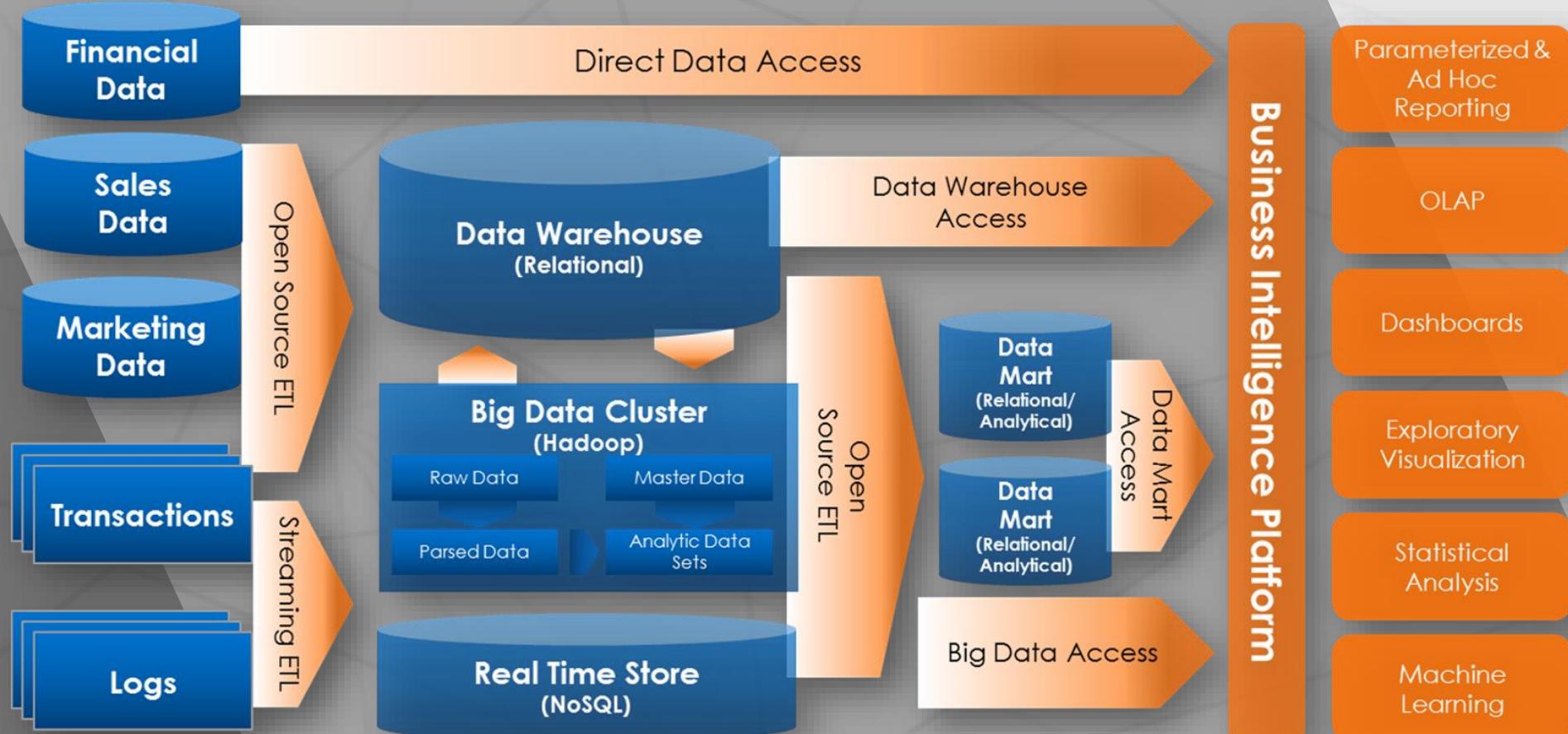


Diferencias entre Big
Data and Business
Intelligence





Diferencias entre Business Intelligence y Big Data

- Los sistemas de BI se convirtieron en la solución ideal para empresas que necesitaban estructurar la acumulación de datos o información. Fue la primera respuesta y el acercamiento más conveniente para manipular datos de distintas fuentes a través de paneles útiles; con capacidades de generación de informes.
- Al ser la primera opción ofrecida para dar una solución práctica a una necesidad sentida; muchas empresas se decantaron y siguen esta opción. Sin embargo; tanto el enfoque como la cantidad de datos han ido mutando al ritmo de las demandas tecnológicas actuales.
- A raíz de esto fue requirió un nuevo enfoque para abordar los mismos problemas del pasado; pero con un acercamiento fresco, actual. Y allí fue donde se concibió el Big Data. Dicho en otras palabras, el Big Data representa un cambio de paradigma en la forma como se maneja la información en los nuevos tiempos.

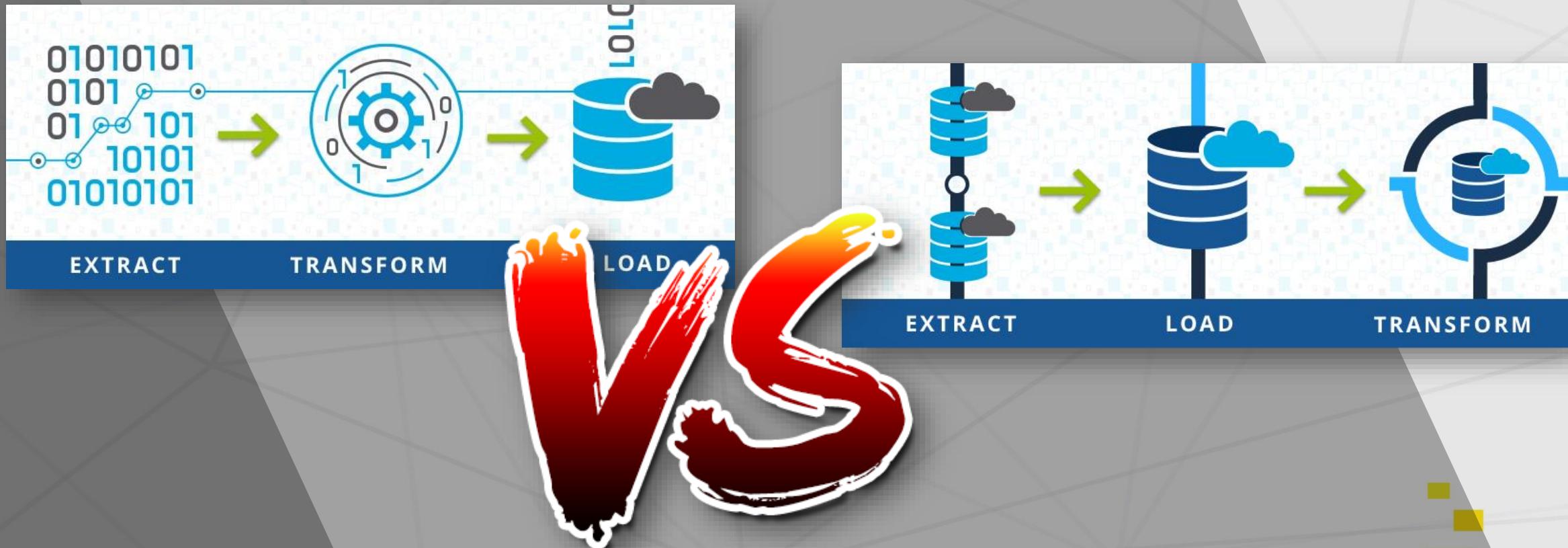
¿Pueden coexistir en un mismo sistema aplicaciones de Business Intelligence y Big Data?

- Business Intelligence y Big Data manejan grandes volúmenes de datos.
- Business Intelligence y Big Data responden y exploran incógnitas previas relacionadas con los objetivos de negocios.
- Business Intelligence y Big Data generan reportes para ofrecer indicadores de gestión que sirven para medir la consecución de objetivos de negocio.

ETL Vs ELT: La diferencia
está en el cómo



ETL Vs ELT: La diferencia está en el cómo



ETL (Extract, Transform and Load)

En palabras simples, mediante alguna **herramienta de ETL** podemos realizar lo siguiente:

- Conectarse a la fuente de los datos.
- Hacer la transformación dentro de la misma herramienta.
- Cargar los datos a la **base de datos** destino.

ELT (Extract, Load and Transform)

E-LT podría definirse siguiendo el orden de las iniciales que lo denominan. Así se puede decir que consiste en la extracción, carga y transformación de datos, y se resume en los siguientes tres pasos :

- Extraer y cargar los datos de manera “BULK” directamente a una Base de Datos o a unas tablas especialmente creadas para los **datos de paso** (staging). Esto supone que este medio servirá solo temporalmente, por lo que podrá ser limpiado en cada proceso de carga.
- Cuando la información se encuentre en el staging se elaborará el **proceso de transformación de los datos**, que posteriormente pasará a la base de datos del **Data Warehouse**. Esta transformación se hará con el lenguaje propio de la base de datos, por ejemplo T-SQL, PL/SQL.
- Una vez que se tienen los datos transformados en los procesos propios de la base de datos, se insertarían en el **Data Warehouse**. Terminada esta acción, se pueden limpiar los datos de paso, si se cree conveniente.

Ventajas de E-LT sobre ETL

- **Velocidad de proceso y transformación.** La principal ventaja de E-LT es la forma en que trabaja cada herramienta implicada. En el caso de ETL las herramientas de transformación evalúan registro por registro, mientras que en E-LT la transformación se hace en la base de datos que evalúa los registros en lotes.
- Uso de recursos. Otra **ventaja de E-LT**, es que una **base de datos** está preparada para la optimización de recursos ya sean de disco, memoria o proceso y esto hace que el rendimiento del proceso sea administrado por la configuración de la base de datos. Sin embargo, las herramientas de ETL no toman ventaja de la **configuración del disco (RAID)** ni de la distribución de la memoria y procesador, ya que hacen transformaciones temporales y en muchos casos redundantes.



Comenzando con la
integración de datos de
Pentaho





TEMAS

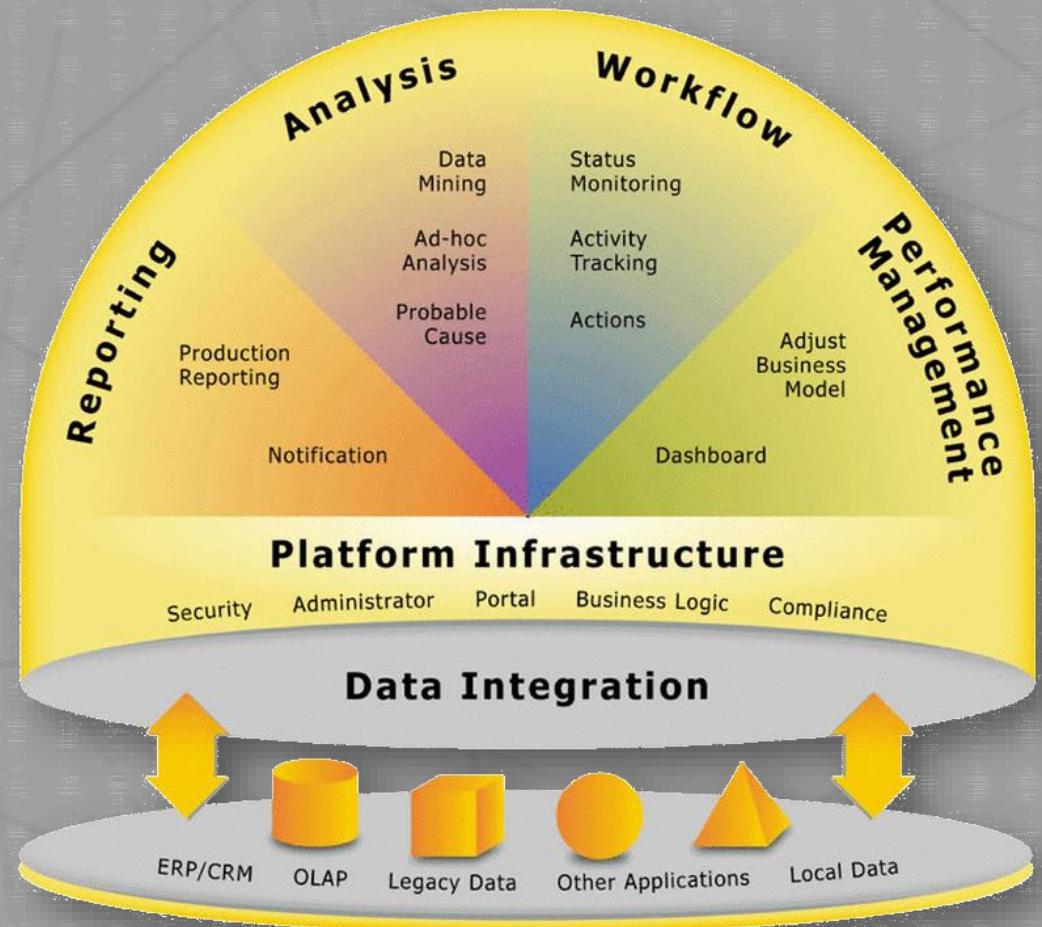
- Aprende qué es la integración de datos de Pentaho
- Instale el software y comience a trabajar con el diseñador gráfico PDI (**Spoon**)
- Explora la interfaz de Spoon
- Configure su entorno instalando otro software relacionado útil

Pentaho Data Integration y Pentaho BI Suite

Comencemos con PDI, pero antes hablemos de Pentaho BI Suite. El Pentaho Business Intelligence Suite es una colección de aplicaciones de software destinadas a crear y entregar soluciones para la toma de decisiones.

Las principales áreas funcionales cubiertas por la suite son:

Pentaho Data Integration y Pentaho BI Suite



Nota

Pueden encontrar más información sobre la plataforma en <https://community.hds.com/community/products-and-solutions/pentaho/>. También hay una edición Enterprise con características adicionales y soporte. Puede encontrar más información sobre esto en <http://www.pentaho.com/>.

Introducción Pentaho Data Integration

- La mayoría de los motores Pentaho, incluidos los motores mencionados anteriormente, se crearon como proyectos comunitarios y luego fueron adoptados por Pentaho. El motor PDI no es una excepción; Pentaho Data Integration es la nueva denominación para la herramienta de inteligencia empresarial nacida como Kettle.



Nota

El nombre Kettle no proviene del acrónimo recursivo Kettle Extraction, Transportation, Transformation and Loading Environment que tiene ahora. Se originó en el entorno de extracción, transporte, transformación y carga de KDE. (KDE ETTL Environment)

Introducción Pentaho Data Integration

Junio 2006	Noviembre 2007	Abril de 2009	Junio de 2010
PDI 2.3 fue lanzado. Numerosos desarrolladores se unieron al proyecto y hubo correcciones de errores proporcionadas por personas en diversas regiones del mundo. La versión incluía, entre otros cambios, mejoras para entornos de gran escala y capacidades multilingües.	PDI 3.0 emergió totalmente rediseñado. Su biblioteca principal cambió para obtener mejoras de rendimiento masivas. La apariencia también había cambiado por completo.	PDI 3.2 se lanzó con una gran cantidad de cambios para una versión menor: nuevas funcionalidades, mejoras de visualización y rendimiento, y una gran cantidad de correcciones de errores.	Se lanzó PDI 4.0, que ofrece principalmente mejoras con respecto a las características de la empresa, por ejemplo, el control de versiones. En la versión comunitaria, el foco estaba en varias mejoras visuales.
Noviembre de 2013	Diciembre 2015	Noviembre 2016	Noviembre 2017
Se lanzó PDI 5.0, que ofrece una mejor vista previa de los datos, un bucle más fácil, muchas mejoras de big data, un mercado de complementos mejorado y cientos de correcciones de errores y mejoras de funciones, como en todas las versiones. En su versión Enterprise, ofrecía interesantes características de bajo nivel, como el equilibrio de carga gradual, las transacciones de trabajo y la capacidad de reinicio.	PDI 6.0 llega con nuevas características como servicios de datos, linaje de datos, mayor soporte para Big Data, cambios en el diseñador gráfico mejorando la experiencia del usuario. Meses después, llega PDI 6.1, que incluye metadatainjection, una característica que permite modificar Transformaciones en tiempo de ejecución.	PDI 7.0 surgió con muchas mejoras en la versión empresarial, que incluyen capacidades de inspección de datos, más soporte para tecnologías de Big Data y mejor administración de repositorio. En la versión de la comunidad, el cambio principal fue un soporte expandido de inyección de metadatos.	Se lanza Pentaho 8.0. Los aspectos más destacados de esta última versión son la optimización de los recursos de procesamiento, una mejor experiencia de usuario y la mejora de la conectividad a las fuentes de datos de transmisión, el procesamiento en tiempo real.

Usando PDI en escenarios del mundo real

Al prestar atención a su nombre, Pentaho Data Integration, podría pensar en PDI como una herramienta para integrar datos.

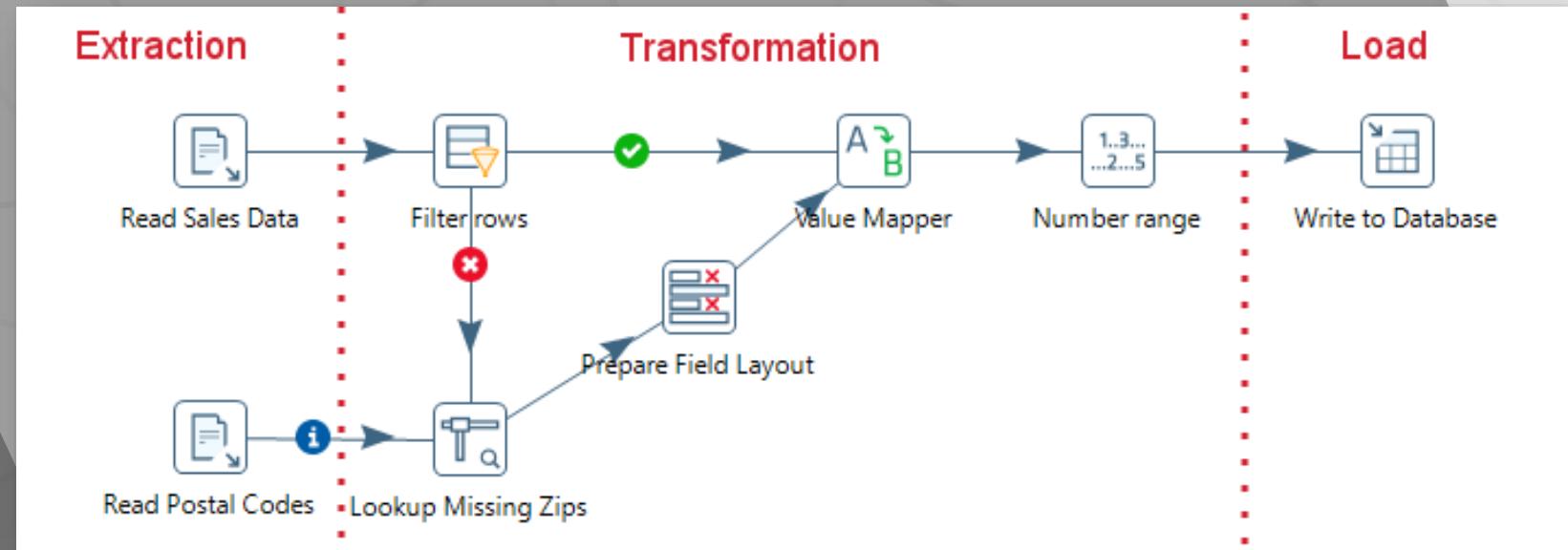
De hecho, PDI no solo sirve como un integrador de datos o una herramienta ETL. PDI es una herramienta tan poderosa que es común ver que se usa para estos y para muchos otros propósitos.

La carga de un data warehouse o un data mart implica muchos pasos y hay muchas variantes según el área de negocio o las reglas de negocio. Sin embargo, en todos los casos, sin excepción, el proceso implica los siguientes pasos:

Usando PDI en escenarios del mundo real

1. Extraer información de una o más bases de datos, archivos de texto, archivos XML y otras fuentes. El proceso de extracción puede incluir la tarea de validar y descartar datos que no coinciden con los patrones o reglas esperados.
2. Transformar los datos obtenidos para satisfacer las necesidades comerciales y técnicas requeridas en el objetivo. Transformar incluye tareas tales como convertir tipos de datos, hacer algunos cálculos, filtrar datos irrelevantes y resumir.
3. Cargando los datos transformados en la base de datos de destino o en el almacén de archivos. Dependiendo de los requisitos, la carga puede sobrescribir la información existente o puede agregar nueva información cada vez que se ejecuta.

Usando PDI en escenarios del mundo real



Nota

Kettle viene listo para realizar cada etapa de este proceso de carga. La captura de pantalla muestra un ETL simple diseñado con la herramienta

Usando PDI en escenarios del mundo real

Integrando datos

Imaginemos 2 empresas similares que necesitan combinar sus bases de datos para tener una vista unificada de los datos, o una sola compañía que debe combinar información de una aplicación principal de Planificación de recursos empresariales (ERP) y una aplicación de gestión de relaciones con el cliente (CRM), aunque no estén conectados. Estos son solo dos de cientos de ejemplos en los que se necesita integración de datos. La integración no es solo una cuestión de recopilar y mezclar datos; Se deben realizar algunas conversiones, validación y transferencia de datos. PDI está destinado a hacer todas estas tareas.

