

CIFF Trustees:



Business Intelligence & Data Mining

Profesor:
Pedro Pasquau

Octubre 2015

MASTER EN BUSINESS ANALYTICS & BIG DATA
2015

Objetivos

- **Sesión 1A: Business Intelligence**
 - ☐ Business Intelligence
 - ☐ BI, BA y Big Data: Contexto
 - ☐ Pentaho BA
- **Sesión 1B: ETL y Adquisición de Datos**
 - ☐ ETL
 - ☐ Pentaho Data Integration
 - ☐ Data Integration y Data Mining: Realimentación
- **Sesión 2A: Data Mining – Machine Learning**
 - ☐ Algoritmos
 - ☐ Ejemplos
- **Sesión 2B: Pentaho Data Scientist (PDS)**
 - ☐ PDS (Weka)
 - ☐ DW→ETL→ PDS
 - ☐ PDS → ETL→DW

- 1.- ETL
- 2.- Pentaho Data Integration
- 3.- Ejemplo Data Integration y Data Mart.
- 4.- Data Integration y Data Mining:
Realimentación



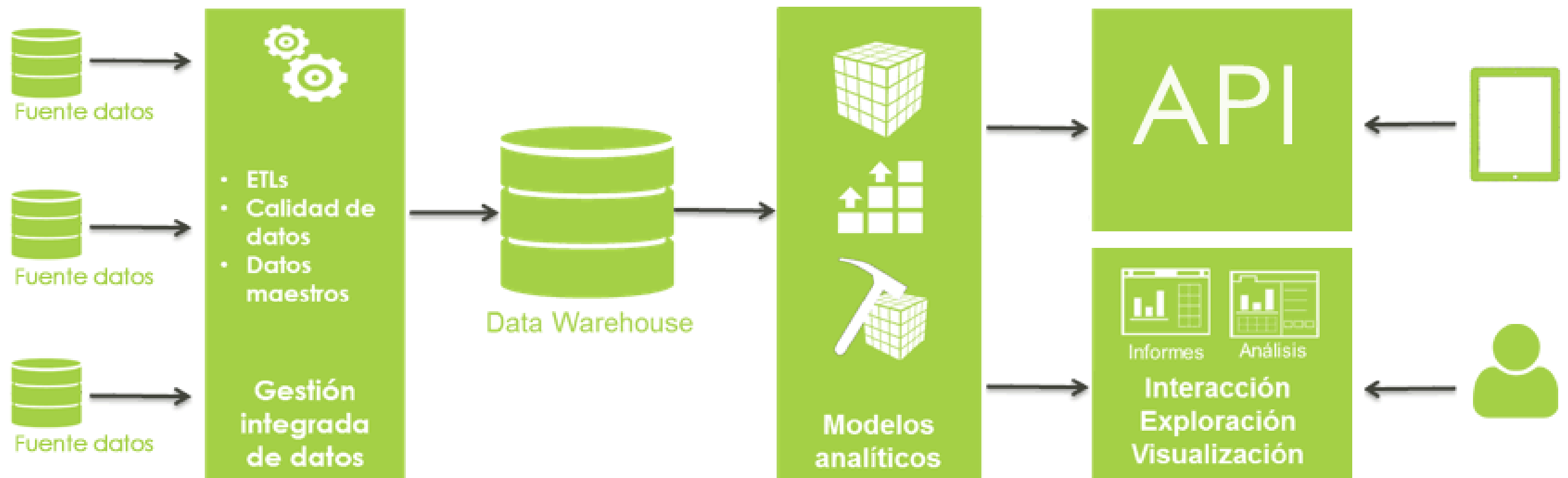
• Llamamos **business intelligence** (BI) al conjunto de estrategias y herramientas enfocadas a la administración y creación de conocimiento mediante el análisis de datos existentes en una organización.

