

Spring Batch

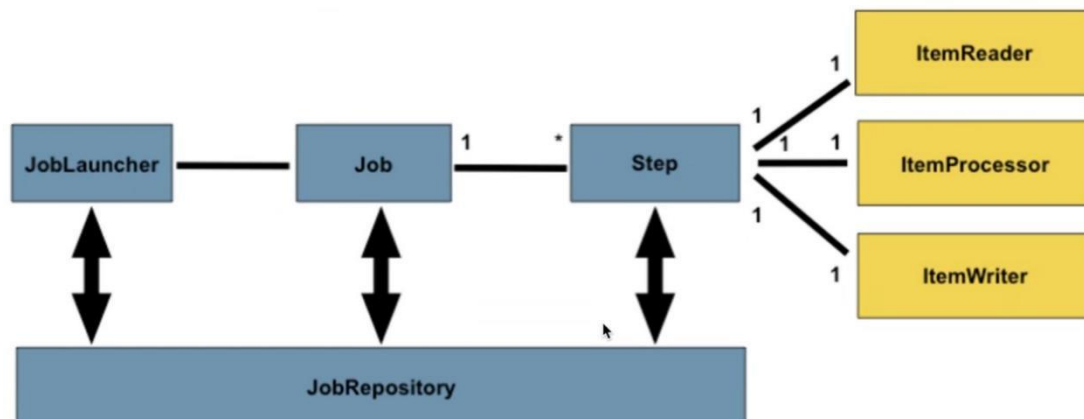
Spring Batch es un framework de procesamiento por lotes (batch processing) que permite ejecutar tareas grandes y estructuradas en pasos (Steps), tal como:

- ◆ Procesar archivos grandes
- ◆ Cargar datos a bases de datos
- ◆ Generar reportes
- ◆ Transformar información en masa

¿Cuándo usar Spring Batch?

Cuando se realizarán tareas que:

- Procesarán una gran cantidad de datos.
- Se ejecuten de forma automática (Periódicamente, en horarios o eventos).
- Tienen pasos tales como: leer -> procesar -> escribir.
- Son propensos a fallar, reiniciarse o reintentarse.



Se muestra cómo **Spring divide el trabajo** en pasos y cómo cada parte se comunica para realizar tareas de forma eficiente y controlada.

JobLauncher:

Inicia la ejecución de un Job. El desarrollador o el sistema lo llaman cuando hay que procesar.

Job:

Es la **tarea completa**, un conjunto de pasos. (El Job puede tener uno o mas Steps)

Step:

Cada Step es una parte del Job, cuenta con 3 partes fundamentales:

- 1) **ItemReader**: lee los datos (de archivo, DB, API, etc.)
- 2) **ItemProcessor**: transforma o valida los datos
- 3) **ItemWriter**: guarda los datos en el destino final

JobRepository:

Es la base de datos interna que guarda el estado de las ejecuciones (por si falla, poder reiniciar desde donde se quedó).

El **JobRepository** está conectado a todos los componentes, ya que:

- Guarda cuándo se ejecutó un Job
- Guarda si falló y dónde
- Guarda estadísticas de progreso

Una forma más clara de verlo es la siguiente:

“Lee un archivo CSV con 10,000 productos, limpia los datos y guárdalos en MySQL”.

- ✓ **ItemReader**: abre y lee el archivo línea por línea.
- ✓ **ItemProcessor**: valida cada producto (¿precio correcto?, ¿categoría válida?).
- ✓ **ItemWriter**: guarda productos válidos en la base de datos.

¿Porqué usar Spring batch?

Ventaja	Importancia
Control total del flujo	Puedes definir paso a paso el proceso
Reinicio automático	Si falla, no comienza desde cero
Escalabilidad	Puede dividir el trabajo en chunks (por bloques)
Transacciones seguras	Guarda el progreso y evita perder datos

Alumno: Edgar Uriel Marín Gómez

Academia Xideral

Fecha: 03 de Agosto de 2025

Spring Batch es ideal para **procesar datos en bloques (batch)**, donde se necesita fiabilidad, monitoreo y estructura.