
cm_starttable()	Esta función se llama al principio del proceso de cada tabla.	No
cm_endtable()	Esta función se llama al final del proceso para cada tabla.	No
cm_transform()	Esta función se llama para cada fila procesada.	Sí

Cada procedimiento de correlación de columnas debe contener una función `cm_transform()`. Si la función `cm_transform()` no existe, el código se evalúa como una expresión compleja.

Funciones globales

Las funciones globales están disponibles en todos los contextos de ejecución de procedimientos de correlación de columnas:

Nombre	Descripción
<code>numparms()</code>	Obtener el número de parámetros que se pasan al procedimiento de correlación de columnas.
<code>parms.get(<i>n</i>)</code>	Obtener el valor del parámetro en el índice <i>n</i> .
<code>print()</code>	Imprimir mensajes en el informe de procesos.
<code>rejectrow()</code>	Saltar fila e ir a la fila siguiente.

Funciones de almacén de datos

Utilice las funciones siguientes para obtener información sobre los almacenes de datos de origen y destino:

Nombre	Descripción
<code>source.getdbalias()</code>	Obtener el alias de base de datos del almacén de datos de origen.
<code>source.getcreatorid()</code>	Obtener el ID de creador del almacén de datos de origen.
<code>target.getdbalias()</code>	Obtener el alias de base de datos del almacén de datos de destino.
<code>target.getcreatorid()</code>	Establecer el ID del almacén de datos de destino.

Funciones de tabla

Utilice las funciones siguientes para obtener información sobre las tablas de origen y destino:

Nombre	Descripción
<code>source.gettablename()</code>	Obtener el nombre de la tabla de origen.
<code>target.gettablename()</code>	Obtener el nombre de la tabla de destino.

Funciones de columna

Utilice las funciones siguientes para obtener información sobre las columnas de origen y destino, transformar datos de columna, escribir los resultados en la columna de destino y salir:

Nombre	Descripción
<code>source.column.getvalue()</code>	Obtener un valor no numérico de una columna de origen. Se genera un error de tiempo de ejecución si utiliza la función <code>source.column.getvalue()</code> para obtener valores de columnas de tipo de datos <code>PST_SQL_BINARY</code> .

Nombre	Descripción
<code>source.column.getasdouble()</code>	Obtener un valor numérico de una columna de origen (los procedimientos de correlación de datos procesan los datos numéricos en formato de precisión doble). Utilice esta función para las columnas de los tipos de datos siguientes: <code>PST_SQL_DECIMAL</code> , <code>PST_SQL_DOUBLE</code> , <code>PST_SQL_FLOAT</code> , <code>PST_SQL_INFX_DEC_FLOAT</code> y <code>PST_SQL_ORA_NUMBER</code> .
<code>source.column.getlength()</code>	Obtener la longitud de la serie en una columna de origen.
<code>source.column.getname()</code>	Obtener el nombre de la columna de origen.
<code>source.column.gettype()</code>	Obtener el tipo de datos de la columna de origen.
<code>target.column.setvalue()</code>	Establecer el valore de una columna de destino. Se genera un error de tiempo de ejecución si utiliza la función <code>target.column.setvalue()</code> para establecer valores en columnas de tipo de datos <code>PST_SQL_BINARY</code> .
<code>target.column.getlength()</code>	Obtener la longitud de la serie en una columna de destino.
<code>target.column.getname()</code>	Obtener el nombre de la columna de destino.
<code>target.column.gettype()</code>	Obtener el tipo de datos de la columna de destino.
<code>optimmask()</code>	Llamar a un proveedor de Optim Data Privacy Providers (ODPP).
<code>userexit()</code>	Llamar una salida de usuario.

Funciones no soportadas

Las categorías de funciones siguientes no están soportadas en los procedimientos de correlación de columnas.

- Recursos de entrada y salida incorporados en la biblioteca de Lua `io`
- `string.dump()`

Limitaciones de procedimientos de correlación de columnas

Tenga en cuenta las limitaciones siguientes al escribir un procedimiento de correlación de columnas.

Datos numéricos procesados en formato de precisión doble

Los procedimientos de correlación de columnas procesan datos numéricos en formato de precisión doble. Por lo tanto, debe utilizar la función `source.column.getasdouble()` para obtener datos numéricos de columnas de origen.

Codificación

Los procedimientos de correlación de columnas utilizan la codificación UTF-16 en su proceso interno.

Tipos de datos `PST_SQL_BINARY` no soportados

Se genera un error de tiempo de ejecución si utiliza la función `source.column.getvalue()` para obtener valores de columnas de tipo de datos `PST_SQL_BINARY`. También se genera un error de tiempo de ejecución si utiliza la función `target.column.setvalue()` para establecer valores en columnas de tipo de datos `PST_SQL_BINARY`.

Funciones no soportadas

Las categorías de funciones siguientes no están soportadas en los procedimientos de correlación de columnas.

- Recursos de entrada y salida incorporados en la biblioteca de Lua `io`
- `string.dump()`

Ejemplo de procedimiento de correlación de columnas: procedimiento genérico

El ejemplo de procedimiento de correlación de columnas ilustra la estructura de un procedimiento de correlación de columnas con sus funciones estándar: `cm_load`, `cm_unload`, `cm_starttable`, `cm_endtable`, y `cm_transform`.

```
-----
--
-- Procedimiento de correlación de columnas de ejemplo de IBM Optim
--
-- Nombre:      OptimSample
--
-- Revisión:    1.0
--
-- Descripción: Muestra todas las prestaciones de los procedimientos de
--              correlación de columnas de Optim/Lua
--
-- Entrada:     Cero o más parámetros que simplemente se devolverán al informe
--              de proceso de Optim
--
--
-- Salida:      Sección del informe de proceso de Optim que muestra
--              información de este procedimiento de correlación de columnas. --
--
--
-----

-----
-- cm_load function - Llamada antes de procesar las tablas
-----
function cm_load()

    print(" *** Start of Process ***")

    colinfoshown = false

    -- Visualizar parámetros pasados de Correlación de columnas

    print(" Argument Count: " .. string.format("%d", numparms()))
    for i = 1, numparms(), 1 do
        print("   Argument " .. string.format("%d", i-1) .. " " .. parms.get(i-1))
    end

end

-----
-- Función cm_unload - Llamada después de procesar todas las tablas
-----
function cm_unload()

    print(" *** End of Process ***")

end

-----
-- Función cm_starttable - Llamada al inicio del proceso para cada tabla
```

```

-----
function cm_starttable()

    print(" \nStart of processing table")
    fullname = source.getdbalias() .. "." .. source.getcreatorid() .. "." .. source.gettablename()
    print("      Source Table: " .. fullname)
    fullname = target.getdbalias() .. "." .. target.getcreatorid() .. "." .. target.gettablename()
    print("      Target Table: " .. fullname)

end

-----
-- Función cm_endtable - Llamada al final del proceso para cada tabla
-----
function cm_endtable()

    print(" \nEnd of processing table")

end

-----
-- Función cm_transform - Llamada para cada fila procesada
-----
function cm_transform()

    if (not colinfoshown) then
        colinfoshown = true
        print(" Processing column " .. source.column.getname())
        print("      Type: " .. source.column.gettype())
        print("      Length: " .. source.column.getlength())
    end

    -- Esta sentencia obtiene el valor en la columna para el que se llamó cm_transform
    -- Opcionalmente, se puede especificar el nombre de otra columna, por ejemplo:
    -- source.column.getvalue("COL1") devolverá el valor en la columna COL1
    oldvalue = source.column.getvalue()

    -- Este código establece la columna destino en el mismo valor para la columna origen
    -- La lógica para cambiar el valor se colocaría aquí.
    -- Si NO desea insertar esta fila en la tabla de destino, llame a la función rejectrow()
    newvalue = oldvalue
    target.column.setvalue(newvalue)

end

```

Ejemplo de procedimiento de correlación de columnas: búsqueda conmutada

El ejemplo de procedimiento de correlación de columnas ilustra cómo la función mask_parms enmascara una columna utilizando datos de una tabla de búsqueda.

```

-----
--
-- Procedimiento de correlación de columnas de ejemplo de IBM Optim
--
-- Nombre:      OptimSwitchedLookup
--
-- Revisión:    1.0
--
-- Descripción: Enmascara una columna utilizando la búsqueda de tabla. La búsqueda
--              de tabla a utilizar está determinada por el valor de otra columna.
--
-- Entrada:     Parámetro 1 (necesario):
--              Una serie que indica el tipo de búsqueda a utilizar:
--              HASH, RANDOM o LOOKUP
--
--              Parámetro 2 .... n-1 (necesario)