• **Características:**

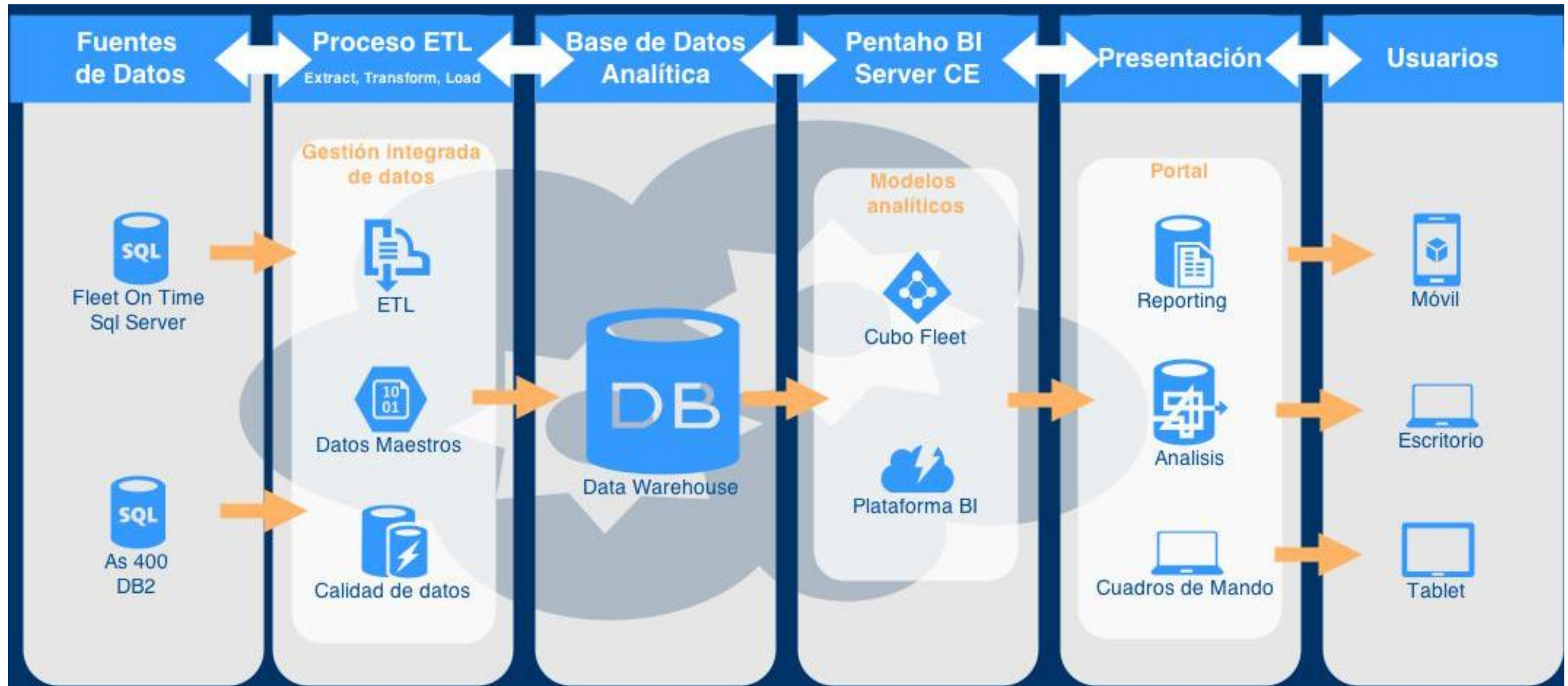


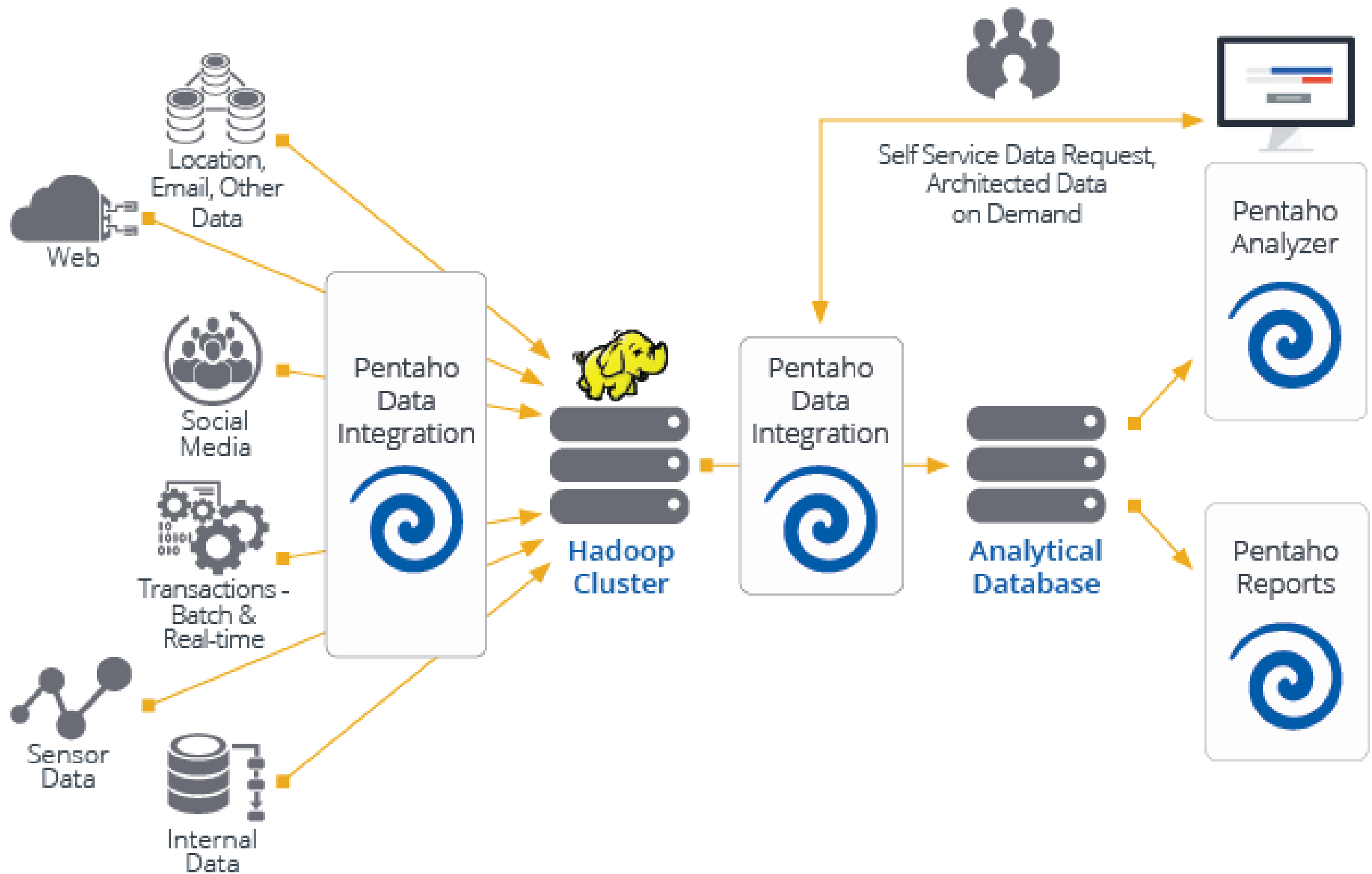
- **Accesibilidad a la información.** Los datos son la fuente principal de este concepto. Lo primero que deben garantizar este tipo de herramientas y técnicas será el acceso de los usuarios a los datos con independencia de la procedencia de estos.
- **Apoyo en la toma de decisiones.** Se busca ir más allá en la presentación de la información, de manera que los usuarios tengan acceso a herramientas de análisis que les permitan seleccionar y manipular sólo aquellos datos que les interesen.
- **Orientación al usuario final.** Se busca independencia entre los conocimientos técnicos de los usuarios y su capacidad para utilizar estas herramientas.

■ Estructura de una solución BI

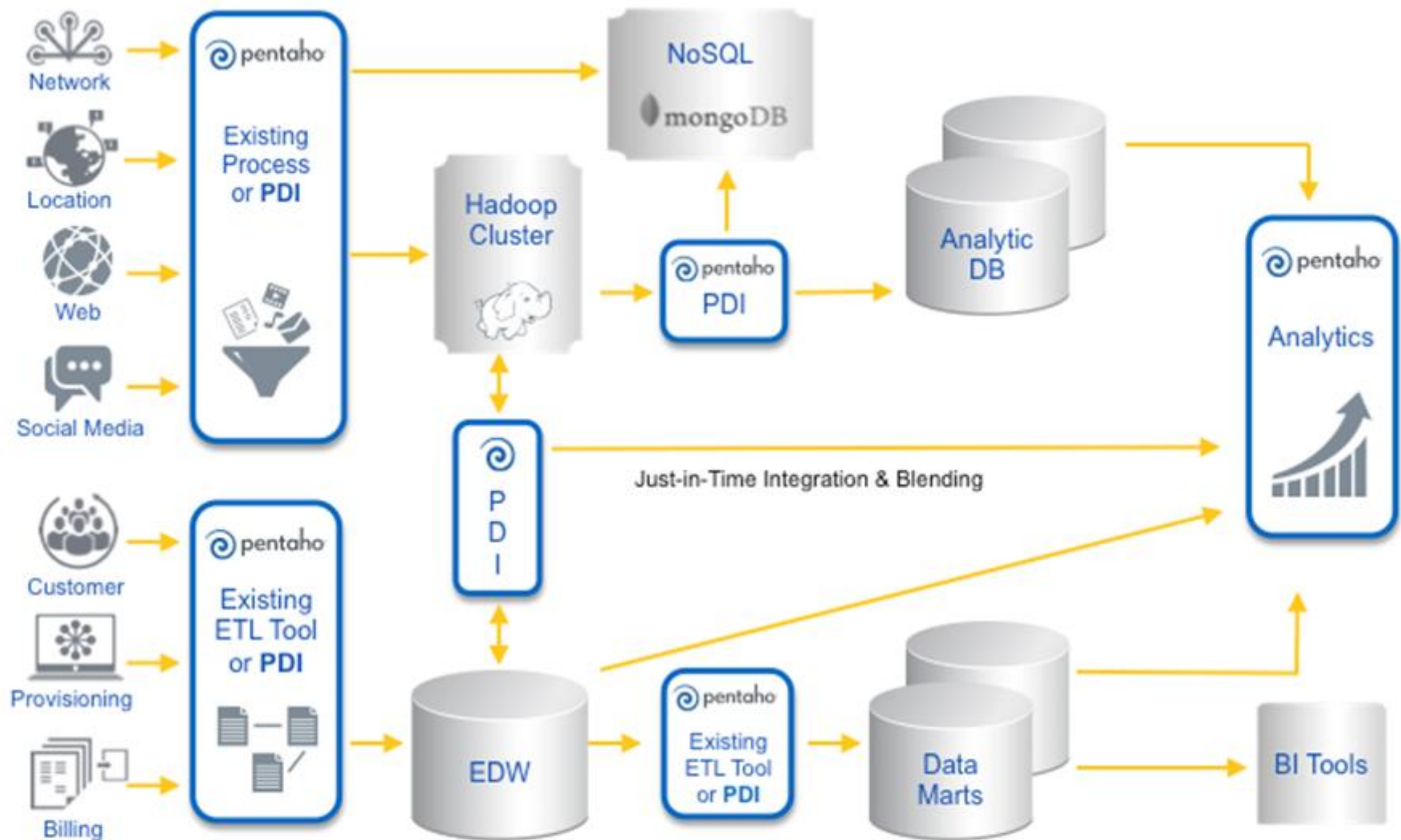


■ Proyecto Real Business Intelligence: Fleet Mangement





Evolving Big Data Architectures



■ ¿Qué es ETL?

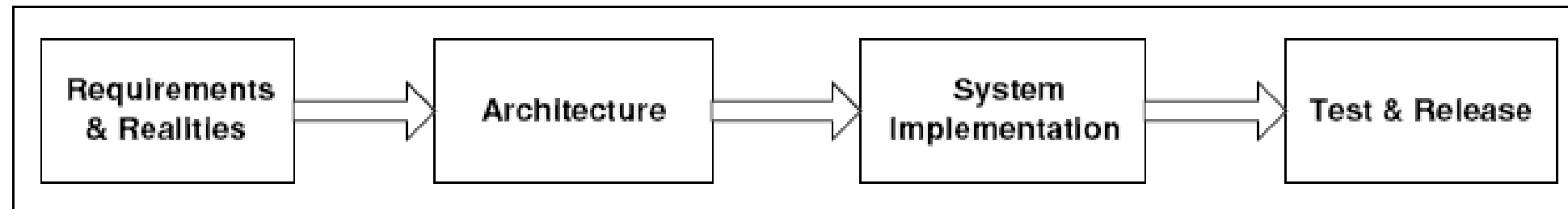
- Extraer, Transformar y Cargar (Load)
- ¿Que realmente hace una ETL?(The Data Warehouse ETL Toolkit 2004):
 - Removes mistakes and corrects missing data.
 - Provides documented measures of confidence in data.
 - Captures the flow of transactional data for safekeeping.
 - Adjusts data from multiple sources to be used together.
 - Structures data to be usable by end-user tools.
- En definitiva:
 - **Extraer** datos de múltiples fuentes
 - Aplicar **calidad y consistencia (limpiar)** a los datos
 - **Conformar (unificar)** los datos
 - **Cargar** los datos en un DW
- Actividad inicial y con ejecución periódica/programable.



