

Tema 4: Preparación de datos en el contexto de la IA

Adquisición y preparación de datos

mpilar.escobar@ua.es



Universitat d'Alacant
Universidad de Alicante

Tema 4: Preparación de datos en el contexto de la IA

Contextualizar

- Ciclo vida proyecto IA
- Fases de la preparación de datos
- Adquisición de datos
- Introducción al diseño de procesos ETL con Pentaho

¿Qué es la preparación de los datos?

¿Por qué es importante?

¿Qué hace que los datos estén preparados para la IA?

Preparación de datos

Proceso: conjunto de pasos necesarios para la transformación de los datos en un formato adecuado, limpios y de calidad para que puedan ser utilizados por los modelos de IA.

Tareas clave: recolectar, limpiar, estructurar y enriquecer los datos.

Objetivo es garantizar la calidad, la coherencia y la relevancia de los datos para maximizar el rendimiento y la fiabilidad de los modelos de IA.

El 30% de los proyectos de IA generativa serán abandonados tras la prueba de concepto debido a la baja calidad de los datos, a controles de riesgo insuficientes, al aumento de los costes o a un valor empresarial poco claro (Gartner).

Preparación de datos

¿Están preparados para los modelos de IA?

Datos preparados para la IA



Alineación	Calificación	Gobernanza
<ul style="list-style-type: none">• Cuantificación• Semántica• Calidad• Confianza y equidad• Diversidad• Linaje	<ul style="list-style-type: none">• Evaluación de la coherencia• Validación y verificación• Acuerdos de nivel de servicio (SLA) operativos• Control de versiones• Pruebas de regresión continuas• Indicadores de observabilidad	<ul style="list-style-type: none">• Administración de datos• Inferencia y derivación• Regulación y cumplimiento• Apoyo a las normas de IA• Intercambio de datos

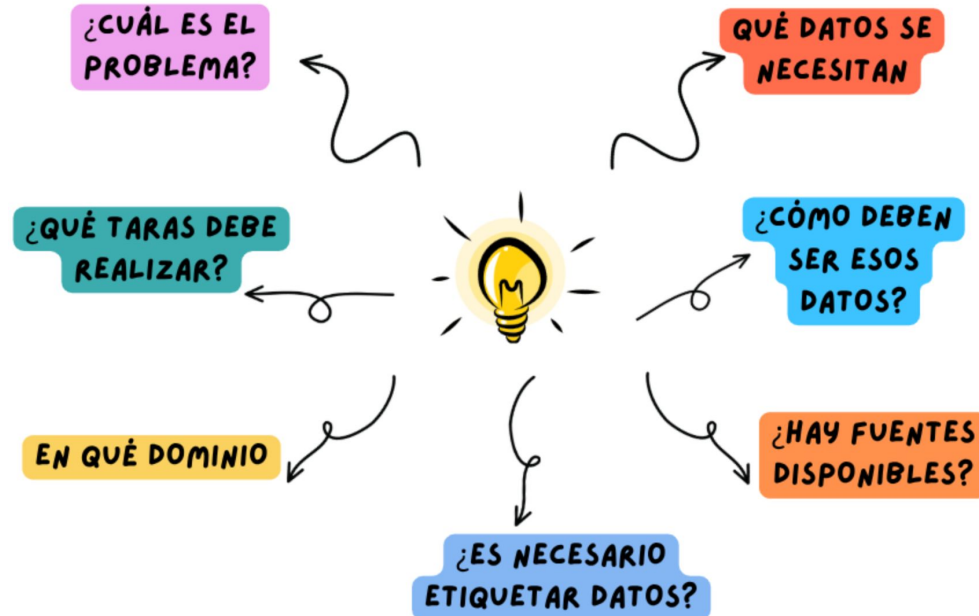
Ciclo de vida del proyecto IA



Ciclo de vida del proyecto IA

Etapa 1: Definición del proyecto centrado en los datos

Contextualización
de los datos



Ciclo de vida del proyecto IA

Etapas 1: Definición del proyecto centrado en los datos

Determinar el alcance del problema a resolver.

Lienzo del problema (4W)

- **Who**, quiénes son las partes interesadas
- **What**, cuál es el problema o necesidad y cuál es su naturaleza
- **Where**, la situación, momento y contexto dónde se produce
- **Why**, cuál es el beneficio y su impacto, tanto en las partes interesadas como en la sociedad

Ciclo de vida del proyecto IA

Etapas 1: Definición del proyecto centrado en los datos

Determinar el alcance del problema a resolver

Definir los objetivos y los casos de uso.

Analizar las necesidades a nivel de datos:

- Qué datos son necesarios: tipo, formato ...
- Qué fuentes de datos se van a utilizar

Definir métricas clave y evaluar en función de los objetivos y casos de uso:

- Expectativas de los casos de uso: cuantificación, semántica, calidad, confianza y diversidad
- Requisitos de confianza: validación, rendimiento, coste y otros requisitos no funcionales
- Requisitos de gobernanza de datos asegura que los datos y los modelos se gestionen de forma legal, ética y eficiente

Ciclo de vida del proyecto IA

Etapas 2: Adquisición y preparación de datos

Pasos para la preparación de los datos:

1. Adquisición de datos
2. Limpieza de datos: unificar, reducir ...
3. Transformación de datos
4. Normalización de los datos
5. Etiquetado de datos
6. Enriquecimiento de datos
7. Validación de datos
8. Equilibrar el conjunto
9. Particionamiento de datos

Ciclo de vida del proyecto IA

Etapas 3: Modelización. Proceso iterativo

- Selección/desarrollo del modelo
- Entrenamiento usando los datos preparados en la fase anterior
 - Pruebas y ejecución para tratar de predecir comportamientos o descubrir ideas en los datos
- Afinado del modelo para resolver el problema planteado
 - Proceso iterativo → retroalimentación con los datos
 - Alineación con los objetivos del proyecto
- Evaluación de otros modelos
 - Precisión, robustez, eficacia, etc.

Ciclo de vida del proyecto IA

Etapas 4: Evaluación

Evaluar en función de las métricas definidas, por ejemplo:

- Precisión
- Recuperación (sensibilidad)
- Fiabilidad del modelo

Alineación con los objetivos definidos previamente y su impacto.

Responsabilidad ética y legal.

Ciclo de vida del proyecto IA

Etapas 5: Despliegue en producción

No todos los proyectos de IA pueden o deben ser puestos en producción.

El coste de poner un modelo en producción puede ser elevado incluso superar el valor que aportaría.

Hacer previsión antes de construir el modelo, aunque esto no siempre es posible.