```



```

--      Una expresión que indica la tabla de búsqueda a utilizar. El formato
--      es COND(nombre-columna=valor) DATASOURCE(parámetros_datasource)
--      Este parámetro se puede repetir muchas veces. Si una fila no satisface
--      ninguno de los parámetros de COND, no se insertará en la tabla de
--      destino.
--
--      Parámetro n (opcional):
--      Una serie que contiene parámetros adicionales a copiar en la
--      invocación de optimmask.
--      Esto es además del valor datasource_parameters en la cláusula COND
--      COND del Parámetro 2 y el campo mask_parms_constant que está
--      declarado al inicio de este script.
--
--
--
-- Salida:      - Los datos de columna enmascarada establecidos por la función target.column.setvalue
--              - Texto dirigido al informe de Optim por la función de impresión
--
-- Cód. retorno: 0 - Ejecución satisfactoria
--               1 - Rechazar fila (Utilizar en cm_transform; la fila no se insertará
--                   en la tabla de destino)
--               2 - Terminar proceso anormalmente
--
--      Se especifican dos funciones de ayudante para especificar otras
--      condiciones que no sean el código de retorno 0.
--      No hay necesidad de codificar una sentencia de retorno al utilizar
--      estas funciones.
--      error(string) - Esta llamada hace que el proceso de Optim
--                      termine anormalmente y se muestra una serie
--                      en el informe de Optim como mensaje de error.
--      rejectrow()   - Esta llamada hace que Optim rechace la fila
--                      que se está procesando actualmente. La fila
--                      no se inserta en la tabla de destino.
--
--
--
-----
function cm_transform()

-- Cambie este campo para que contenga los parámetros
-- que se deben colocar en todas las llamadas de optimmask
local mask_parms_constant = 'CACHE=Y,WHENNF=PRE'

--
-- Validar número de parámetros:
--
nparm = numparms()
if (nparm < 2) then
    process_error("Call to column map procedure OptimSwitchedLookup must have 2 or more parameters")
end

--
-- Parámetro de proceso 1 (tipo lookup)
parm = parms.get(0)
if (string.lower(parm) == "hash") then
    provalue = "HASH_LOOKUP"
elseif (string.lower(parm) == "random") then
    provalue = "RANDOM_LOOKUP"
elseif (string.lower(parm) == "lookup") then
    provalue = "LOOKUP"
else
    process_error("Invalid parameter. Expected HASH, RANDOM or LOOKUP. Found " .. parm)
end

--

```

```

-- Procesar parámetros de COND/DATASOURCE
--
gotcond = false
for parmptr = 1, nparm-1 do
    parm = parms.get(parmptr)
    if (string.lower(string.sub(parm, 1, 5)) == "cond(") then
        gotcond = true
        datasource_parameters = process_cond()
        if (datasource_parameters > "") then
            break;
        end
    end
end
if (not gotcond) then
    process_error("No COND parameter found")
end

-- Ninguna COND coincide con esta fila, así que se rechaza la fila
if (datasource_parameters <= "") then
    rejectrow()
end

--
-- Procesar parámetro optimmask adicional opcional
--
lastparm = parms.get(nparm-1)
if (string.lower(string.sub(lastparm, 1, 5)) == "cond(") then
    optimmask_additional_parms = ""
else
    optimmask_additional_parms = lastparm
end

--
-- Construir llamada a optimmask, hacer la llamada,
-- y colocar el valor nuevo en la columna de destino
--
mask_parms = "PRO=" .. provalue .. "," .. mask_parms_constant

-- Este uso de variables de entorno para almacenar el ID de usuario y la
-- contraseña para la tabla del sistema sólo a efectos ilustrativos.
-- Para una mayor seguridad, almacene esta información en variables de
-- entorno en un formato cifrado.
userid = os.getenv("optimmaskuserid")
if (userid) then
    mask_parms = mask_parms .. ",USER=" .. userid
end
password = os.getenv("optimmaskpswd")
if (password) then
    mask_parms = mask_parms .. ",PASS=" .. password
end

mask_parms = mask_parms .. "," .. optimmask_additional_parms

oldvalue = source.column.getvalue()
newvalue = optimmask(oldvalue, mask_parms)
target.column.setvalue(newvalue)
end

function process_cond()

    strptr = 6    -- Señalar el primer carácter después de "COND("

    -- Obtener el nombre de columna
    equalsign = string.find(parm, "=", strptr, true)
    if (not equalsign) then

```

```

    process_error("Syntax error around character " .. strptr .. " in expression: " .. parm)
end
colname = string.sub(parm, strptr, equalsign-1)

-- Obtener el valor de columna
strptr = equalsign + 1
closeparen = string.find(parm, ")", strptr, true)
if (not closeparen) then
    process_error("COND expression is missing closing parenthesis in expression: " .. parm)
end
colvalue = string.sub(parm, strptr, closeparen-1)

--Depurar - Imprimir resultados de exploración:
--print ("Found colname=" .. colname .. " in parm: " .. parm)
--print ("Found colvalue=" .. colvalue .. " in parm: " .. parm)

-- Si COND(nombre-columna=valor) no se cumple para esta fila, no seguir
-- procesando este parámetro. Devolver "" para indicar sin coincidencia.
if (source.column.getvalue(colname) ~= colvalue) then
    return ""
end

-- Coincidencia obtenida en COND, así que obtener el valor de DATASOURCE
strptr = closeparen + 1
datasourceValuePtr = string.find(parm, "DATASOURCE(", strptr, true)
if (not datasourceValuePtr) then
    process_error("DATASOURCE clause not found in expression: " .. parm)
end
-- (Nota: esta exploración es muy simple; presupone que no hay
-- texto errante entre las cláusulas COND y DATASOURCE)
closeparen = string.find(parm, ")", datasourceValuePtr, true)
if (not closeparen) then
    process_error("COND expression is missing closing parenthesis in expression: " .. parm)
end
return string.sub(parm, datasourceValuePtr+11, closeparen-1)
end
function process_error(msg)
    errprefix = "Error in column map procedure for column " .. source.column.getname()
    errprefix = errprefix .. " in table " .. source.getdbalias() .. "."
    .. source.getcreatorid() .. "." .. source.gettablename()
    error(errprefix .. ":\n " .. msg)
end

```

Ejemplo de procedimiento de correlación de columnas: enmascaramiento de ID nacional

El ejemplo de procedimiento de correlación de columnas ilustra cómo la función `mask_parms` enmascara una columna que contiene un ID de nacional.

```

-----
--
-- Procedimiento de correlación de columnas de ejemplo de IBM Optim
--
-- Nombre:      OptimNID
--
-- Revisión:    1.0
--
-- Descripción:  Enmascara una columna de ID nacional basada en el valor de otra
--               columna o un argumento de script con el código de país.
--
-- Entrada:     Parámetro 1 (necesario):
--               Un código de país de (p. ej. US para Estados Unidos)
--               --0--
--               COL(nombre-columna) donde nombre-columna es el nombre de una
--               columna que contiene el código de país
--