Usando PDI en escenarios del mundo real

Limpieza de datos

La limpieza de datos consiste en garantizar que los datos sean correctos y precisos. Esto se puede lograr verificando si los datos cumplen ciertas reglas, descartando o corrigiendo aquellos que no siguen el patrón esperado, estableciendo valores predeterminados para los datos faltantes, eliminando la información duplicada, normalizando los datos para que se ajusten a los valores mínimos y máximos, y pronto. Estas son tareas que Kettle hace posibles, gracias a su amplio conjunto de capacidades de transformación y validación.

Usando PDI en escenarios del mundo real

Migración de información

Piense en una empresa, de cualquier tamaño, que utiliza una aplicación ERP comercial. Un día, los propietarios se dan cuenta de que las licencias consumen una parte importante de su presupuesto. Así que deciden migrar a un ERP de código abierto. La empresa ya no tendrá que pagar licencias, pero si desea cambiar, tendrá que migrar la información. Obviamente, no es una opción comenzar desde cero o escribir la información a mano. Kettle hace posible la migración, gracias a su capacidad para interactuar con la mayoría de los tipos de fuentes y destinos, como archivos sin formato, bases de datos comerciales y gratuitas, y hojas de cálculo, entre otros.

Usando PDI en escenarios del mundo real

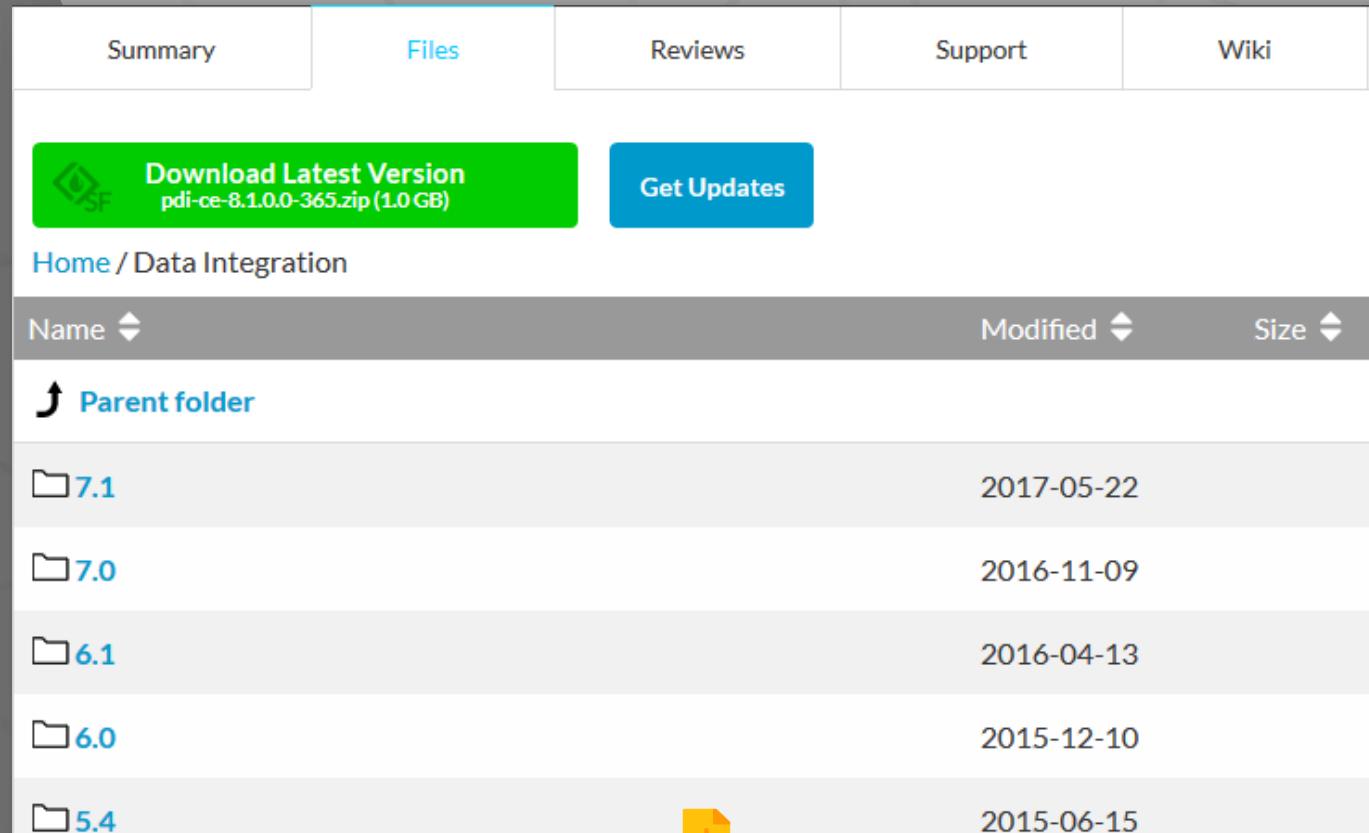
Exportando datos

Los datos pueden necesitar ser exportados por varias razones:

- Para crear informes comerciales detallados.
- Permitir la comunicación entre diferentes departamentos dentro de la misma empresa.
- Para entregar datos de sus sistemas heredados para obedecer las regulaciones gubernamentales, etc.

Kettle se puede utilizar integrado como parte de un proceso o un flujo de datos. Algunos ejemplos son el preprocesamiento de datos para un informe en línea, el envío de correos electrónicos de manera programada, la generación de informes de hoja de cálculo, la alimentación de un panel con datos provenientes de servicios web, etc.

Instalar PDI



Summary Files Reviews Support Wiki

Download Latest Version
pdi-ce-8.1.0.0-365.zip (1.0 GB)

Get Updates

Home / Data Integration

Name	Modified	Size
Parent folder		
7.1	2017-05-22	
7.0	2016-11-09	
6.1	2016-04-13	
6.0	2015-12-10	
5.4	2015-06-15	

Advertencia

Este ejemplo se basa en una instalación CE, pero una EE la herramienta se incluye dentro del instalador y se encuentra en:
<DIR-INST>/design-tools/data-integration

Nota

El único requisito previo para instalar PDI es tener JRE 8.0 instalado. Puedes bajarlo desde:
<https://www.oracle.com/technetwork/java/javase/downloads/index.html>

Pasos 1

Ir a [https://sourceforge.net/projects/pentaho/files/Data Integration/](https://sourceforge.net/projects/pentaho/files/Data%20Integration/)
Elija la versión estable más reciente. En este momento, es 8.0

Pasos 2

Descargue el archivo zip disponible, que le servirá para todas las plataformas.

Pasos 3

Descomprima el archivo descargado en una carpeta de su elección, como, por ejemplo: c:\kettle o /home/usuario/kettle.

Levantemos la “cuchara”

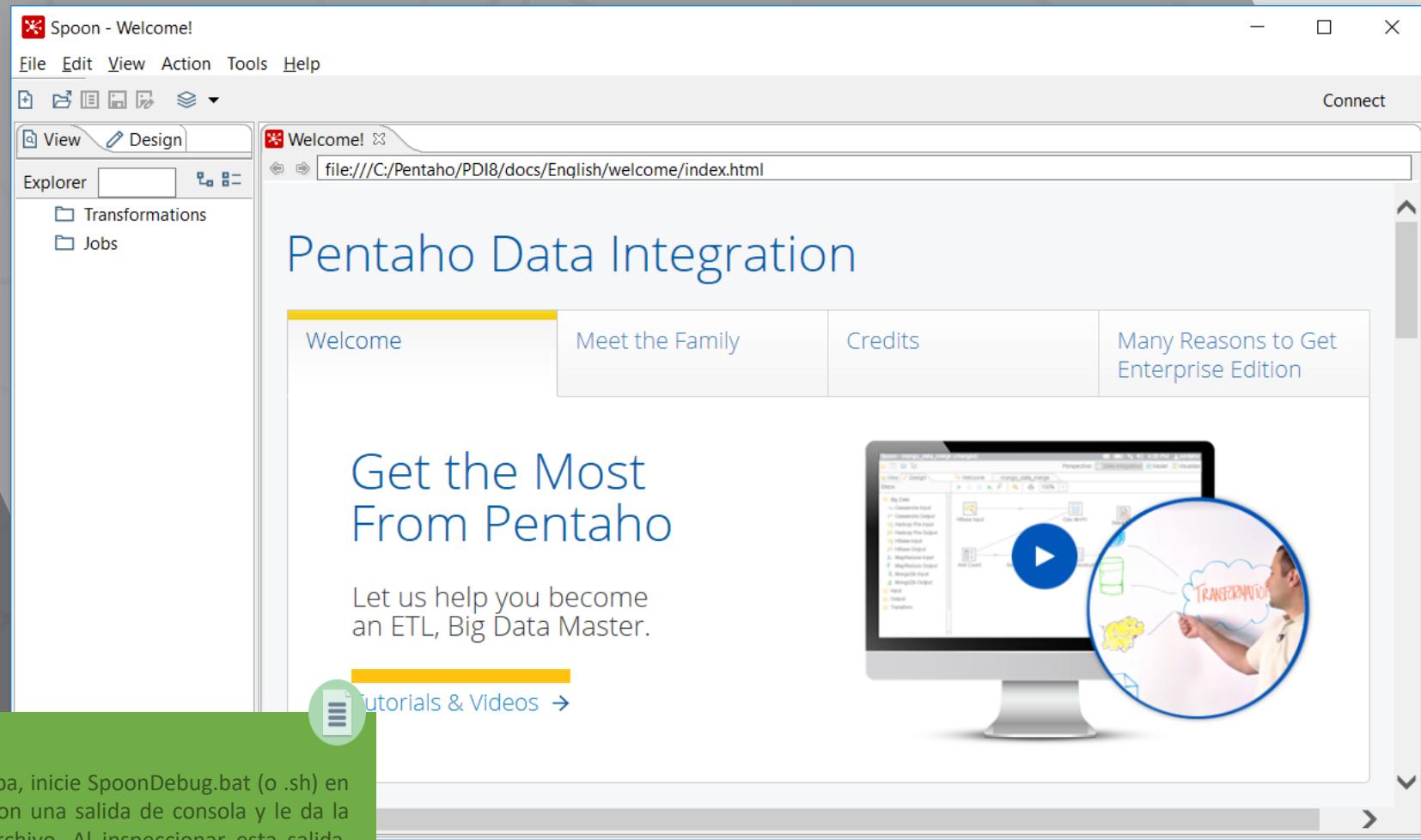
Diseñador Gráfico PDI – SPOON

Spoon es la herramienta de diseño de escritorio de PDI, con la cual podemos diseñar, previsualizar y probar todo nuestro trabajo, es decir, transformaciones y jobs. Cuando vemos capturas de pantalla de PDI, lo que realmente estamos viendo son capturas de pantalla de Spoon.

Ejecutar Spoon

- Si su sistema es Windows, ejecute Spoon.bat desde el directorio de instalación de PDI. En otras plataformas, como Unix, Linux, etc., abrimos un terminal, y ejecutamos spoon.sh.
- Aparece la ventana principal.

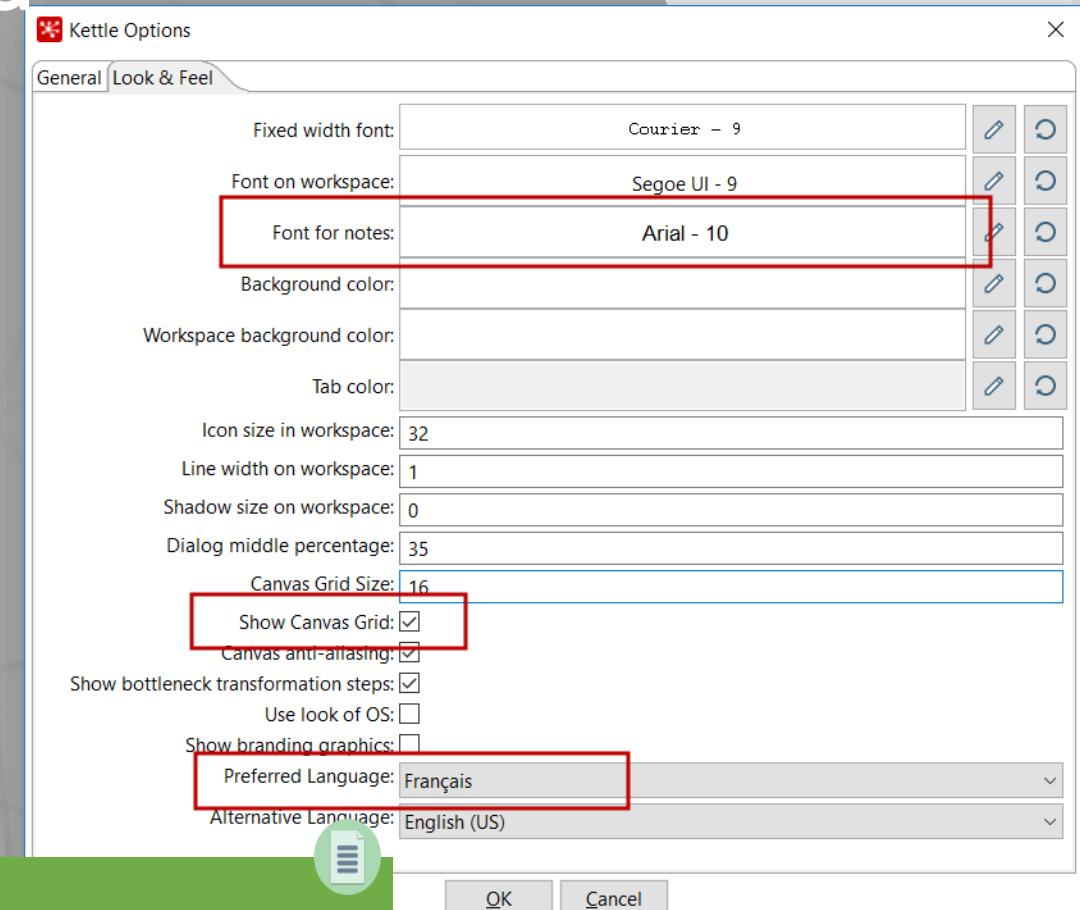
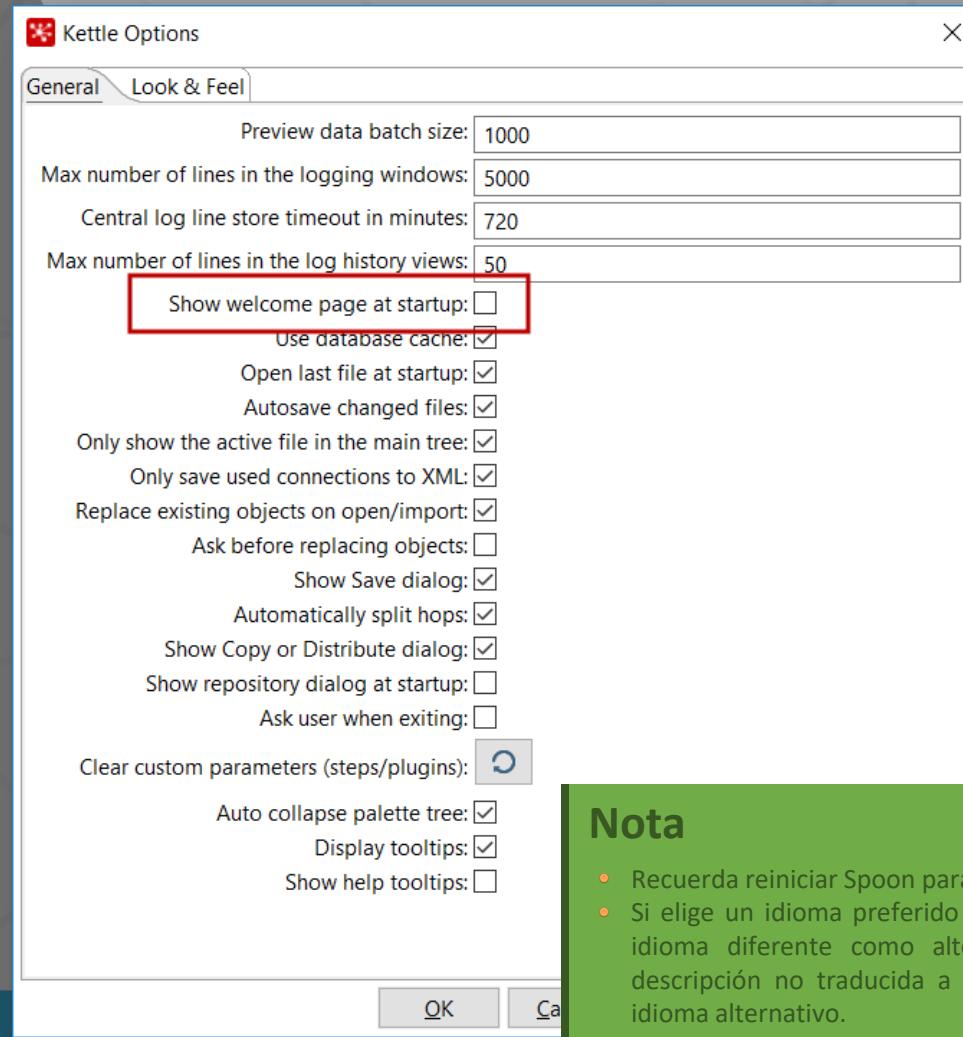
Levantemos la “cuchara”



Nota

Si Spoon no se inicia como se esperaba, inicie SpoonDebug.bat (o .sh) en su lugar. Esta utilidad inicia Spoon con una salida de consola y le da la opción de redirigir la salida a un archivo. Al inspeccionar esta salida, podrá averiguar qué sucedió y solucionar el problema.

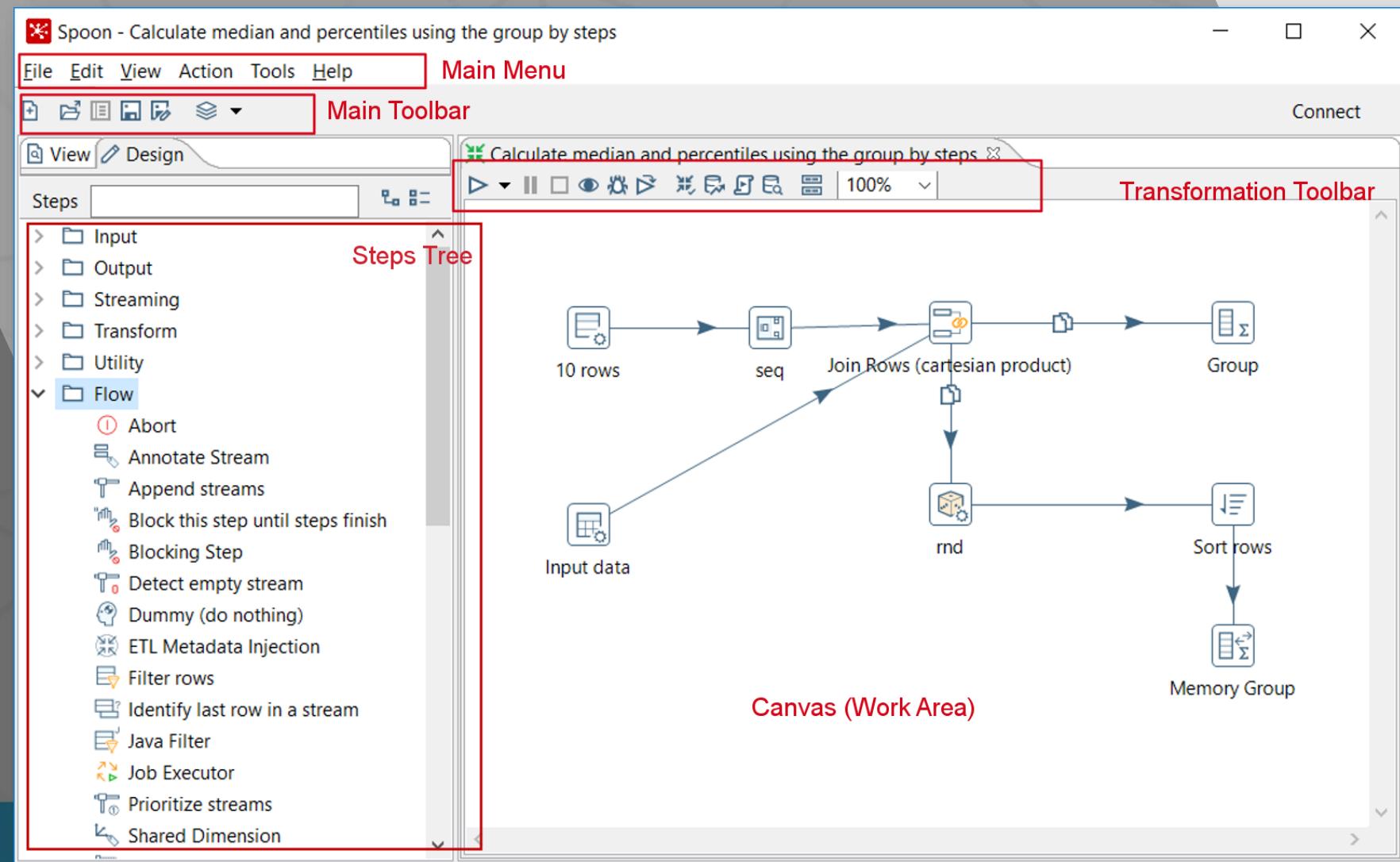
Levantemos la “cuchara”