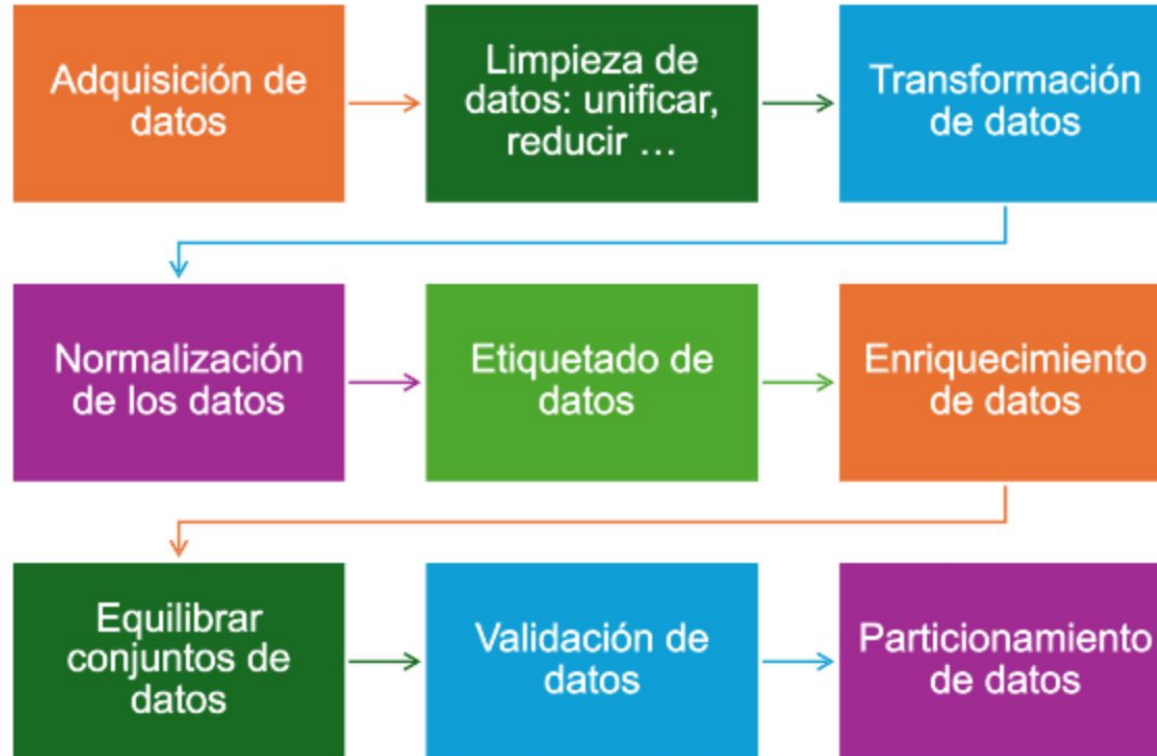
Monitorizar modelo comprobando que siga siendo preciso y eficaz y se adapta a los nuevos datos y cambio en las condiciones.

Ciclo de vida del proyecto IA

Retos más comunes

- Escasez o desequilibrio de datos
- Sesgo en datos y modelos
- Cumplimiento y normativa
- Sobreajuste y pérdida de generalizabilidad
- Latencia en aplicaciones de tiempo real
- Complejidad de la integración
- Gestión de costes

Preparación de datos

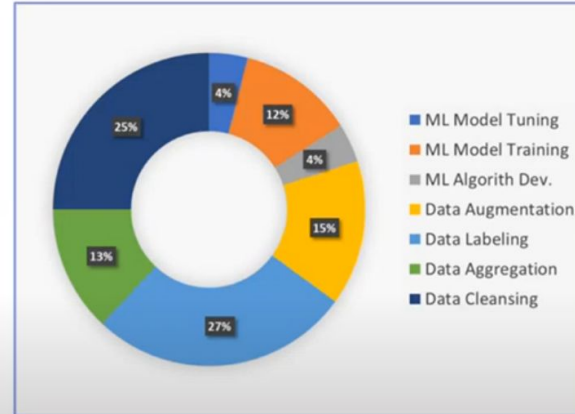


Preparación de datos

3. PREPARACIÓN DE DATOS

SIGMA • COGNITION

- 80% del tiempo en proyectos de IA
- Garantizar que los datos que alimenten a los modelos de AI cumplan los siguientes requisitos:
 - ✓ Limpios
 - ✓ Precisos
 - ✓ Completos
 - ✓ Relevantes



Fuente: <https://www.cognilytica.com>

Adquisición de datos

Objetivo: Obtener datos relevantes y confiables para el entrenamiento de los modelos de IA.

Aspectos relevantes en la recolección de datos:

- Identificar las necesidades del proyecto
- Seleccionar fuentes de datos adecuadas y confiables
- Seleccionar datos relevantes, representativos y diversos
- Verificar el cumplimiento legal y ético FAIR-R + CARE
- Automatizar la recolección si es necesario
- Evaluar la calidad de los datos recopilados

Adquisición de datos

Seleccionar fuentes de datos, variedad y veracidad

Fuentes externas			Fuentes internas
<i>Datasets</i> públicos	<i>Datasets</i> privados	Datos generados	
Datos abiertos, datos abiertos enlazados y API públicas	Plataformas de datos de terceros.	Capturas de sensores, datos de IoT o simulaciones.	BD corporativas, registros de usuarios o historiales
Licencia Creative Commons	Normalmente bajo licencia o acuerdo con el responsable	Comprobar la credibilidad y la puntualidad de estas fuentes para garantizar que los datos sean pertinentes y precisos	Administración propia de los datos, coste añadido
Primeras aproximaciones del modelo	Administración propia de los datos		Cubre casuística
Adaptaciones más ligeras al dominio	Adaptación más cercana al dominio y situación real donde operará el modelo.		Datos personalizados según las necesidades
Se puede aplicar a datos experimentales			

Adquisición de datos

Clasificación de los datos según formato y esquema

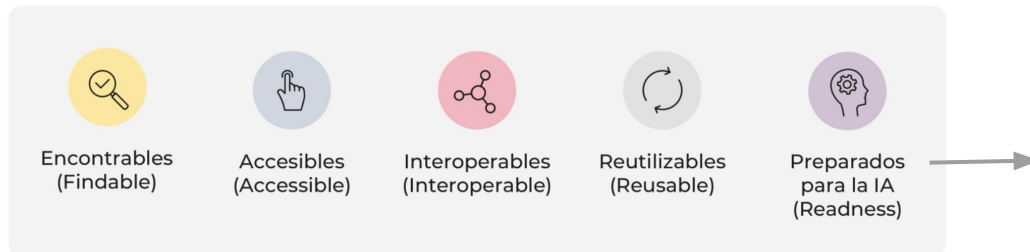
	Datos Estructurados	Datos No Estructurados	Datos Semiestructurados
Estructura	Modelo de datos predefinido y estricto	Sin modelo de datos predefinido	Estructura más flexible y jerárquica Uso de metadatos
Almacenamiento	Formato tabular Esquema rígido bases de datos relacionales y almacenes de datos*	Formato nativo, bases de datos NoSql, data lakes	Bases de datos NoSQL
Análisis	Fácil y rápido	Más complejo y requieren habilidades y herramientas especializadas	Moderadamente complejo
Ejemplos	Tablas, bases de datos Sql, hojas de cálculo, csv*	Textos, imágenes, videos, multimedia, IoT, social media	JSON, XML
Casos de uso	Machine learning (ML)	PLN Modelos IA generativa RAG (Generación Aumentada por Recuperación) Análisis predictivos	

