Atlanta Java Users Group
20 Aug 2013

Integrating Spring Batch and Spring Integration

By Gunnar Hillert
Twitter: @ghillert

What we will cover...

- Spring **Batch**
- Spring **Integration**
- Spring **Batch Integration**
- Spring **XD**

Pivotal™

The Pivotal One Platform

- **Application** Fabric
  - Languages, Frameworks, Services, Analytics

PIVOTAL LABS

GRAILS

spring

Groovy

- **Data** Fabric
  - High Capacity, Real-time, Ingest & Query, Scale-out, Storage

hadoop

PIVOTAL HD

redis

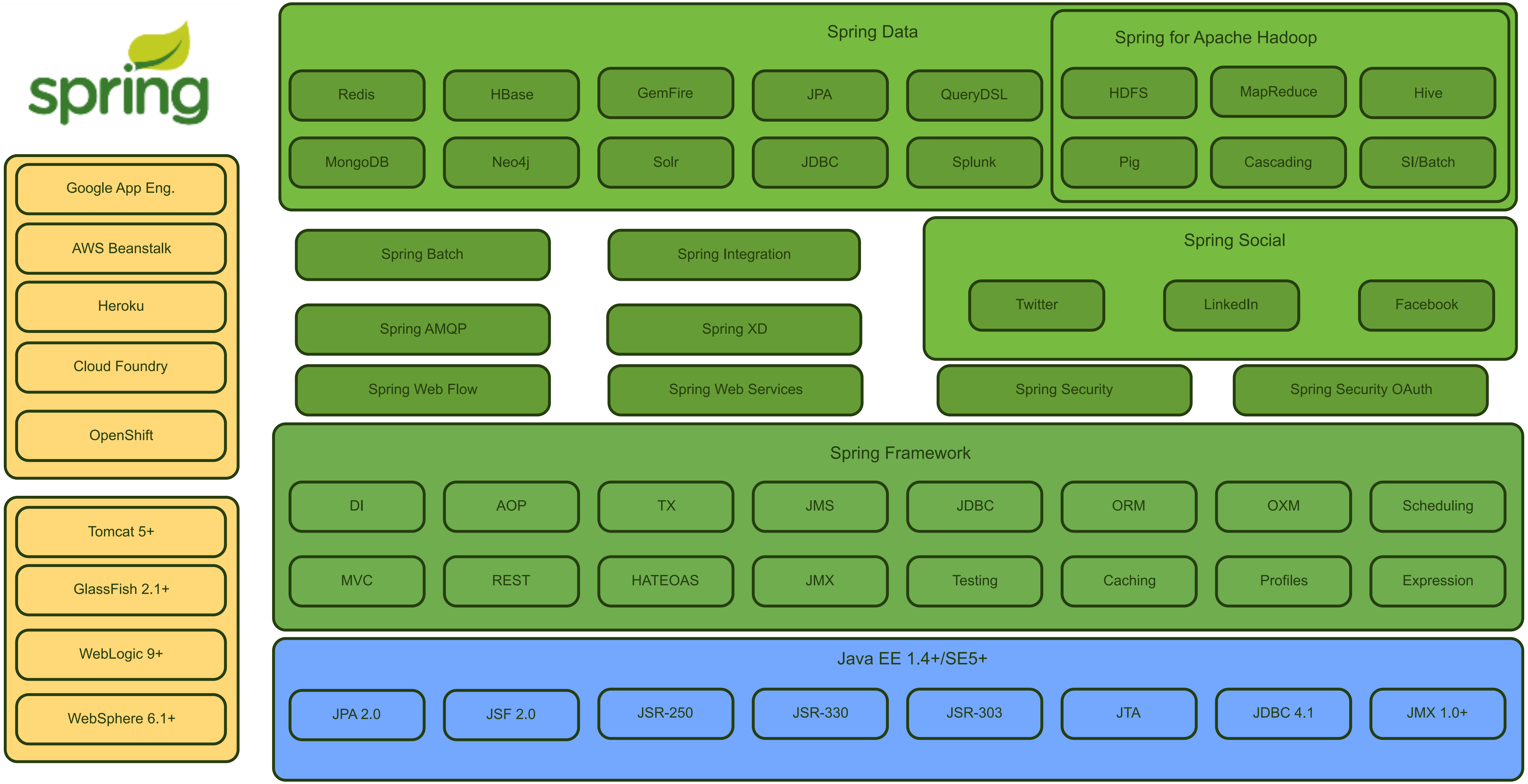
- **Cloud** Fabric
  - Automation, Service Registry, Cloud Independence

CLOUD FOUNDRY™

GemFire

RabbitMQ
Open Source Enterprise Messaging

Spring Stack



Spring Data

Redis, HBase, GemFire, JPA, QueryDSL

MongoDB, Neo4j, Solr, JDBC, Splunk

Spring for Apache Hadoop

HDFS, MapReduce, Hive

Pig, Cascading, SI/Batch

Spring Batch

Spring Integration

Spring Social

Twitter, LinkedIn, Facebook

Spring AMQP

Spring XD

Spring Web Flow

Spring Web Services

Spring Security

Spring Security OAuth

Spring Framework

DI, AOP, TX, JMS, JDBC, ORM, OXM, Scheduling

MVC, REST, HATEOAS, JMX, Testing, Caching, Profiles, Expression

Java EE 1.4+/SE5+

JPA 2.0, JSF 2.0, JSR-250, JSR-330, JSR-303, JTA, JDBC 4.1, JMX 1.0+

Google App Eng.

AWS Beanstalk

Heroku

Cloud Foundry

OpenShift

Tomcat 5+

GlassFish 2.1+

WebLogic 9+

WebSphere 6.1+

Spring Batch

http://www.springsource.org/spring-batch

> “ **Batch processing** ... is defined as the processing of data without interaction or interruption.