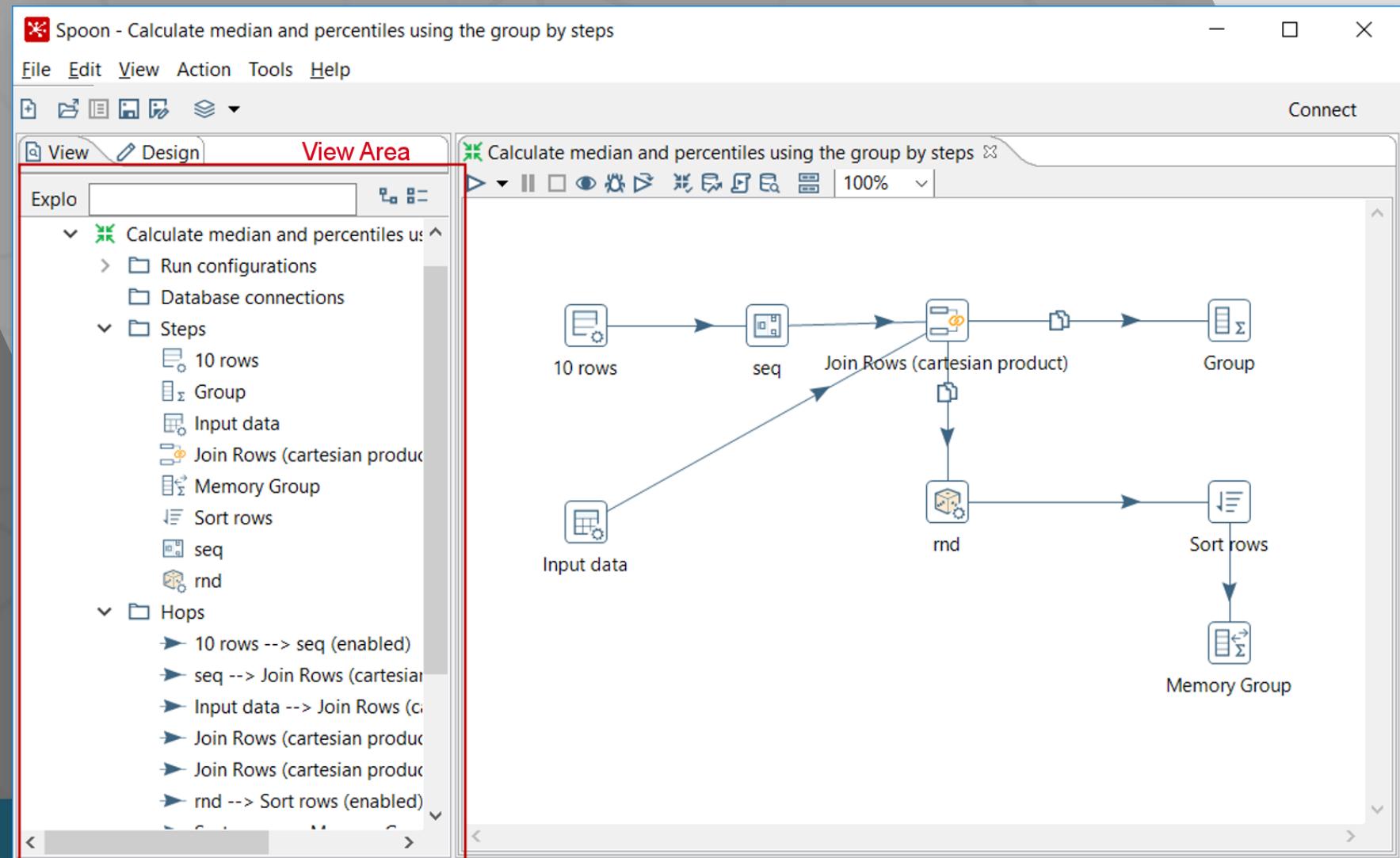
Nota

- Recuerda reiniciar Spoon para ver los cambios aplicados.
- Si elige un idioma preferido que no sea inglés, debe seleccionar un idioma diferente como alternativa. Si lo hace, cada nombre o descripción no traducida a su idioma preferido se mostrará en el idioma alternativo.

Explorando la interfaz de Spoon



Explorando la interfaz de Spoon



Extendiendo la funcionalidad PDI a través de Marketplace

- **CommunityLane:** Para proyectos comunitarios y patrocinados por clientes.
- **CustomerLane:** Para proyectos que forman parte de la oferta oficial de Pentaho. Los proyectos en el carril de cliente pueden comenzar como proyectos desarrollados en el carril comunitario que crean valor para los clientes de suscripción Pentaho.

Nota

Para obtener una explicación completa del modelo y las etapas de madurez, visitar la siguiente URL
<https://community.hds.com/docs/DOC-1009876>.

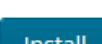


Extendiendo la funcionalidad PDI a través de Marketplace

Marketplace

Available Installed

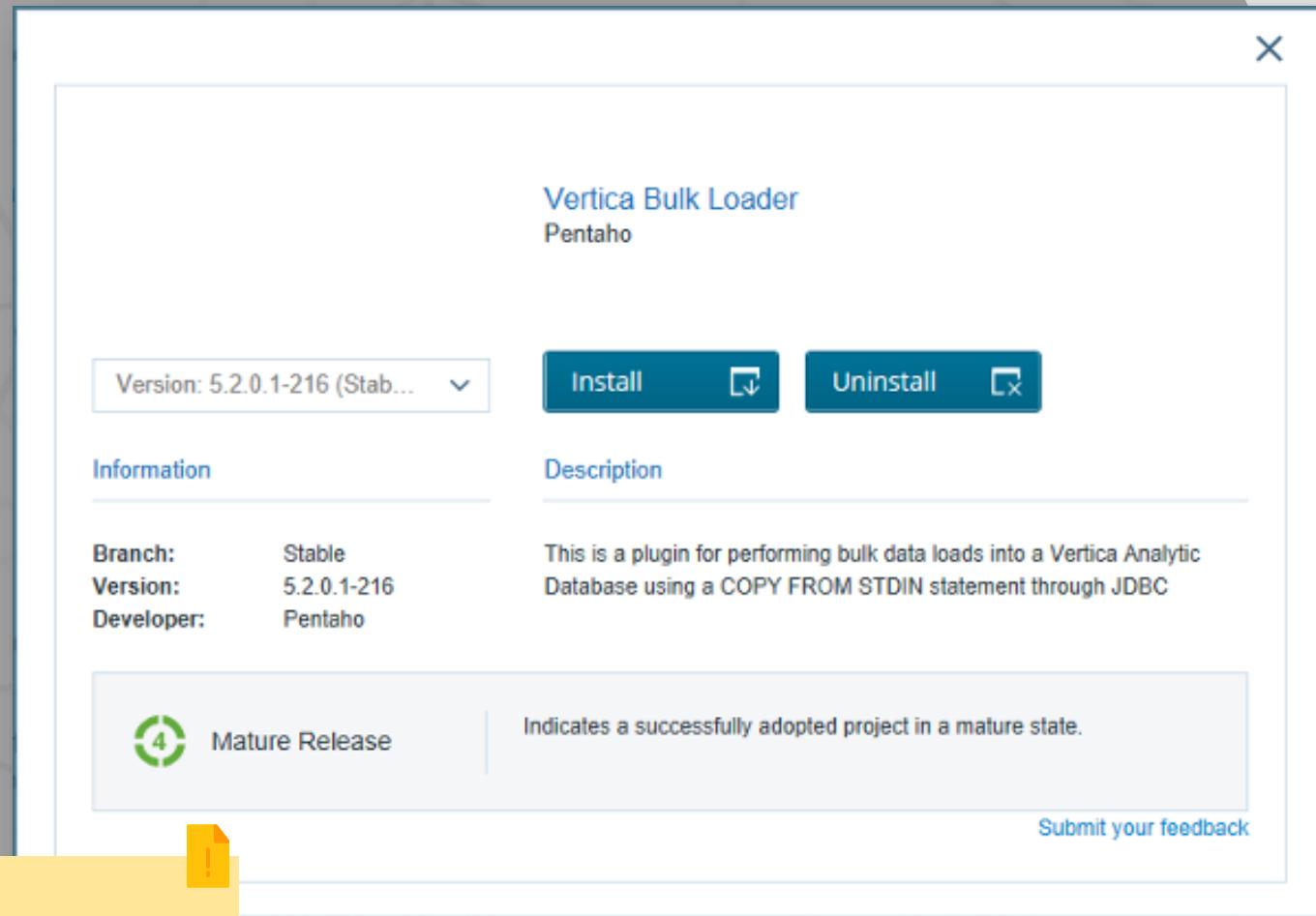
Type: All Stage: All Search  What are stages?

PDI MySQL Plugin Pentaho	 Available TRUNK-SNAPSHOT (3)	
CPython Script Executor Mark Hall	 Available 1.1 (Trunk) (2)	
PDI MQTT Steps Mark Hall	 Available 1 (Trunk) (2)	
PDI NuoDB Plugin NuoDB	 Available 1.0-SNAPSHOT (Stable) (4)	
Apple Push Notification Joel Latino	 Available 1.0.1 (Stable) (3)	
Android Push Notification Joel Latino	 Available 1.0.1 (Stable) (3)	

- **Etapa 1 (Development)**
Significa que el complemento está en desarrollo (experimental).
- **Etapa 2 (Snapshot)**
Versión probable inestable, buena para pruebas y evaluación pero no se recomienda para uso de producción.
- **Etapa 3 (Stable / Limited)**
La adopción está aumentando y el producto podría usarse en entornos de producción (**Community lane**).

Asistencia prestada por Servicios de Desarrollo sin soporte contractual para entornos de producción (**Customer lane**).
- **Etapa 4 (Mature / Production)**
La última etapa de un proyecto, indica un proyecto adoptado con éxito en un estado maduro (**Community lane**).
Lanzamiento de producción con PM asignado, totalmente compatible como parte del ciclo de lanzamiento de Pentaho (**Customer lane**).

Detalle plugins



A screenshot of a plugin details page for the "Vertica Bulk Loader" by Pentaho. The page has a header with the plugin name and developer. It shows the current version (5.2.0.1-216) and two main buttons: "Install" and "Uninstall". Below this, there are two tabs: "Information" and "Description". Under "Information", there are fields for Branch (Stable), Version (5.2.0.1-216), and Developer (Pentaho). Under "Description", there is a brief text explaining the plugin's function: "This is a plugin for performing bulk data loads into a Vertica Analytic Database using a COPY FROM STDIN statement through JDBC". A badge indicates it is a "Mature Release". At the bottom right, there is a link to "Submit your feedback".

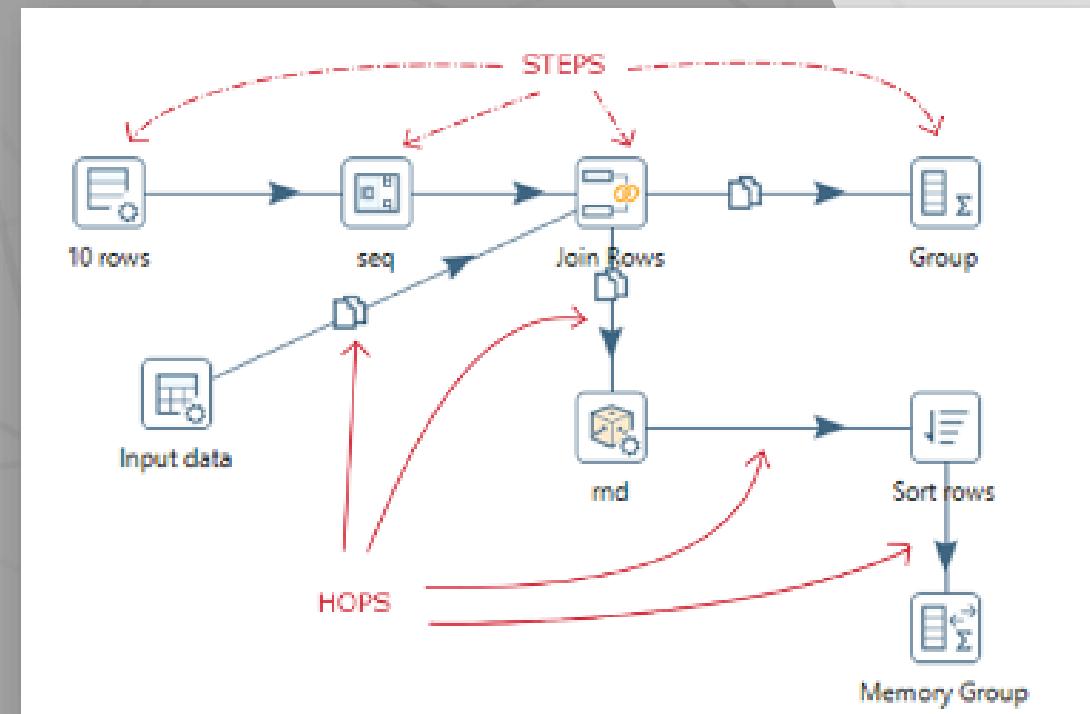
Advertencia

Algunos complementos solo están disponibles para Pentaho Enterprise Edition. Esto se detalla en la descripción completa del complemento.

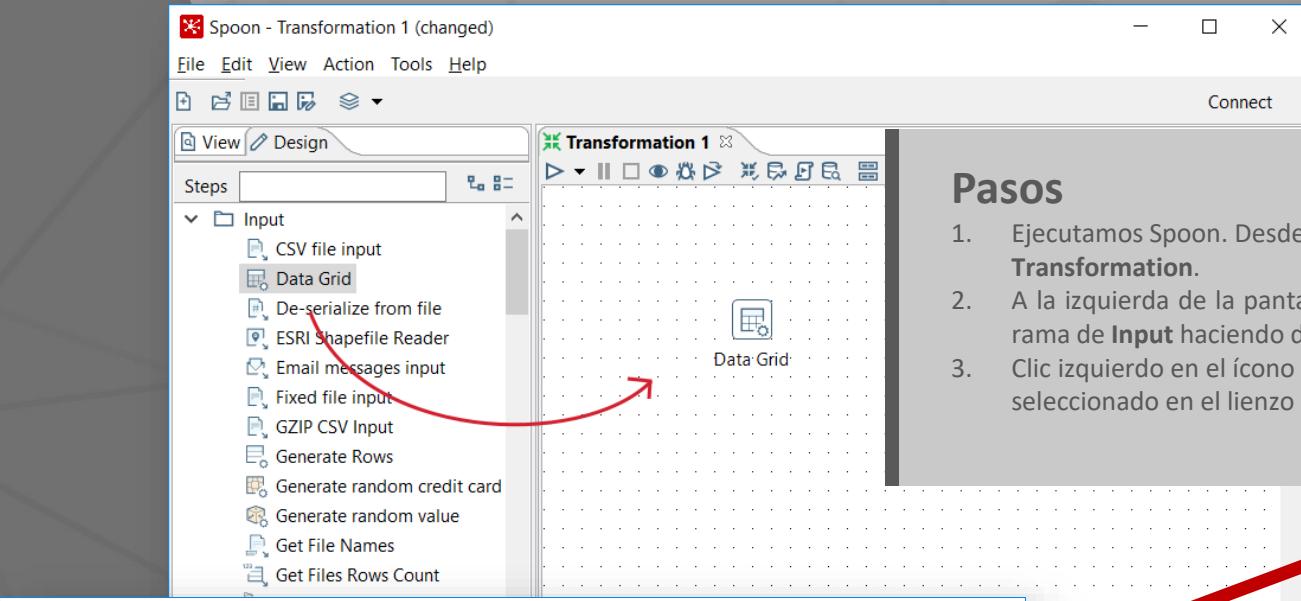
Introduciendo transformaciones

Los fundamentos de las transformaciones.

Una transformación es una entidad hecha de pasos vinculados por saltos. Estos pasos y saltos crean rutas a través de las cuales fluyen los datos: los datos ingresan o se crean en un paso, el paso aplica algún tipo de Transformación y, finalmente, los datos salen de ese paso. Por lo tanto, se dice que una transformación está orientada al flujo de datos. Gráficamente, los pasos se representan con cuadros pequeños, mientras que los saltos se representan con flechas direccionales.

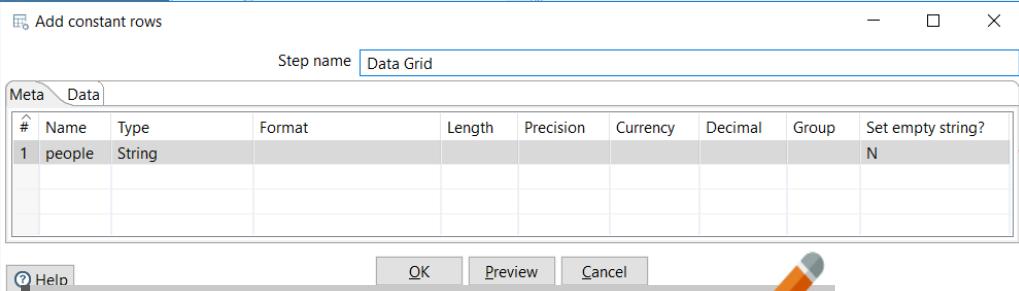


Creando una transformación Hola Mundo!



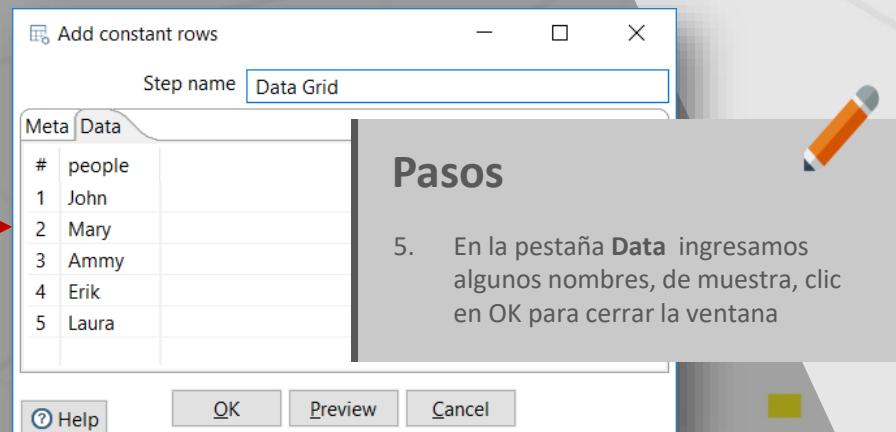
Pasos

1. Ejecutamos Spoon. Desde el menú principal seleccionamos **File | New | Transformation**.
2. A la izquierda de la pantalla, en la pestaña **Design**, verás un árbol de Pasos. Expanda la rama de **Input** haciendo doble clic en ella.
3. Clic izquierdo en el ícono de la "Data Grid" y, sin soltar el botón, arrastre y suelte el ícono seleccionado en el lienzo principal.



Pasos

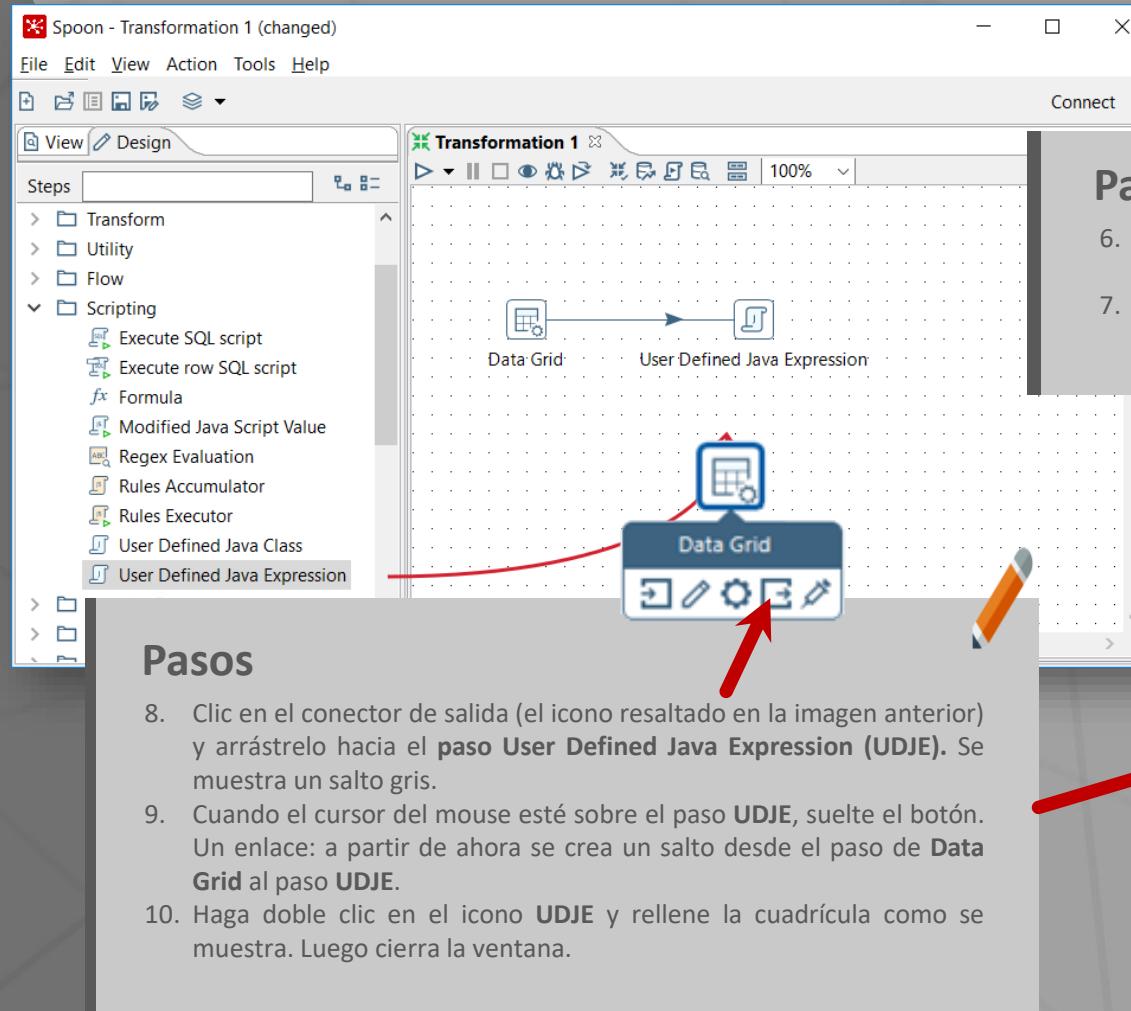
4. Doble clic en el paso “Data Grid” que acaba de colocar en el lienzo y complete la pestaña **Meta** de la siguiente manera



Pasos

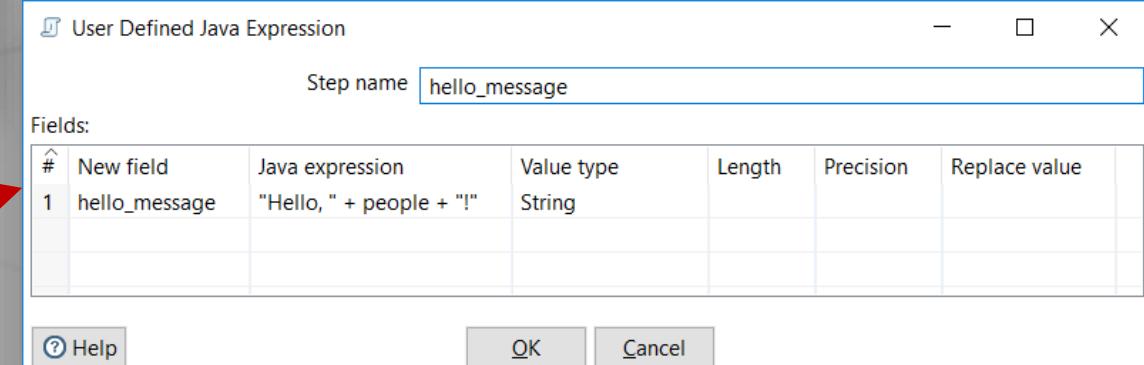
5. En la pestaña **Data** ingresamos algunos nombres, de muestra, clic en OK para cerrar la ventana

Creando una transformación Hola Mundo!