Adquisición de datos

Seleccionar fuentes de datos, datos abiertos y **FAIR-R**

- Ámbito de datos abiertos y ciencia abierta
- FAIR Enfocado en las características del dato

PRINCIPIOS FAIR-R



Aspectos de FAIR-R

- Etiquetado, anotado y enriquecimiento
- Transparencia sobre el origen, linaje y tratamiento de los datos
- Estándares, metadatos, esquemas y formatos óptimos para su uso por parte de la IA
- Cobertura y calidad suficientes para evitar sesgos o falta de representatividad

Adquisición de datos

Seleccionar fuentes de datos, datos abiertos y CARE

- Ámbito de datos abiertos y ciencia abierta
- Complementan FAIR considerando tanto a las personas como al propósito



Adquisición de datos

Requisitos para la adquisición de los datos

Datasets existentes	Anotación de datos
Dominio de la aplicación	Calidad
Representa la variabilidad	Regulación ética y legal (GDPR, LOPD, CARE)
Clases están balanceadas	Seguridad
	Escalabilidad

Adquisición de datos

Ejemplos de fuentes de datos

- Data science community
<https://www.kaggle.com/>
- World Bank o Banco Mundial
[World Bank Open](#)
- [datos.gob.es](#)
- [European data](#)
- [FAO catalog datasets](#)
- [Geonames.org](#)
- [Copernicus EU](#)
- [TData CV datos sobre el turismo](#)
- [INE](#)
- [Open food facts](#)
- [AEMET OpenData](#)
- [OpenWeatherMap](#)

Adquisición de datos

Casos de uso

- [Google Dataset Search](#)
- [DataMed](#)
- [Copernicus](#)
- Administraciones públicas:
 - [datos.gob.es](#), [guía práctica para la publicación de datos abierto usando APIs](#)
 - EDP dataset search interface [Portal europeo de datos](#)
 - <https://data.europa.eu/es/publications/use-cases>
 - [Open Data Barcelona](#) → Ejemplo [RDF Data Cube](#)
- Patrimonio cultural
 - Bibliotecas digitales [CBVMC](#)
 - [British Library datasets](#)
 - [Gallica, la biblioteca digital francesa](#)
 - [Library of Congress](#)

Procesos ETL

Procesos ETL

En el contexto de la IA los procesos ETL son una opción para la adquisición y preparación de los datos para los modelos.

Herramienta de integración de datos

Facilita:

- Adquisición y gestión de datos
- Gestión y administración de servicios
- Extracción de datos
- Transformación de datos
- Carga de datos en el sistema destino

Procesos ETL

Extract

- Extraer datos desde las distintas fuentes de origen - fusión de datos
 - Identificar fuentes
 - Identificar conjunto de datos
 - Conectores para la adquisición
 - Fuentes como Hadoop, datos abiertos, Sql, Nosql (MongoDB, Cassandra), repositorios RDF, Wikidata, ...
- Analizar los datos obtenidos
 - Tipos de datos en qué formato origen
 - Codificación origen
- Interpretar y comprobar que cumplen con la estructura esperada
- Convertir al formato más adecuado para nuestro objetivo

Procesos ETL

Transform

En esta fase se aplican las reglas de negocio que guiarán los datos obtenidos aplicando transformaciones para el formato final, datos en contenido y forma

- Selección de datos (columnas, filas...) - Solo lo necesario el resto es ruido
- División de columnas según necesidades
- Traducción de valores
- Codificación
- Combinación de fuentes
- Combinación de datos para obtener nuevos
- Tratamiento de textos (operaciones sobre ellos expresiones regulares)

Procesos ETL

Load

Cargar los datos obtenidos y transformados en el sistema destino

- Selección sistema destino: fichero, BD relacional, índice, NoSql
- Nuevo o sobrescribir (ejemplo BD, Excel)
- Historial de cambios

Tipos de proceso de carga:

- Acumulación simple
- Rolling - mayor granularidad

Flujos de trabajo de procesos ETL

Existen diferentes herramientas que permiten trabajar con grandes volúmenes de datos y facilitan su análisis.

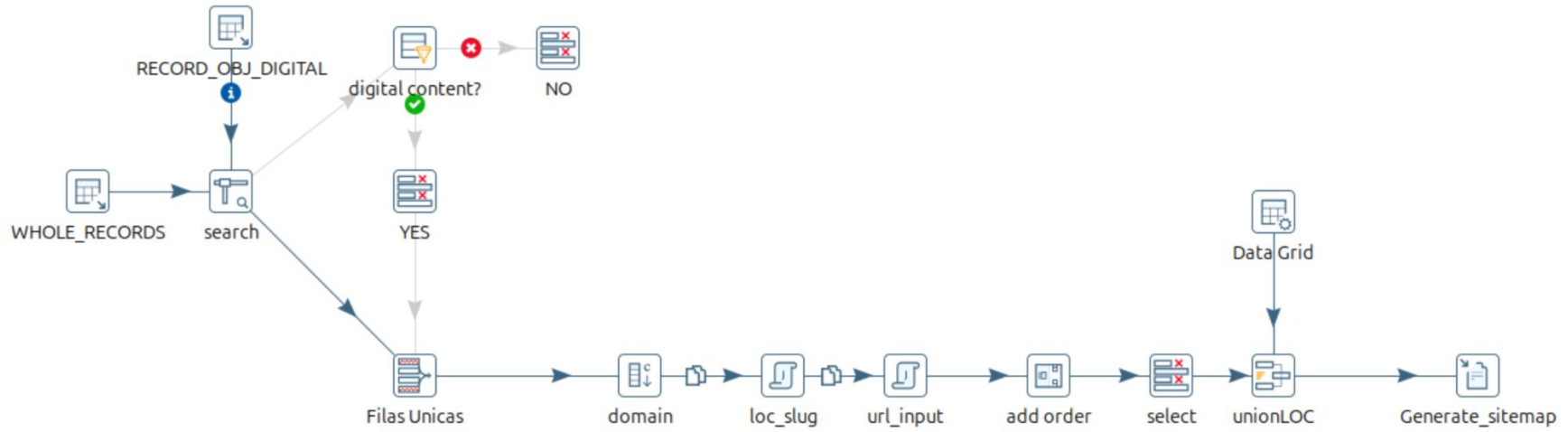
Para ello se diseñan y desarrollan flujos de trabajo que se pueden crear utilizando pasos o entradas unidas por saltos que pasan datos de un elemento al siguiente.