Michael T. Minella, Pro Spring Batch

Batch Jobs

- Long-running
  - Often outside office hours
- Non-interactive
  - Often include logic for handling errors or restarts
- Process large volumes of data
  - More than fits in memory or a single transaction

Batch and offline processing

- Close of business processing
  - Order processing, Business reporting, Account reconciliation
- Import/export handling
  - a.k.a. ETL jobs (Extract-Transform-Load)
  - Instrument/position import
  - Data warehouse synchronization
- Large-scale output jobs
  - Loyalty scheme emails, Bank statements

Features

- Transaction management
- Chunk based processing
- Declarative I/O
- Start/Restart/Skip capabilities
- Web administration interface
- Based on the Spring framework
- JSR 352: Batch Applications for the Java Platform

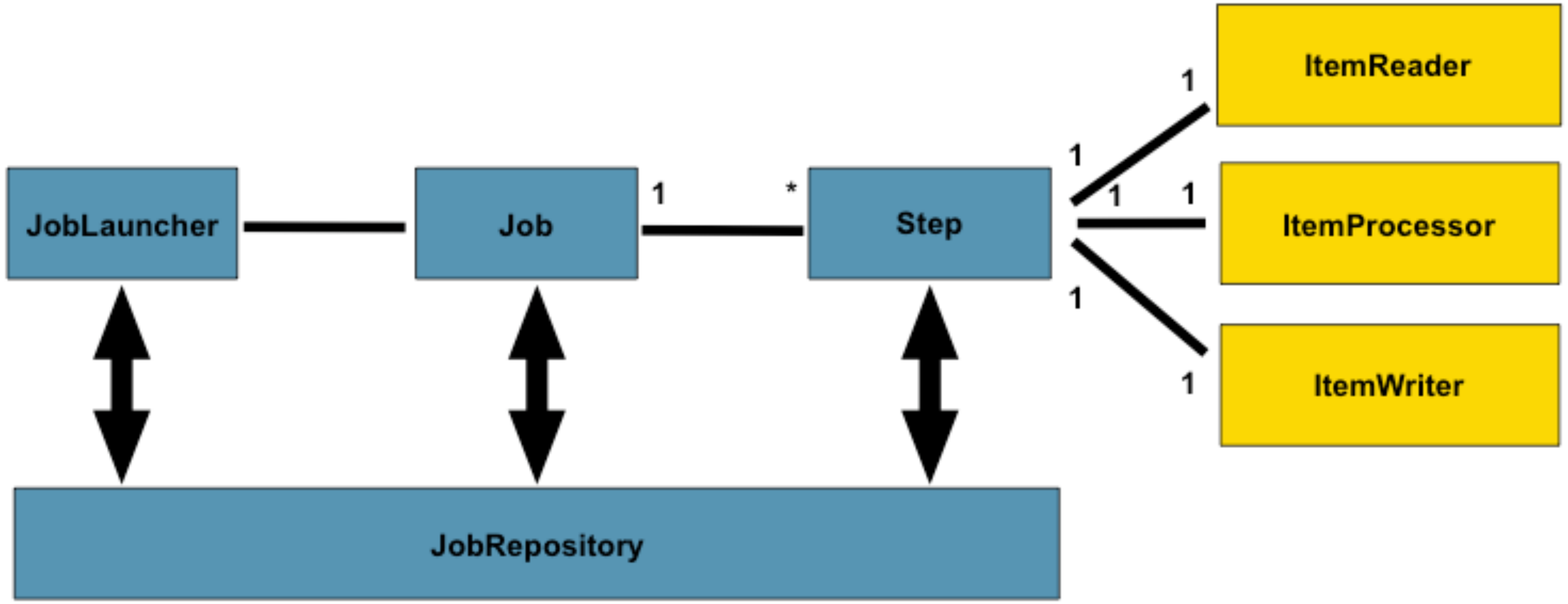
Demo

Hello Spring Batch

https://github.com/ghillert/spring-batch-sample-hello

Concepts

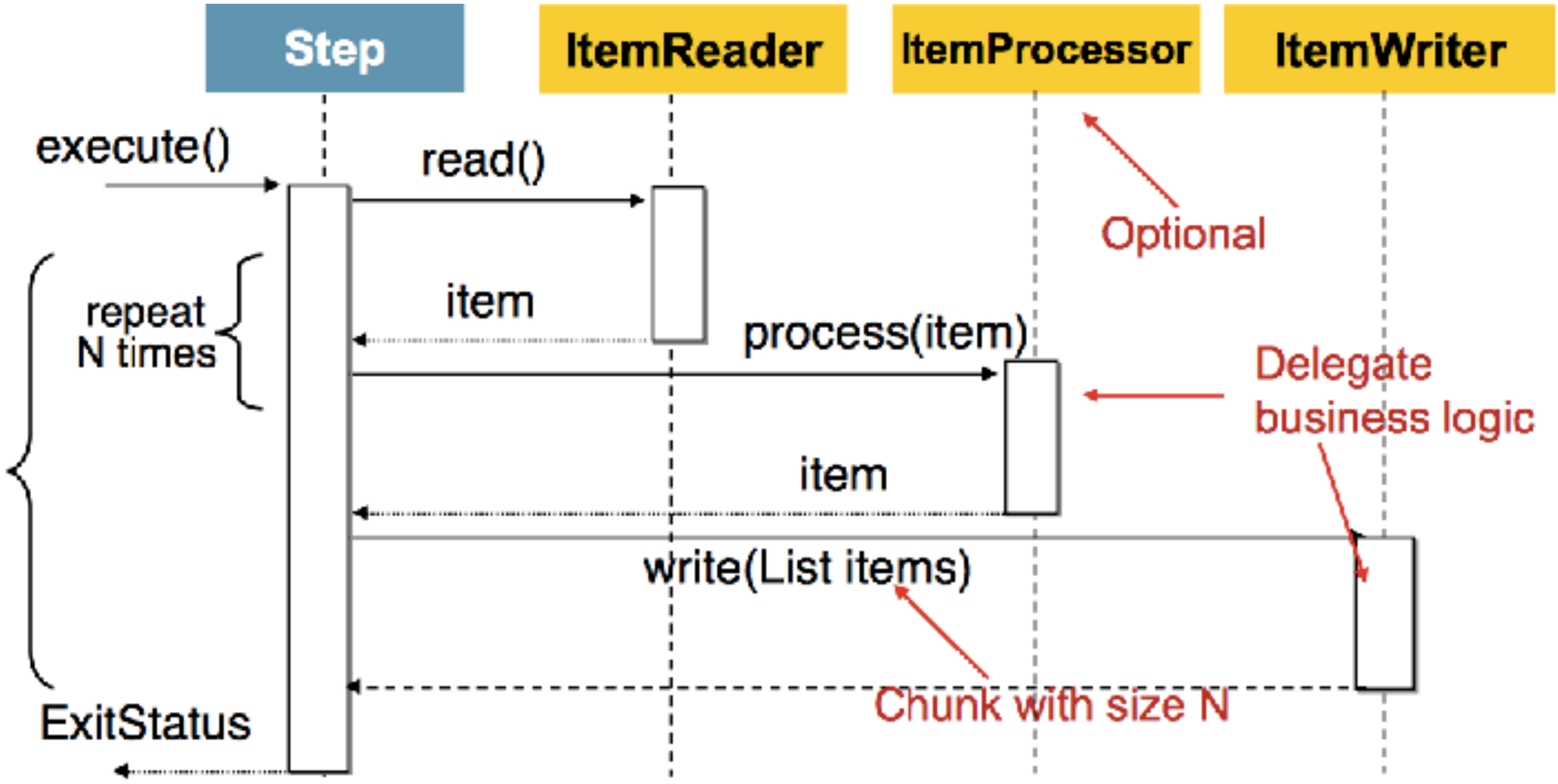
- Job
- Step
- Item
- Chunk



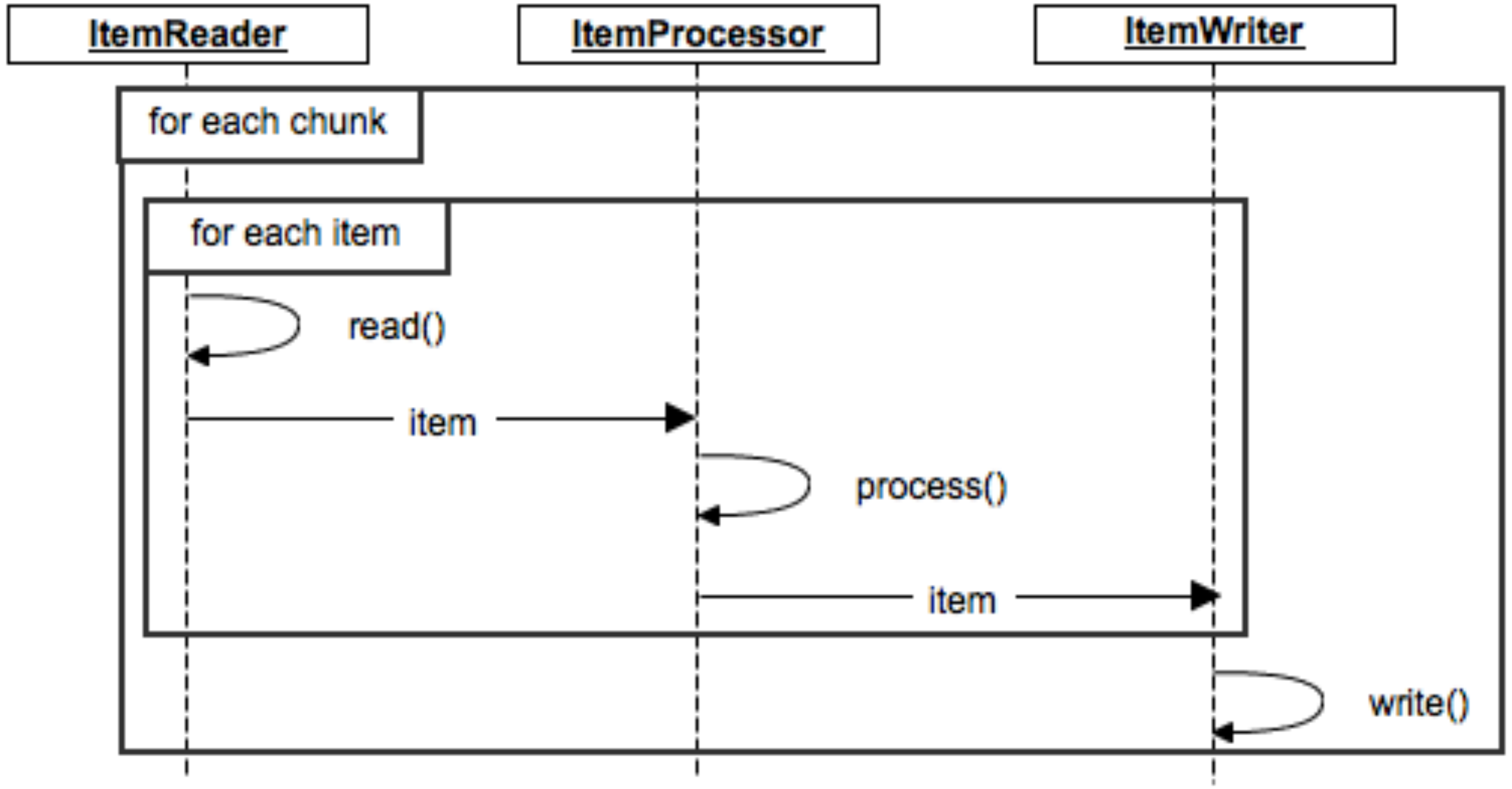
Repeat | Retry | Skip | Restart

Chunk-Oriented Processing

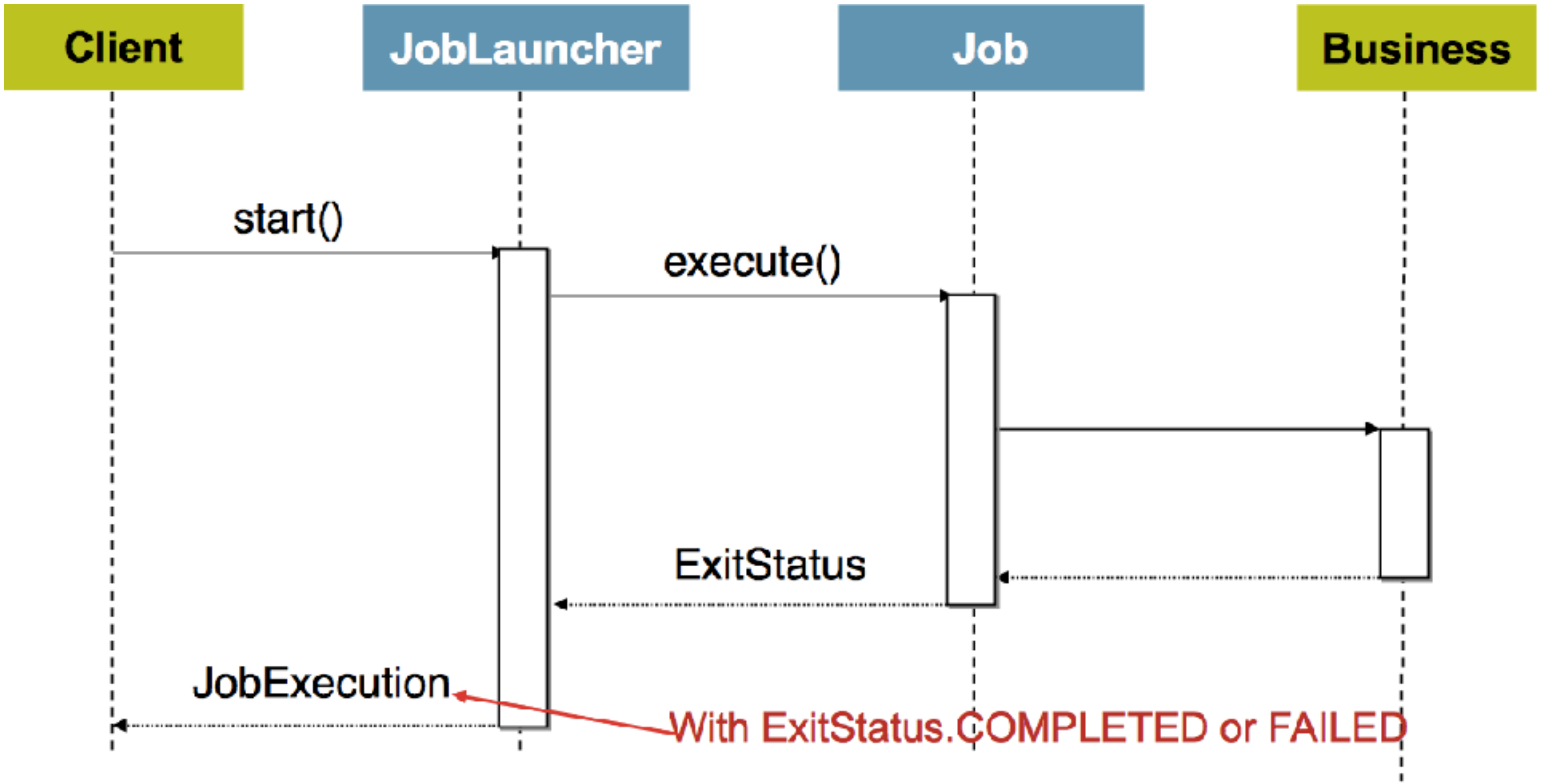
- Input-output can be grouped together
- Input collects Items before outputting: Chunk-Oriented Processing
- Optional ItemProcessor