Pasos

6. Desde el árbol de Pasos, hacer doble-clic en **Scripting**, clic en el ícono **User Defined Java Expression** y arrástrelo y suéltelo en el lienzo principal.
7. Coloque el cursor del mouse sobre el paso **Data Grid** y espere hasta que una pequeña barra de herramientas se muestre sucesivamente en el ícono de **Data Grid**.



Vista previa y ejecución de una transformación



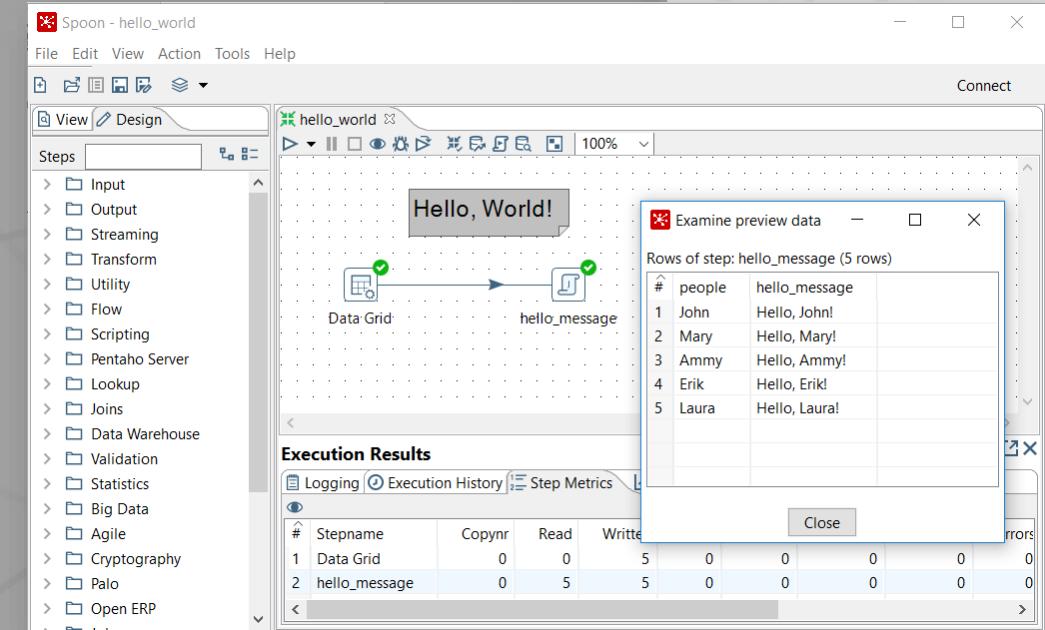
Pasos

1. Seleccione el paso **User Defined Java Expression** haciendo clic izquierdo en él.
2. Clic en el ícono **Preview** en el menú de la barra que aparece en el lienzo principal



Pasos

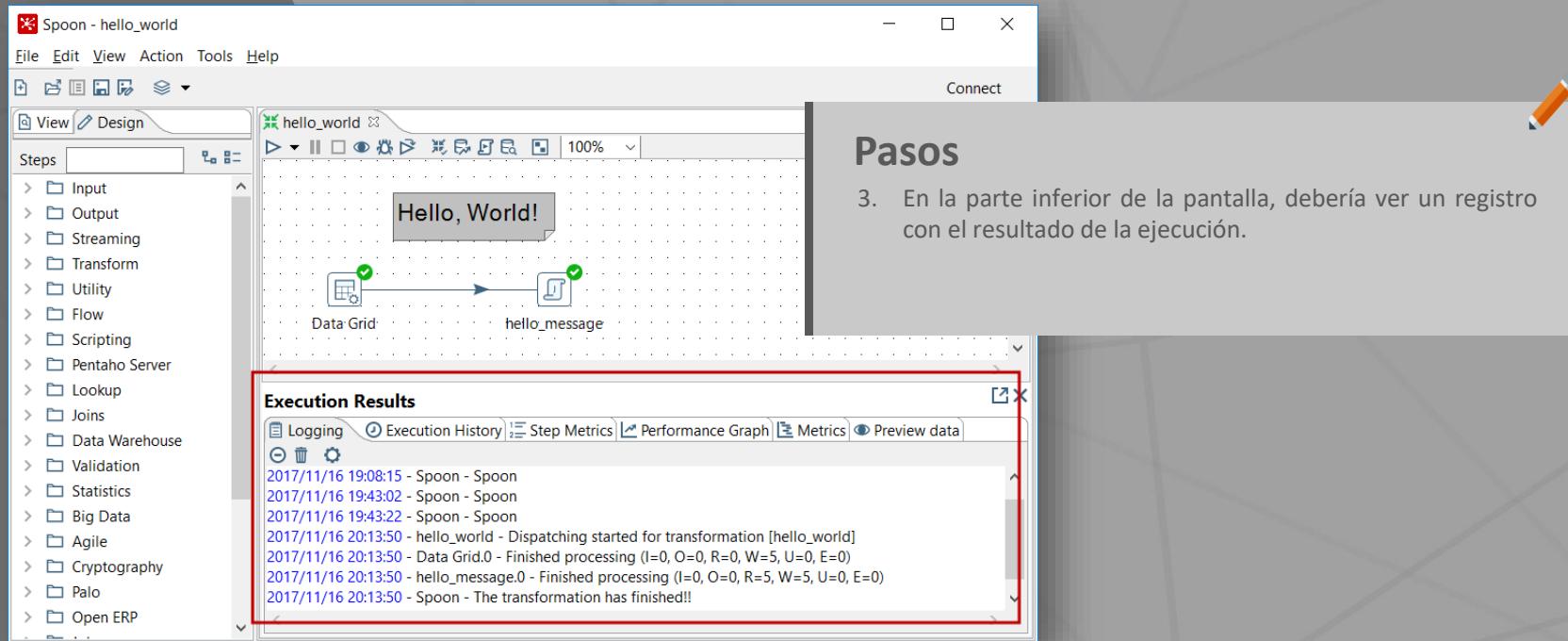
3. Aparecerá la ventana de diálogo de depuración de transformación. Haga clic en el botón **Quick Launch**.
4. Aparecerá una ventana para previsualizar los datos generados por la transformación.



Nota

Puede previsualizar la salida de cualquier paso en la transformación en cualquier momento del proceso de diseño. También puede obtener una vista previa de los datos incluso si aún no ha guardado el trabajo.

Vista previa y ejecución de una transformación



Pasos

1. Clic en el ícono Run
2. Aparece una ventana llamada **Run Options**. Haga clic en Run.

Pasos

3. En la parte inferior de la pantalla, debería ver un registro con el resultado de la ejecución.

Nota

Debe guardar la transformación antes de ejecutarla. Si ha modificado la transformación sin guardarla, se le pedirá que lo haga.



Comenzando con
transformaciones





TEMAS

- Cómo transformar datos.
- El proceso de diseñar, depurar y probar una transformación.
- Características disponibles para ejecutar transformaciones de Spoon.
- Terminología de PDI relacionada con datos y metadatos.
- Introducción al manejo de errores en tiempo de ejecución.

Diseñando transformaciones

El proceso de diseño de PDI no solo consiste en poner pasos, configurarlos y vincularlos con saltos. También implica obtener una vista previa de los datos a medida que agrega más pasos, golpear y corregir errores en el camino, y avanzar y retroceder hasta que todo funcione como se espera.

Diseñando transformaciones

Esta nueva Transformación leerá la lista de proyectos del archivo, y luego calculará el tiempo que tomó completar cada proyecto.

En primer lugar, leeremos el archivo para que su contenido se convierta en nuestro conjunto de datos de entrada. Aquí están las instrucciones:

```
project_name,start_date,end_date
Project A,2016-01-10,2016-01-25
Project B,2016-04-03,2016-07-21
Project C,2017-01-15,???
Project D,2015-09-03,2015-12-20
Project E,2016-05-11,2016-05-31
Project F,2011-12-01,2013-11-30
```

Diseñando transformaciones



Pasos

1. Iniciar Spoon.
2. Desde el menú principal, vaya a **File | New | Transformation**.
3. Expanda la rama de **Input** del árbol de pasos. Recuerde que el árbol de Pasos se encuentra en la pestaña **Design** a la izquierda del área de trabajo.
4. Arrastre y suelte el ícono **CSV file input** en el lienzo.
5. Haga doble clic en el ícono de **CSV file input**, enter **projects** en el campo **Step name**.
6. Bajo **Filename**, haga clic en Examinar ... y busque el archivo en su disco.
7. Rellene la cuadrícula, como se muestra en la siguiente captura de pantalla

#	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Trim type
1	project_name	String							none
2	start_date	Date	yyyy-MM-dd						none
3	end_date	Date	yyyy-MM-dd						none

Pasos

8. Haga clic en Vista previa, y en la pequeña ventana que aparece, haga clic en Aceptar para ver el conjunto de datos definido. Debería ver una ventana de vista previa con las seis filas de datos que provienen del archivo
9. Cerramos la ventana

Examine preview data

Rows of step: CSV file input (6 rows)

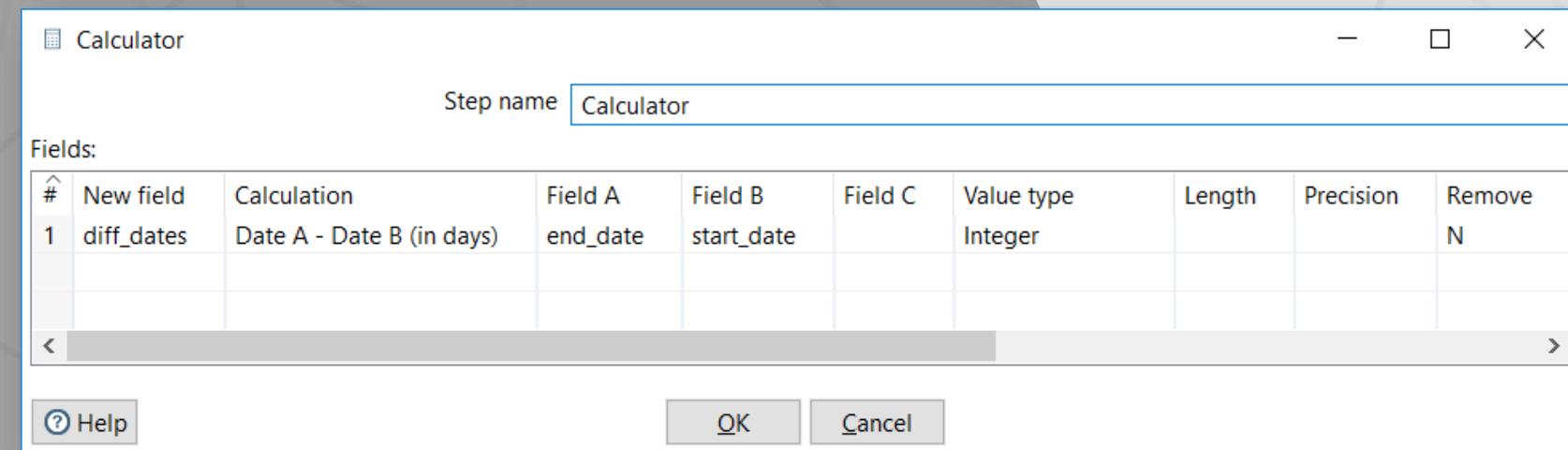
#	project_name	start_date	end_date
1	Project A	2016-01-10	2016-01-25
2	Project B	2016-04-03	2016-07-21
3	Project C	2017-01-15	<null>
4	Project D	2015-09-03	2015-12-20
5	Project E	2016-05-11	2016-05-31
6	Project F	2011-12-01	2013-11-30

< > Close Show Log

Diseñando transformaciones

Actividades

- Un nombre para el nuevo campo: **diff_dates**
- El cálculo a aplicar, que será la diferencia entre fechas (en días).
- Los parámetros para el cálculo: **start_date** y **end_date**
- El tipo para el resultado: **Integer**



Pasos

1. Expanda la rama Transformar de pasos. Busque el paso de la Calculadora y arrástrelo y suéltelo en el área de trabajo.
2. Cree un salto desde el paso de entrada del archivo CSV hacia el paso de la Calculadora. Aparecerá un pequeño menú que le pedirá el tipo de salto. Entre las opciones, seleccione la salida principal del paso.
3. Haga doble clic en el paso de la Calculadora y complete la primera fila de la cuadrícula con la siguiente información
4. Clic en **OK** para cerrar la ventana

Diseñando transformaciones

Pasos

1. Agregue un nuevo paso **Number ranges**, y vincule el paso de **Calculator** al paso de **Number ranges** con un nuevo salto. Asegúrese de que la flecha vaya desde el paso de la **Calculator** hacia el paso del **Number ranges** y no al revés.
2. Con el paso de **Number ranges**, creará un nuevo campo, **performance**, basado en el valor de un campo entrante, **diff_dates**. Haga doble clic en el paso y complete la cuadrícula como se muestra en la pantalla. Luego haga clic en **OK**

Number ranges

Step name: Performance Range
Input field: diff_dates
Output field: performance
Default value(if no range matches): unknown

Ranges (min <= x < max):

#	Lower Bound	Upper Bound	Value
1		30.0	excellent
2	30.0	80.0	very good
3	80.0	160.0	good
4	160.0		poor

Help OK Cancel

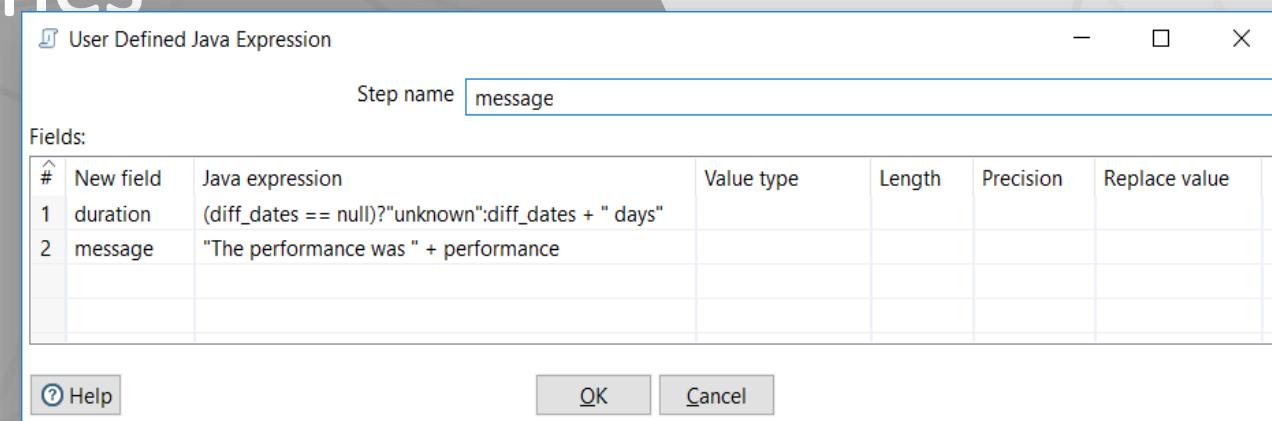
Actividades

- Finalmente, evaluaremos el desempeño del proyecto.

Diseñando transformaciones

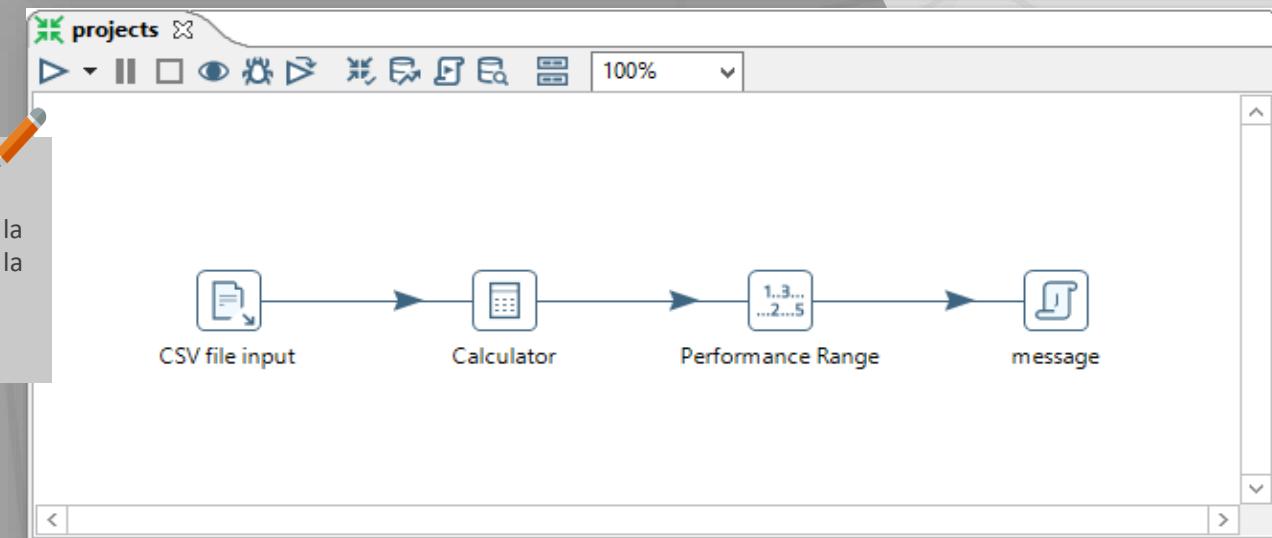
Pasos

3. Desde la rama de Scripting, agregue un paso de **User Defined Java Expression**, y creamos un salto desde el paso de **Number range** hacia este nuevo paso. Cuando cree el salto, se le indicará el tipo de salto. Seleccione **Main output of step**.
4. Con el UDJE, creará dos mensajes informativos: **duration** y **message**. Como en el paso **Calculator**, este paso también le permite crear un nuevo campo por fila. Haga doble clic en el paso y complete la cuadrícula como se muestra en la siguiente pantalla.