Los **flujos de trabajo** utilizan dos tipos de archivos:

1. Transformaciones que realizan tareas ETL.
2. Trabajos que organizan actividades de ETL, como la definición del flujo, dependencias y ejecución.

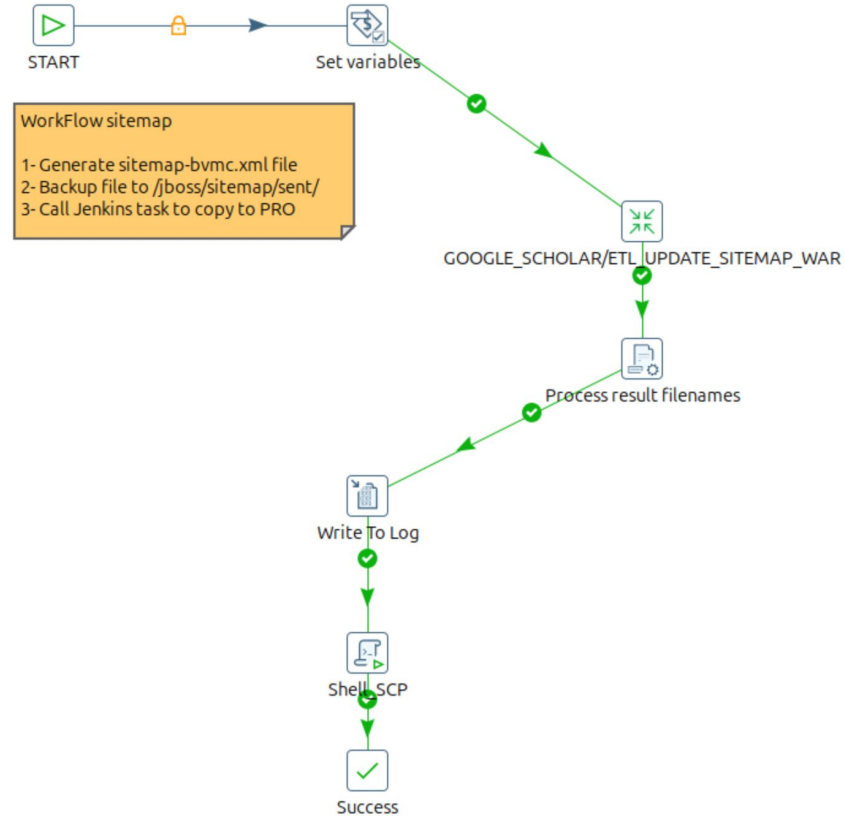
Flujos de trabajo de procesos ETL

Transformation



Flujos de trabajo de procesos ETL

Job



Pentaho, herramientas para diseño de ETLs

[Pentaho](#) suite Business Intelligence para los procesos de gestión de datos.

Open source multiplataforma creada en 2004, plataforma Java.

Disponible en dos ediciones: **Community Edition** (CE) y **Enterprise Edition** (EE).

Múltiples plugins creado por la comunidad de código abierto.

Pentaho Community wiki

<https://pentaho-public.atlassian.net/wiki/spaces/COM/overview?mode=global>

Pentaho, herramientas para diseño de ETLs

Pentaho Data Integration (Kettle)

Entorno gráfico para diseño flujos de datos.

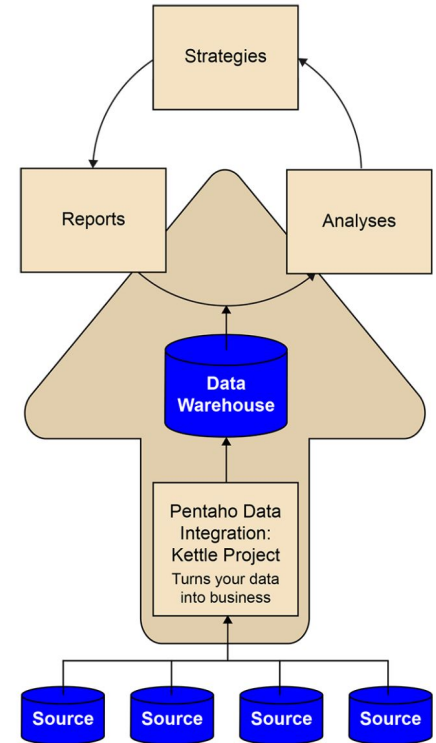
Servicio de integración de datos ETL (extract, transform, load).

Facilita el acceso a una amplia gama de fuentes de datos, incluidas bases de datos de código abierto y propietarias.

Arquitectura extensible mediante plugins y perspectivas

Algunos usos de Pentaho Data Integration

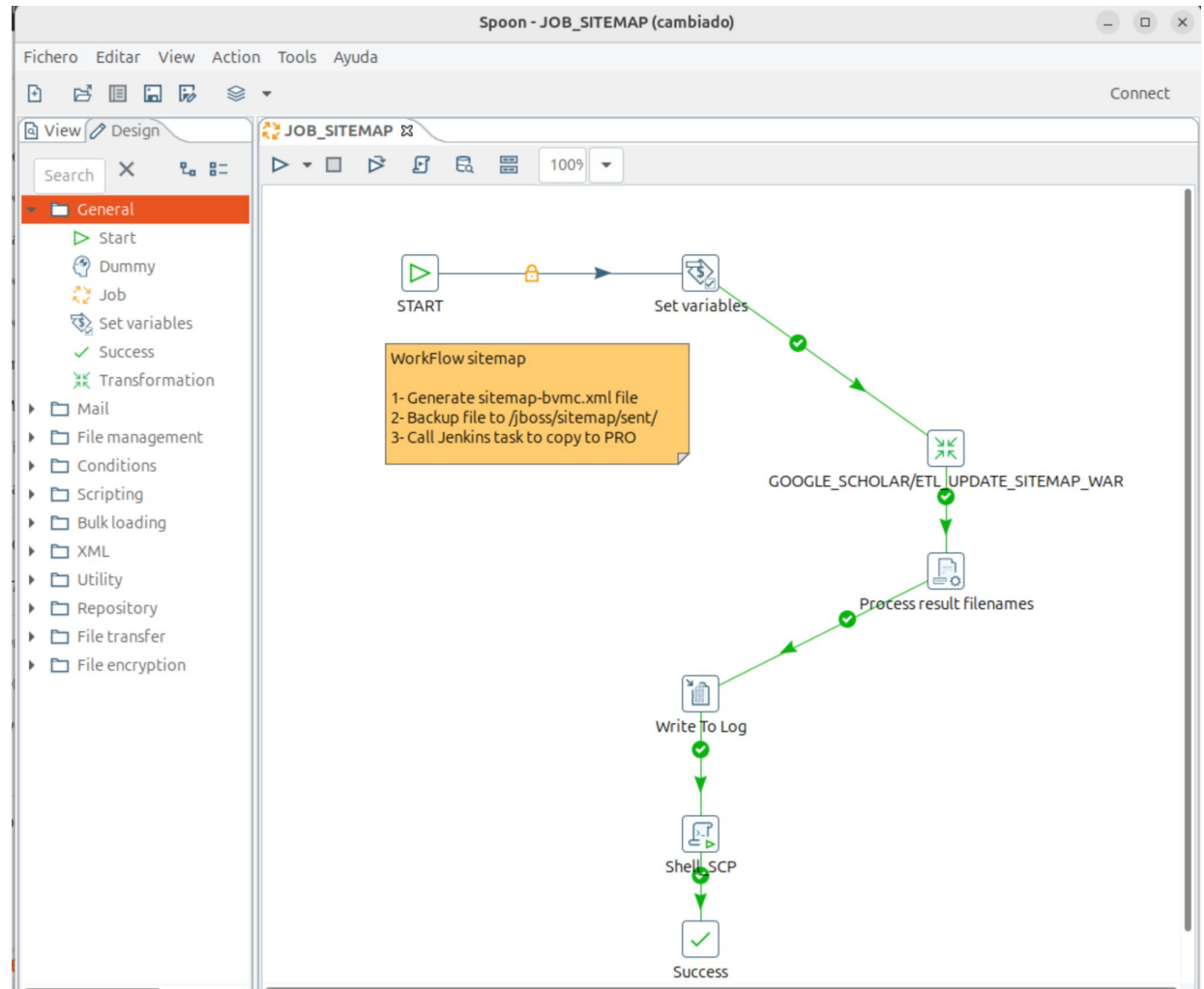
- Migración de datos entre diferentes sistemas
- Carga de grandes volúmenes de datos
- Limpieza de datos mediante transformaciones
- Integración de datos mediante herramientas ETL en tiempo real