```

```
--      Parámetro 2 (Opcional):
--      Una serie que contiene parámetros adicionales a copiar en la
--      invocación de optimmask
--      Esto es además del campo mask_parms_constant declarado al inicio de
--      este script.
--
--
--
-- Salida:      - Los datos de columna enmascarada establecidos por la función target.column.setvalue
--              - Texto dirigido al informe de Optim por la función de visualización
--
-- Cód. retorno: 0 - Ejecución satisfactoria
--               1 - Rechazar fila (Utilizar en cm_transform; la fila no se insertará
--                   en la tabla de destino)
--               2 - Terminar proceso anormalmente
--
-- Se especifican dos funciones de ayudante para especificar otras
-- condiciones que no sean el código de retorno 0.
-- No hay necesidad de codificar una sentencia de retorno al utilizar
-- estas funciones.
-- error(string) - Esta llamada que hará que el proceso de Optim
--                 termine anormalmente y se mostrará una serie
--                 en el informe de Optim como mensaje de error.
-- rejectrow()   - Esta llamada hará que Optim rechace la fila
--                 procesada actualmente. La fila no se
--                 insertará en la tabla de destino.-
--
--
--
-----
```

```
function cm_transform()

-- Cambie este campo para que contenga los parámetros
-- que se deben colocar en todas las llamadas de optimmask
local mask_parms_constant = 'MTD=REPEATABLE'

--
-- Obtener parámetros:
--
nparm = numparms()
if nparm ~= 1 and nparm ~= 2 then
    msg = "Call to column map procedure for column "
    msg = msg .. source.column.getname()
    msg = msg .. " must have 1 or 2 parameters"
    error(msg)
end
parm1 = parms.get(0)
if (nparm == 2) then
    parm2 = parms.get(1)
end

--
-- Obtener código de país en el campo swivalue basado en parámetros
--
if string.lower(string.sub(parm1, 1, 4)) == 'col(' then
    closeparen = string.find(parm1, ')', 5, true)
    if (not closeparen) then
        error("No closing parenthesis found in expression: " .. parm1)
    end
    colname = string.sub(parm1, 5, closeparen-1)
    swivalue = source.column.getvalue(colname)
else
    swivalue = parm1
end

--
-- Construir llamada a optimmask, hacer la llamada,
```

```

        -- y colocar el valor nuevo en la columna de destino
    --
    mask_parms = 'PRO=NID, SWI=' .. swivalue .. ', ' .. mask_parms_constant
    if (parm2) then
        mask_parms = mask_parms .. ', ' .. parm2
    end
    oldvalue = source.column.getvalue()
    newvalue = optimmask(oldvalue, mask_parms)
    target.column.setvalue(newvalue)
end

```

Capítulo 6. Gestión de datos

Utilice los programas de utilidad de Optim Designer para editar, examinar y comparar datos. También puede crear tablas.

Examinar datos

Utilice la utilidad Examinar para revisar el contenido de un almacén de datos de archivo sin tener que restaurarlo a la base de datos.

Utilice el recurso de ayuda de la utilidad Examinar para obtener más información sobre un tema o función en concreto. Puede abrir una ventana de ayuda seleccionando **Ayuda > Contenido**, pulsando con el botón derecho del ratón en un elemento y seleccionando **¿Qué es esto?** o pulsando F1.

Para examinar datos:

1. En Optim Designer, pulse **Herramientas > Examinar**. Se abre la ventana Examinar.
2. Pulse **Archivo > Abrir** para abrir la ventana Abrir y seleccione un archivo de datos creado por un servicio. Los archivos se almacenan en el directorio de datos especificado en la utilidad de configuración.
3. Pulse **Abrir**. Las tablas del archivo de datos se listan en la ventana Examinar.
4. Pulse **Herramientas > Crear todos los objetos seleccionados**. La tabla se abre en la ventana Examinar datos.

Edición de datos

Utilice el Editor de tablas para examinar y editar conjuntos de datos relacionamente intactos en las tablas de base de datos. El editor maneja un modelo de datos arbitrariamente complejo formado por cualquier número de tablas y relaciones y que asegura un conjunto relacionamente intacto de datos.

Utilice el recurso de ayuda del Editor de tablas para obtener más información sobre un tema o función en concreto. Puede abrir una ventana de ayuda seleccionando **Ayuda > Contenido**, pulsando con el botón derecho del ratón en un elemento y seleccionando **¿Qué es esto?** o pulsando F1.

Para editar datos:

1. En Optim Designer, pulse **Herramientas > Editar**. Se abre la ventana Opciones de edición.
2. Especifique un **Nombre de tabla** o **Definición de acceso nombrada**, seleccione las opciones **Visualización inicial** o **Modalidad** y pulse **Aceptar**. Se abre la ventana Editor de tablas.
3. Edite una o más tablas al mismo tiempo.

Confirme los cambios en la base de datos moviendo el puntero a una fila distinta o utilizando las opciones de menú. Cada instancia de una confirmación cuenta como un nivel de deshacer. Un nivel de deshacer se define como un cambio en una fila que se confirma en la base de datos.

Si se produce una condición de error cuando se intentan confirmar los datos en la base de datos, los datos no se confirman, pero el intento todavía cuenta como un nivel de deshacer. El editor le permite restaurar los datos que modifique en un punto de confirmación específico.

Un conjunto de recuperación es el conjunto de filas que el editor lee de una sola tabla de la base de datos.

- Para deshacer los cambios y restaurar las filas a las versiones originales obtenidas en el conjunto de recuperación actual para todas las tablas, seleccione **Herramientas > Deshacer**.
- Para deshacer un cambio para una fila, pulse con el botón derecho en la fila y pulse **Deshacer**.

- Para visualizar cada versión sucesiva de la fila confirmada en el conjunto de recuperación actual, y la versión original de la fila en el conjunto de recuperación, pulse con el botón derecho en la fila y pulse **Deshacer....**

Comparación de datos

Utilice la utilidad de comparación para comparar datos de un conjunto de tablas de origen con los datos de otra. Puede definir una solicitud de comparación que está almacenada en el repositorio de Optim y guardar los resultados de una comparación en un archivo de comparación.

Definición de una solicitud de comparación

Utilice el Editor de solicitudes de comparación para definir orígenes de datos para comparar y procesar opciones.

Utilice el recurso de ayuda de la utilidad de comparación para obtener más información sobre un tema o función en concreto. Puede abrir una ventana de ayuda seleccionando **Ayuda > Contenido**, pulsando con el botón derecho del ratón en un elemento y seleccionando **¿Qué es esto?** o pulsando F1.

Para definir una solicitud de comparación:

1. En el Explorador de repositorios, pulse con el botón derecho en **Comparar** y pulse **Nueva comparación**. Se abre el Editor de solicitudes de comparación.
2. En la pestaña **General**, lleve a cabo los pasos siguientes:
 - a. Especifique un nombre de archivo de comparación en el campo **Archivo de comparación**.
 - b. Seleccione **Examinar resultados inmediatamente** para ver los resultados del proceso de comparación después de que haya finalizado el proceso.
 - c. Seleccione una modalidad de **Comparación**.

Tabla única

Archivo de origen – Archivo de origen

Compare los datos de una tabla única de un archivo de origen con los datos de otro archivo de origen.

Archivo de origen – Tabla de base de datos

Compare los datos de una tabla única de un archivo de origen con los datos de una tabla única en la base de datos.

Tabla de base de datos – Tabla de base de datos

Compare los datos de una tabla única en una base de datos con los datos de una tabla única en otra base de datos.

Múltiples tablas

Archivo de origen – Archivo de origen

Compare los datos de tablas de un archivo de origen con los datos de otro archivo de origen.

Archivo de origen – Definición de acceso

Compare los datos de tablas de un archivo de origen con los datos especificados en una definición de acceso.

Archivo de origen – Todas las tablas de base de datos

Compare los datos de tablas de un archivo de origen con los datos de una base de datos.

Nota: Esta selección es de gran utilidad cuando se prueba una aplicación de base de datos. El archivo de origen representa la imagen “anterior” en comparación con las tablas de base de datos, que representan la imagen “posterior”.

Definición de acceso – Definición de acceso

Compare los datos especificados en una definición de acceso con los datos especificados en otra definición de acceso.

Definición de acceso – Todas las tablas de base de datos

Compare los datos especificados en una definición de acceso con los datos de una base de datos.

3. En la pestaña **Origen**, seleccione los orígenes de datos que desea comparar en función de la modalidad de comparación.
4. Pulse **Archivo > Ejecutar** para empezar el proceso de comparación.
5. Pulse **Archivo > Guardar** para guardar la solicitud de comparación. Especifique un nombre de solicitud en el formato IDENTIFICADOR.NOMBRE.

El proceso creará un archivo de comparación en el directorio de datos especificado en la utilidad de configuración. Utilice la utilidad Examinar para ver los resultados de la comparación.

Edición de una solicitud de comparación

Utilice el Editor de solicitudes de comparación para editar una solicitud de comparación.