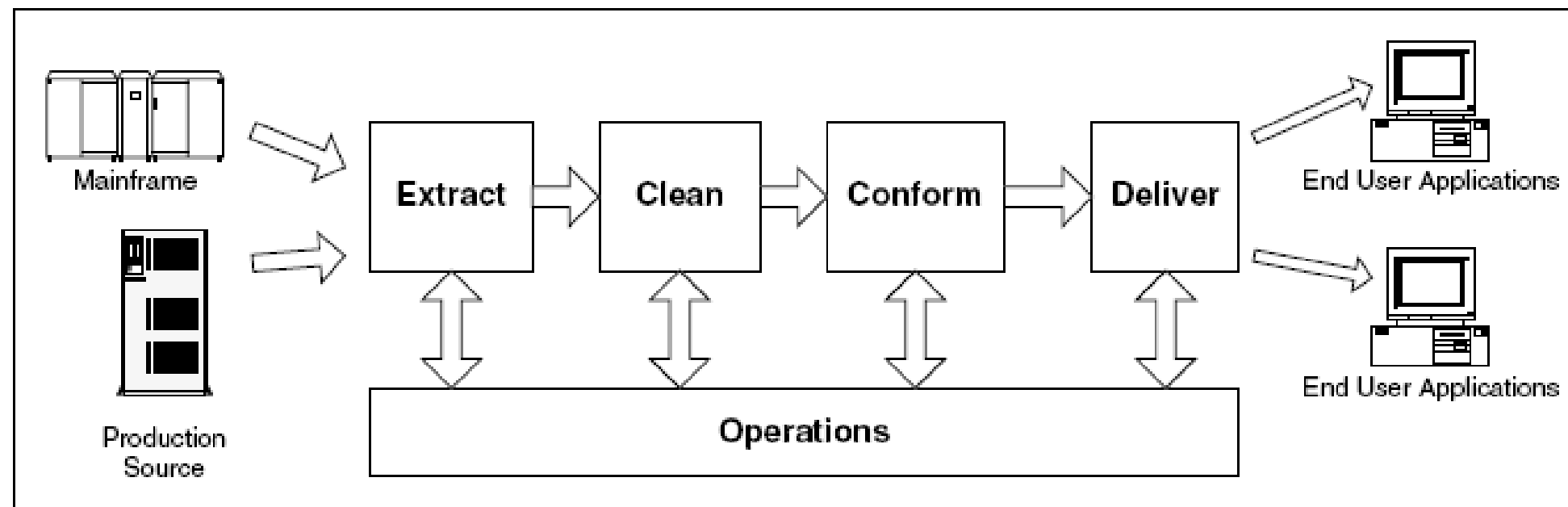
ECCD (extract, clean, conform, deliver)

■ Flujos de un Proyecto ETL

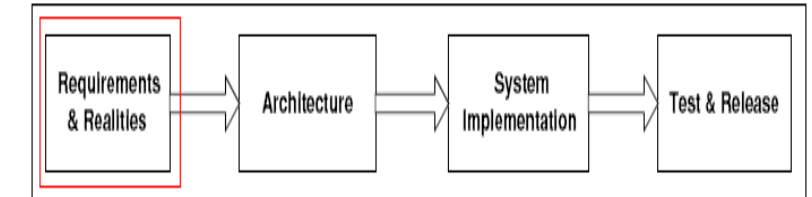
Flujo del proyecto



Flujo de los datos



■ Flujo de Proyecto. Requerimientos



1) Identificar las fuentes

2) Data profiling

- Calidad de los datos para evaluar la fase de limpieza.
- Master Data Management.

3) Requerimientos legales (en entornos muy especificos)

- Algoritmos.
- Almacenamientos secundarios / Copias para auditoria, legislación (Sarbanes Oxley , etc...)

4) Requerimientos de seguridad

- Accesos a fuentes, roles.
- Políticas.
- Permisos.
- Plataformas Fuente.

5) Requerimientos de conformado

- Evaluar la fase de conformado.
- Descubrir heterogeneidad en las fuentes.

6) Latencia del dato

- Describe la velocidad en la que los datos serán “entregados” al usuario final.
- Ventanas de etl.
- Evaluar rendimientos.
- Decidir entre lotes o stream



■ Flujo de Proyecto. Arquitectura

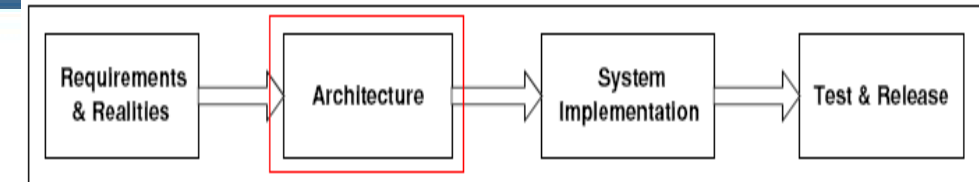
• Decisión crítica: ETL Tool?

Ventajas del ETL Tool

1. Desarrollo del ETL mas **simple**, **rápido** y (posiblemente) **barato**.
2. Técnicos con conocimiento de negocio (no programadores) pueden participar en el desarrollo.
3. Generación de metadata automático
4. Manejo de errores construido
5. Conectores con (casi) cualquier fuente
6. Análisis de impacto
7. Se pueden aumentar con módulos de código
8. Prácticamente auto documentado

Ventajas del ETL programado

1. POO puede hacerlo muy estándar y reusable
2. Control mas granular de todas las actividades.
3. Utilizar programadores ya formados con poco conocimiento del negocio
4. Independencia de proveedor.
5. **Flexibilidad**