Fuente: Pentaho.

Pentaho

Entorno visual



Pentaho, herramientas para diseño de ETLs

Componentes clave de Pentaho Data Integration

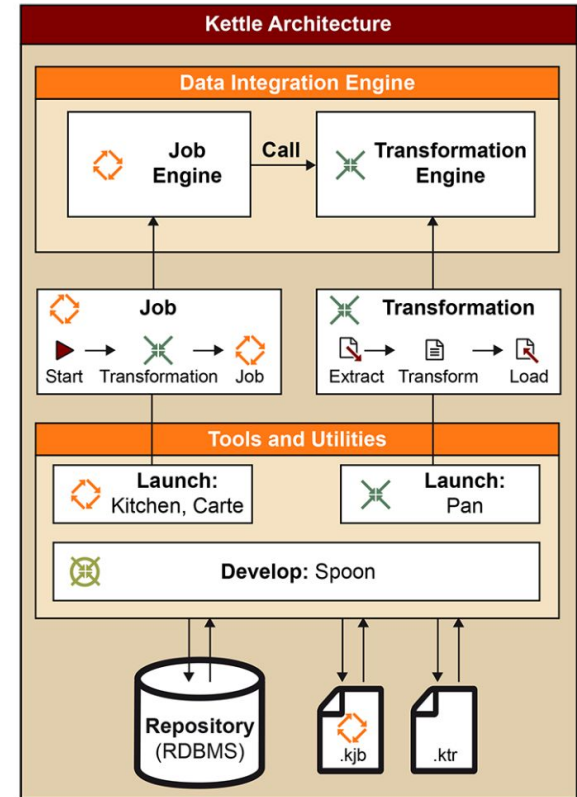
Spoon: entorno gráfico para creación de flujos de trabajo, transformaciones y trabajos.

- Realizar data warehouse
- Operaciones, como crear conexiones de datos, transformaciones o inserción de fórmulas, entre otras.
- Múltiples funcionalidades para realizar procesos de ejecución, transformación y carga de datos.

Pan: creación líneas de comandos para ejecutar transformaciones

Kitchen: ejecución de trabajos por línea de comandos

Carte: servidor web para ejecutar transformaciones y trabajos



Fuente: Pentaho.

```
carte.sh*
encr.sh*
import.sh*
kitchen.sh*
pan.sh*
purge-utility.sh*
runSamples.sh*
set-pentaho-env.sh*
SpoonDebug.sh*
spoon.sh*
yarn.sh*
```

```
#!/bin/bash
```

```
bash ~/Aplicaciones/Pentaho/data-integration/kitchen.sh -file
/home/GS/JOB_SITEMAP.kjb -level:Basic
```

```
bash ~/Aplicaciones/Pentaho/data-integration/pan.sh -file:/home/ETL_GEN_ARK.ktr
-level:Basic
```

```
bash ./carte.sh 127.0.0.1 8083
```

The screenshot shows the Pentaho PDI Status web interface in a browser. The address bar shows 'localhost:8083/kettle/status/'. The page title is 'PDI Status'. There are two main sections: 'Transformations' and 'Jobs'. Each section has a table with columns: Name, Carte Object ID, Status, Last log date, and Last log time. Below these sections is a 'Configuration details:' section with a table showing various settings and their values. At the bottom, a note states: 'These parameters can be set in the slave server configuration XML file: (Using defaults)'.

Name	Carte Object ID	Status	Last log date	Last log time
------	-----------------	--------	---------------	---------------

Name	Carte Object ID	Status	Last log date	Last log time
------	-----------------	--------	---------------	---------------

Configuration details:	
The maximum size of the central log buffer	No limit
The maximum age of a log line	No limit
The maximum age of a stale object	No limit
Repository name	

These parameters can be set in the slave server configuration XML file: (Using defaults)

Pentaho Data Integration

Pasos para el diseño de *jobs*

General, pasos de inicio de trabajo, ejecución transformaciones o trabajos entre otras operaciones