Chunk-Oriented Processing

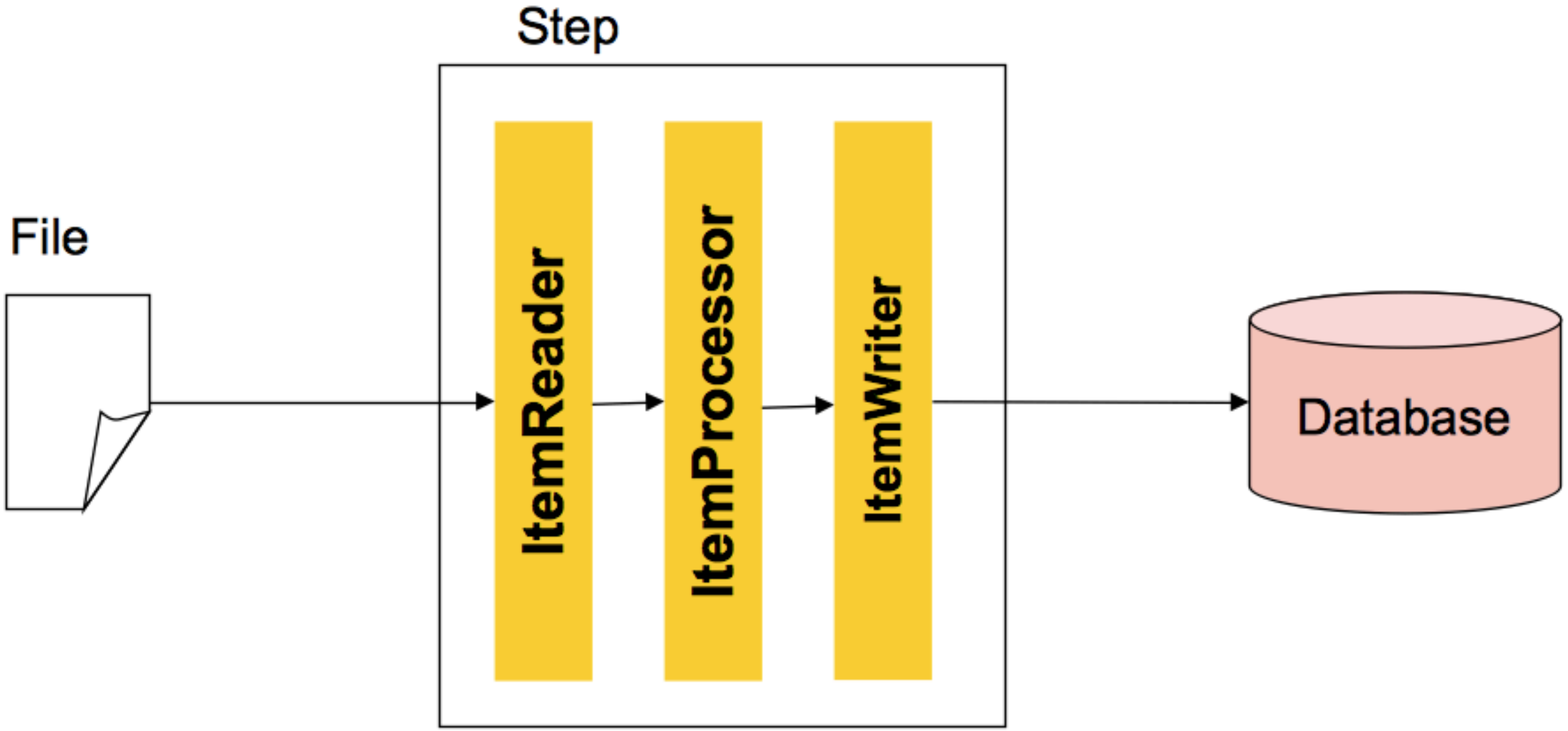


JobLauncher



With ExitStatus.COMPLETED or FAILED

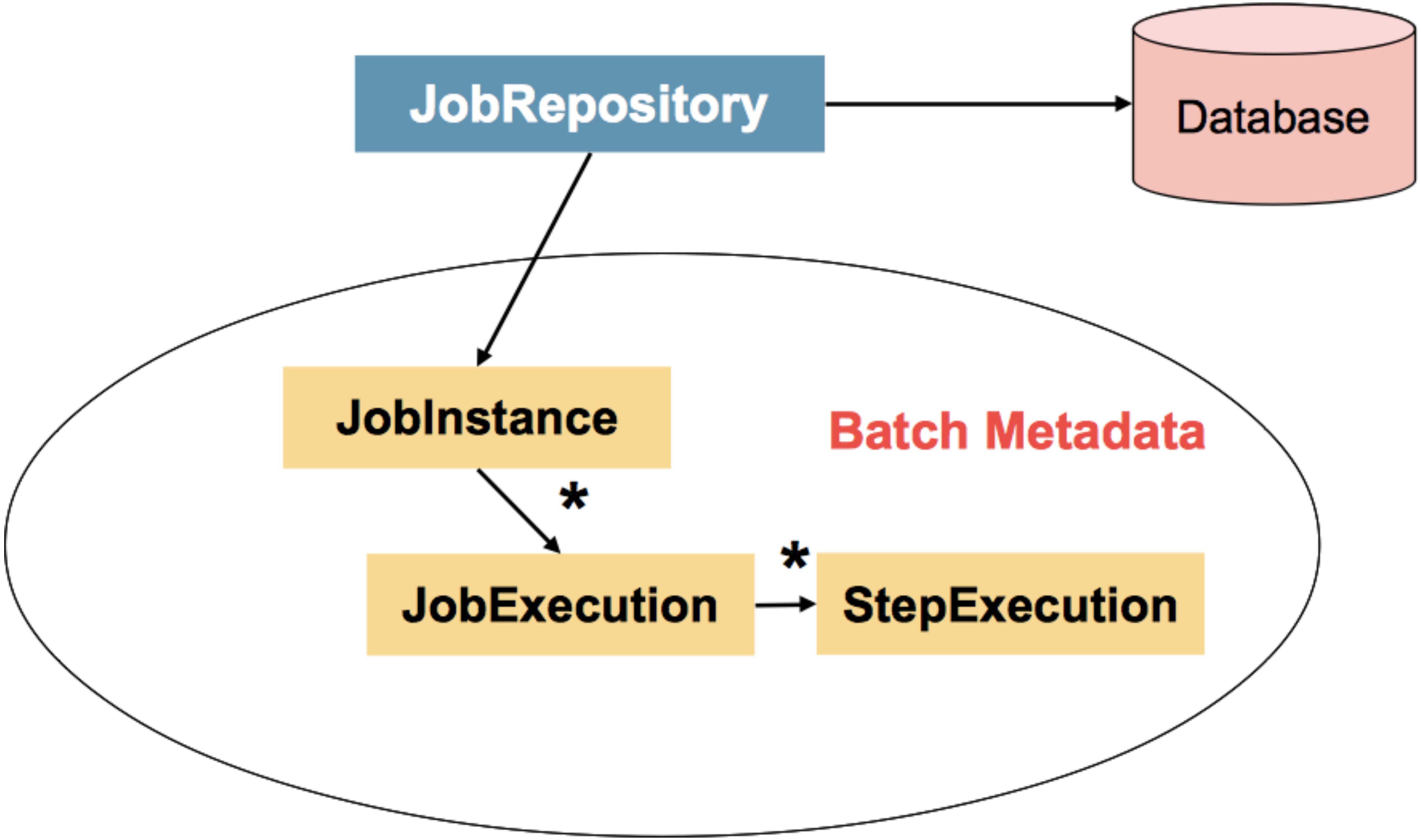
Simple File Load Job



ItemReaders and ItemWriters

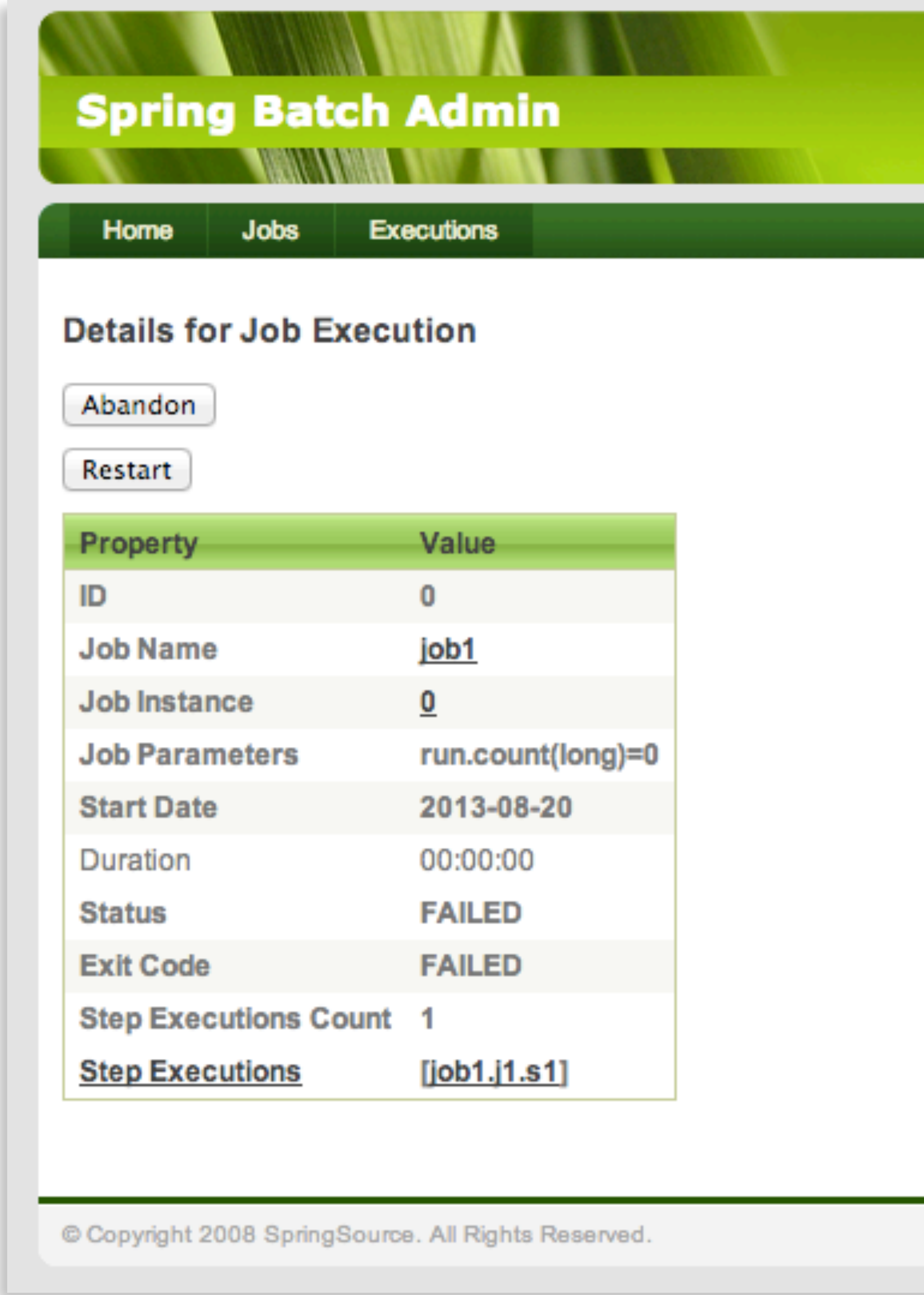
- Flat File
- XML (StAX)
- Multi-File Input
- Database