Utilice el recurso de ayuda de la utilidad de comparación para obtener más información sobre un tema o función en concreto. Puede abrir una ventana de ayuda seleccionando **Ayuda > Contenido**, pulsando con el botón derecho del ratón en un elemento y seleccionando **¿Qué es esto?** o pulsando F1.

Para editar una solicitud de comparación:

1. En el Explorador de repositorios, expanda el nodo **Comparar**, pulse con el botón derecho del ratón en la solicitud que se debe editar y pulse **Editar**. Se abre el Editor de solicitudes de comparación.
2. Edite la solicitud.
3. Pulse **Archivo > Guardar** para guardar la solicitud de comparación.

Ejecución de un proceso de comparación

Puede ejecutar un proceso de comparación para generar un archivo de comparación que puede ver con la utilidad Examinar.

Para ejecutar una solicitud de comparación:

Ejecute la solicitud ejecutando uno de los métodos siguientes:

Para ejecutar el proceso de comparación desde el Explorador de repositorios, pulse con el botón derecho y pulse **Ejecutar**.

Para ejecutar el proceso de comparación desde el Editor de solicitudes de comparación, pulse **Archivo > Ejecutar**.

El proceso creará un archivo de comparación en el directorio de datos especificado en la utilidad de configuración. Utilice la utilidad Examinar para ver los resultados de la comparación.

Creación de tablas

Utilice el programa de utilidad de creación para crear objetos en función de las definiciones de un archivo de datos de origen.

Utilice el recurso de ayuda de la utilidad de creación para obtener más información sobre un tema o función en concreto. Puede abrir una ventana de ayuda seleccionando **Ayuda > Contenido**, pulsando con el botón derecho del ratón en un elemento y seleccionando **¿Qué es esto?** o pulsando F1.

Para crear tablas:

1. En Optim Designer, pulse **Herramientas > Crear** Se abre la ventana Crear opciones.
2. Seleccione un archivo de datos de origen creado por un servicio. Los archivos se almacenan en el directorio de datos especificado en la utilidad de configuración.
3. Seleccione una de las **Opciones de correlación de tablas**.
Si elige una correlación de tablas local, se abrirá el Editor de correlaciones de tablas local. Lleve a cabo los pasos siguientes:
 - a. En el campo **Cualificador**, debe especificar un alias de almacén de datos de destino y el ID de creador en el formato `alias_almacén_datos.id_creador`.
 - b. Modifique los nombres de la tabla de destino.
 - c. Pulse **Archivo > Actualizar y regresar**.Si selecciona una correlación de tablas con nombre, realice el paso siguiente:
 - a. En el campo **Nombre de correlación de tablas**, especifique o seleccione un nombre de correlación de tablas.
4. Pulse **Aceptar**. Se abre la ventana Crear.
5. Edite y seleccione los objetos que desea crear.
6. Pulse **Herramientas > Crear todos los objetos seleccionados** para empezar el proceso de creación.

Capítulo 7. Gestión de datos con Optim Designer

Optim Designer proporciona una interfaz de diseño única para la privacidad de datos de Optim y las soluciones de gestión de datos de prueba.

Extracción de datos con Optim Designer

Esta guía de aprendizaje describe cómo utilizar Optim Designer para crear un servicio de extracción de Optim. En esta guía de aprendizaje, especificará una definición de acceso y un servicio de extracción.

Después de completar esta guía de aprendizaje, podrá ejecutar un servicio de extracción.

Objetivos de aprendizaje

Al finalizar los ejercicios, habrá aprendido a realizar las tareas siguientes:

- Crear una carpeta Explorador de repositorios
- Crear una definición de acceso para definir los datos a extraer
- Editar una definición de acceso para definir los criterios de elección
- Crear un servicio de extracción
- Probar un servicio de extracción

Tiempo necesario

Este módulo debería tardar aproximadamente 60 minutos en completarse.

Requisitos previos

Esta guía de aprendizaje requiere un entorno de Optim Designer que incluye una conexión con un repositorio de Optim y un alias de almacén de datos que contiene los datos de ejemplo de Optim.

Esta guía de aprendizaje se puede completar en el entorno Optim Designer.

Creación de una carpeta Explorador de repositorios

En este ejercicio, se creará una carpeta Explorador de repositorios. Una carpeta Explorador de repositorios contiene servicios, definiciones de acceso, correlaciones de columnas y correlaciones de tablas. Utilice las carpetas para organizar estos objetos.

Para crear un plan de acceso de datos y una política de selección:

1. Con el botón derecho del ratón, pulse la vista Explorador de repositorios y pulse **Nueva > Carpeta**. Se abre la ventana Nueva carpeta.
2. En **Nombre**, escriba Guía de aprendizaje y pulse **Aceptar**.

Ha creado una carpeta Explorador de repositorios denominada Guía de aprendizaje.

Creación de una definición de acceso

En este ejercicio, se creará una definición de acceso. Utilice las definiciones de acceso para especificar las tablas, el cruce de relaciones y los criterios de selección para los datos que desea procesar.

Antes de crear una definición de acceso, debe existir un alias de almacén de datos para la base de datos que contiene la tabla inicial.

Para crear una definición de acceso:

1. Expanda la carpeta Guía de aprendizaje en el Explorador de repositorios, pulse con el botón derecho en **Definiciones de acceso** y pulse **Nueva definición de acceso**. Se abre el asistente Nueva definición de acceso.
2. En la página Especificar nombre de definición de acceso, especifique **SAMPLE.AD**. Pulse **Siguiente**.
3. En la página Seleccionar un alias de almacén de datos, seleccione el alias de almacén de datos que contiene los datos de ejemplo de Optim. Pulse **Siguiente**.
4. En la página Tabla inicial, realice los pasos siguientes:
 - a. En el campo **Tabla inicial**, especifique el siguiente patrón de búsqueda: **SCHEMA.OPTIM_CUSTOMERS**, donde **SCHEMA** es el esquema que contiene los datos de ejemplo de Optim. Por ejemplo: **SAMPLE.OPTIM_CUSTOMERS**.
 - b. Pulse **Buscar tablas**. La lista de tablas muestra las tablas que coinciden con el patrón.
 - c. Seleccione la tabla **OPTIM_CUSTOMERS**.
 - d. Pulse **Siguiente**.
5. En la página Especificar un método de selección de tabla, pulse **Buscar tablas relacionadas**. Pulse **Siguiente**.
6. En la página Especificar un patrón de tabla y seleccionar tablas relacionadas, realice los pasos siguientes:
 - a. Pulse **Buscar tablas**. La lista de tablas muestra las tablas relacionadas con **OPTIM_CUSTOMERS**.
 - b. Pulse **Seleccionar todo**.
 - c. Pulse **Finalizar**.

La nueva definición de acceso se visualiza en el Editor de definiciones de acceso.

Definición de criterios de selección

En este ejercicio, se definirán los criterios de selección en la definición de acceso. Utilice criterios de selección para centrarse en un conjunto específico de datos relacionados definiendo una cláusula WHERE de SQL y utilizando variables de sustitución con los valores predeterminados.

Los criterios de selección deben estar conformes con la sintaxis SQL e incluir operadores lógicos o relacionales. Los operadores lógicos y la sintaxis varían entre los distintos DBMS. Consulte la documentación apropiada del DBMS para obtener información.

Para seleccionar el conjunto deseado de datos de una tabla, quizás necesite una combinación de operadores lógicos AND y OR.

Para definir criterios de selección:

1. Expanda la carpeta Guía de aprendizaje en el Explorador de repositorios, expanda el nodo **Definiciones de acceso** y efectúe una doble pulsación en la definición de acceso **SAMPLE.AD**. Se abre el Editor de definiciones de acceso.
2. Seleccione la ficha **Tablas**.
3. Seleccione la tabla **OPTIM_CUSTOMERS** en la lista de tablas.
4. Pulse **Editar criterios de selección**. Se abre la ventana Editar criterios de selección que muestra una cláusula WHERE para la tabla.
5. Especifique los criterios siguientes: **COUNTRY_CODE='US'**.
Pulse **Comprobar sintaxis** para verificar la sintaxis e identificar errores.
6. Pulse **Aceptar** para volver al Editor de definiciones de acceso.
Si la sintaxis no es válida, se abrirá una solicitud para identificar el error. No se pueden guardar los criterios si contienen errores.
7. En el menú principal, pulse **Archivo > Guardar** para guardar la definición de acceso.