Mail, envío de correos, recuperación y validación de cuentas

File management, gestión de operaciones sobre ficheros y carpetas, HTTP, etc

Conditions, comprobaciones sobre conexiones, BD, ficheros, etc

Scripting, scripts en JavaScript, Shell y SQL

Bulk loading, cargas a MySQL y MSSQL

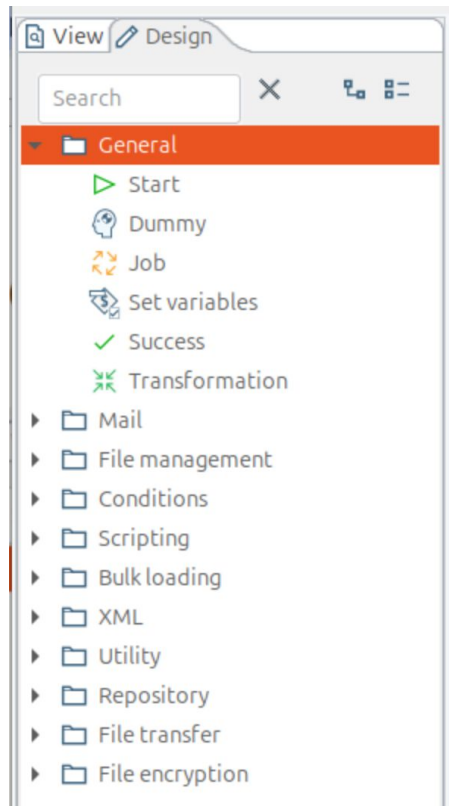
XML, validaciones XML, DTD, XSD y XSL

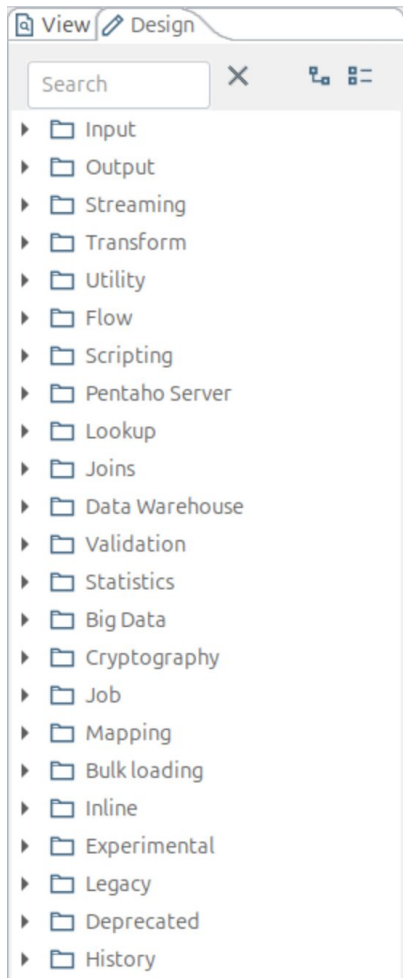
Utility, pasos para ejecución de transformaciones como ping, enviar información a syslogs, Nagios, etc

Repository, operaciones con el repositorio de transformaciones y trabajos.

File transfer, gestión ficheros FTP, FTPS y SFTP

File encryption : sistema PGP (Pretty Good Privacy) para envío y recepción de ficheros





Pentaho Data Integration

Pasos para el diseño de *transformations*

Input, adquisición de datos desde diversas fuentes, BD, CSV, Excel, JSON, repositorios, etc

Output, carga datos en diferentes sistemas

Streaming, obtener datos de Kafka, JMS (Java Message Service) o MQTT (Standard for IoT Messaging)

Transform, ejecución de diversas operaciones con datos, como filtrar, ordenar, partir, añadir nuevos campos, mapear, etc

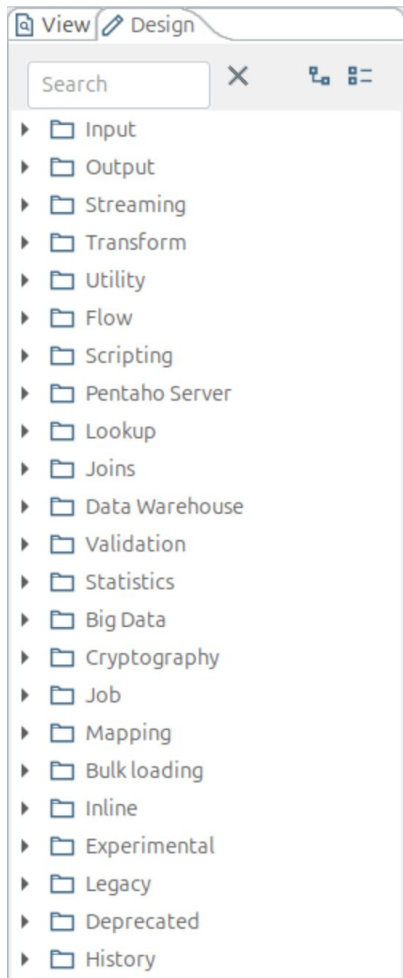
Utility, ejecución ssh commands, envío de mensajes syslog, etc

Flow, control de flujo de datos en las transformaciones

Scripting, SQLScript, JavaScript, expresiones regulares, fórmulas y expresiones java

Pentaho server, conexión al servidor de Pentaho para operaciones como la gestión de variables

Lookup, pasos para añadir información al flujo de datos mediante búsqueda en fuentes como BD, web services, HTTP, etc



Pentaho Data Integration

Pasos para el diseño de *Transformations*

Joins, unión de filas en función de diferentes criterios

Data Warehouse, para trabajar con dimensiones

Validation, validaciones como tarjetas de crédito, datos, email o XSD

Statistics, operaciones estadísticas sobre un flujo de datos como group by

Big Data, cargar y extraer datos de Avro y MongoDB

Cryptography, uso de PGP (Pretty Good Privacy)

Job, realizar operaciones propias de un Job

Mapping, mapeo entre campos de entrada y salida

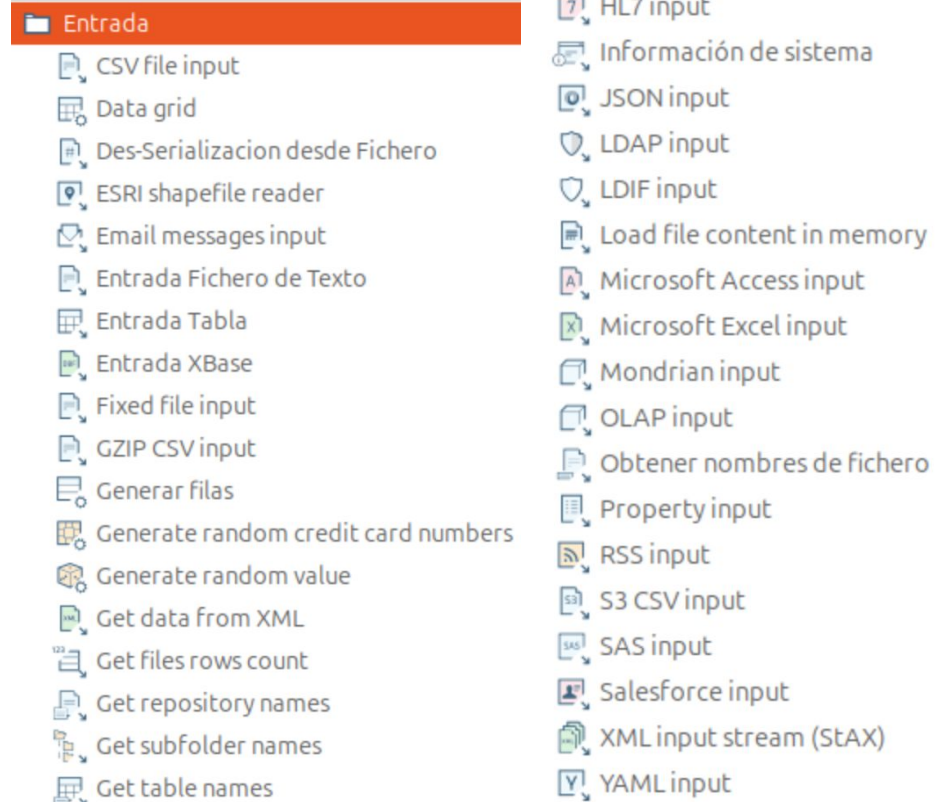
Bulk loading, cargas desde Vectores, MySQL, Oracle, PostgreSQL entre otras