Pasos

5. Haga clic en **OK** para cerrar la ventana y guardar la transformación. Su transformación final debe ser similar a la siguiente captura de pantalla:



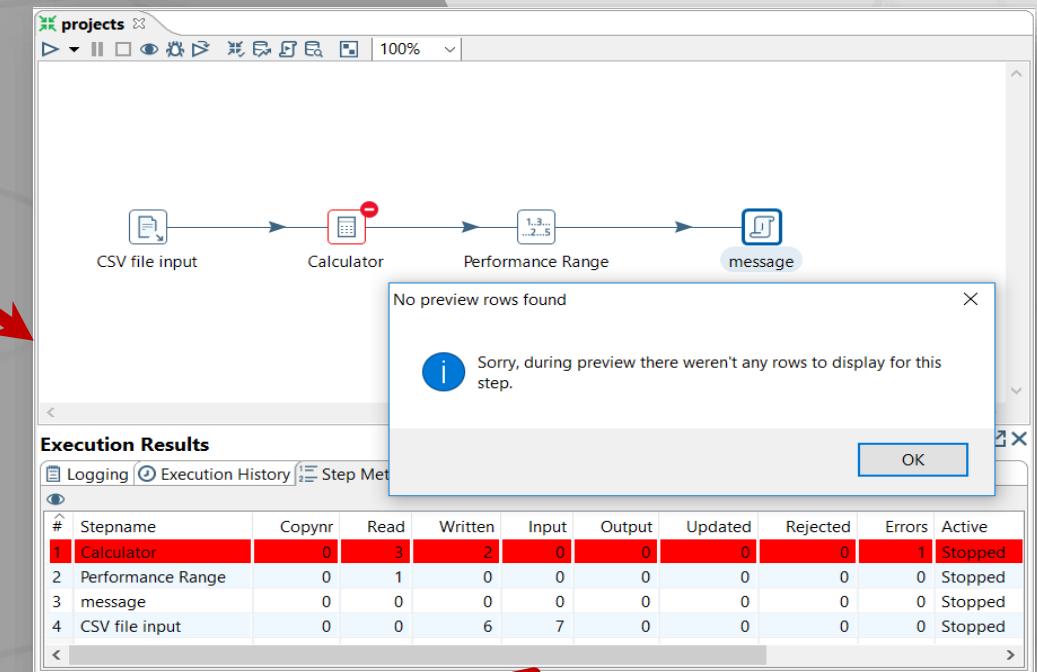
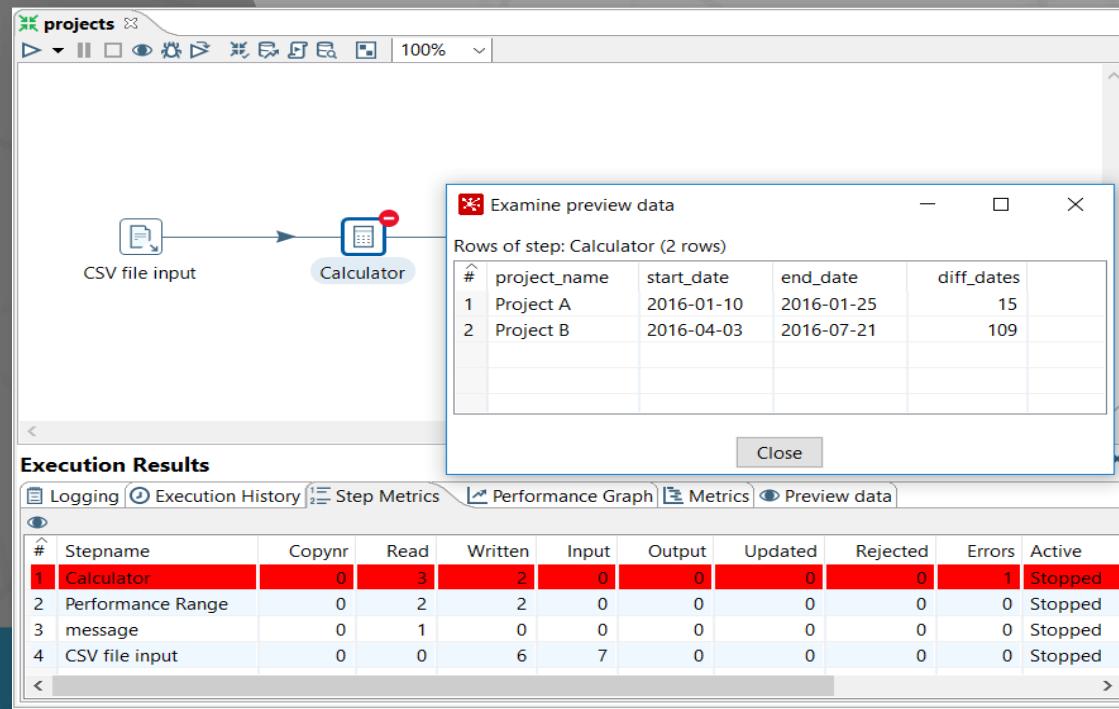
Nota

- Si selecciona involuntariamente la opción incorrecta, no se preocupe. Haga clic derecho en el salto y aparecerá un menú contextual. Seleccione **Delete hop** y vuelva a crear el salto.

Previsualización y corrección de errores a medida que aparecen.

Pasos

1. Seleccione el paso **UDJE** y ejecute una vista previa. Ya sabes cómo hacerlo: haz clic en el ícono **Preview** en la barra de herramientas Transformación y luego haz clic en **Quick Launch**.
2. Clic en el paso **Calculator** y ejecute una vista previa. Esta vez aparece una ventana emergente con resultados, pero solo vemos dos filas



Previsualización y corrección de errores a medida que aparecen.

Pasos

3. Editamos el archivo projects.txt y borramos la fila que está causando el error.
4. Haga clic en el paso **Calculator** y ejecute la vista previa una vez más. Esta vez, verá todas las filas y una nueva columna con el nuevo campo **diff_dates**
5. Cerramos la ventana y damos clic en el paso **UDJE**. Ejecutamos **Preview** nuevamente.



Examine preview data

Rows of step: Calculator (5 rows)

#	project_name	start_date	end_date	diff_dates
1	Project A	2016-01-10	2016-01-25	15
2	Project B	2016-04-03	2016-07-21	109
3	Project D	2015-09-03	2015-12-20	108
4	Project E	2016-05-11	2016-05-31	20
5	Project F	2011-12-01	2013-11-30	730

Close

Examine preview data

Rows of step: message (5 rows)

#	project_name	start_date	end_date	diff_dates	performance	duration	message
1	Project A	2016-01-10	2016-01-25	15	excellent	<null>	<null>
2	Project B	2016-04-03	2016-07-21	109	good	<null>	<null>
3	Project D	2015-09-03	2015-12-20	108	good	<null>	<null>
4	Project E	2016-05-11	2016-05-31	20	excellent	<null>	<null>
5	Project F	2011-12-01	2013-11-30	730	poor	<null>	<null>

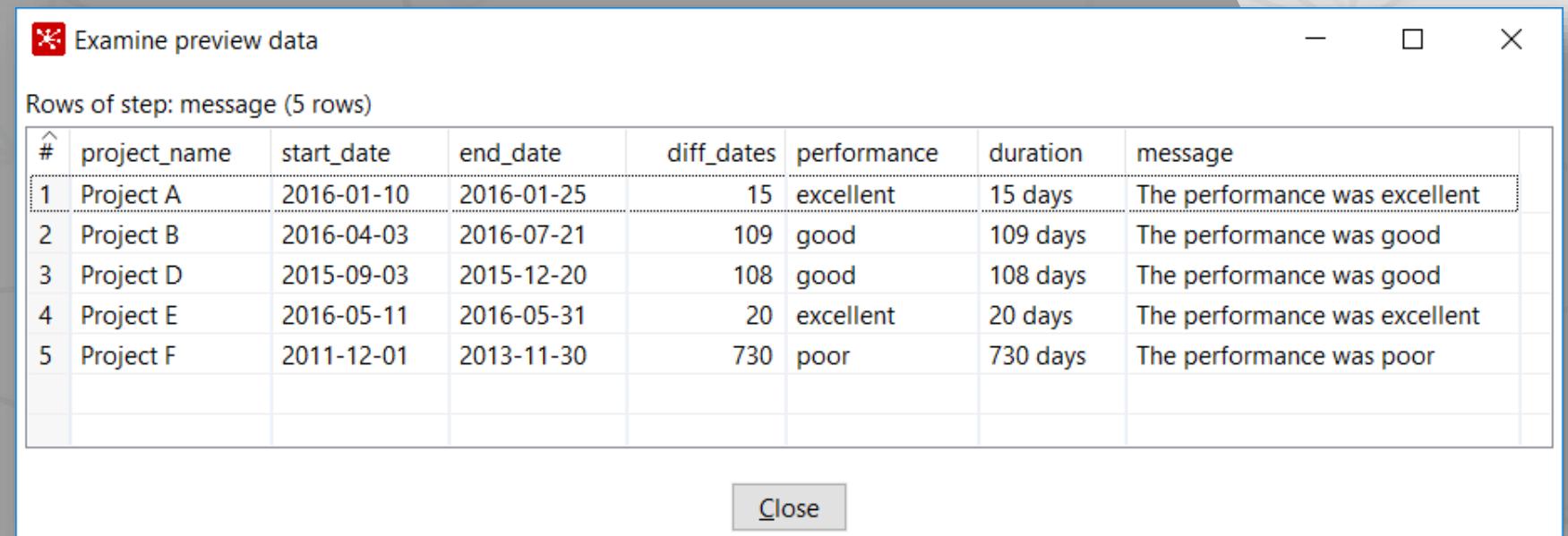
Close



Previsualización y corrección de errores a medida que aparecen.

Pasos

6. Editar el paso UDJE. En la columna **Value Type**, seleccione **String** para los campos de **duration** y **message**. Cerrar la ventana.
7. Asegúrese de que el paso UDJE esté seleccionado y ejecute una vista previa. El error debería haber desaparecido de la pestaña Registro, y la ventana debería mostrar los datos finales:



The screenshot shows a data preview window titled "Examine preview data" with the sub-tittle "Rows of step: message (5 rows)". The window displays a table with the following data:

#	project_name	start_date	end_date	diff_dates	performance	duration	message
1	Project A	2016-01-10	2016-01-25	15	excellent	15 days	The performance was excellent
2	Project B	2016-04-03	2016-07-21	109	good	109 days	The performance was good
3	Project D	2015-09-03	2015-12-20	108	good	108 days	The performance was good
4	Project E	2016-05-11	2016-05-31	20	excellent	20 days	The performance was excellent
5	Project F	2011-12-01	2013-11-30	730	poor	730 days	The performance was poor

Close

Observando los resultados en el panel de resultados de ejecución

- El panel **Execution Results**: muestra lo que sucede mientras obtiene una vista previa o ejecuta una transformación. Este panel se ubica siguiendo el área de trabajo. Si no está visible de inmediato, aparecerá cuando se muestre o se ejecute una Transformación.
- El tab Logging : muestra la ejecución de su transformación, paso a paso. De forma predeterminada, el nivel de los detalles de registro es el registro básico, pero puede elegir entre las siguientes opciones: Nothing at all, Error logging only, Minimal logging, Basic logging, Detailed logging, Debugging, Rowlevel (mucho detalle).

La pestaña de métricas de pasos

Para cada paso en la transformación, la pestaña Métricas del paso muestra varias columnas de estado e información. Por ahora, las columnas más relevantes en esta pestaña son:

Columna	Valor
Read	Número de filas procedentes de pasos anteriores
Written	Número de filas que salen de este paso hacia el siguiente
Input	Número de filas leídas de un archivo o tabla
Outputs	Número de filas escritas en un archivo o tabla
Errors	Número de errores en la ejecución; Si hay errores, toda la fila se pondrá roja.
Active	Estado actual de la ejecución
Speed (r/s)	La velocidad calculada en filas por segundo

Ejecutando transformaciones de una manera interactiva

Hasta ahora, hemos visto conceptos básicos sobre cómo trabajar con Spoon durante el proceso de diseño. Ahora veremos sobre cómo interactuar con la herramienta.

Primero, crearemos una Transformación, con el objetivo de aprender algunos nuevos pasos útiles. Después de eso, adaptaremos esa Transformación para inspeccionar los datos a medida que se crean, el objetivo es generar un conjunto de datos con todas las fechas entre un rango determinado de fechas:

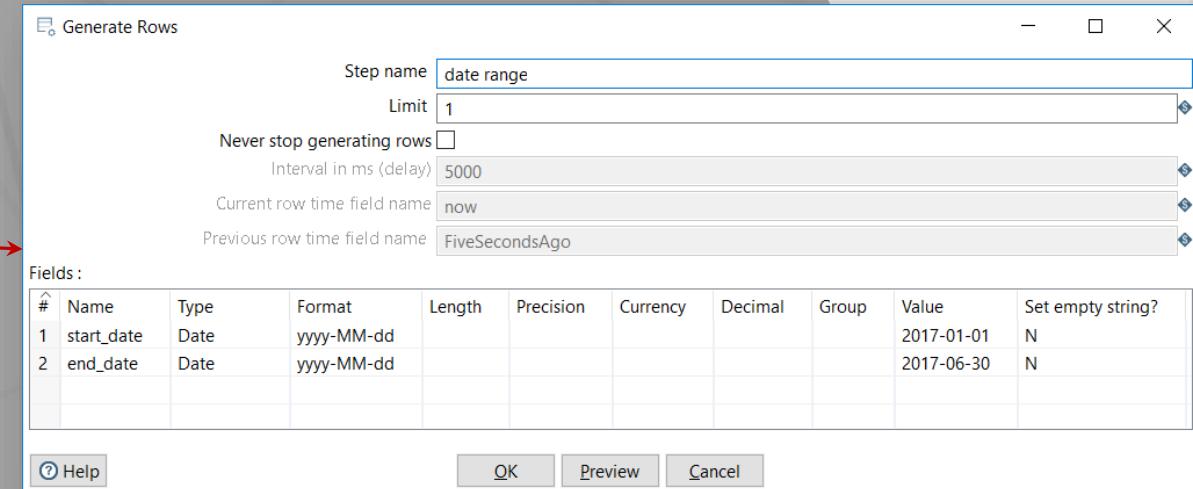
Nota

A medida que avanza, no dude en obtener una vista previa de los datos que se están generando. Esto te ayudará a entender lo que está pasando. Probar cada paso a medida que avanza hace más fácil la depuración y la creación de una transformación funcional.

Nueva transformación

Pasos

1. Creamos una nueva transformación.
2. Desde el grupo de pasos **Input**, arrastramos el paso **Generate Rows** y lo configuramos como se muestra en la imagen. 
3. Cerramos la ventana.
4. En la categoría de pasos **Transform**, agregue el paso **Calculator** y cree un salto que vaya del paso **Generate Rows** a este.
5. Haga doble clic en el paso **Calculator** y agregue el campo **diff_dates** como la diferencia entre **end_date** y **start_date**. Es decir, configúrelo exactamente de la misma manera que lo hizo anteriormente.
6. Ejecutar una vista previa. Debería ver una sola fila con tres campos: la **start date** y **end date** y un campo con el número de días entre ambos.
7. Ahora agregue el paso **Clone rows**. Lo encontrarás dentro del grupo de pasos **Utility**.
8. Crea un salto desde el paso **Calculator** hacia este nuevo paso.
9. Edite el paso **Clone rows**.
10. Seleccione **Nr clone in fields?** Opción para habilitar el cuadro de texto **del campo Nr Clone**. En este cuadro de texto, escriba **diff_dates**.
11. Ahora seleccione **Add clone num to output?** Opción para habilitar el cuadro de texto **Clone num field**. En este cuadro de texto, escriba **delta**.
12. Ejecutar una vista previa. Deberías ver lo siguiente: 



Examine preview data

Rows of step: Clone row (181 rows)

#	start_date	end_date	diff_dates	delta
1	2017-01-01	2017-06-30	180	0
2	2017-01-01	2017-06-30	180	1
3	2017-01-01	2017-06-30	180	2
4	2017-01-01	2017-06-30	180	3
5	2017-01-01	2017-06-30	180	4
6	2017-01-01	2017-06-30	180	5
7	2017-01-01	2017-06-30	180	6
8	2017-01-01	2017-06-30	180	7
9	2017-01-01	2017-06-30	180	8
10	2017-01-01	2017-06-30	180	9

Close

Nueva transformación

Pasos

13. Agregue otro paso **Calculator** y cree un salto desde el paso **Clone row** hasta este.
14. Edite el nuevo paso y agregue el campo **a_single_date**. Como **Calculation**, seleccione **Date A + B Days**. Como Campo A, seleccione **start_date** y como Campo B, seleccione **delta**. Finalmente, como **Value type**, seleccione **Date**. Para el resto de las columnas, deje los valores por defecto.
15. Ejecute la vista previa



Examine preview data

Rows of step: single date (181 rows)

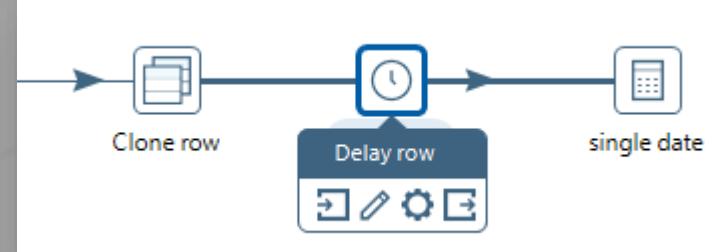
#	start_date	end_date	diff_dates	delta	a_single_date
1	2017-01-01	2017-06-30	180	0	2017-01-01
2	2017-01-01	2017-06-30	180	1	2017-01-02
3	2017-01-01	2017-06-30	180	2	2017-01-03
4	2017-01-01	2017-06-30	180	3	2017-01-04
5	2017-01-01	2017-06-30	180	4	2017-01-05
6	2017-01-01	2017-06-30	180	5	2017-01-06
7	2017-01-01	2017-06-30	180	6	2017-01-07
8	2017-01-01	2017-06-30	180	7	2017-01-08
9	2017-01-01	2017-06-30	180	8	2017-01-09
10	2017-01-01	2017-06-30	180	9	2017-01-10

Close

Nueva transformación

Pasos

1. Edite el paso **Generate Rows** y cambie el rango de fechas. En **end_date**, escriba 2023-12-31.
2. Desde el grupo de pasos **Utility**, arrastre al área de trabajo el paso **Delay row**. Con este paso, retrasaremos deliberadamente cada fila de datos.
3. Arrastre el paso hasta el salto entre el paso **Clone row** y el segundo paso **Calculator**, hasta que el salto cambie el ancho
4. Aparecerá una ventana que le preguntará si desea dividir el salto. Haga clic en **Yes**. El salto se dividirá en dos: uno desde el paso **Clone row** hasta el paso **Delay row**, y el segundo desde este paso hasta el paso **Calculator**.
5. Haga doble clic en el paso **Delay row** y configúrelo con la siguiente información: **TimeOut** en 500 y en la lista desplegable, seleccione **Milliseconds**. Cierre la ventana.
6. Guarde la transformación y ejecútelo. Verás que corre a un ritmo más lento.



Nota

Puede configurar PDI para dividir los saltos automáticamente. Puedes hacerlo seleccionando la opción **Don't ask again?** casilla de verificación en esta misma ventana, o ir a **Tools | Options ...** y marcar la opción “**Automatically Split hops**”.



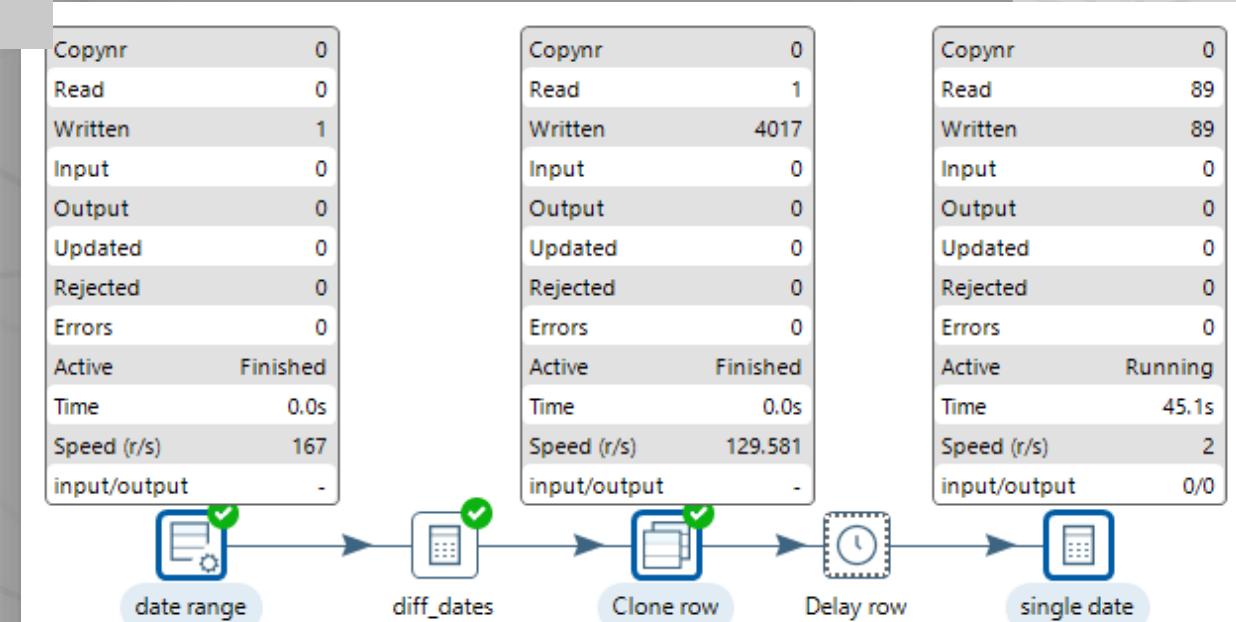
Nueva transformación

Pasos

1. Sin detener la ejecución, haga clic en el segundo paso **Calculator**. Aparecerá una ventana emergente que describe los resultados de la ejecución de este paso en tiempo real. Presione la tecla Ctrl y haga clic en dos pasos más: el paso **Generate Rows** y el paso **Clone row**. Para cada paso seleccionado, verá las métricas de pasos en tiempo de ejecución
2. Ahora, vamos a inspeccionar los datos en sí. Haga clic con el botón derecho en el segundo paso **Calculator** y navegue hasta **Sniff Test During Execution | Sniff Test output rows**. Aparecerá una ventana que muestra los datos a medida que se generan.