Se han definido criterios de selección que sólo seleccionarán filas de la tabla OPTIM_CUSTOMERS en la que el valor de la columna COUNTRY_CODE es 'US'.

Creación de un servicio de extracción

En este ejercicio, se creará un servicio de extracción basado en los datos especificados en una definición de acceso. Utilice un servicio de extracción para copiar un conjunto de filas relacionadas de una o más tablas y guardar las filas en un almacén de datos de archivo.

Para crear un servicio de extracción:

1. Expanda la carpeta Guía de aprendizaje en el Explorador de repositorios, pulse con el botón derecho en **Servicios** y pulse **Servicio nuevo**. Se abre el asistente Nuevo servicio.
2. En la página Escribir un nombre de servicio y seleccionar un servicio, lleve a cabo los pasos siguientes:
 - a. En el campo **Nombre**, especifique SAMPLE.EXTRACT.
 - b. En la lista de tipos de servicio, seleccione **Extraer**.
 - c. Pulse **Siguiente**.
3. En la página Seleccionar una definición de acceso, seleccione SAMPLE.AD. Pulse **Siguiente**.
4. En la página Especificar un nombre de almacén de datos de archivo de destino, especifique SAMPLE_EXT en el campo **Nombre de almacén de datos de archivo de destino**.
5. Pulse **Finalizar**.

El nuevo servicio de extracción se visualiza en el Editor de servicios de extracción. Puede utilizar este servicio para crear un almacén de datos de archivo para un servicio de conversión, inserción o carga. También puede utilizar el almacén de datos de archivo para crear una correlación de columnas o una correlación de tablas.

Enmascaramiento de datos con Optim Designer

Esta guía de aprendizaje describe cómo utilizar Optim Designer para crear un servicio de conversión que transforme los datos de un almacén de datos de archivo.

Esta guía de aprendizaje requiere el almacén de datos de archivo SAMPLE_EXT creado en la guía de aprendizaje “Extracción de datos con Optim Designer” en la página 125.

Después de completar esta guía de aprendizaje, podrá aplicar una función de privacidad y ejecutar un servicio de conversión.

Objetivos de aprendizaje

Al finalizar los ejercicios, habrá aprendido a realizar las tareas siguientes:

- Crear una carpeta Explorador de repositorios
- Crear una definición de acceso para definir los datos a extraer
- Editar una definición de acceso para definir los criterios de elección
- Crear un servicio de extracción
- Probar un servicio de extracción

Tiempo necesario

Este módulo debería tardar aproximadamente 60 minutos en completarse.

Requisitos previos

Esta guía de aprendizaje requiere un entorno de Optim Designer que incluye una conexión con un repositorio de Optim y un alias de almacén de datos que contiene los datos de ejemplo de Optim.

Esta guía de aprendizaje se puede completar en el entorno Optim Designer.

Creación de una correlación de tablas

En este ejercicio, se creará una correlación de tablas. Utilice una correlación de tablas para definir especificaciones para correlacionar tablas de origen y destino de datos compatibles. Puede correlacionar tablas que tienen diferentes nombres, modificar nombres de tablas, excluir tablas de un proceso o incluir correlaciones de columnas para un mayor control sobre los datos.

Para crear una correlación de tablas:

1. Expanda la carpeta Guía de aprendizaje en el Explorador de repositorios, pulse con el botón derecho en **Correlación de tablas** y pulse **Nueva correlación de tablas**. Se abre el asistente Nueva correlación de tablas.
2. En la página Especificar nombre de correlación de tablas, especifique SAMPLE.TMAP en el campo **Nombre**. Pulse **Siguiente**.
3. En la página Seleccionar un alias de almacén de datos, seleccione SAMPLE.EXT.
4. Pulse **Finalizar**.

La nueva correlación de tablas se visualiza en el Editor de correlaciones de tablas.

Debe utilizar el editor para definir los almacenes de datos de destino y los esquemas antes de guardar la correlación de tablas.

Edición de una correlación de tablas

En este ejercicio utilizará el Editor de correlaciones de tablas para definir almacenes de datos de destino y esquemas para tablas de origen en una correlación de tablas.

Para editar los datos de destino de una correlación de tablas:

1. Expanda la carpeta Guía de aprendizaje en el Explorador de repositorios, expanda el nodo **Correlaciones de tablas** y efectúe una doble pulsación en la correlación de tablas SAMPLE.TMAP. Se abre el Editor de correlaciones de tablas.
2. Seleccione la ficha **Correlación de tablas**.
3. En el área **Alias del almacén de datos y correlación de esquemas**, seleccione la fila que contiene el almacén de datos de origen.
4. Pulse la casilla **Alias de almacén de datos de destino** y seleccione el alias de almacén de datos que contiene los datos de ejemplo de Optim.
5. Pulse la casilla **Esquema de destino** y seleccione el esquema que contiene los datos de ejemplo de Optim.
6. En el menú principal, pulse **Archivo > Guardar** para guardar la correlación de tablas.

La correlación de tablas utiliza las mismas tablas de origen y de destino para enmascarar datos de origen y mantener el mismo esquema y alias de almacén de datos.

Creación de una correlación de columnas

En este ejercicio, se creará una correlación de columnas. Una correlación de columnas proporciona las especificaciones necesarias para encontrar coincidencias con columnas o para excluir columnas del proceso en un servicio de gestión de datos. Los servicios de conversión, inserción y carga deben hacer

referencia a una correlación de tablas, que puede hacer referencia a una o más correlaciones de columnas. Puede utilizar una correlación de columnas para definir transformaciones de datos con funciones o procedimientos de correlación de columnas.

Para crear una correlación de columnas:

1. Expanda la carpeta Guía de aprendizaje en el Explorador de repositorios, pulse con el botón derecho en **Correlación de columnas** y pulse **Nueva correlación de columnas**. Se abre el asistente Nueva correlación de columnas.
2. En la página Especificar nombre de correlación de columnas, especifique `SAMPLE.CMAP` en el campo Nombre. Pulse **Siguiente**.
3. En la página Seleccionar un alias de almacén de datos de archivo de origen, seleccione `SAMPLE_EXT`. Pulse **Siguiente**.
4. En la página Seleccionar una tabla de origen, lleve a cabo los pasos siguientes:
 - a. En el campo **Patrón de búsqueda de tabla**, especifique el patrón de búsqueda siguiente: `DATA_STORE_ALIAS.SCHEMA.OPTIM_CUSTOMERS`, donde `DATA_STORE_ALIAS.SCHEMA` son el alias de almacén de datos y el esquema que contienen los datos de ejemplo de Optim. Por ejemplo: `OPTIM.SAMPLE.OPTIM_CUSTOMERS`.
 - b. Pulse **Buscar tablas**. La lista de tablas muestra las tablas que coinciden con el patrón.
 - c. Seleccione la tabla `OPTIM_CUSTOMERS`.
 - d. Pulse **Siguiente**.
5. En la página Seleccionar un alias de almacén de datos de destino, seleccione el alias de almacén de datos que contiene los datos de ejemplo de Optim. Pulse **Siguiente**.
6. En la página Seleccionar una tabla de destino, realice los pasos siguientes:
 - a. En el campo **Patrón de búsqueda de tabla**, escriba el siguiente patrón de búsqueda: `SCHEMA.OPTIM_CUSTOMERS`, donde `SCHEMA` es el esquema que contiene los datos de ejemplo de Optim. Por ejemplo: `SAMPLE.OPTIM_CUSTOMERS`.
 - b. Pulse **Buscar tablas**. La lista de tablas muestra las tablas que coinciden con el patrón.
 - c. Seleccione la tabla `OPTIM_CUSTOMERS`.
7. Pulse **Finalizar**.

La nueva correlación de columnas se visualiza en el Editor de correlaciones de columna.

Aplicación de una función de enmascaramiento de datos

En este ejercicio, se aplicará una función de enmascaramiento de datos a una columna de una correlación de columnas. Puede enmascarar datos como, por ejemplo, números de ID nacionales, número de tarjetas de crédito, fechas, valores numéricos e información personal.

Para aplicar una función de enmascaramiento de datos.