Inline, embebido para operaciones con sockets

Experimental, pasos en fase de validación

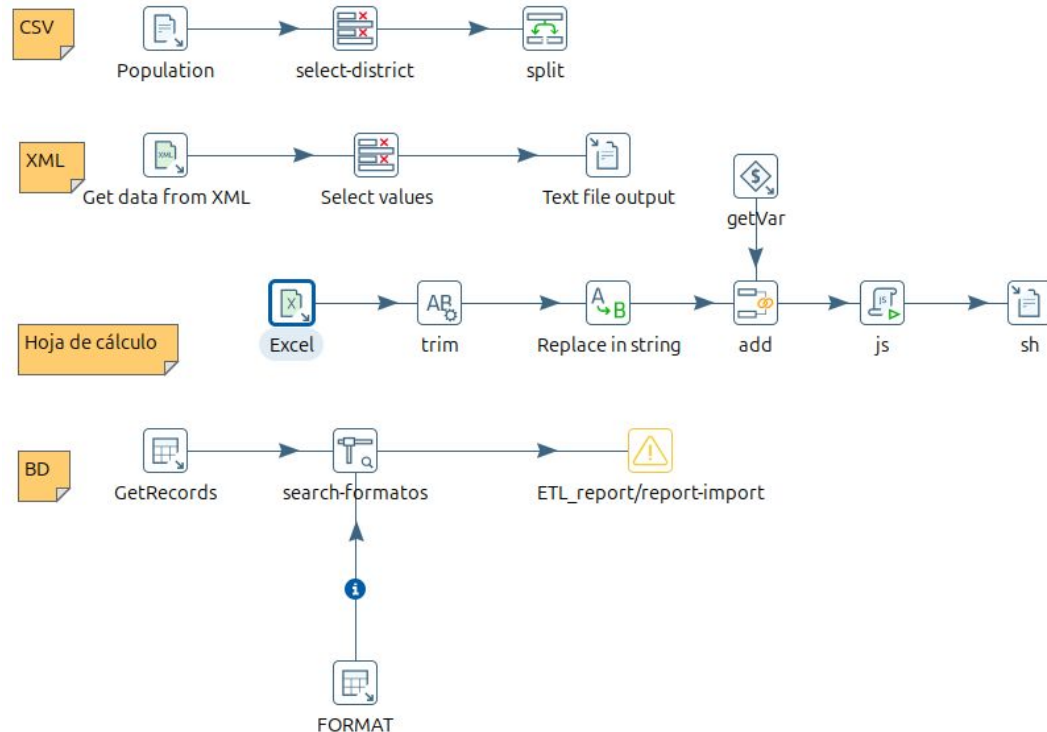
Pentaho Data Integration

Extract, adquisición de datos y distintos conectores para el acceso a las fuentes.



Pentaho Data Integration

Ejemplo 1: conexión a distintas fuentes de datos



Tema 4: Caso práctico

Con el objetivo de entender el diseño de procesos ETL, se propone un caso práctico real, pero simplificado. Donde la explotación y reutilización de los datos proporcionados por plataformas Open Data hace posible realizar previsiones, adelantándose a posibles problemas y minimizando así su impacto. En muchos casos es posible llegar a evitarlos.

Se utilizará datos abiertos publicados por la plataforma Open Data BCN en el contexto de Open Government Data. Este escenario se centra en el estado de los puntos críticos de limpieza en esta ciudad. En primer lugar, se realizó un análisis de los datos disponibles en esta plataforma seleccionando archivos sobre entorno urbano y estado de las áreas críticas de limpieza en la ciudad de Barcelona así como información sobre límites administrativos.

A continuación se listan los conjuntos de datos seleccionados para esta experimentación, junto con el formato en el que estaban disponibles originalmente en la fecha en la que se realizó la prueba real.

- renta-poblacion.csv datos sobre la distribución de los ingresos y la población en la ciudad
- edm.csv datos sobre la población en la ciudad
- puntscriticsbarcelona.csv información sobre el entorno urbano y estado de las áreas críticas de limpieza en la ciudad de Barcelona así como información sobre límites administrativos

*Algunos datos estaban en formato texto y se transformaron a CSV, se ha simplificado para la realización de este ejemplo

Tema 4: Caso práctico

La estrategia en el diseño del proceso ETL será la siguiente:

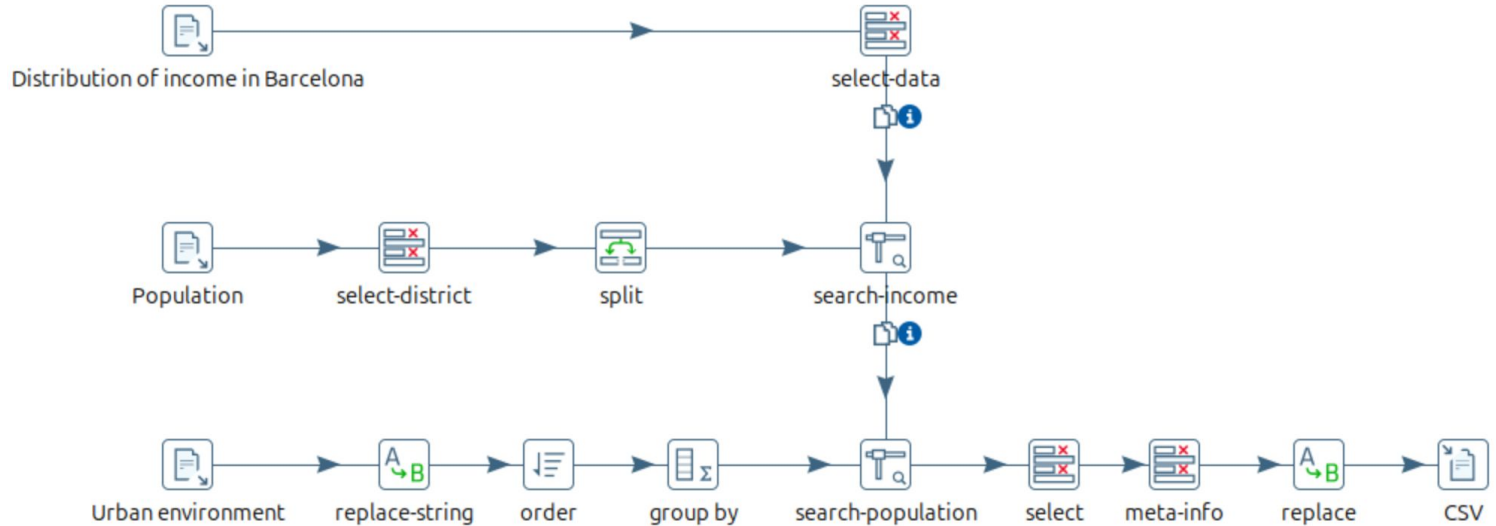
1. Cargar los datos a partir de los ficheros anteriores seleccionando los datos a utilizar en cada caso.
2. Se identifican los conjuntos de datos que comparten puntos comunes (distrito, ubicación geográfica, etc.) y por tanto, permiten un análisis más detallado.
3. Complementar las dimensiones sobre información urbana a partir de la información del fichero edm.csv y el de renta-poblacion.csv.
4. Crear un trabajo para lanzar todas las transformaciones de una manera única.