Nota

Como alternativa para ejecutar vistas previas en pasos individuales, puede utilizar el modo de vista previa continua. En lugar de ejecutar una vista previa, puede ejecutar la transformación y ver el resultado en la pestaña Datos de vista previa de la ventana Resultados de ejecución.



Entendiendo datos y metadatos de PDI

- Daremos definiciones formales para la terminología básica de PDI relacionada con datos y metadatos.
- También le daremos una lista práctica de pasos que expandirán su caja de herramientas para Transformar datos.

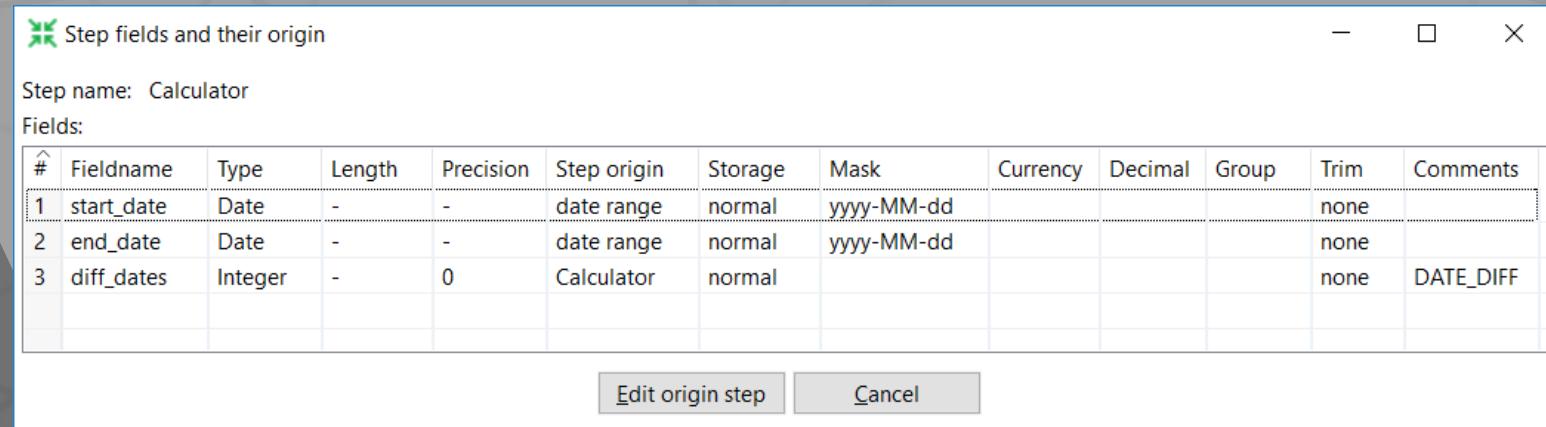
Entendiendo el conjunto de filas PDI

La transformación tiene que ver con conjuntos de datos o conjuntos de filas, es decir, filas de datos con metadatos predefinidos. Los metadatos nos informan sobre la estructura de los datos, es decir, la lista de campos y sus definiciones. La siguiente tabla describe los metadatos de un conjunto de datos PDI:

Elemento de metadatos	Descripción
Field name	Nombre del campo; puede ser cualquier texto (no nulo)
Type	Uno de los tipos admitidos, como, por ejemplo, String o Integer
Length	Longitud del campo
Precision	Aplicable para campos numéricos.
Position	Posición del campo con respecto a los otros campos, comenzando en 1

Entendiendo el conjunto de filas PDI

Para obtener todos los detalles de los metadatos podemos por ejemplo mover el cursor del mouse sobre Calculator y presione la barra espaciadora. Aparecerá una ventana llamada **Step fields and their origin**



Agregar o modificar campos utilizando diferentes pasos de PDI

Una vez que los datos se crean en el primer paso, viajan de un paso a otro a través de los saltos que enlazan esos pasos. La función del salto es solo para dirigir los datos desde un búfer de salida a uno de entrada. La manipulación real de los datos, así como la modificación de un flujo, agregando o eliminando campos, se produce en los pasos. En la última Transformación que creó, usó el paso de la Calculadora para crear nuevos campos y agregarlos a su conjunto de datos.

Agregar o modificar campos utilizando diferentes pasos de PDI

Paso	Descripción	Ejemplo
Add constants	Aggrega uno o más campos con valores constantes.	Si la fecha de inicio era la misma para todos los proyectos, podría agregar ese campo con un paso Add constants .
Add sequence	Aggrega un campo con una secuencia. De forma predeterminada, la secuencia generada será 1, 2, 3 ... pero puede cambiar los valores de inicio, incremento y máximo para generar diferentes secuencias.	Podría haber creado el campo delta con un paso Add sequence en lugar de usar la opción Clone num field en el paso Clone row .
Number range	Crea un nuevo campo basado en rangos de valores. Se aplica a un campo numérico.	Utiliza este paso para crear el campo de rendimiento en función de la duración del proyecto.
Replace in string	Reemplaza todas las apariciones de un texto en un campo de cadena con otro texto.	El valor del campo project_name incluye la palabra project . Con este paso, puede eliminar la palabra o reemplazarla por una más corta. El nombre final para el Project A podría ser Proj A o simplemente A .

Agregar o modificar campos utilizando diferentes pasos de PDI

Paso	Descripción	Ejemplo
Split Fields	Divide un solo campo en dos o más campos nuevos. Tienes que especificar qué personaje actúa como separador.	Divida el nombre del proyecto en dos campos: la primera palabra (que en este caso siempre es Proyecto) y el resto. El separador sería un carácter de espacio.
String operations	Aplica algunas operaciones en cadenas: recortar y eliminar caracteres especiales, entre otros.	Podrías convertir el nombre del proyecto a mayúsculas.
Value Mapper	Crea una correspondencia entre los valores de un campo y un nuevo conjunto de valores.	Podría definir un nuevo campo basado en el campo de rendimiento. El valor podría rechazarse si el rendimiento es bajo o desconocido, y Aprobado para el resto de los valores de rendimiento.
User Defined Java Expression	Crea un nuevo campo utilizando una expresión Java que involucra uno o más campos. Este paso puede eventualmente reemplazar cualquiera de los pasos anteriores.	Usamos este paso en la primera sección para crear dos cadenas: duration y message

Nota

Documentación de los pasos en

https://help.pentaho.com/Documentation/8.0/Products/Data_Integration/Transformation_Step_Reference.

Además, para todos los pasos, hay un útil botón de Ayuda en su ventana de configuración.



Explicando los tipos de datos PDI

Tipo dato PDI	Tipo de dato Java	Descripción
String	java.lang.String	Texto de longitud ilimitada
Integer	java.lang.Long	Un entero largo firmado (64 bits)
Number	java.lang.Double	Un valor de punto flotante de doble precisión.
BigNumber	java.math.BigDecimal	Número de precisión ilimitada
Date	java.util.Date	Un valor de fecha y hora con milisegundos de precisión.
Timestamp	java.sql.Timestamp	Un valor de fecha y hora con nanosegundos de precisión.
Boolean	java.lang.Boolean	Un valor booleano (verdadero / falso, S / N)
Binary	java.lang.byte[]	Una matriz de bytes que contiene cualquier tipo de datos binarios (imágenes, sonidos y otros)
Internet Address	java.net.InetAddress	Una dirección de protocolo de Internet (IP)

Errores de manejo

- Los datos reales tienen errores, es un hecho que no se puede evitar. Si no le presta atención, las transformaciones que se ejecutan con datos de prueba o de muestra probablemente se bloquearán cuando se ejecuten con datos reales.
- En la mayoría de los casos, su trabajo final es ejecutado por un proceso automatizado y no por un usuario de Spoon. Por lo tanto, si una Transformación falla, no habrá nadie que se dé cuenta y reaccione ante esa situación.

Implementando la funcionalidad de manejo de errores.

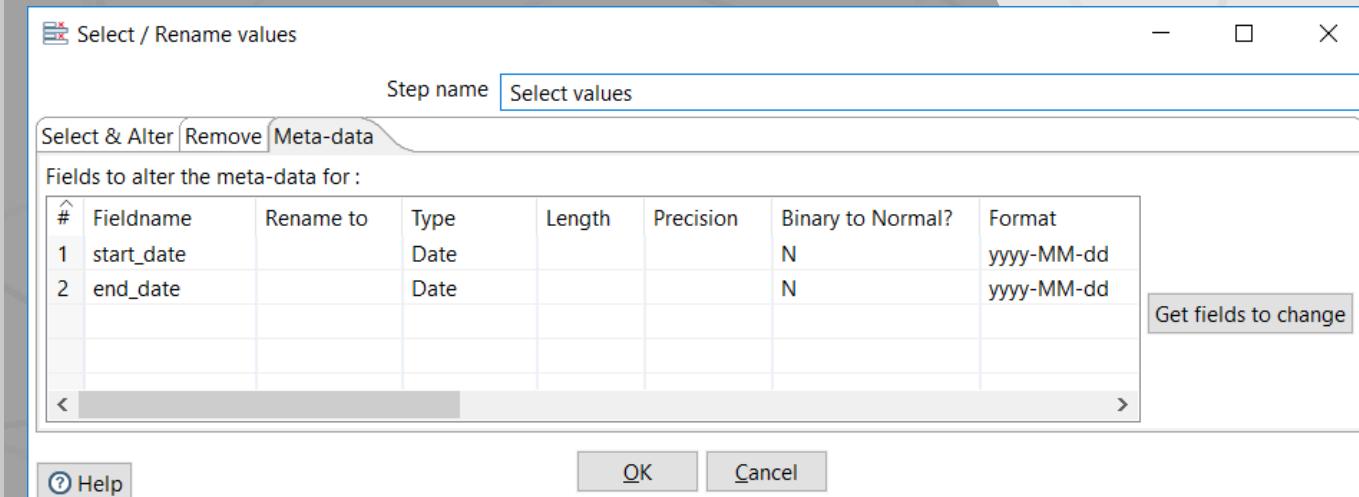
- Con la funcionalidad de manejo de errores, puede capturar errores que de otra manera harían que la Transformación se detuviera. En lugar de abortar, las filas que causan los errores se envían a un flujo diferente para un tratamiento adicional.
- La funcionalidad de manejo de errores se implementa a nivel de paso. No es necesario implementar el manejo de errores en cada paso. De hecho, no puede hacerlo porque no todos los pasos admiten el manejo de errores. El objetivo del manejo de errores es implementarlo en los pasos donde es más probable que tenga errores.

Implementando la funcionalidad de manejo de errores.

Pasos

1. Abra la Transformación de proyectos y guárdelo con un nombre diferente. Puede hacerlo desde el menú principal navegando a **File | Save as ...** o desde la barra de herramientas principal.
2. Edite el paso de **CSV file input** y cambie todos los tipos de datos de **Date** a **String**. Además, elimine los valores en la columna **Format**.
3. Ahora agregue un paso **Select values** e insértelo en el paso **CSV file input** y el paso de la **Calculator**. Lo utilizaremos para convertir los **String** al formato de **Date**.
4. Haga doble clic en el paso **Select values** y seleccione la pestaña Metadatos. Rellene la pestaña de la siguiente manera (ver imagen)
5. Cierre la ventana y ejecute una vista previa. Hay un error en el paso Select Values al intentar convertir el valor no válido:

```
Select values.0 - end_date String<binary-string> : couldn't
convert string [???] to a date using format [yyyy-MM-dd] on
offset location
```



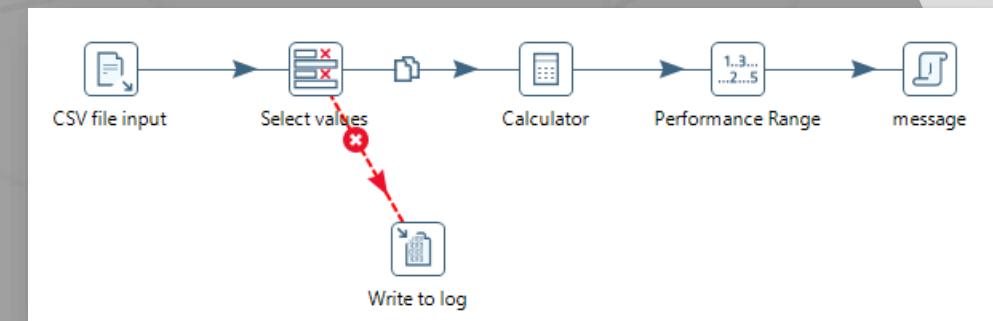
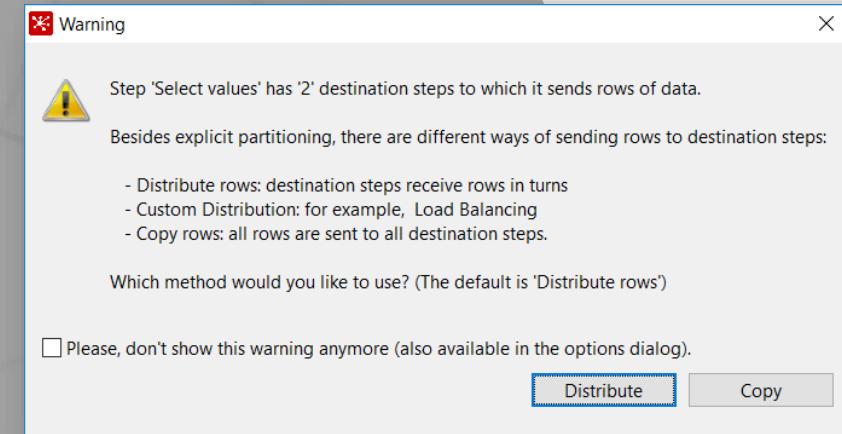
Implementando la funcionalidad de manejo de errores.

Pasos

1. Arrastre al lienzo el paso **Write to log**. Lo encontrarás en la categoría **Utility** de pasos.
2. Cree un nuevo salto desde el paso **Select values** hacia el paso **Write to log**. Cuando se le solicite el tipo de salto que se debe crear, seleccione **Error handling of step**. A continuación, aparecerá una Advertencia
3. Clic en **Copy**
4. Ahora su transformación debe verse como se muestra en la siguiente pantalla de abajo

Nota

Por ahora, no tiene que preocuparse por estas dos opciones ofrecidas. Aprenderá sobre ellos en el Capítulo **Control del flujo de datos**.



Implementando la funcionalidad de manejo de errores.

Pasos

5. Haga doble clic en el paso **Write to log**. En el cuadro de texto de **Write to log**, escriba “**There was an error changing the metadata of a field**”.
6. Haga clic en **Get Fields**. La cuadrícula se llenará con los nombres de los campos que provienen del paso anterior.
7. Cierra la ventana y guarda la transformación.
8. Ahora ejecútalo. Mire la pestaña de Logging en la ventana **Execution Results**. El registro se verá así
9. Ejecutar una vista previa del paso **Calculator**. Verá todas las líneas excepto la línea que contiene la fecha no válida. Esta salida es exactamente la misma que la de la Vista previa de captura de pantalla de una transformación.
10. Ahora ejecute una vista previa en el paso **Write to log**. Solo verá la línea que tenía el valor no válido de fecha de finalización:

```

- Write to log.0 - -----> Line nr 1-----
- Write to log.0 - There was an error changing the metadata of a field
- Write to log.0 -
- Write to log.0 - project_name = Project C
- Write to log.0 - start_date = 2017-01-15
- Write to log.0 - end_date = ???
- Write to log.0 -
- Write to log.0 - =====
- Write to log.0 - Finished processing (I=0, O=0, R=1, W=1, U=0, E=0)

```

 Examine preview data

Rows of step: Write to log (1 rows)

#	project_name	start_date	end_date	error_desc
1	Project C	2017-01-15	???	end_date String<binary-string> : couldn't convert string [???] to a date using format [yyy]

< >

Close

Personalizando el manejo de errores.

Por un lado, PDI le permite agregar nuevos campos a su conjunto de datos que describen los errores:

- Número de errores
- Descripción de los errores.
- Nombre de los campos que causaron los errores.
- Código de error

Nota

 Solo configura el nombre de los campos que contendrán estos valores. Los valores en sí son calculados y establecidos por la herramienta. Usted no define descripciones de errores y códigos; Son internos al PDI.

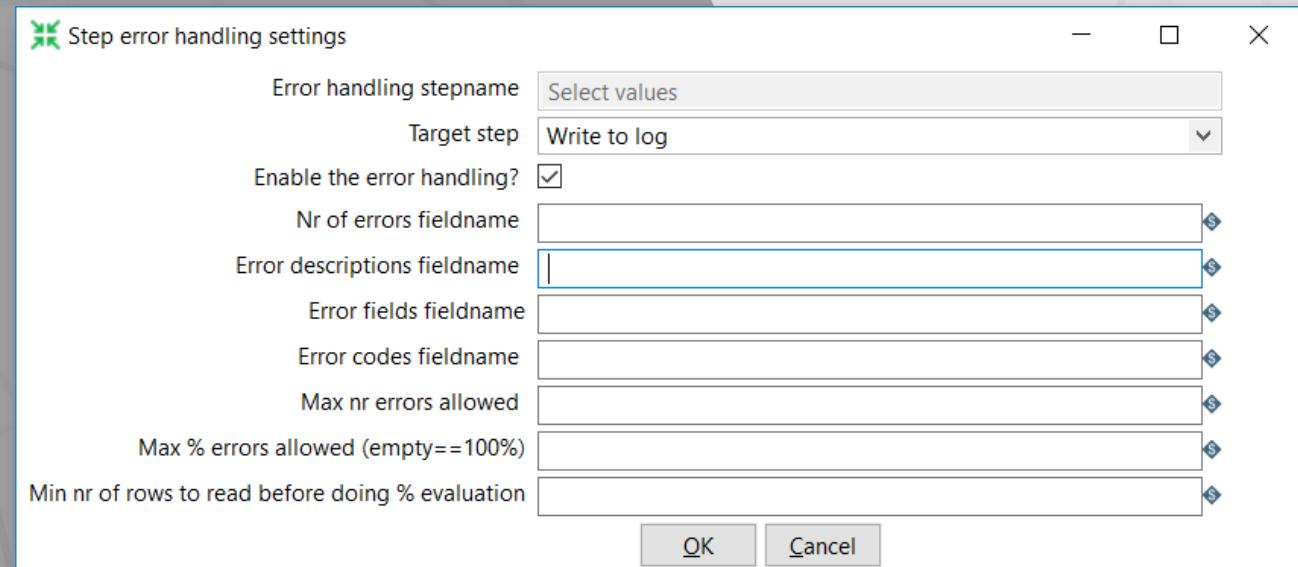
Personalizando el manejo de errores.

- **El número máximo de errores permitido:** Si el número de errores excede este valor, la transformación se cancela. Si este valor está ausente, todos los errores están permitidos.
- Porcentaje máximo de errores permitido: Igual que el punto anterior, pero el umbral para abortar es un porcentaje de filas en lugar de un número absoluto. La evaluación no se hace justo después de leer la primera fila. Junto con esta configuración, debe especificar otro valor: el número mínimo de filas para leer antes de hacer el % de evaluación. Si esta configuración está ausente, no hay control de porcentaje. Como ejemplo, suponga que establece un porcentaje máximo del 20% y un número mínimo de filas para leer antes de realizar la evaluación del porcentaje a 100. Cuando el número de filas con errores excede el 20 por ciento del total, el PDI dejará de capturar los errores y abortará . Sin embargo, este control se realiza solo después de haber procesado 100 filas.

Implementando la funcionalidad de manejo de errores.

Pasos

1. Abra la transformación de la sección anterior.
2. Haga clic con el botón derecho en el paso **Select values** y seleccione **Define Error handling...** Aparecerá la siguiente ventana de diálogo, que le permitirá establecer todas las configuraciones descritas anteriormente:
3. En el cuadro de texto **Error descriptionsfieldname**, escriba **error_desc** y haga clic en **OK**.
4. Haga doble clic en el paso **Write to log** y, después de la última fila, escriba o seleccione **error_desc**.
5. Guarde la transformación y ejecute una vista previa en el paso **Write to log**. Verá un nuevo campo llamado **error_desc** con la descripción del error.
6. Ejecutar la transformación. En la ventana de ejecución, verá lo siguiente



```

- There was an error changing the metadata of a field
- Write to log.0 -
- Write to log.0 - project_name = Project C
- Write to log.0 - start_date = 2017-01-15
- Write to log.0 - end_date = ???
- Write to log.0 - error_desc =
- Write to log.0 -
- Write to log.0 - end_date String<binary-string> : couldn't convert string [????] to a date using format [yyyy-MM-
- Write to log.0 - ???
- Write to log.0 - =====

```

Trabajando con Datos





TEMAS

- **Modelado de datos:** modelado multidimensional, modelado relacional, Streamlined Data Refinery
- **Big Data:** Hadoop, Spark.

Modelado de datos

- Puede refinar sus metadatos relacionales Pentaho y los modelos de datos multidimensionales Mondrian. También puedes aprender a trabajar con big data.
- Estas secciones son útiles si usted es un diseñador de datos, un científico de datos o un desarrollador.

Modelado de datos

Modelado multidimensional

Pentaho Analyzer y Report Designer se basan en el motor de procesamiento analítico en línea (OLAP) de Mondrian, que se basa en un modelo de datos multidimensional.

Modelado relacional

Un modelo de datos relacionales de Pentaho asigna la estructura física de su base de datos a un modelo de negocio lógico.

Streamlined Data Refinery

Streamlined Data Refinery (SDR) aumenta y combina los datos sin procesar a través de un formulario de solicitud y luego los publica para su uso en Analyzer.