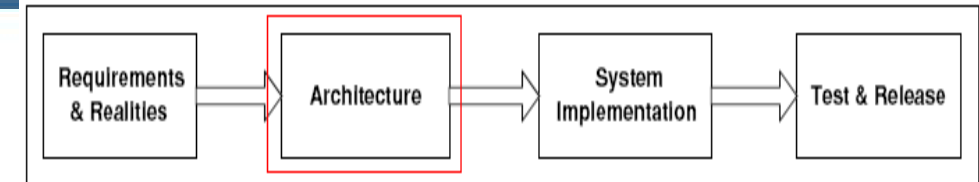


¿Los datos corporativos son sencillos? **X**

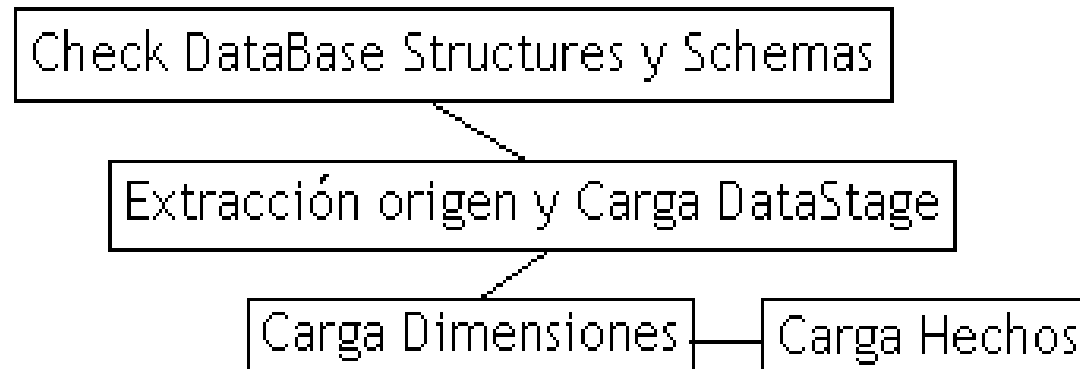
¿Va a cambiar la información? **X**

¿Puede cambiar el roadmap de la extracción? **X**

■ Flujo de Proyecto. Arquitectura



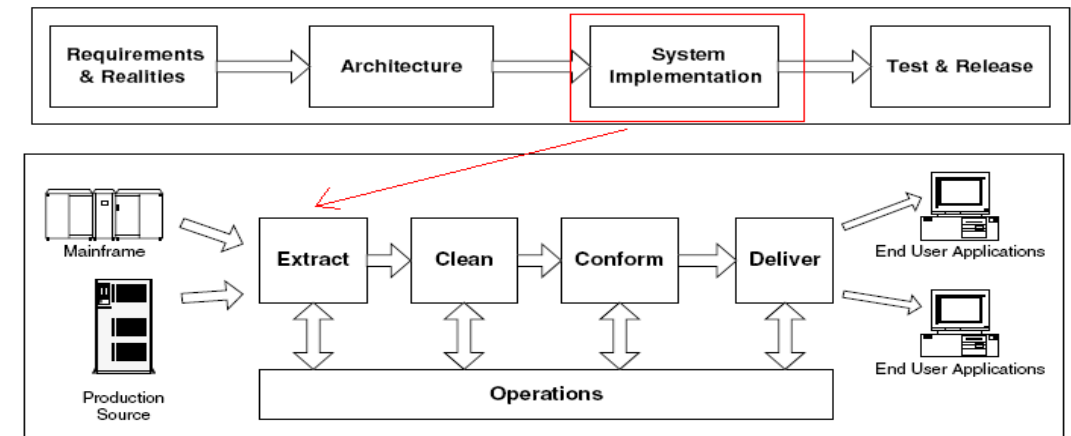
- Metadatos de Flujo de ejecución:



- Programación de las ETL
 - Manuales (históricos (carga 0), cargar segmentos, rollbacks...)
 - Automáticas
 - Diarias
 - Ante eventos (fallos, umbrales, alarmas..)
- Manejo de excepciones
- Manejo de informes de calidad
- Generar informes de accesos y seguridad

■ Flujo de Proyecto. Implementación

- DW y ETL están íntimamente ligados.
 - Se han detectado las fuentes, y el equipo de DW ha definido los modelos y tablas.
 - Se planifica, diseña e implementa en cada paso.
- Responsabilidades del equipo ETL
 - Extraer los datos de las fuentes
 - Limpiar y asegurar la calidad de los datos
 - Conformar los datos para conseguir consistencia y aplicar las reglas de negocio.
 - Entregar los datos en el formato físico determinado.



- ❖ How do we properly design an ETL system?
- ❖ How do we extract data from sources systems?
- ❖ How do we enforce data quality?
- ❖ How do we enforce consistency standards?
- ❖ How do we conform data?
- ❖ How do we make sure that data from separate sources can be used together?
- ❖ How do we make the data presentation-ready?

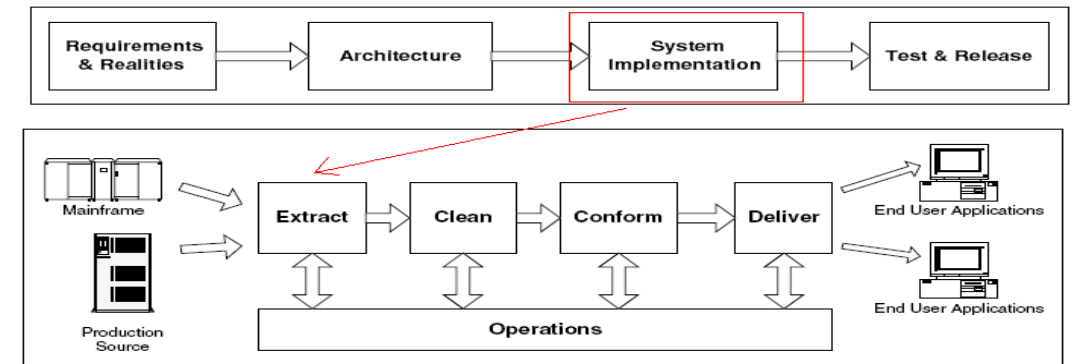
■ Flujo de Proyecto. Implementación

• Data Stage:

- Escribir a disco los datos (fichero o base de datos)
- La otra opción es procesar en memoria (reduciendo así I/O)

• Ventajas de la Staging Area

- **Recuperación**
 - Se hace stage tras cada fase mayor. (extracción !!!)
- **Back-up**
 - Antes del delivery (comprimir, guardar, prevé catástrofes)
- **Auditar el proceso**
 - Fácil detectar en que fase se produjo el error
- **Evitar sobrecarga del operacional.**



• La Staging Area es un sitio de trabajo !!!!!

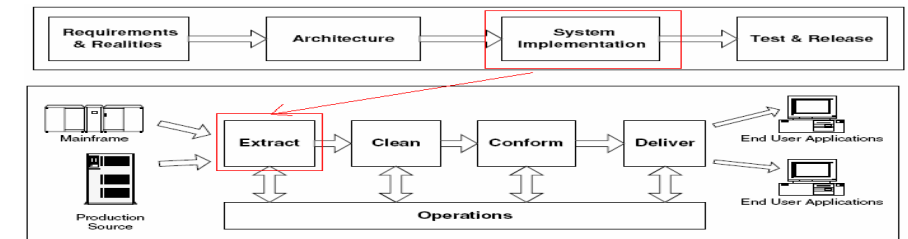
- Los usuarios no acceden
- No se lanzan informes finales
- Solo los ETL acceden



■ Flujo de Proyecto. Implementación

Flujo de datos. Extracción

1. Profundizar en la lógica de las fuentes
 1. Identificar PK
 2. Comprobar los tipos de datos y tipos de fuente
 3. Comprobar relaciones
 4. Comprobar cardinalidades (tanto en relaciones como en columnas)
2. Profundizar en el contenido de las fuentes
 1. NULL y Formatos de fecha.
 2. Volumen de datos.
3. Comprender las reglas (técnicas) de negocio con los administradores de datos
 - Ej: El código del artículo en esta fuente es de 3 dígitos, pero en la otra fuente es de 6, completando con 0 a la izquierda..., los clientes sólo se extraen de esta fuente,...
4. Profundizar en el modelo en estrella
5. Validar los cálculos (KPIs, columnas derivadas, implícitos...)



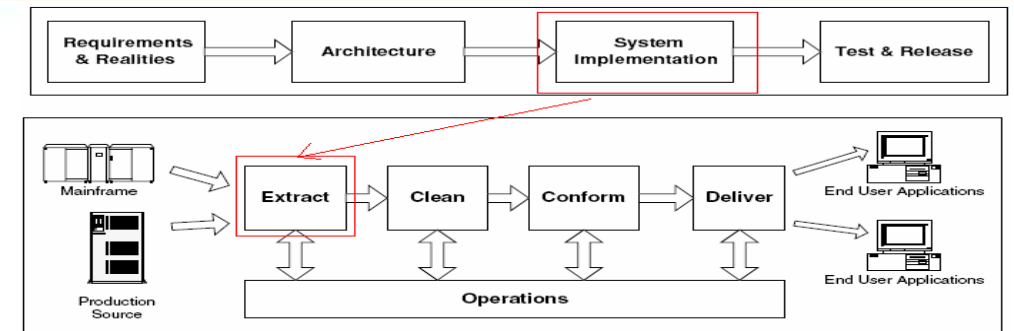
■ Flujo de Proyecto. Implementación

Flujo de datos. Extracción

• Identificación de fuentes:

- Tablas de datos
- Excel
- Ficheros planos
 - Delimitados
 - Longitud determinada
- XML
- Logs (como ficheros planos)
- ERP (tablas)

• ETC !!!