  - JDBC, JPA/Hibernate, Stored Procedures
- Implement your own...

Job Repository



Spring Batch Admin

- Sub project of Spring Batch
- Provides Web UI and REST interface to manage batch processes
- Manager, Resources, Sample WAR
  - Deployed with batch job(s) as single app to be able to control & monitor jobs
  - Or monitors external jobs only via shared database



Details for Job Execution

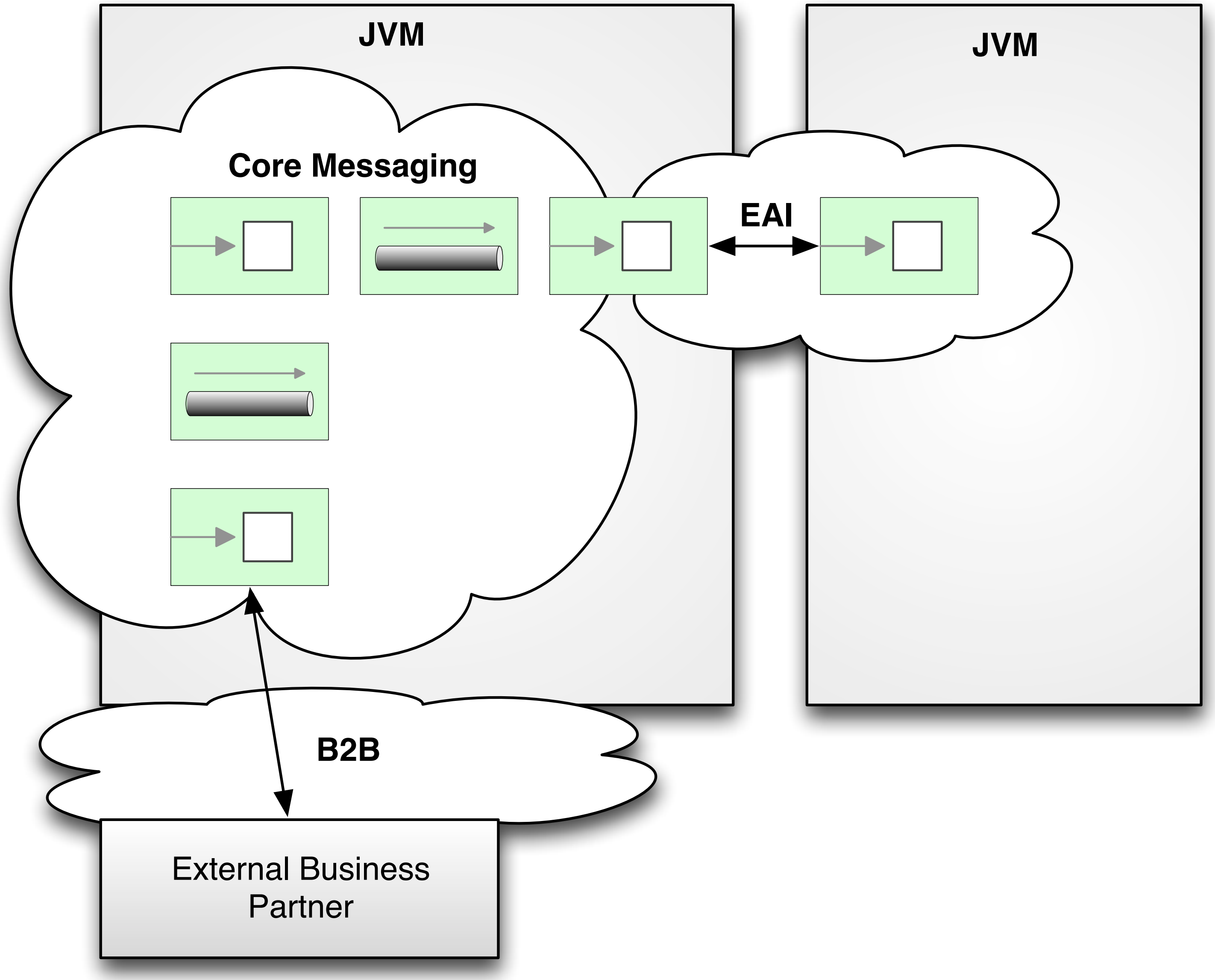
Property	Value
ID	0
Job Name	job1
Job Instance	0
Job Parameters	run.count(long)=0
Start Date	2013-08-20
Duration	00:00:00
Status	FAILED
Exit Code	FAILED
Step Executions Count	1
Step Executions	[job1.j1.s1]