Big data

Pentaho es compatible con Hadoop y Spark para todo el proceso de análisis de big data, desde la agregación, preparación e integración de big data hasta la visualización interactiva, el análisis y la predicción.

Hadoop

Pentaho Data Integration (PDI) puede ejecutarse tanto fuera de un clúster Hadoop como dentro de los nodos de un clúster Hadoop.

Spark

PDI puede ejecutar trabajos Spark a través de una entrada Spark Submit o la Capa de ejecución adaptable (AEL).

Modelado Multidimensional de Datos en Pentaho

Pentaho Business Analytics se basa en el motor de procesamiento analítico en línea (OLAP) de Mondrian. OLAP se basa en un modelo de datos multidimensional que, cuando se consulta, devuelve un conjunto de datos que se asemeja a una cuadrícula. Las filas y columnas que describen y dan sentido a los datos en esa cuadrícula son dimensiones, y los valores numéricos en cada celda son las medidas o los hechos. En Pentaho Analyzer, las dimensiones se muestran en amarillo y las medidas en azul.

Modelado Multidimensional de Datos en Pentaho

OLAP requiere un origen de datos adecuadamente preparado en forma de estrella o esquema de copo de nieve que defina una base de datos lógica multidimensional y la asigne a un modelo de base de datos física. Una vez que tenga su estructura de datos inicial en su lugar, debe diseñar una capa descriptiva para ella en forma de un esquema de Mondrian, que consta de uno o más cubos, jerarquías y miembros. Solo cuando tiene un esquema Mondrian probado y optimizado, sus datos se preparan en un nivel básico para las herramientas de usuario final como Pentaho Analyzer.

Modelado Multidimensional de Datos en Pentaho

Pentaho también ofrece una funcionalidad ampliada para los clientes de Pentaho Analysis Enterprise Edition, que incluye:

- La herramienta de visualización Pentaho Analyzer.
- Un Enterprise Cache conectable con soporte para implementaciones de caché distribuibles y altamente escalables, incluyendo Infinispan y Memcached.

El uso de estas funciones requiere una licencia de Pentaho Analysis Enterprise Edition instalada en el servidor Pentaho y estaciones de trabajo que tengan Schema Workbench y Metadata Editor.

Nota

También se debe instalar un paquete especial de Servidor Pentaho; Este proceso está cubierto en la documentación de instalación.

https://help.pentaho.com/Documentation/8.1/Setup/Installation/Tools/BA_Design_Tools



Diseña una estrella o un esquema de copo de nieve

Use las notas que tomó durante la fase de prueba para rediseñar adecuadamente su almacén de datos y el esquema de Mondrian. Ajustar jerarquías y métodos de agregación de medidas relacionales. Cree cubos virtuales para analizar múltiples tablas de hechos por dimensiones conformes. Vuelva a probar la nueva implementación y continúe refinando el modelo de datos hasta que se adapte perfectamente a las necesidades de su negocio.

Poblar esquemas estrella / copo de nieve

Una vez que su modelo de datos está diseñado, el siguiente paso es rellenarlo con datos reales, creando así su almacén de datos. La mejor herramienta para este trabajo es Pentaho Data Integration, una aplicación de extracción, transformación y carga (ETL) de nivel empresarial.

Construir un esquema de Mondrian

Ahora que su proyecto de almacenamiento de datos inicial está completo, debe crear un esquema de Mondrian para organizarlo y describirlo en términos que Pentaho Analysis pueda entender. Puede utilizar Pentaho Schema Workbench para crear un esquema de análisis.

Pruebas iniciales

En este punto, debe tener una estructura de datos multidimensional con una capa de metadatos adecuada. Ahora puede comenzar a utilizar las herramientas de inspección de datos para profundizar en sus datos y ver si su primer intento de modelado de datos fue exitoso. Con toda probabilidad, necesitará algún ajuste, así que tome nota de todas las limitaciones del esquema con las que no está satisfecho durante esta fase de prueba inicial. No se preocupe por los problemas de rendimiento en este momento; solo concéntrese en la integridad y la exhaustividad del modelo de datos.

Ajustar y repetir hasta que esté satisfecho

En este punto, debe tener una estructura de datos multidimensional con una capa de metadatos adecuada. Ahora puede comenzar a utilizar las herramientas de inspección de datos para profundizar en sus datos y ver si su primer intento de modelado de datos fue exitoso. Con toda probabilidad, necesitará algún ajuste, así que tome nota de todas las limitaciones del esquema con las que no está satisfecho durante esta fase de prueba inicial. No se preocupe por los problemas de rendimiento en este momento; solo concéntrese en la integridad y la exhaustividad del modelo de datos.

Prueba de rendimiento

Una vez que esté satisfecho con el diseño y la implementación de su modelo de datos, debe tratar de encontrar problemas de rendimiento y resolverlos ajustando la base de datos del almacén de datos y creando tablas de agregación. La prueba solo puede hacerse razonablemente a mano, utilizando el analizador Pentaho. Tome nota de todas las medidas que toman un tiempo irrazonablemente largo para calcular. Además, habilite el registro de SQL y localice las consultas de rendimiento lento, y cree índices para optimizar el rendimiento de las consultas.

Crear tablas de agregación

Utilizando sus notas como guía, cree tablas de agregación en Pentaho Aggregation Designer para almacenar los informes de Analizador calculados con frecuencia. Vuelva a probar y cree nuevas tablas de agregación según sea necesario. Si está trabajando con un almacén de datos relativamente pequeño o con un número limitado de dimensiones, es posible que no tenga una necesidad real de tablas de agregación. Sin embargo, tenga en cuenta la posibilidad de que surjan problemas de rendimiento en el futuro. Consulte con sus usuarios de vez en cuando para ver si tienen alguna inquietud particular sobre la velocidad de su contenido de BI.

Implementar en producción

Su almacén de datos y el esquema de Mondrian han sido creados, probados y refinados. Ahora estás listo para poner todo en producción.

Modelado dimensional

Con la estructura de datos inicial en su lugar, puede utilizar el modelado dimensional para diseñar una capa descriptiva. El modelado dimensional es el proceso de transformación de datos de múltiples fuentes en formatos no amigables para el ser humano en una única fuente de datos que está organizada para soportar análisis de negocios. A continuación se muestra un flujo de trabajo típico para desarrollar un modelo dimensional

Modelado dimensional

- Reúne los requisitos del usuario para la lógica de negocios y procesos.
- Teniendo en cuenta la totalidad de sus datos, divídaluos por temas.
- Aislar grupos de hechos en una o más tablas de hechos.
- Diseñar tablas dimensionales que dibujen relaciones entre niveles (grupos de hechos).
- Determine qué miembros de cada nivel son útiles para cada tabla dimensional.
- Cree y publique un esquema de Mondrian (Pentaho Analysis) y recopile los comentarios de los usuarios.
- Refine su modelo según los comentarios de los usuarios, continúe iterando a través de esta lista hasta que los usuarios sean productivos.

Modelado dimensional

- Reúne los requisitos del usuario para la lógica de negocios y procesos.
- Teniendo en cuenta la totalidad de sus datos, divídaluos por temas.
- Aislar grupos de hechos en una o más tablas de hechos.
- Diseñar tablas dimensionales que dibujen relaciones entre niveles (grupos de hechos).
- Determine qué miembros de cada nivel son útiles para cada tabla dimensional.
- Cree y publique un esquema de Mondrian (Pentaho Analysis) y recopile los comentarios de los usuarios.
- Refine su modelo según los comentarios de los usuarios, continúe iterando a través de esta lista hasta que los usuarios sean productivos.

Modelado dimensional

O, expresado como una serie de preguntas:

- ¿Qué tema o temas son importantes para los usuarios que están analizando los datos?
¿Qué necesitan tus usuarios aprender de los datos?
- ¿Cuáles son los detalles importantes que sus usuarios necesitarán examinar en los datos?
- ¿Cómo debe relacionarse cada columna de datos con otras columnas de datos?
- ¿Cómo se deben agrupar y organizar los conjuntos de datos?
- ¿Cuáles son algunas descripciones breves útiles para cada nivel dimensional en una jerarquía (para cada elemento, decida qué es útil dentro de ese elemento; por ejemplo, en una tabla dimensional que representa el tiempo, sus niveles pueden ser año, mes y día, y sus miembros) para el año el nivel podría ser 2003, 2004, 2005).
- ¿Qué tan efectivo es este modelo dimensional para la base de usuarios previstos? ¿Cómo puede ser mejorado?

Modelado dimensional

Pentaho Data Integration ofrece herramientas de inspección de datos para que el modelado dimensional sea mucho más fácil que los métodos más tradicionales. A través de PDI, puede ajustar rápidamente su lógica empresarial, la granularidad de sus tablas de hechos y los atributos de sus tablas de dimensiones, luego generar un nuevo modelo y enviarlo a un entorno de prueba para su evaluación.

Entendiendo los cubos de datos

Otro nombre para un modelo dimensional es un cubo. Cada cubo representa una tabla de hechos y varias tablas dimensionales. Este modelo debería ser útil para informar y analizar el tema de los datos en la tabla de hechos. Sin embargo, si desea realizar una referencia cruzada de estos datos con otro cubo, si necesita analizar datos en dos o más cubos, o si necesita combinar información de dos tablas de hechos sobre el mismo tema pero con una granularidad diferente, debe crear un cubo virtual. Los elementos XML que componen un cubo virtual se explican en detalle a continuación.

Nota

Los cubos virtuales no se pueden crear actualmente a través de la perspectiva del modelo de Pentaho Data Integration; debes usar Schema Workbench en su lugar.

Entendiendo los cubos de datos

El elemento **<CubeUsages>** especifica los cubos que se importan en el cubo virtual. Contiene elementos **<CubeUsage>**.

El elemento **<CubeUsage>** especifica el cubo base que se importa al cubo virtual. Alternativamente, puede definir un **<VirtualCubeMeasure>** y usar importaciones similares desde el cubo base sin definir un **<CubeUsage>**. El atributo **cubeName** especifica el nombre del cubo base. El atributo **ignoreUnrelatedDimensions** determina si las medidas de este cubo base tendrán miembros de dimensión sin unión empujados al miembro de nivel superior. Este atributo es falso por defecto porque todavía es experimental.

Entendiendo los cubos de datos

El elemento **<VirtualCubeDimension>** importa una dimensión de uno de los cubos constituyentes. Si no especifica el atributo **cubeName**, esto significa que está importando una dimensión compartida.

El elemento **<VirtualCubeMeasure>** importa una medida de uno de los cubos constituyentes. Se importa con el mismo nombre. Si desea crear una fórmula o cambiar el nombre de una medida a medida que la importa, use el elemento **<CalculatedMember>** en su lugar.

Nota

Si una dimensión compartida se usa más de una vez en un cubo, no hay forma de determinar qué uso de la dimensión compartida pretende importar.

Entendiendo los cubos de datos

Los cubos virtuales son útiles para situaciones en las que hay tablas de hechos de diferentes granularidades (por ejemplo, una tabla de datos de Tiempo puede configurarse en un nivel de Día, otra en el nivel de Mes), o tablas de hechos de diferentes dimensionalidades (por ejemplo, una de productos, tiempo y cliente, otro sobre productos, tiempo y almacén), y debe presentar los resultados a los usuarios que no saben cómo están estructurados los datos.

Todas las dimensiones comunes (dimensiones compartidas que utilizan los dos cubos constituyentes) se sincronizan automáticamente. En este ejemplo, [Tiempo] y [Productos] son dimensiones comunes. De modo que si el contexto es ([Tiempo]. [2005]. [P2], [Productos]. [Nombre del producto]. [P-51-D Mustang]), las medidas de cada cubo se relacionarán con este contexto.

Entendiendo los cubos de datos

Las dimensiones que solo pertenecen a un cubo se denominan dimensiones no conformes. La dimensión [Género] es un ejemplo de esto; existe en el cubo Ventas, pero no en Almacén. Si el contexto es ([Género]. [F], [Tiempo]. [2005]. [P1]), tiene sentido preguntar el valor de la medida [Ventas de unidades] (que proviene del cubo [Ventas]) pero no la medida [Unidades ordenadas] (de [Almacén]). En el contexto de [Género]. [F], [Unidades ordenadas] tiene un valor NULL.

Mapear un modelo con Schema Workbench

Con un modelo de datos multidimensional físico en su lugar, debe crear un modelo lógico que se asigne a él. Un esquema de Mondrian es esencialmente un archivo XML que realiza este mapeo, definiendo así una estructura de base de datos multidimensional. Puede crear esquemas Mondrian utilizando Pentaho Schema Workbench.

Configurar motor Mondrian

El motor Pentaho Analysis (Mondrian) se puede configurar a través de un archivo de propiedades. Las opciones de Mondrian permiten un mejor rendimiento y funcionalidad del motor y la fuente de datos bajo ciertas condiciones.

Mondrian Cache Control

Puede configurar y controlar la infraestructura de caché que utiliza el motor de análisis de Pentaho para los datos OLAP.

La mayoría de las funciones avanzadas de caché que se explican son solo para implementaciones de Enterprise Edition. Dentro de eso, la mayoría de las características de Enterprise Edition del motor de análisis solo son beneficiosas para implementaciones grandes de OLAP de múltiples nodos que tienen un bajo rendimiento.

Modelado de datos relacionales en Pentaho

Pentaho permite construir dominios de metadatos y modelos de datos relacionales. Un modelo de metadatos de Pentaho asigna la estructura física de su base de datos a un modelo de negocio lógico. Estas asignaciones se almacenan en un repositorio de metadatos centralizado y permiten a los administradores:

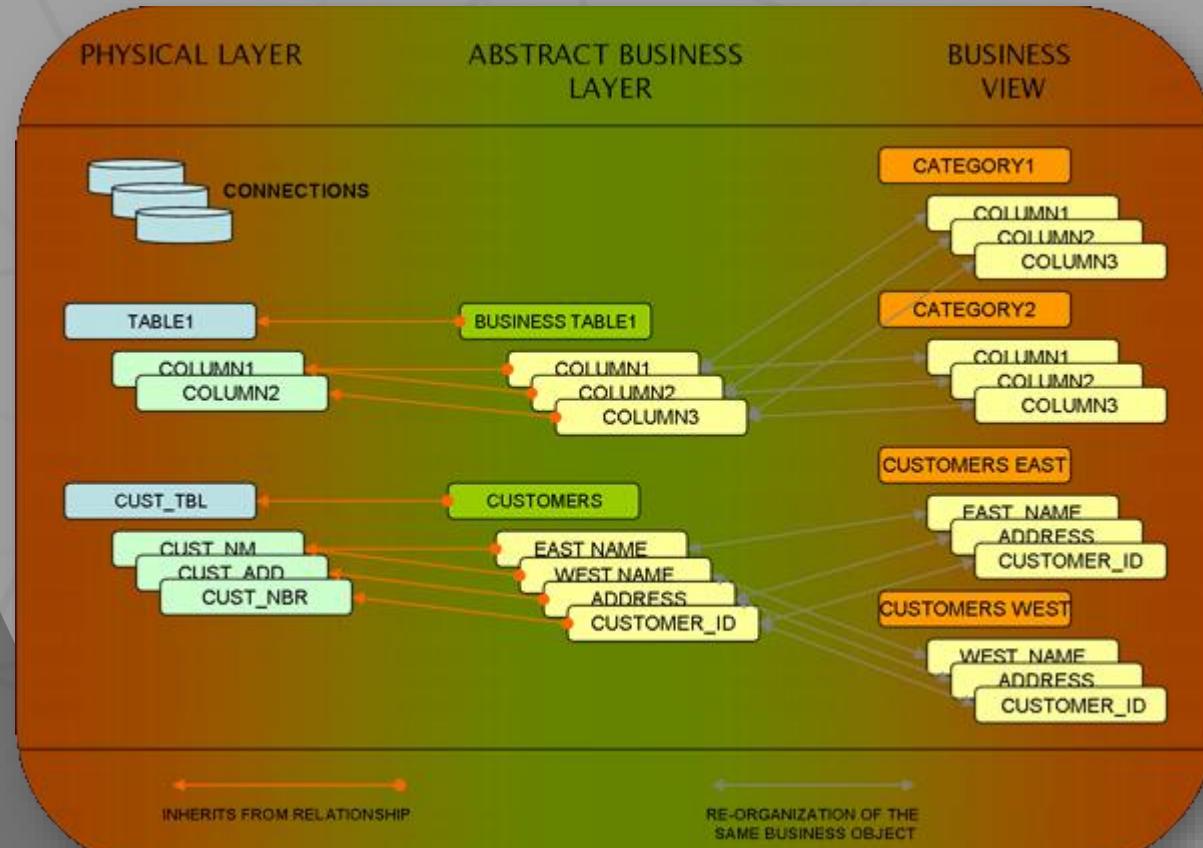
- Crear definiciones de lenguaje empresarial para tablas de base de datos complejas o crípticas
- Disminuir el costo y el impacto asociado con los cambios en la base de datos de bajo nivel
- Establecer los parámetros de seguridad que limitan el acceso del informe a los datos del usuario.
- Formato de la unidad en texto, fecha y datos numéricos que mejoran el mantenimiento del informe
- Localiza la información a la configuración regional del usuario.

Modelado de datos relacionales en Pentaho

El objetivo del modelado de datos relacionales en Pentaho es simplificar la experiencia de los usuarios de negocios cuando crean informes.

El modelo de negocio de metadatos es en realidad un componente importante en un dominio de metadatos de Pentaho. El dominio encapsula tanto las descripciones físicas de los objetos de su base de datos como el modelo lógico (el modelo de negocio), la representación abstracta de la base de datos.

Modelado de datos relacionales en Pentaho



La capa física

La capa física de un dominio Pentaho abarca conexiones, tablas físicas y columnas físicas. Estos objetos representan las bases de datos que intenta modelar y enriquecer con metadatos. La capa física no se considera parte del modelo de negocio, porque no todas las conexiones definidas en la capa física se utilizarán en todos los modelos de negocio.

La vista de negocios

Business View es la parte del modelo de negocio con el que las aplicaciones operarán y los usuarios finales verán. La vista empresarial no es más que "grupos" (denominados categorías) para que usted pueda reorganizar y reorganizar las columnas de su negocio de una manera que tenga sentido para los consumidores de los datos.

La capa abstracta de negocios

La Capa de negocios abstracta es el levantador pesado en el modelo de negocio de metadatos. El modelo de negocio abarca la capa de negocio abstracta y la vista de negocio. En la Capa de negocio abstracta, tiene tablas de negocios, columnas de negocios y relaciones comerciales.

Puede crear tablas de negocios para cualquier tabla física que haya definido en la capa física. También puede crear más de una tabla de negocios para hacer referencia a la misma tabla física. Las mismas reglas se aplican a las columnas de negocios. Esto puede ser útil en una multitud de escenarios, como ejemplo, filtrar la seguridad o incluso datos en este nivel

Incorporar metadatos

Cada objeto de negocio en el dominio puede tener metadatos asociados con él, con la excepción de las categorías. En la terminología de Pentaho, una colección de propiedades de metadatos se denomina concepto.

Cada objeto de negocio puede tener tres niveles de conceptos: propio (concepto propio o secundario), un concepto principal y un concepto heredado. Todas las propiedades de metadatos definidos de los tres niveles están disponibles (para aplicaciones conscientes de metadatos, usuarios finales) en un objeto comercial. Es importante entender qué es la jerarquía de anulación, cuando más de un nivel de concepto ha definido la misma propiedad de metadatos.

Construye tus modelos con el editor de metadatos de Pentaho

Use el Editor de metadatos de Pentaho para construir sus dominios y modelos de metadatos. Hay datos de muestra disponibles si desea probar el Editor de metadatos antes de importar sus propios datos. Esta muestra de datos se incluye en la descarga de Pentaho.

Para usar el Editor de metadatos de Pentaho, debe tener habilidades de administrador de base de datos (DBA). Debe saber cómo realizar las siguientes tareas: importar tablas, crear relaciones entre tablas, asignar agregaciones, agregar categorías y asignar seguridad. Debe tener un amplio conocimiento de sus bases de datos y saber qué tipo de datos desean los usuarios de negocios. Estas habilidades le permitirán asignar un modelo de usuario empresarial (modelo lógico) a una base de datos relacional compleja. Esto permite a sus usuarios crear informes de Pentaho sin la ayuda de un DBA.