1. Expanda la carpeta Guía de aprendizaje en el Explorador de repositorios, expanda el nodo **Correlaciones de columnas** y efectúe una doble pulsación en la correlación de columnas `SAMPLE.CMAP`. Se abre el Editor de correlaciones de columnas.
2. Seleccione la columna `PHONE_NUMBER`.
3. Pulse **Aplicar función**. Se abre la ventana Aplicar función.
4. Seleccione la función **Reorganizar**. Pulse **Aceptar**. El nombre de la función aparece en la columna **Columna de origen** y el editor de funciones se abre en el Editor de correlaciones de columnas.
5. En el campo **Expresión de correlación de columnas** del editor de funciones, especifique `SHUFFLE(RETRY=12)`.
6. En el menú principal, pulse **Archivo > Guardar** para guardar la correlación de columnas.

Ha aplicado la función de reorganización a la columna PHONE_NUMBER. La función enmascarará los datos sustituyendo valores de la columna por otros valores de la columna. La función buscará hasta doce veces un valor de sustitución que no coincida con el valor de origen.

Adición de una correlación de columnas a una correlación de tablas

En este ejercicio, se añadirá una correlación de columnas a una correlación de tablas. Se requiere una correlación de tablas para un servicio de conversión. Utilice la correlación de columnas asociada para llevar a cabo la función de enmascaramiento de datos definida en la correlación de columnas.

Para añadir una correlación de columnas a una correlación de tablas:

1. Expanda la carpeta Guía de aprendizaje en el Explorador de repositorios, expanda el nodo **Correlaciones de tablas** y efectúe una doble pulsación en la correlación de tablas SAMPLE.TMAP. Se abre el Editor de correlaciones de tablas.
2. Seleccione la ficha **Correlación de tablas**.
3. En el área **Correlación de tablas**, seleccione la tabla OPTIM_CUSTOMERS.
4. Pulse **Añadir correlación de columnas**. Se abrirá la ventana Nueva correlación de columnas con una lista de correlaciones de columnas que contienen las tablas seleccionadas.
5. Seleccione la correlación de columnas SAMPLE.CMAP. Pulse **Aceptar**.
6. En el menú principal, pulse **Archivo > Guardar** para guardar la correlación de tablas.

Ha añadido la correlación de columnas SAMPLE.CMAP (y su función de enmascaramiento de datos asociada) a la correlación de tablas SAMPLE.TMAP, que permitirá a un servicio de conversión transformar los datos.

Creación de un servicio de conversión

En este ejercicio, se creará un servicio de conversión para enmascarar datos.

Para crear un servicio de conversión:

1. Expanda la carpeta Guía de aprendizaje en el Explorador de repositorios, pulse con el botón derecho en **Servicios** y pulse **Servicio nuevo**. Se abre el asistente Nuevo servicio.
2. En la página Escribir un nombre de servicio y seleccionar un servicio, lleve a cabo los pasos siguientes:
 - a. En el campo **Nombre**, especifique SAMPLE.CONVERT.
 - b. En la lista de tipos de servicio, seleccione **Convertir**.
 - c. Pulse **Siguiente**.
3. En la página Seleccionar una correlación de tablas, seleccione la correlación de tablas SAMPLE.TMAP. Pulse **Siguiente**.
4. En la página Especificar propiedades del destino, especifique SAMPLE_CONV en el campo **Nombre de almacén de datos de archivo de destino**.
5. Pulse **Finalizar**.

El nuevo servicio de conversión se visualiza en el Editor de servicios de conversión. Puede utilizar este servicio para enmascarar la columna PHONE_NUMBER en la tabla OPTIM_CUSTOMERS.

Avisos

Esta información se ha desarrollado para productos y servicios ofrecidos en España.

Es posible que IBM no ofrezca en otros países los productos, servicios o características que se describen en este documento. Póngase en contacto con el representante local de IBM para obtener información sobre los productos y servicios disponibles actualmente en su área. Las referencias a programas, productos o servicios de IBM no pretenden establecer ni implicar que sólo puedan utilizarse dichos productos, programas o servicios de IBM. En su lugar, se puede utilizar cualquier producto, programa o servicio funcionalmente equivalente que no infrinja ninguno de los derechos de propiedad intelectual de IBM. Sin embargo, es responsabilidad del cliente evaluar y verificar el funcionamiento de cualquier producto, programa o servicio que no sea de IBM.

IBM puede tener patentes o solicitudes de patentes pendientes que cubran el tema principal descrito en este documento. La entrega de este documento no le otorga ninguna licencia sobre dichas patentes. Puede enviar consultas sobre licencias, por escrito, a:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
EE.UU.

Para consultas sobre licencias relacionadas con información de doble byte (DBCS), póngase en contacto con el departamento de propiedad intelectual de IBM de su país o envíe sus consultas, por escrito, a:

IBM World Trade Asia Corporation
Licensing 2-31 Roppongi 3-chome, Minato-ku
Tokio 106-0032, Japón

El párrafo siguiente no se aplica al Reino Unido ni a ningún otro país donde estas disposiciones sean incompatibles con la legislación vigente: INTERNATIONAL BUSINESS MACHINES CORPORATION PROPORCIONA ESTA PUBLICACIÓN "TAL CUAL" SIN GARANTÍAS DE NINGÚN TIPO, NI EXPLÍCITAS NI IMPLÍCITAS, INCLUYENDO PERO NO LIMITÁNDOSE A ELLAS, LAS GARANTÍAS IMPLÍCITAS DE NO VULNERACIÓN, COMERCIALIZACIÓN O IDONEIDAD PARA UN FIN CONCRETO. Algunos países no permiten la renuncia a garantías explícitas o implícitas en determinadas transacciones, por lo que puede que esta declaración no sea aplicable en su caso.

Esta información puede incluir imprecisiones técnicas o errores tipográficos. Periódicamente se efectúan cambios en la información aquí contenida; estos cambios se incorporarán en nuevas ediciones de la publicación. IBM puede realizar en cualquier momento mejoras o cambios en los productos o programas descritos en esta publicación sin previo aviso.

Cualquier referencia incluida en esta información a sitios Web que no sean de IBM sólo se proporciona para su comodidad y en ningún modo constituye una aprobación de dichos sitios Web. Los materiales de dichos sitios Web no forman parte de los materiales para este producto de IBM y el uso de dichos sitios Web corre a cuenta y riesgo del Cliente.

IBM puede utilizar o distribuir cualquier información que se le proporcione en la forma que considere adecuada, sin incurrir por ello en ninguna obligación para con el remitente.

Los titulares de licencias de este programa que deseen obtener información sobre el mismo con el fin de permitir: (i) el intercambio de información entre programas creados independientemente y otros programas (incluido éste) y el uso mutuo de información que se haya intercambiado, deben ponerse en contacto con:

IBM Corporation
Software Interoperability Coordinator, Department 49XA
3605 Highway 52 N
Rochester, MN 55901
EE.UU.

Dicha información puede estar disponible, sujeta a los términos y condiciones correspondientes, incluyendo, en algunos casos, el pago de una tarifa.

El programa bajo licencia descrito en esta información y en todo el material con licencia disponible los ofrece IBM bajo los términos del Acuerdo de Cliente IBM, el Acuerdo de Licencia de Programa Internacional de IBM o cualquier acuerdo equivalente.

Los datos de rendimiento aquí contenidos se han determinado en un entorno controlado. Por lo tanto, los resultados que se obtengan en otros entornos operativos pueden variar significativamente. Pueden haberse realizado algunas mediciones en sistemas a nivel de desarrollo y no existe ninguna garantía de que estas mediciones vayan a ser equivalentes en sistemas disponibles generalmente. Además, puede que se haya estimado alguna medida mediante la extrapolación. Los resultados reales pueden variar. Los usuarios de este documento deben verificar los datos aplicables a su entorno específico.

La información concerniente a productos no IBM se ha obtenido de los suministradores de esos productos, de sus anuncios publicados o de otras fuentes de información pública disponibles. IBM no ha comprobado dichos productos y no puede afirmar la exactitud en cuanto a rendimiento, compatibilidad u otras características relativas a productos no IBM. Las consultas acerca de las posibilidades de productos no IBM deben dirigirse a los suministradores de los mismos.

Todas las declaraciones relativas a la dirección o intención futura de IBM están sujetas a cambios o anulación sin previo aviso y representan únicamente metas y objetivos.