Flujo de Proyecto. Implementación

Flujo de datos. Limpieza y Conformado

- Son los pasos que mas valor aportan

(requiere un trabajo previo muy fuerte para definir las reglas)



- Limpieza

- Forzado de columnas:

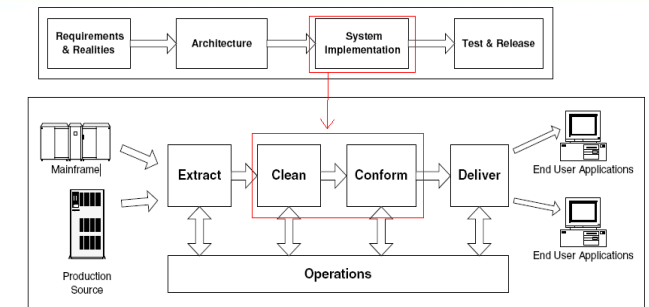
- Nulos donde no debe
 - Valores que se salen de los rangos
 - Tamaños de columna
 - Columnas con valores diferentes a los set discretos

- Forzado de estructuras:

- Relaciones entre tablas

- Forzado de datos y reglas de valor:

- ej.: si un cliente es VIP, su saldo asociado ha de ser XXXX.

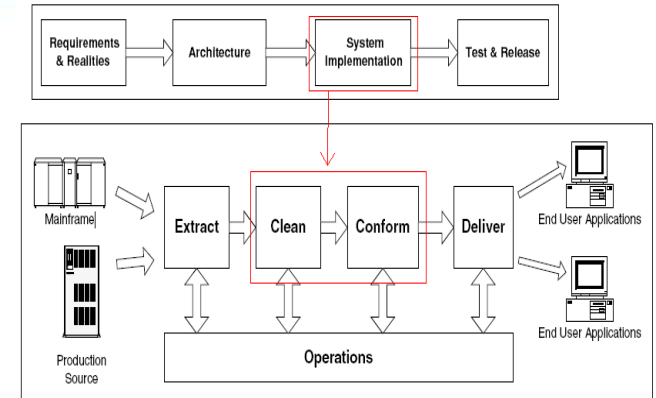


Flujo de Proyecto. Implementación

Flujo de datos. Extracción

● Conformado:

- ¿Qué es?
 - Un departamento llama al sexo de las personas (H,M), otro (1,2), etc...
- Importante: Una **dimensión**, en todas las estrellas tiene el mismo significado y atributos, y viene de las mismas fuentes.
- A nivel de ETL, el objetivo es, recibiendo las definiciones, **implementar los procesos**, no definir la dimensión.
- Importante: Un hecho esta conformado cuando **significa lo mismo para todos, se calcula igual** en todas las estrellas que interviene, y puede **intervenir directamente** en comparaciones y cálculos.
 - Ej.: Ingresos, ventas directas lo calcula al mes, suscripciones al año, etc... ¿Cómo los comparo?

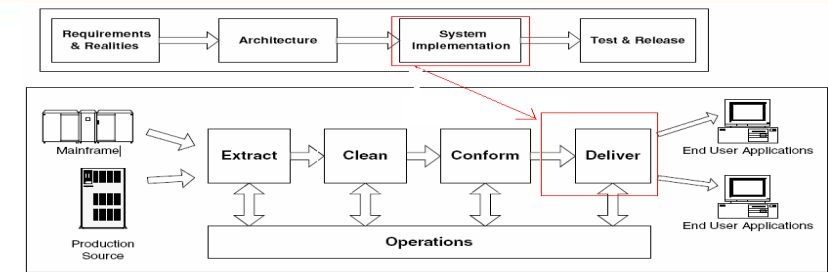


Dimensions

	Time	Products	Channels	Branches	Promotions	Customer	Shipping	Training
Sales	X	X	X	X		X		
Distribution	X		X				X	
Marketing	X	X			X	X		
HR	X			X			X	

■ Flujo de Proyecto. Implementación

Flujo de datos. Entrega. Dimensiones

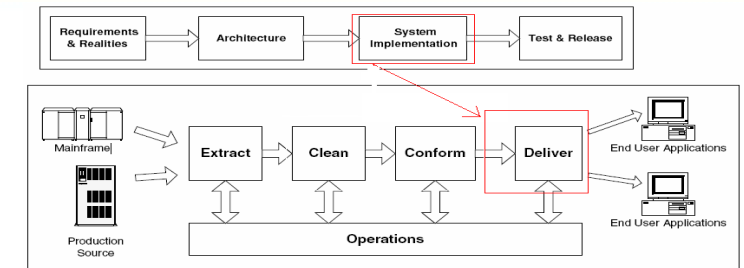


- Tantas combinaciones de acciones a tomar que es muy difícil generalizar:
 - Tipo de dimensión en cuanto a distribución
 - Tipo de dimensión en cuanto a cambio
- Recomendaciones generales:
 - Generar las claves surrogadas con secuencias
 - Utilizar estrategias Insert/Update
 - Seleccionar que campos hacen diferente a un elemento de dimensión de otro
 - Para las SCD, si la herramienta lo permite, dejar en sus manos el control de versiones.

■ Flujo de Proyecto. Implementación

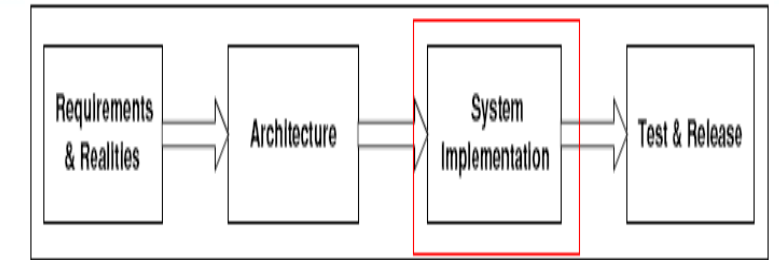
Flujo de datos. Entrega. Hechos

- A considerar:
 - Tipo de tabla de hechos
- Recomendación general:
 - Mantener la integridad referencial en el etl (lookup)
 - Tabla Lookup vs Dimension lookup (in memory solutions)
 - O con apoyo a una tabla del DataStage mediante Joins.
- Hechos:
 - Aditivos, Semi aditivos, No Aditivos. (Junk Dimensions)
 - Tipo de Agregación.
 - Hecho calculado en la ETL o al Vuelo.



■ Flujo de Proyecto. Implementación

Flujo de datos. Consideraciones de la Limpieza de Datos

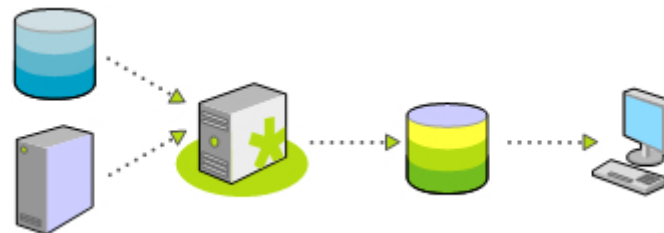
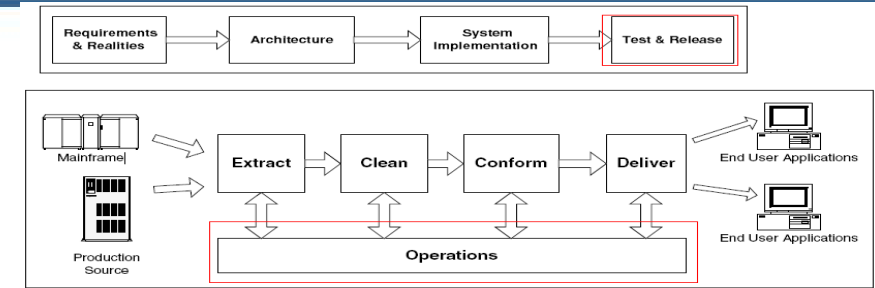


- **Filtrar las filas/columnas lo antes posible (no extraer filas/columnas inútiles)**
- **Particionar y paralelizar la ETL**
- **Reducir el tráfico de red (distribución de máquinas)**
- **Prepara la carga como BULK, algunos gestores lo permiten.**
- **Tunning de base de datos destino, elección del motor de almacenamiento.**
- **Programar la ETL, mediante la herramienta, o comandos (crontab -e)**

■ Flujo de Proyecto. Pruebas y Entregaas

Operaciones de la ETL son:

- Planificaciones (actuar ante eventos)
- Recibir las notificaciones del ETL y actuar en consecuencia
- Parametrizar las ETL (paso entre entornos, cambios)
 - Servidores, bases de datos, esquemas, directorios, emails, informes
- Monitorizar y minimizar fallos “físicos”
 - Red, memoria, base de datos, disco
- Monitorizar el rendimiento
 - Duración, filas leídas, escritas y procesadas por segundo, Throughput