Spring Integration

http://www.springsource.org/spring-integration

Integration Styles

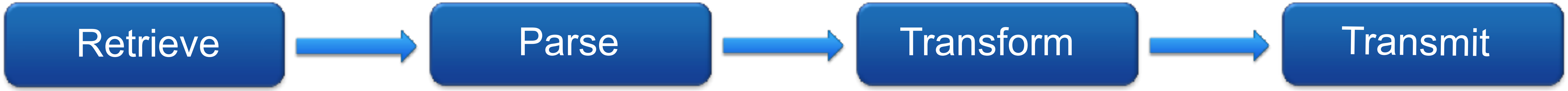
- Business to Business Integration (B2B)
- Inter Application Integration (EAI)
- Intra Application Integration



Integration Styles

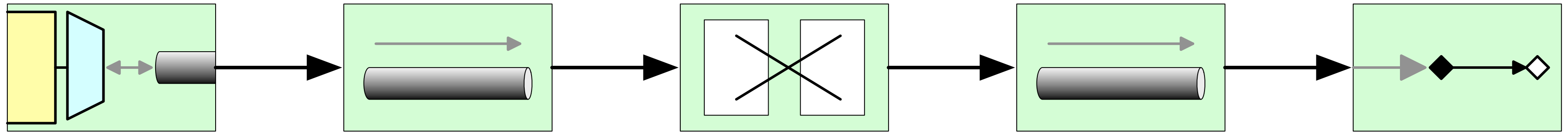
- File Transfer
- Shared Database
- Remoting
- Messaging

Common Patterns

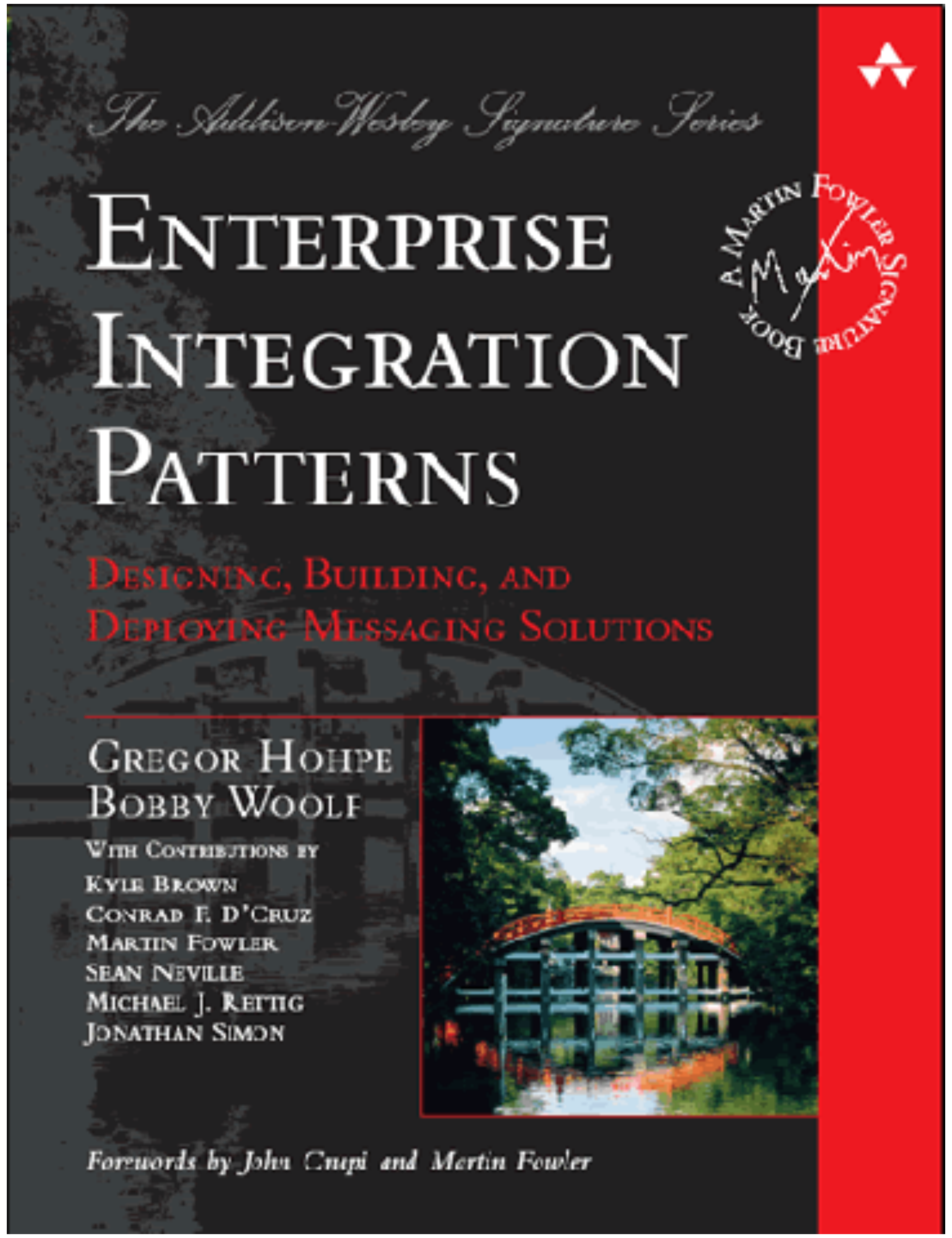


Enterprise Integration Patterns

- By Gregor Hohpe & Bobby Woolf
- Published 2003
- Collection of well-known patterns
- Icon library provided