Notación utilizada:

Para las transformaciones: ETL_INI_{Nombre de la dimensión, tabla de hecho que cargar o funcionalidad}.

Para los trabajos: JOB_INI_{Nombre del proceso}.

Tema 4: Caso práctico



Tema 4: Caso práctico

CSV file input

Step name

Distribution of income in Barcelona

Filename

\${Internal.Entry.Current.Directory}/barcelona/renta-poblacion.c

Browse...

Delimiter

|

Insert TAB

Enclosure

"

NIO buffer size

50000

Lazy conversion?

☒

Header row present?

☒

Add filename to result

☐

The row number field name (optional)

Running in parallel?

☐

New line possible in fields?

☐

Format

File encoding

	Name	Type	Format	Length	Precision	Currency	Decimal	Group
1	cod	Integer	#		0	€	,	.
2	barri	String				€	,	.
3	població resident	String				€	,	.
4	Índex RFD BCN=100	String				€	,	.
5	posició	String				€	,	.

Help

OK

Get Fields

Preview

Cancel

Tema 4: Caso práctico

CSV file input

Step name

Population

Filename

\$(Internal.Entry.Current.Directory)/barcelona/edm.csv

Browse...

Delimiter

,

Insert TAB

Enclosure

"

NIO buffer size

50000

Lazy conversion?

☒

Header row present?

☒

Add filename to result

☐

The row number field name (optional)

Running in parallel?

☐

New line possible in fields?

☐

Format

File encoding

	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Trim type
1	Dte.	String				€	,	.	both
2	Barri de baixa	String				€	,	.	both
3	Districte d'alta: Mitjana edat homes	Number	##		0	€	.	,	none
4	Districte d'alta: Mitjana edat dones	Number	##		0	€	.	,	none
5	Districte de baixa: Mitjana edat homes	Number	##		0	€	.	,	none
6	Districte de baixa: Mitjana edat dones	Number	##		0	€	.	,	none

Help

OK

Get Fields

Preview

Cancel

Tema 4: Caso práctico

CSV file input

Step name:

Filename:

Delimiter:

Enclosure:

NIO buffer size:

Lazy conversion? ☒

Header row present? ☐

Add filename to result ☐

The row number field name (optional)

Running in parallel? ☐

New line possible in fields? ☐

Format:

File encoding:

	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Trim type
1	Número_punt_crític	Integer	#		0	€	,	.	none
2	Codi_Districte	Integer	#		0	€	,	.	none
3	Nom_Districte	String				€	,	.	none
4	Codi_Barri	Integer	#		0	€	,	.	none
5	Nom_Barri	String				€	,	.	none
6	Nom_Carrer	String				€	,	.	none
7	Número	String	#		0	€	,	.	none
8	Latitud	String			0	€	,	.	none
9	Longitud	String			0	€	,	.	none

Tema 4: Caso práctico

Stream lookup

Step name

search-population

Lookup step

search-income

the key(s) to look up the value(s):

	Field	LookupField
1	Codi_Barri	codBarrio

specify the fields to retrieve :

	Field	New name	Default	Type
1	Districte d'alta: Mitjana edat dones			Number
2	Districte d'alta: Mitjana edat homes			Number
3	Districte de baixa: Mitjana edat dones			Number
4	Districte de baixa: Mitjana edat homes			Number
5	població residente			Number
6	Índex RFD BCN=100			Number
7	posició			Integer

Preserve memory (costs CPU) ☒

key and value are exactly one integer field ☐

Use sorted list (i.s.o. hashtable) ☐

Help

OK

Cancel

Get Fields

Get lookup fields

Tema 4: Caso práctico

Extracto de los datos que contiene el fichero con información sobre los puntos críticos de limpieza en la ciudad de Barcelona. Datos extraídos de Open Data BCN.

Martina Castells ;1;41.379772;2.166827;Mala praxis comercial;20;15.8;18;7.4;18;10.8;11;4.3;18;4.2
Malnom ;1;41.380627;2.167949;Problemàtica social;19;4;18;4.9;18;3.3;16;4.6;19;3.7
Picalquers ;1;41.380665;2.168612;Problemàtica social;19;12.1;18;8;18;12.7;16;13.1;19;12.3
Rubió i Luch;1;41.38073;2.170163;Problemàtica social;21;6.9;19;4.3;18;4.7;16;3.2;19;2.9
Egipcíques;3;41.38089;2.168989;Mal ús de contenidors i/o papereres;19;7.6;19;9.3;18;10.4;16;7.3;20;6.9
Gardunya ;1;41.381252;2.17052;Força afluència de gent;19;7.1;19;5.8;17;5.3;16;4.1;19;5.4
Sant Agustí;0;41.38073;2.171528;Mala praxis comercial;19;9.2;19;14.6;18;6.2;16;6.7;19;11.9
Robador;53;41.378761;2.171453;Efecte crida;23;3.4;19;3.6;18;5.8;16;6.4;20;8.3
Salvador Seguí;1;41.37878;2.170497;Mal ús de contenidors i/o papereres;23;5.3;19;5.5;18;3.6;16;4.1;20;7.6
Aurora ;11;41.378757;2.168208;Efecte crida;23;7.3;18;11.1;18;15.1;16;18.5;19;16.1
Aurora ;25;41.377934;2.167251;Efecte crida;23;10.8;18;7.5;18;16.1;16;15.7;19;18.9
Sant Rafael ;42;41.378326;2.1683;Mal ús de contenidors i/o papereres;23;13.3;17;14.8;18;17.1;16;17.1;19;17.8

01|el Raval|47.274|74,6|48|
38|la Teixonera|11.281|75,8|47
02|el Barri Gòtic|15.729|110,5|16|
39|Sant Genís dels Agudells|6.828|86,3|33
03|la Barceloneta|15.068|84,8|35|
40|Montbau|5.101|82,2|41
04|Sant Pere, Santa Caterina i la Ribera|22.380|97,8|24|
41|la Vall d'Hebron|5.687|87,5|32