■ Metadatos

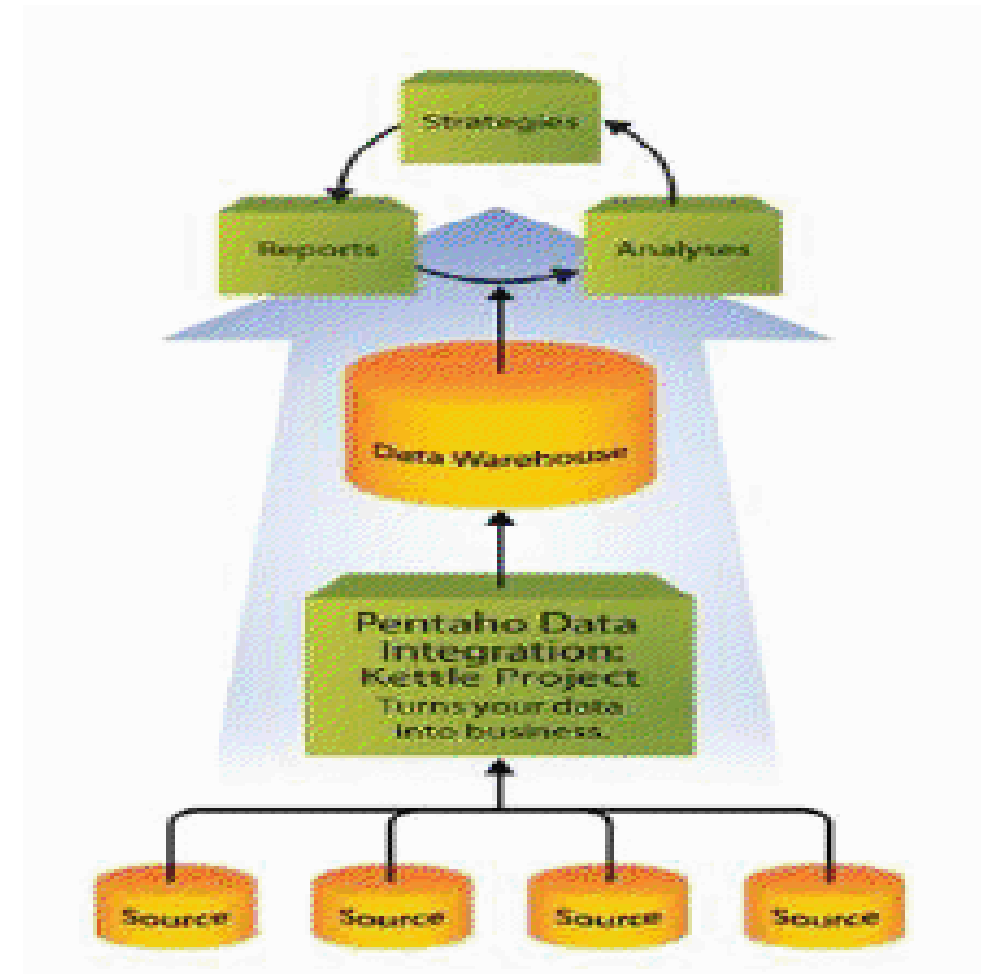
- Orígenes
 - Tablas, campos, reglas, modelos, relaciones, etc...
- Trabajos
 - Nombre, propósito, fuentes, destinos, tabla de rechazos, pre-procesos y post-procesos
- Transformaciones
 - Fuentes, lookups, filtros, rutas, agregaciones, uniones, etc...

Proceso:

- Resultados de la ejecución
- Tablas de auditoria
- Manejo de excepciones

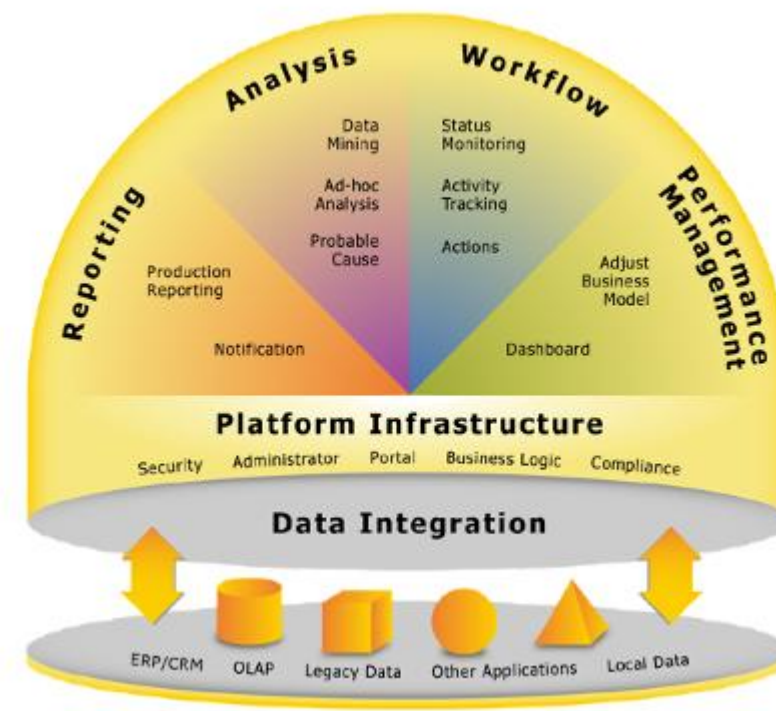
■ Pentaho Data Integration

- ◆ ¿Qué es Pentaho Data Integration?
- ◆ Características y beneficios
- ◆ Trabajando con PDI
 - Pestañas
 - Menú de Iconos
 - Componentes de PDI
 - Pasos de las transformaciones
 - Variables de Entorno
 - Ejemplos



■ ¿Qué es Pentaho Data Integration?

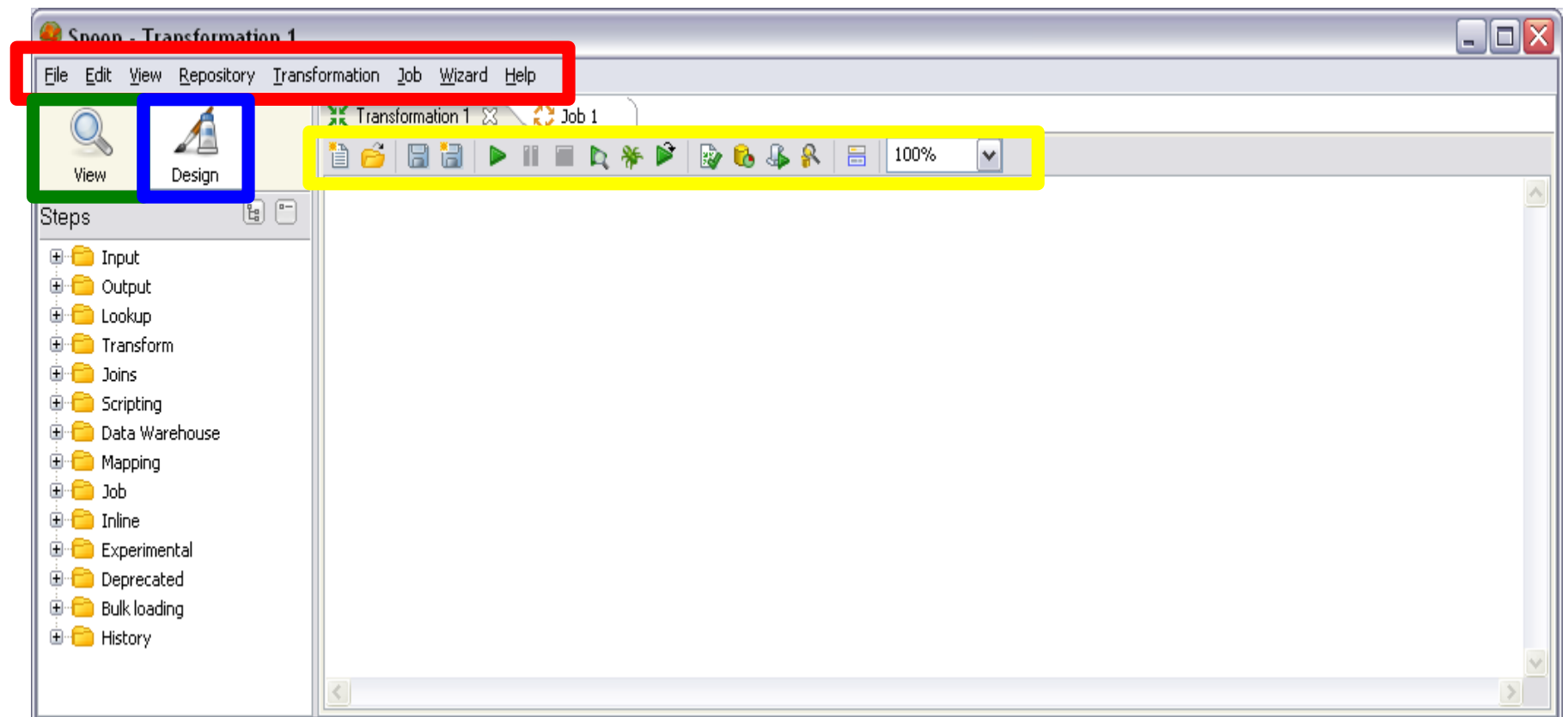
- **PDI** es un set de herramientas, que permite diseñar ETLs, mediante transformaciones y trabajos que pueden ser ejecutadas por las herramientas de **Spoon**, **Pan** y **Kitchen**. Antes se le conocía con el nombre de **Kettle**.
- **Spoon** interfaz gráfica para diseño de transformaciones y trabajos ETL.
- **Pan** es un motor capaz de ejecutar múltiples transformaciones de datos como leer, manipular y escribir desde y en distintos orígenes de datos.
- **Kitchen** es un programa que ejecuta los trabajos diseñados por **Spoon**. Normalmente estos trabajos son planificados en modo batch para ejecutar automáticamente a periodos regulares (crontab -e).