Nota

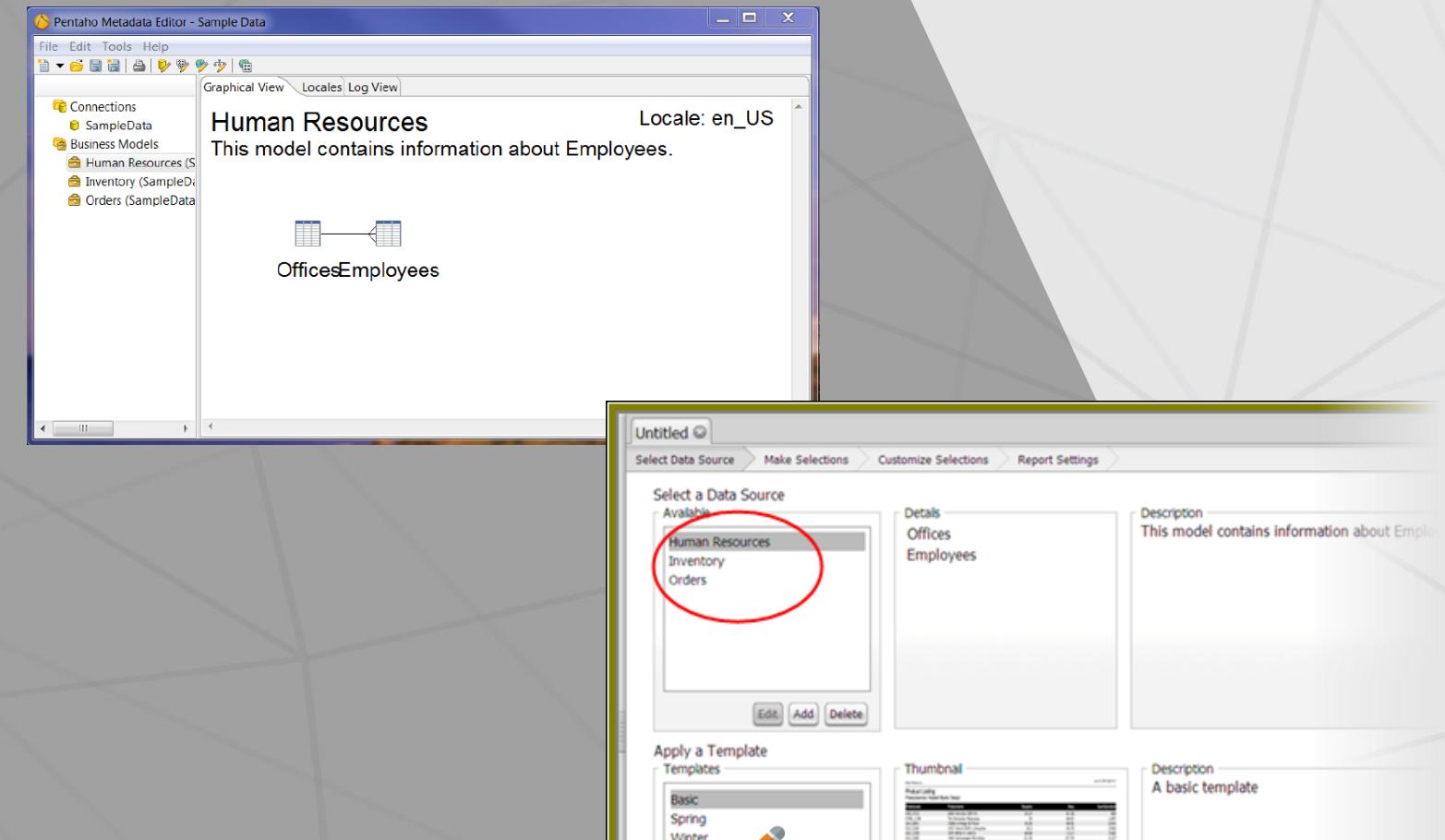
Una descripción conceptual del modelado de datos relacionales

Los metadatos en Pentaho se basan en el modelado de datos relacionales, que mapea la estructura física de su base de datos en un modelo de negocio lógico.

Comienza con el editor de metadatos de Pentaho

Pasos

1. En la ventana principal del Editor de metadatos, vaya a Archivo> Importar desde archivo XMI.
2. Navegue a su instalación del servidor Pentaho. Por ejemplo, si usó el instalador, navegue a pentaho / server / pentaho-server / pentaho-solutions. Puede haber varias carpetas de solución pentaho raíz y cada una de ellas puede contener un repositorio de modelo de metadatos.
3. Abra la carpeta de ruedas de acero y haga clic en [MyBusinessModel] .xmi, donde [myBusinessModel] es el nombre que le ha dado al modelo. El archivo .xmi es el repositorio de los metadatos relacionados con Pentaho y las vistas empresariales.
4. Escriba Datos de muestra en el cuadro de diálogo Guardar modelo. Este paso procesa el archivo y muestra la estructura del repositorio (conexiones y modelos de negocio) en el panel de navegación de la izquierda. Si ve el mensaje "Este modelo ya existe ...", haga clic en Sí para continuar.

The screenshot illustrates the initial setup of a metadata model in the Pentaho Metadata Editor. The main window shows a 'Human Resources' business model with a description: 'This model contains information about Employees.' Below the main window, a smaller screenshot of the Pentaho Reporting interface shows the 'Select a Data Source' dialog, where the 'Human Resources' option is highlighted with a red circle, indicating it is the selected data source for a report.

Pasos

5. Inicie sesión en la Consola de usuario y haga clic en Nuevo informe. (Se muestran los modelos de negocio Recursos humanos, Inventario y Pedidos.)



Crear un dominio

Un dominio de metadatos es un término de Pentaho que representa todos los objetos comerciales creados, almacenados y utilizados en la capa de metadatos. Un dominio puede constar de una o más conexiones, uno o más modelos, información de seguridad, tablas de negocios, vistas de negocios, categorías, columnas y conceptos. Puede crear y guardar múltiples dominios de metadatos utilizando el Editor de metadatos.

Aplicar propiedades y conceptos de metadatos

Cuando creó su dominio, modeló su base de datos de una manera más intuitiva que su representación física. A continuación, debe definir sus metadatos. El paradigma de metadatos de Pentaho utiliza el término concepto para significar una colección de propiedades de metadatos que se pueden aplicar a un objeto de negocio determinado (tabla de negocios o columna, por ejemplo).

Publicar un dominio

Puede compartir una representación XML de su dominio con una solución Pentaho que su servidor Pentaho reconozca, de modo que el servidor pueda acceder al dominio de metadatos y su contenido.

Importar y exportar dominios

Cuando guarda un dominio, se almacena en un repositorio de metadatos. El servidor Pentaho no utiliza el repositorio de metadatos. En su lugar, accede a un archivo XML exportado desde el editor de metadatos de Pentaho. Exportar su dominio es una buena manera de garantizar copias de seguridad seguras de sus dominios. Cuando importa un nuevo dominio, se convierte en el dominio activo en el Editor de metadatos de Pentaho.

Realice los siguientes pasos para importar un dominio:

1. En el Editor de metadatos de Pentaho, vaya a Archivo> Importar desde archivo XMI.
2. En el explorador de archivos, seleccione su archivo de dominio y haga clic en Aceptar.
3. En el cuadro de diálogo Guardar modelo, escriba un nombre para el dominio. Si ingresa el nombre de un dominio existente, la importación sobrescribe ese dominio.

Nota

Es posible que se le solicite que guarde el modelo actualmente activo si tiene algún cambio pendiente de guardar.

Importar y exportar dominios

Realice los siguientes pasos para exportar un dominio:

1. En el Editor de metadatos, vaya a Archivo> Exportar a archivo XMI.
2. Escriba un nombre de archivo y seleccione una ubicación para guardar su archivo. La extensión predeterminada para un archivo XML de dominio de metadatos es .xmi.
3. Clic en Guardar. Una vez que haya ingresado un nombre para su archivo de exportación, el dominio se exporta a ese archivo. Puede inspeccionar el archivo de exportación utilizando un editor de texto para ver el código XML subyacente.

Copia de seguridad y recuperación de dominio

Cada dominio se puede guardar en el repositorio de Metadatos de Almacén Común (CWM) con el nombre que desee. Las opciones Guardar y Guardar como están disponibles tanto en el menú Archivo del editor de metadatos de Pentaho como en la barra de herramientas. Cada vez que se guarda un dominio en el repositorio, se guarda un archivo de exportación de recuperación del dominio en el sistema de archivos, bajo el directorio .pentaho-meta. Este directorio se encuentra normalmente en el directorio de inicio. El archivo de recuperación contiene el último estado guardado con éxito del dominio. Los archivos se denominan recovery_[studio: domainname].xmi.

La restauración del dominio puede ser necesaria si necesita volver al último buen estado conocido de su dominio en caso de errores en el repositorio o daños. Puede restaurar un dominio al último estado guardado importando un archivo de recuperación.

Trabajando con la Streamlined Data Refinery

La refinería de datos optimizada (SDR) es una refinería ETL simplificada y específica compuesta por una serie de trabajos de integración de datos Pentaho (PDI) que toman datos sin procesar, aumentan y combinan a través del formulario de solicitud y luego lo publican para que los diseñadores de informes lo utilicen en Analizador.

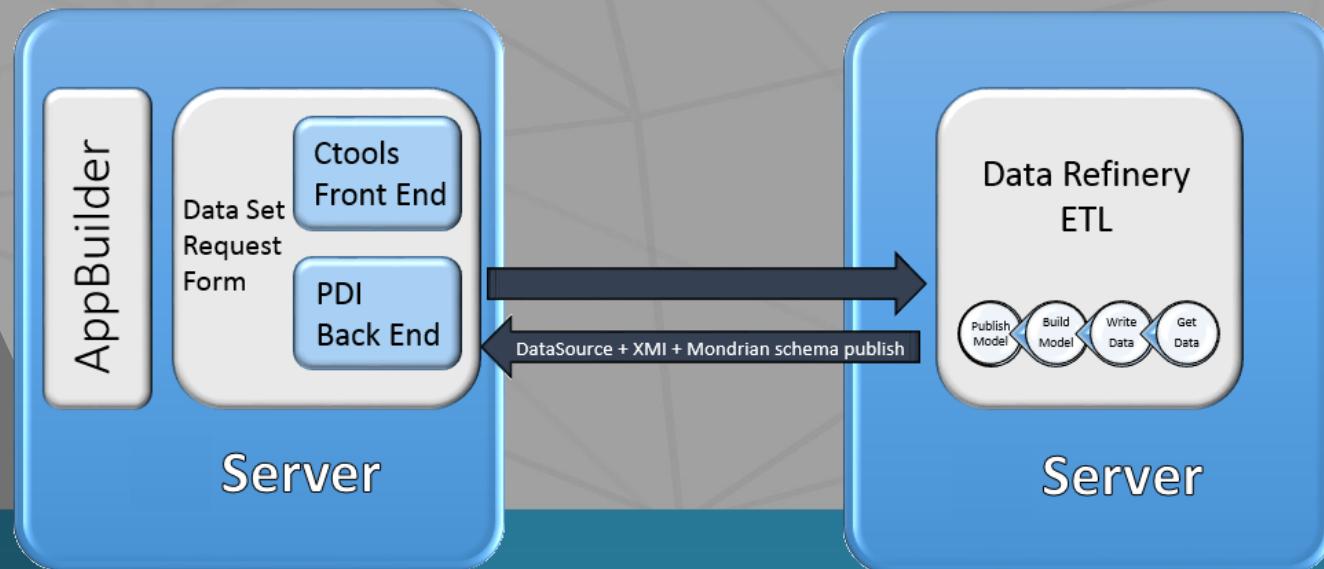
Nota

Ejemplo de refinería:
<https://pentaho.app.box.com/SDRSample60>

Esta muestra fue desarrollada por Pentaho y se basa en CTools.

¿Cómo funciona el SDR?

Los componentes que forman la refinería de datos son PDI, que se utiliza para la entrada de parámetros, y funciona en conjunto con una aplicación para refinar los datos. Esta aplicación llama al servidor Pentaho para el trabajo principal: refina los datos a través de Spoon utilizando la nueva entrada de trabajo [Build Model](#), y luego publica la fuente de datos en el servidor Pentaho a través de la entrada del trabajo [Publish Model](#). Una vez que se publica, los datos refinados están disponibles para su uso en la creación de informes de Analyzer. Este proceso se muestra en el gráfico.



App Builder, Community Dashboard Editor y CTools

App Builder es un generador de aplicaciones para personas que pueden no tener conocimientos de Java, pero que pueden tener muchas ideas interesantes para nuevos complementos. Todo lo que se requiere para usar App Builder es conocimiento de CTools y PDI.

El Editor del panel de la comunidad (CDE), cuando se integra con la Consola de usuario de Pentaho (PUC), simplifica el proceso de creación, refinamiento y vista previa de los paneles de Pentaho. Puede utilizar CDE para diseñar paneles, ya sea desde cero o utilizando una plantilla.

Hadoop con Pentaho

Pentaho proporciona una solución completa de análisis que soporta todo el proceso de análisis de big data. Desde la agregación, la preparación y la integración de big data hasta la visualización interactiva, el análisis y la predicción, Pentaho le permite cosechar los patrones significativos ocultos en grandes almacenes de datos. El análisis de sus grandes conjuntos de datos le brinda la capacidad de identificar nuevas fuentes de ingresos, desarrollar relaciones leales y rentables con los clientes y administrar su organización de manera más eficiente y rentable.

Nota

Las soluciones de Big Data rediseñan los componentes de las bases de datos tradicionales (almacenamiento de datos, recuperación, consulta, procesamiento) y los escala de forma masiva.

Resumen de Pentaho Big Data

Pentaho aumenta el análisis de la velocidad de pensamiento incluso en los almacenes de big data más grandes al centrarse en las características que ofrecen rendimiento.

- **Acceso instantáneo:** Pentaho proporciona herramientas visuales para facilitar la definición de los conjuntos de datos que son importantes para el análisis interactivo. Estos conjuntos de datos y los análisis asociados pueden compartirse fácilmente con otros y, a medida que surgen nuevas preguntas comerciales, se pueden definir nuevas vistas de datos para el análisis interactivo.
- **Plataforma de alto rendimiento:** Pentaho se basa en una plataforma moderna, ligera y de alto rendimiento. Esta plataforma aprovecha al máximo los procesadores de múltiples núcleos de 64 bits y los grandes espacios de memoria para aprovechar de manera eficiente la potencia del hardware contemporáneo.

Resumen de Pentaho Big Data

- **Caché en memoria a gran escala:** Pentaho es único en el aprovechamiento de tecnologías de cuadrícula de datos externa, como Infinispan y Memcached, para cargar grandes cantidades de datos en la memoria de modo que esté disponible al instante para el análisis de velocidad de pensamiento.
- **Integración de datos federados:** los datos se pueden extraer de múltiples fuentes, incluidos los almacenes de datos tradicionales y de datos grandes, se pueden integrar y luego fluir directamente a los informes, sin necesidad de un almacén de datos empresarial o un centro de datos.

Acerca de Hadoop

- The Apache Hadoop software library es un framework que permite el procesamiento distribuido de grandes conjuntos de datos en grupos de computadoras utilizando modelos de programación simples. Está diseñado para escalar desde servidores individuales a miles de máquinas, cada una ofrece computación y almacenamiento locales. En lugar de confiar en el hardware para ofrecer alta disponibilidad, la biblioteca en sí está diseñada para detectar y manejar fallas en la capa de aplicación, por lo que ofrece un servicio de alta disponibilidad sobre un grupo de computadoras, cada una de las cuales puede ser propensa a fallas.
- Hadoop consta de un kernel Hadoop, un modelo MapReduce, un sistema de archivos distribuidos y, a menudo, una serie de proyectos relacionados, como Apache Hive, Apache HBase y otros.
- Un sistema de archivos distribuidos de Hadoop, comúnmente conocido como HDFS, es un sistema de archivos basado en Java, distribuido, escalable y portátil para el marco de Hadoop.

Comience con Hadoop y PDI

- Pentaho Data Integration (PDI) puede funcionar en dos modos distintos, orquestación de trabajos y transformación de datos. Dentro de PDI se les conoce como trabajos y transformaciones.
- Los trabajos PDI secuencian un conjunto de entradas que encapsulan acciones. Un ejemplo de un trabajo de datos grandes de PDI sería verificar la existencia de nuevos archivos de registro, copiar los nuevos archivos a HDFS, ejecutar una tarea de MapReduce para agregar el weblog en una secuencia de clics y organizar los datos del flujo de clics en una base de datos analítica.
- Las transformaciones de PDI consisten en un conjunto de pasos que se ejecutan en paralelo y operan en un flujo de columnas de datos. A través del motor Pentaho predeterminado, las columnas generalmente fluyen desde un sistema donde se pueden calcular nuevas columnas o se pueden buscar valores y agregarlos a la secuencia. El flujo de datos se envía a un sistema receptor como un clúster de Hadoop, una base de datos o incluso el motor de informes de Pentaho.

Comience con Hadoop y PDI

- También puedes ejecutar transformaciones usando el motor Spark. Pentaho utiliza la Capa de ejecución adaptable (AEL) para ejecutar transformaciones en diferentes motores. AEL crea una definición de transformación para Spark, que traslada la ejecución directamente al clúster, aprovechando la capacidad de Spark para coordinar grandes cantidades de datos en múltiples nodos. Ver Capa de ejecución adaptable para más detalles.

Plugin de Big Data de PDI

El complemento Pentaho Big Data contiene todas las entradas de trabajo y los pasos de transformación necesarios para trabajar con Hadoop, Cassandra y MongoDB. PDI se puede configurar para comunicarse con las distribuciones de Hadoop más populares. Consulte la sección Configurar Pentaho para conectarse a Hadoop Cluster para obtener más información. Para obtener una lista de la tecnología de big data admitida, incluida la configuración actual de Hadoop, consulte la Referencia de componentes.

Usando Spark con PDI

Puede ejecutar un job con Spark utilizando **Spark Submit** o ejecutar una transformación PDI en Spark a través de [run configuration](#).

Instala el cliente Spark

Antes de comenzar, debe instalar y configurar el cliente Spark de acuerdo con las instrucciones en la entrada Spark Submit job, que puede encontrar aquí: [Spark Submit](#).

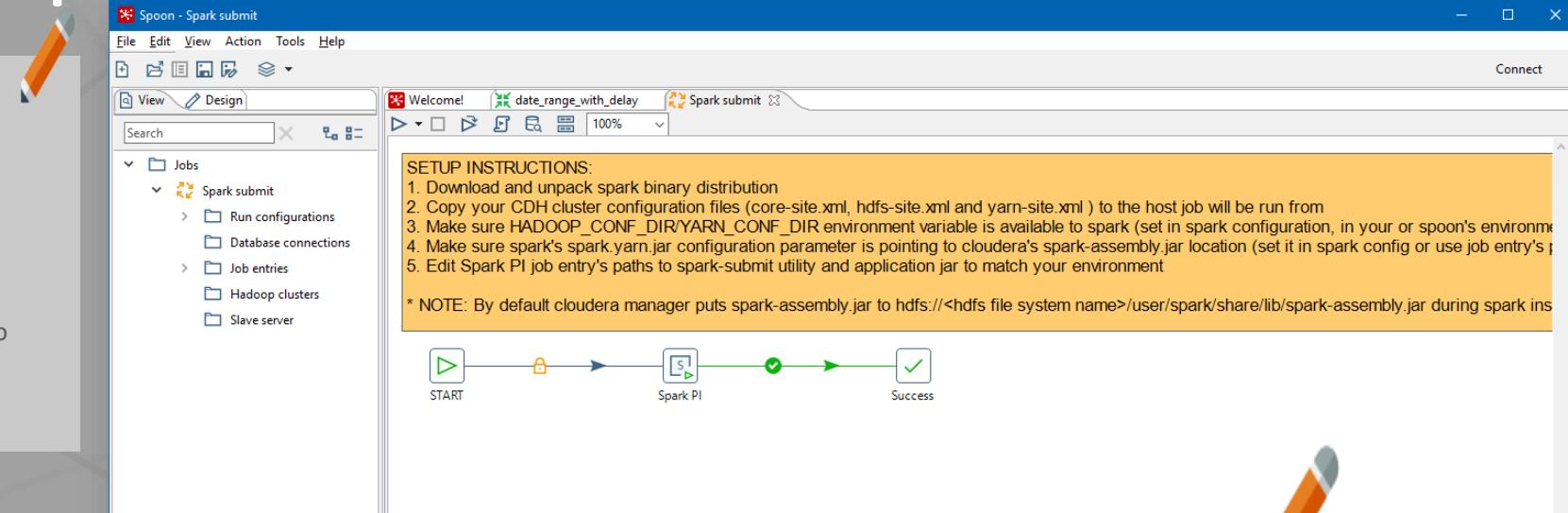
Modificar el ejemplo de Spark

Pasos (Abrir y renombrar el trabajo Job)

1. Copie un archivo de texto que contenga palabras que le gustaría contar al HDFS en su grupo.
2. Inicia Spoon.
3. Abra el trabajo Spark Submit.kjb, que se encuentra en <pentaho-home> / design-tools / data-integration / samples / jobs.
4. Seleccione Archivo> Guardar como, luego guarde el archivo como Spark Submit Sample.kjb.

Nota

Para copiar archivos en estas instrucciones, use el paso de trabajo Copiar archivos de Hadoop o las herramientas de línea de comandos de Hadoop. Para ver un ejemplo de cómo hacer esto usando PDI, consulte nuestro tutorial en <http://wiki.pentaho.com/display/BAD>Loading+Data+into+HDFS>.



Pasos (Submit Spark Job)

1. Abra la entrada **Spark PI job**. Spark PI es el nombre dado a la entrada **Spark Submit** en el ejemplo.
2. Indique la ruta en **spark-submit** en el campo **Spark Submit Utility**. Se encuentra en el lugar donde instaló el cliente Spark.
3. Indique la ruta al contenedor de ejemplos de Spark jar (ya sea la versión local o la del clúster en el HDFS) en el campo de la aplicación jar. El ejemplo de Word Count está en este jar.
4. En el campo Nombre de clase, agregue lo siguiente: **org.apache.spark.examples.JavaWordCount**.
5. Le recomendamos que establezca la URL maestra en **yarn-client**. Para leer más sobre otros modos de ejecución, consulte <https://spark.apache.org/docs/1.2.1/submitting-applications.html>.
6. En el campo Argumentos, indique la ruta al archivo en el que desea ejecutar Word Count.
7. Haga clic en el botón OK.
8. Guarde el trabajo.
9. Ejecutar el trabajo. A medida que se ejecuta el programa, verá los resultados del programa de conteo de palabras en el panel Ejecución.

Consultas

