

Spring Batch

Danna Valeria Morales Aguilar

JobLauncher

ooo



Job

Repository



Step

¿QUÉ ES SPRING BATCH?

Es un framework ligero enfocado en la creación de procesos batch, marca directrices para el diseño de procesos y da una gran cantidad de componentes que intentan dar soporte a las distintas necesidades que surgen a la hora de crear programas.



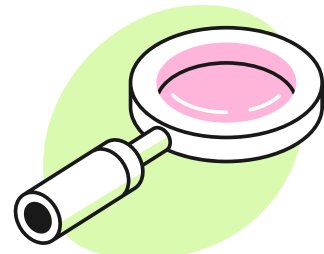
spring
Batch
Spring Boot Based



COMPONENTES PRINCIPALES

```
<job-repository id = "jobRepository"
  data-source = "dataSource"
  transaction-manager = "transactionManager"
  isolation-level-for-create = "SERIALIZABLE"
  table-prefix = "BATCH_"
  max-varchar-length = "1000"/>
```

JOBREPOSITORY



Se encarga de la persistencia de metadatos relativos a los procesos en curso o estados de las ejecuciones.

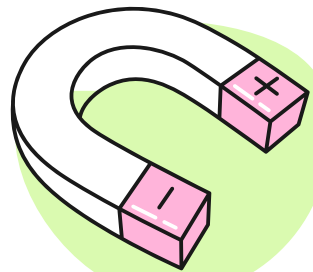
```
<bean id = "jobLauncher"
  class = "org.springframework.batch.core.launch.support.Simp
  <property name = "jobRepository" ref = "jobRepository" />
</bean>
```

JOBLAUNCHER



Encargado de lanzar los procesos suministrando los parámetros de entrada deseados.

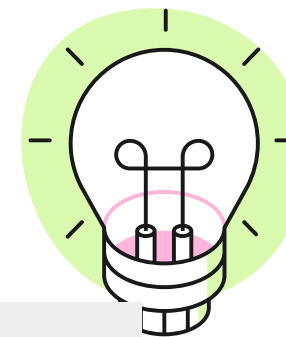
JOB



Es la representación del proceso. Un proceso es un contenedor de pasos (steps).

```
<job id = "jobid">
  <step id = "step1" next = "step2"/>
  <step id = "step2" next = "step3"/>
  <step id = "step3"/>
</job>
```

STEP



Elemento independiente dentro de un Job que es una de las fases de las que esta compuesto dicho proceso.

ELEMENTOS DE UN STEP

ITEMREADER

Elemento responsable de leer datos de una fuente de datos como BBDD, fichero, cola de mensajes, etc.

ITEMPROCESSOR

Elemento responsable de tratar la información obtenida por el reader, no es obligatorio su uso.

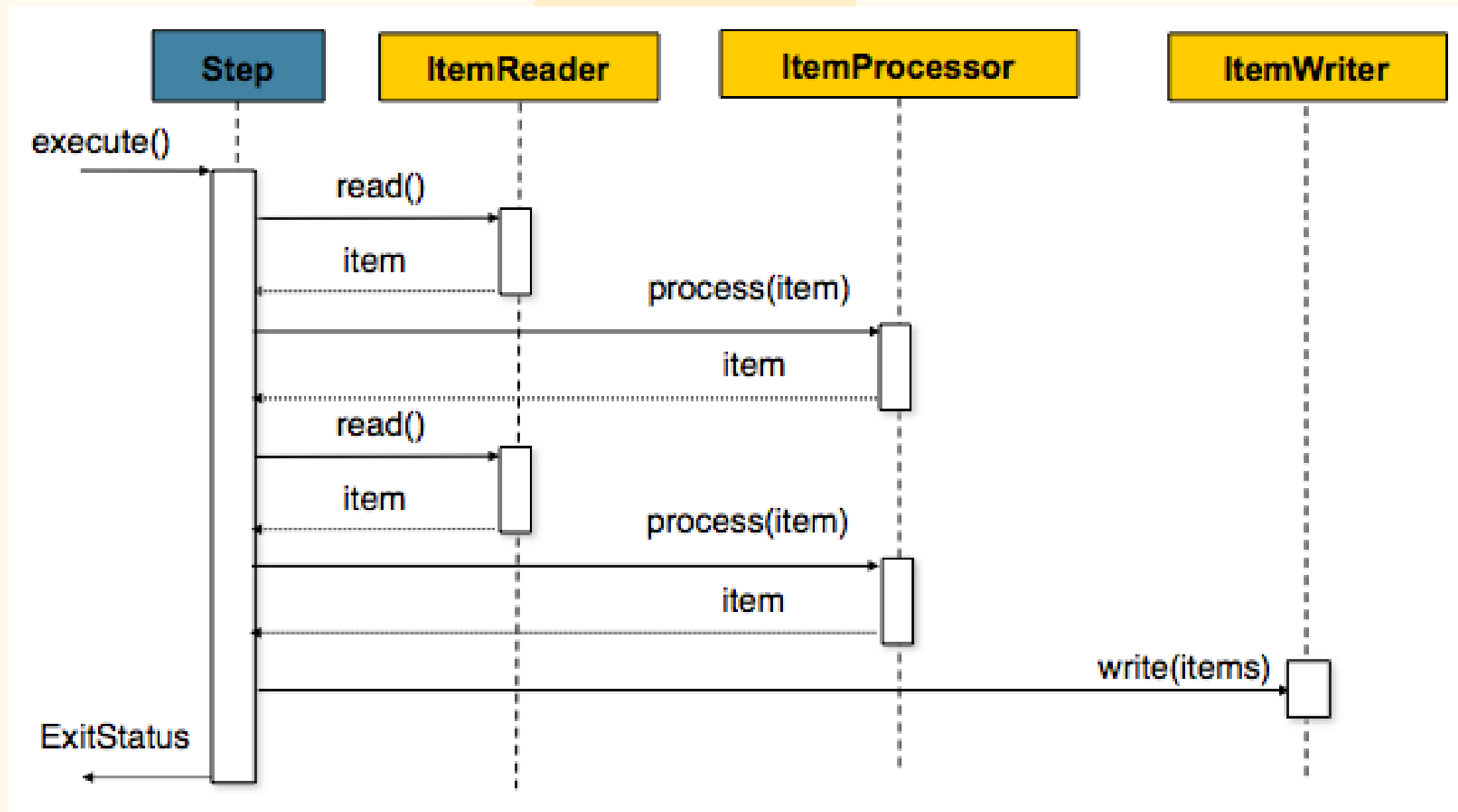
ITEMWRITER

Elemento responsable de guardar la información leída por el reader o tratada por el processor, si hay un reader debe haber un writer.

STEP



DIAGRAMA





CHUNK-ORIENTED PROCESSING

Es una técnica utilizada para la ejecución de las fases de un proceso.

Su funcionamiento es el siguiente:

- El reader lee una porción de datos de la fuente de datos y los convierte en un "chunk".
- Si existe un processor, ese chunk pasa al processor para que lo trate.
- Todo esto se realiza dentro de un limite transaccional, o sea leemos y tratamos tantos chunks como queramos antes de que sean persistidos por el writer.



EJEMPLO

Suponiendo que se tiene un fichero en texto plano, debemos tratar cada línea para luego mantener la información en una base de datos.

Si configuramos nuestro step con un intervalo de commit igual a 10, lo que haría sería leer una línea del fichero y tratarla, hasta 10 veces.

Una vez que ya hemos leído y tratado 10 líneas, el writer recibe la información (10 chunks) y los mantiene en la base de datos.