Todos los precios de IBM mostrados son precios actuales de venta al por menor sugeridos por IBM y sujetos a modificaciones sin previo aviso. Los precios de los concesionarios pueden ser diferentes.

Esta información está pensada a efectos de planificación. La información aquí contenida está sujeta a cambios antes de que los productos descritos estén disponibles.

Esta información contiene ejemplos de datos e informes utilizados en operaciones comerciales diarias. Para ilustrarlas de la forma más completa posible, los ejemplos incluyen nombres de personas, empresas, marcas y productos. Todos estos nombres son ficticios y cualquier parecido con nombres y direcciones utilizados por una empresa real es mera coincidencia.

Si está viendo esta información en copia software, es posible que las fotografías y las ilustraciones en color no aparezcan.

Marcas registradas

Los términos siguientes son marcas registradas de International Business Machines Corporation en Estados Unidos y/o en otros países:

IBM
Logotipo de IBM
DB2

AIX
Informix
InfoSphere
Optim

Netezza es una marca registrada de IBM International Group B.V., una empresa de IBM.

Adobe, Acrobat, PostScript y todas las marcas basadas en Adobe son marcas registradas de Adobe Systems Incorporated en Estados Unidos y/o en otros países.

Apache Derby es una marca registrada de The Apache Software Foundation.

Eclipse es una marca registrada de Eclipse Foundation, Inc.

Microsoft, Windows, Windows NT, y el logotipo de Windows son marcas registradas de Microsoft Corporation en Estados Unidos y/o en otros países.

Java y todas las marcas basadas en Java son marcas registradas de Sun Microsystems, Inc. en Estados Unidos y/o en otros países.

Linux es una marca registrada de Linus Torvalds en Estados Unidos y/o en otros países.

UNIX es una marca registrada de The Open Group en Estados Unidos y/o en otros países.

Otros nombres de empresas, productos o servicios pueden ser marcas registradas de terceros.

Índice

A

- alias de almacén de datos
 - asistente Nuevo alias de almacén de datos 15
 - asistente Propiedades de conexión 15
 - conexión 16
 - definición 15
 - edición 15
 - visión general 15
- asistente Añadir tablas 20
- asistente Clave primaria nueva 32
- asistente Nueva conexión de repositorio 20
- asistente Nueva correlación de columnas 37
- asistente Nueva correlación de tablas 34
- asistente Nueva relación Optim 30
- asistente Nuevo alias de almacén de datos 15
- Asistente Nuevo servicio 41, 44, 46, 49
- asistente Propiedades de conexión 15
- asistente Transformar solicitud en servicio Optim 17

B

- búsqueda conmutada
 - ejemplo 114

C

- carpeta Explorador de repositorios 3
- claves primarias de Optim
 - asistente Clave primaria nueva 32
 - creación 32
 - edición 32
 - Editor de claves primarias 32, 33
 - explícitas 31
 - genéricas 31, 33
 - selección de columnas de claves 32
 - valor de propagación 92
 - visión general 31
- codificación
 - procedimientos de correlación de columnas 112
- comparación de datos 122
- correlaciones de columnas
 - adición de correlaciones de tablas 35
 - aplicación de una función de enmascaramiento de datos 38, 53
 - asistente Nueva correlación de columnas 37
 - correlación de columna de origen 38
 - creación 37
 - crear procedimiento de correlación de columnas 38
 - edición 38
 - editar parámetros en procedimiento de correlación de columnas 39

- correlaciones de columnas (*continuación*)
 - editar procedimiento de correlación de columnas 39
 - Editor de correlaciones de columnas 38
 - editor de scripts de Lua 38, 39
 - procedimiento de correlación de columnas 38
 - reglas de compatibilidad de datos 36
 - rutinas de salida 101
 - tipos de datos 36
 - ventana Aplicar función 38, 53
 - visión general 35
- correlaciones de tablas
 - adición de correlación de columnas 35
 - asistente Nueva correlación de tablas 34
 - correlaciones de columnas 35
 - creación 34
 - datos de destino predeterminados 34
 - edición 34
 - edición de datos de destino 34
 - Editor de correlaciones de tablas 34, 35
 - visión general 33
- criterios de selección
 - agrupación de datos 22
 - cruce de relaciones 23
 - definición 22
 - gestión de relaciones 23
 - lista de apuntar y disparar 25, 26
 - opciones de acceso a tabla 24
 - opciones de muestreo de datos y de límite de filas 23
 - variables 24, 25
 - ventana Editar criterios de selección 22
 - visión general 22

D

- datos de ejemplo
 - creación de tablas de ejemplo de Optim 12
 - creación de tablas de privacidad de datos 12
 - datos de ejemplo de Optim 5
 - tabla OPTIM_CUSTOMERS 6
 - tabla OPTIM_DETAILS 8
 - tabla OPTIM_FEMALE_RATES 11
 - tabla OPTIM_ITEMS 8
 - tabla OPTIM_MALE_RATES 10
 - tabla OPTIM_ORDERS 7
 - tabla OPTIM_SALES 5
 - tabla OPTIM_SHIP_INSTR 10
 - tabla OPTIM_SHIP_TO 9
 - tabla OPTIM_STATE_LOOKUP 11
 - tablas de privacidad de datos 11
 - visión general 5

- definiciones de acceso
 - adición de tablas 20
 - asistente Añadir tablas 20
 - asistente Nueva conexión de repositorio 20
 - cambiar tablas a tablas de referencia o relacionadas 20
 - creación 20
 - criterios de selección 22
 - cruce de relaciones 23
 - Editor de definiciones de acceso 20, 21, 22, 23
 - eliminación de tablas 21
 - gestión de relaciones 23
 - gestión de tablas 20
 - lista de apuntar y disparar 25, 26
 - opciones de acceso a tabla 24
 - pasos de cruce 21
 - variables 24, 25
 - ventana Editar criterios de selección 22
 - ventana Pasos de cruce 21
 - visión general 19
- directorio Optim
 - asistente Transformar solicitud en servicio Optim 17
 - conexión 17
 - transformar la solicitud en servicio 17
 - ventana Asociar directorio 17

E

- edición de datos 121
- Editor de claves primarias 32, 33
- Editor de correlaciones de columnas 38
 - editor de scripts de Lua 38, 39
 - ventana Aplicar función 38, 53
- Editor de correlaciones de tablas 34
 - pestaña Correlación de tablas 34
 - ventana Nueva correlación de columnas 35
- Editor de definiciones de acceso 20, 21
 - pestaña Propiedades del grupo de datos 22
 - pestaña Relaciones 23, 24
 - pestaña Tablas 23
 - separador Apuntar y disparar 26
- Editor de relaciones 30, 31
 - ventana Añadir expresión de columna 30
 - ventana Seleccionar una columna hijo 30
 - ventana Seleccionar una columna padre 30
- editor de scripts de Lua 38, 39
- Editor de servicios de carga 50
 - asistente Cambiar correlación de tablas 51
 - pestaña Opciones de carga 50
 - pestaña Opciones de proceso 50

- Editor de servicios de conversión 45
 - asistente Cambiar correlación de tablas 45
 - pestaña Opciones de proceso 45
- Editor de servicios de extracción 42
 - asistente Cambiar definición de acceso 44
 - pestaña Conversión 42
 - pestaña Datos y objetos 42
 - pestaña Opciones de compresión de archivos 43
 - pestaña Propiedades de servicio 42
 - pestaña Variables 44
- Editor de servicios de inserción 46
 - asistente Cambiar correlación de tablas 47
 - pestaña Opciones de proceso 46
- Editor de solicitudes de comparación 122, 123
- Editor de tablas 121
- enmascaramiento de datos
 - aplicación de una función de enmascaramiento de datos 38, 53
 - constante booleana 94
 - constante numérica 94
 - crear procedimiento de correlación de columnas 38
 - editar parámetros en procedimiento de correlación de columnas 39
 - editar procedimiento de correlación de columnas 39
 - expresiones concatenadas 93
 - expresiones numéricas 94
 - función aleatoria 89
 - función de antigüedad 95
 - función de búsqueda 53
 - Función de búsqueda aleatoria 60
 - función de búsqueda hash 56
 - función de identidad 91
 - función de moneda 98
 - función de nombre de correo electrónico generado automáticamente 101
 - Función de número aleatorio 101
 - Función de número secuencial 101
 - función de propagación 92
 - función de reorganización 63
 - Función de secuencia de Oracle 91
 - función de serie 91
 - función de subserie 88
 - función secuencial 90
 - función TRANS CCN 68
 - función TRANS COL 75
 - función TRANS EML 71
 - función TRANS NID 77
 - CF italiano 82
 - INSEE de Francia 80
 - NIF español 84
 - NINO del Reino Unido 85
 - SIN de Canadá 79
 - SSN de Estados Unidos 87
 - función TRANS SSN 65
 - funciones 53
 - funciones de búsqueda 53
 - funciones de literal y valor 94
 - literal de fecha/hora 94
 - literal de serie 94