■ Características y Beneficios

- Permite trabajar con un repositorio en Base de Datos o en Ficheros.
- Su interfaz gráfica te permitirá crear de transformaciones y trabajos de manera intuitiva mediante pasos modulares ya creados, conexiones con múltiples fuentes, etc...
- Distribución y combinación de diferentes fuentes, en diferentes hosts.
- Interfaz SQL y generador de código automático.
- Crear cálculos de una manera muy sencilla.
- Define que quieres hacer, no como quieres hacerlo.
- Genera código XML y Java.
- Instalación sencilla - sólo extraer los ficheros, aplicación Java. (ojo con la versión java -version)
- Fácil de mantener, con alto rendimiento y escalabilidad.
- Es posible parametrizar bastantes configuraciones (directorios, conexiones, mail).
- Posee una arquitectura de Plug-in que te permitirá expandir sus funcionalidades.

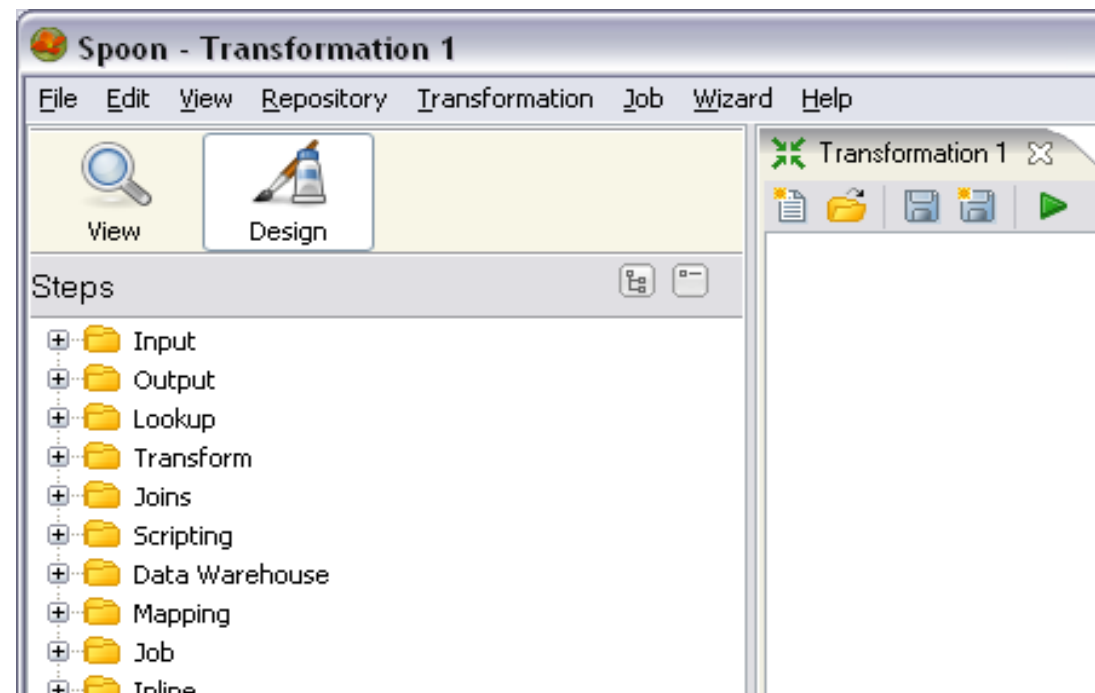
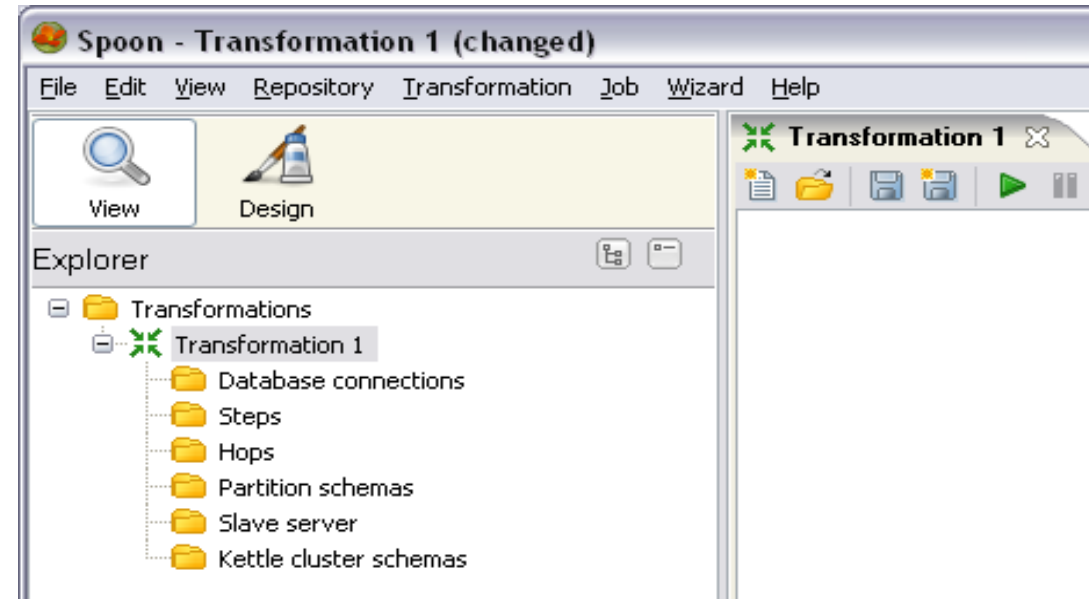
- Menú principal (rojo)
- Pestaña vista (verde)
- Pestaña design (azul)
- Menú iconos (amarillo)
- Zona de Trabajo















■ Pestañas

- Pestaña Vista (View):
 - Orígenes de Datos.
 - Pasos
 - Saltos
 - Esquemas
 - Servidores Esclavos
 - Esquemas en Cluster.