- enmascaramiento de datos (*continuación*)
 - literal hexadecimal 94
 - nombre de correo electrónico formateado 101
 - NULL 94
 - registro especial 94
 - visión general 53
- enmascaramiento de ID nacional
 - ejemplo 117
- examinar datos 121
- Explorador de directorios 3
- Explorador de repositorios
 - carpeta Explorador de repositorios 3
 - visión general 2
- expresiones concatenadas 93
- expresiones numéricas 94

F

- función aleatoria 89
- función de antigüedad 95
 - antigüedad incremental 96
 - antigüedad semántica 96
 - salida de formato de destino 102
 - salida de formato de origen 101
- función de búsqueda 53
- Función de búsqueda aleatoria 60
- Función de búsqueda hash 56
- función de identidad 91
- función de moneda 98
 - conversión directa 98
 - triangulación 99
- función de nombre de correo electrónico formateado 101
- función de nombre de correo electrónico generado automáticamente 101
- Función de número aleatorio 101
- Función de número secuencial 101
- función de propagación 92
- función de reorganización 63, 64
- Función de secuencia de Oracle 91
- función de serie 91
- función de subserie 88
- función secuencial 90
- función TRANS CCN 68
- función TRANS COL 75
- función TRANS EML 71
- función TRANS NID
 - CF italiano 82
 - INSEE de Francia 80
 - NIF español 84
 - NINO del Reino Unido 85
 - SIN de Canadá 79
 - SSN de Estados Unidos 87
 - visión general 77
- función TRANS SSN 65
- Funciones de búsqueda 53
- funciones de literal y valor 94

G

- gestión de datos
 - comparación de datos 122
 - creación de tablas 123
 - definición de una solicitud de comparación 122

- gestión de datos (*continuación*)
 - edición de datos 121
 - edición de una solicitud de comparación 123
- Editor de solicitudes de comparación 122, 123
- Editor de tablas 121
- ejecución de una solicitud de comparación 123
- examinar datos 121
- utilidad de comparación 122
- utilidad de creación 123
- utilidad Examinar 121
- utilidades de Optim Designer 121

L

- lista de apuntar y disparar
 - creación 25
 - selección 26
- separador Apuntar y disparar 26
- ventana Nuevo archivo de apuntar y disparar 25
- visión general 25

M

- máscara de datos 53
- modalidad incluida de Optim Manager 4, 51
- modelado de datos
 - definiciones de acceso 19
 - visión general 19
- modelos de datos
 - definiciones de acceso 19
 - visión general 19

O

- Optim Designer
 - conexiones de base de datos 15
 - Explorador de directorios 3
 - Explorador de repositorios 2
 - funciones de accesibilidad 13
 - iniciación 1
 - perspectiva Optim 1
 - soporte de base de datos 13
 - visión general 1

P

- perspectiva Optim
 - Explorador de directorios 3
 - Explorador de repositorios 2
 - visión general 1
- probar servicios 51
- procedimiento de correlación de columnas 110
 - codificación 112
 - correlaciones de columnas 38
 - creación 38
 - edición 39
 - editar parámetros 39
 - ejemplo 113, 114, 117
 - formato de datos numéricos 112

- procedimiento de correlación de columnas (*continuación*)
 - funciones 110
 - limitaciones 112
 - nombres reservados 110
- propagar valor de clave primaria 92

R

- relaciones de Optim
 - ampliadas 26
 - asistente Nueva relación Optim 30
 - compatibilidad de datos 28
 - creación 30
 - edición 30
 - edición de columnas 30
 - Editor de relaciones 30
 - explícitas 26
 - expresiones de columna 30
 - genéricas 26
 - orden de columnas 31
 - relación genérica 31
 - restricciones 27
 - visión general 26
- repositorio de Optim
 - cambio de conexión 16
 - conexión 16
 - conexión de directorio Optim 17
 - edición de conexión 16
 - transformar la solicitud en servicio 17
 - ventana Conmutar repositorio 16
 - ventana Editar conexión de repositorio 16
 - ventana Nueva conexión de repositorio 16
 - visión general 16
- rutinas de salida
 - archivos de cabecera de ejemplo 102
 - escribir 102
 - requisitos 103
 - salida de formato de destino
 - códigos de retorno 109
 - entrada a la función de antigüedad 102
 - formatos 108
 - función 107
 - función de devolución de llamada 108
 - modalidades de terminación
 - anormal 109
 - parámetros 108
 - proceso 108
 - salida de formato de origen
 - códigos de retorno 107
 - entrada a la función de antigüedad 101
 - función 105
 - función de devolución de llamada 106
 - modalidades de terminación
 - anormal 106
 - parámetros 105
 - proceso 106
 - salida estándar
 - códigos de retorno 105

- rutinas de salida (*continuación*)
 - salida estándar (*continuación*)
 - funciones de devolución de llamada 104
 - parámetros 104
 - proceso 104
 - Salida estándar 103
 - utilización de DLL 103
 - visión general 101

S

- servicio de carga
 - archivos de salida 48
 - Asistente Nuevo servicio 49
 - cambiar correlación de tablas 51
 - creación 49
 - edición 50
 - Editor de servicios de carga 50, 51
 - opciones de cargador DBMS 50
 - opciones de proceso 50
 - visión general 48
- servicio de conversión
 - Asistente Nuevo servicio 44
 - cambiar correlación de tablas 45
 - creación 44
 - edición 45
 - Editor de servicios de conversión 45
 - opciones de proceso 45
 - visión general 44
- servicio de extracción
 - Asistente Nuevo servicio 41
 - cambiar definición de acceso 44
 - creación 41
 - edición 42
 - Editor de servicios de extracción 42, 43, 44
 - objetos y opciones de agrupación 42
 - opciones de compresión de
 - archivos 43
 - opciones de conversión 42
 - opciones de proceso 42
 - opciones de variables 44
 - visión general 41
- servicio de inserción
 - Asistente Nuevo servicio 46
 - cambiar correlación de tablas 47
 - creación 46
 - edición 46
 - Editor de servicios de inserción 46, 47
 - opciones de proceso 46
 - visión general 45
- servicios de gestión de datos
 - modalidad incluida de Optim Manager 51
 - probar 51
 - servicio de carga 48
 - servicio de conversión 44
 - servicio de extracción 41
 - servicio de inserción 45
 - visión general 41

U

- utilidad de comparación
 - visión general 122
- utilidad de creación 123
- utilidad Examinar 121
- utilidades de Optim Designer
 - comparación de datos 122
 - creación de tablas 123
 - definición de una solicitud de comparación 122
 - edición de datos 121
 - edición de una solicitud de comparación 123
 - Editor de solicitudes de comparación 122, 123
 - Editor de tablas 121
 - ejecución de una solicitud de comparación 123
 - examinar datos 121
 - utilidad de comparación 122
 - utilidad de creación 123
 - utilidad Examinar 121
 - visión general 121

V

- variables
 - creación 25
 - ventana Añadir una variable 25
 - visión general 24
- ventana Añadir una variable 25
- ventana Aplicar función 38, 53
- ventana Asociar directorio 17
- ventana Conmutar repositorio 16
- ventana Editar conexión de repositorio 16
- ventana Editar criterios de selección 22
- ventana Ejecutar servicio 51
- ventana Nueva conexión de repositorio 16
- ventana Nueva correlación de columnas 35
- ventana Nuevo archivo de apuntar y disparar 25
- ventana Pasos de cruce 21



Impreso en España