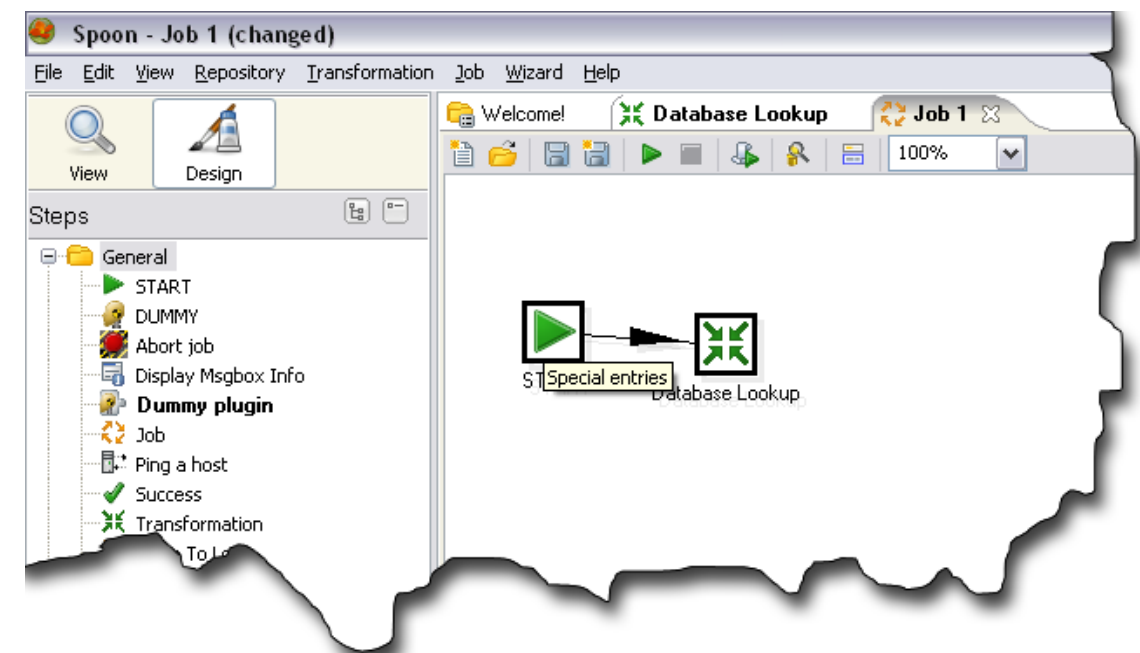
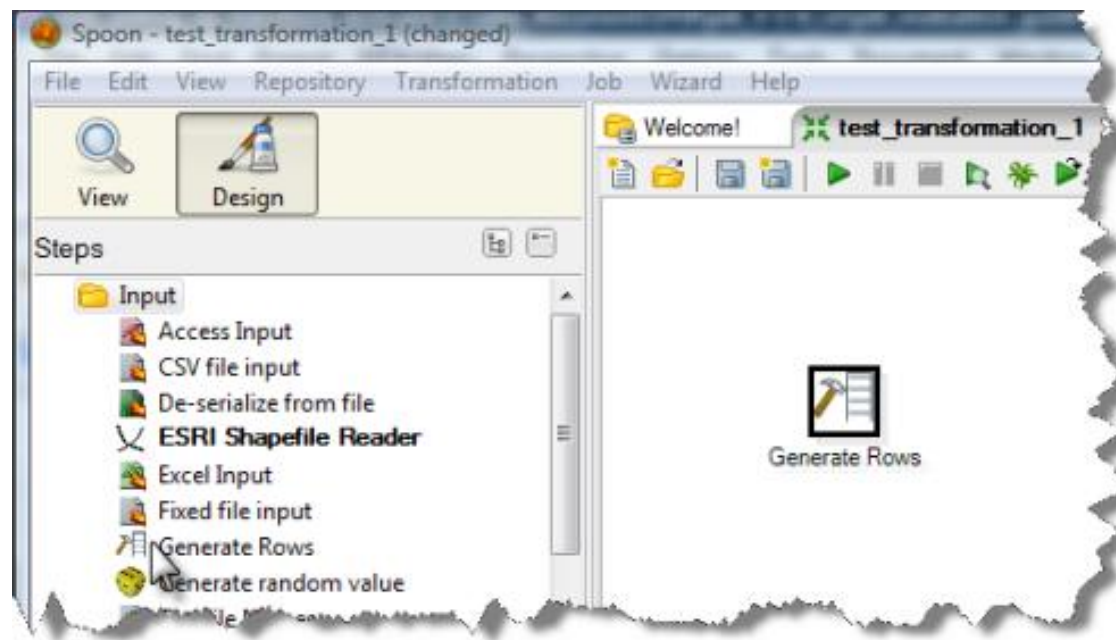
- Pestaña Diseño (Design):
 - Entrada
 - Salida
 - Búsqueda
 - Transformar
 - Uniones
 - Scripting
 - Data Warehouse
 - Mapeado
 - Trabajo
 - Embebido
 - Experimental



Icono	Descripción
	Crear un nuevo trabajo o transformación o CNTRL - N
	Abrir un trabajo/transformación de un fichero o del repositorio si estas conectado a él.
	Guardar el trabajo/transformación a un fichero o al repositorio
	Guardar el trabajo/transformación con un nombre distinto.
	Abrir la ventana de impresora.
	Ejecutar el trabajo/transformación: ejecuta la transformación actual desde el fichero XML o el repositorio.
	Previsualizar la transformación: ejecuta la transformación actual desde memoria. Puedes previsualizar las filas producidas por el paso seleccionado
	Ejecutar la transformación en modo de pruebas permitiéndote la solución de errores de ejecución.
	Repetir el proceso de una transformación para una cierta fecha y hora. Esto causará que ciertos pasos (TextFile Input y Excel Input) sólo procesarán las filas que fallaron para ser interpretadas correctamente a esa fecha y hora particular.
	Ejecutar un análisis de impacto: que impacto tiene la transformación en la base de datos usada.
	Generar el SQL que es necesario para ejecutar la transformación.
	Lanza el explorador de la base de datos permitiéndote previsualizar los datos, ejecutar consultas SQL, generar DDL y más.

■ Componentes de PDI (I)

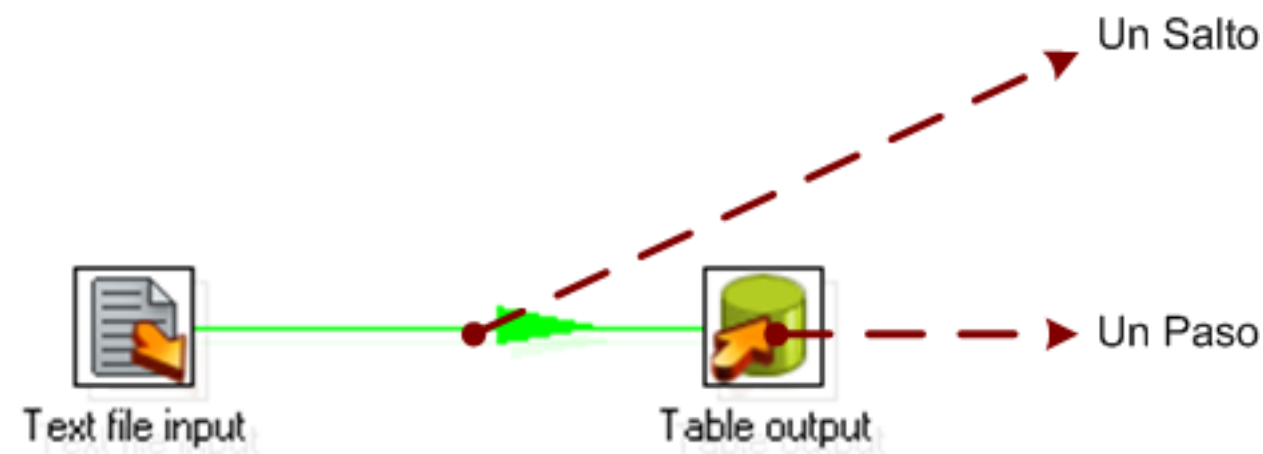
- Los procesos ETL se dividen en dos componentes principales:
 - **Transformaciones (.ktr):** es el conjunto de pasos básicos que componen el nivel más bajo de una ETL.
 - **Trabajos (.kjb):** es un conjunto de pasos, trabajos y transformaciones.



■ Componentes de PDI (II)

- Transformaciones:

- **Paso:** son los elementos atómicos de PDI y cada uno realiza una transformación en el flujo de datos. (Leer datos, escribir en BBDD, crear cálculos, añadir constantes,)
- **Salto:** es la representación gráfica del flujo de datos entre 2 pasos.



■ Componentes de PDI (III)

• Trabajos:

- **Paso:** son los elementos atómicos de PDI y cada uno realiza una trabajo. (No modifican el flujo de datos)
- **Salto:** representa el orden de ejecución de transformaciones y trabajos.
- **Trabajo y Transformación:** dentro de un trabajo podemos incluir llamadas a otras transformaciones y/o trabajos para que sean ejecutadas.
- Un **trabajo** procesa todos los registros antes de continuar, en cambio, una **transformación** es un flujo de datos continuo de manera que los registros avanzan por los pasos según llegan.