http://www.eaipatterns.com/eaipatterns.html


The Addison-Wesley Signature Series
Enterprise Integration Patterns
Designing, Building, and Deploying Messaging Solutions
A Martin Fowler Signature Book
Gregor Hohpe
Bobby Woolf
With Contributions by
Kyle Brown
Conrad F. D'Cruz
Martin Fowler
Sean Neville
Michael J. Rettig
Jonathan Simon
Forewords by John Crupi and Martin Fowler

> “ Spring Integration provides an extension of the Spring programming model to support the well-known enterprise integration patterns.

Spring Integration Website

Spring Integration Components

- Claim Check (In/Out)
- Content Enricher
- Header Enricher
- Payload Enricher
- Control Bus
- Delayer
- JMX Support
- Message Handler Chain
- Messaging Bridge
- Resequencer

- Service Activator
- Scripting support (JSR 223)
- Ruby/JRuby, Javascript ...
- Groovy
- Message History
- Message Store
- JDBC, Redis, MongoDB, Gemfire
- Wire Tap
- ...

Adapters

- AMQP/RabbitMQ
- AWS*
- File/Resource
- FTP/FTPS/SFTP
- GemFire
- HTTP (REST)
- JDBC
- JMS
- JMX
- JPA

- MongoDB
- POP3/IMAP/SMTP
- Print*
- Redis
- RMI
- RSS/Atom
- SMB*
- Splunk*
- Spring Application Events

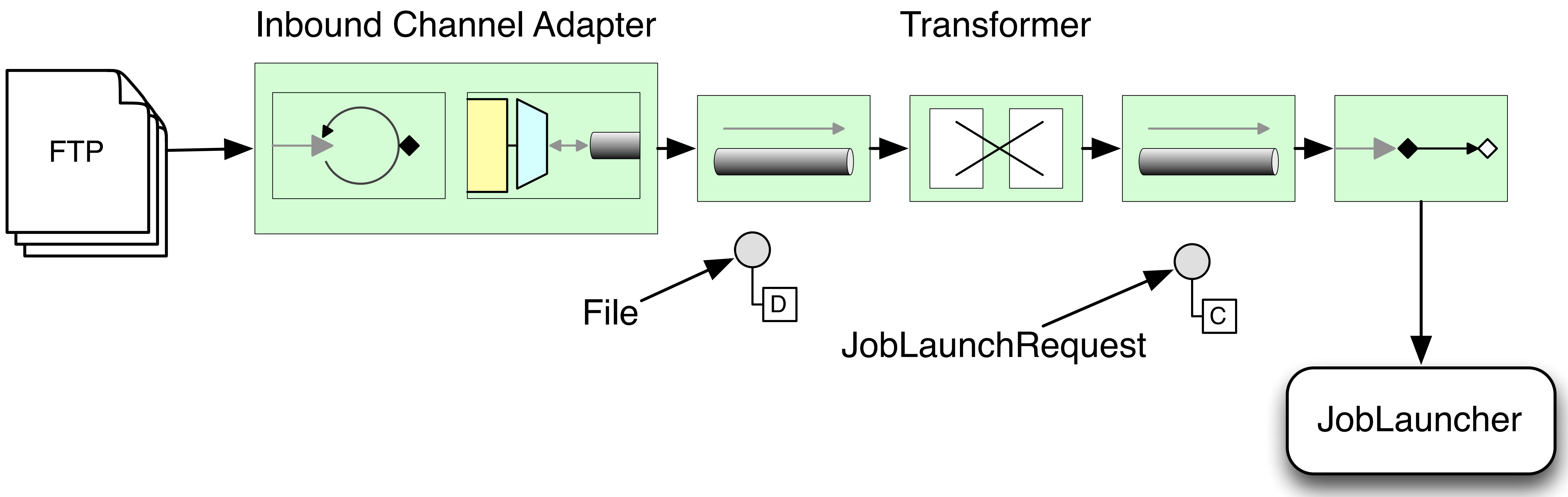
- Stored Procedures
- TCP/UDP
- Twitter
- Web Services
- XMPP
- XPath
- XQuery*
- ...

Spring Batch Integration

https://github.com/SpringSource/spring-batch-admin/

Launching batch jobs through messages

- Event-Driven execution of the JobLauncher
- Spring Integration retrieves the data (e.g. file system, FTP, ...)
- Easy to support separate input sources simultaneously



JobLaunchRequest

```
public class FileMessageToJobRequest {
  private Job job;
  private String fileParameterName;
    ...
  @Transformer
  public JobLaunchRequest toRequest(Message<File> message) {
    JobParametersBuilder jobParametersBuilder = new JobParametersBuilder();
    jobParametersBuilder.addString(fileParameterName,
                                   message.getPayload().getAbsolutePath());
      return new JobLaunchRequest(job, jobParametersBuilder.toJobParameters());
  }
}
```

DefaultJobParametersConverter

- Convert (textual) Properties/Maps to JobParameters
- Provide Typed Parameters
  - Date
  - String
  - Long
  - Double

```
myDateParam(date)=2013/08/20
aStringParameter=Hello AJUG
-stringParamNOTIdentifying=Hello AJUG
aNumberParameter(Long)=123456
```

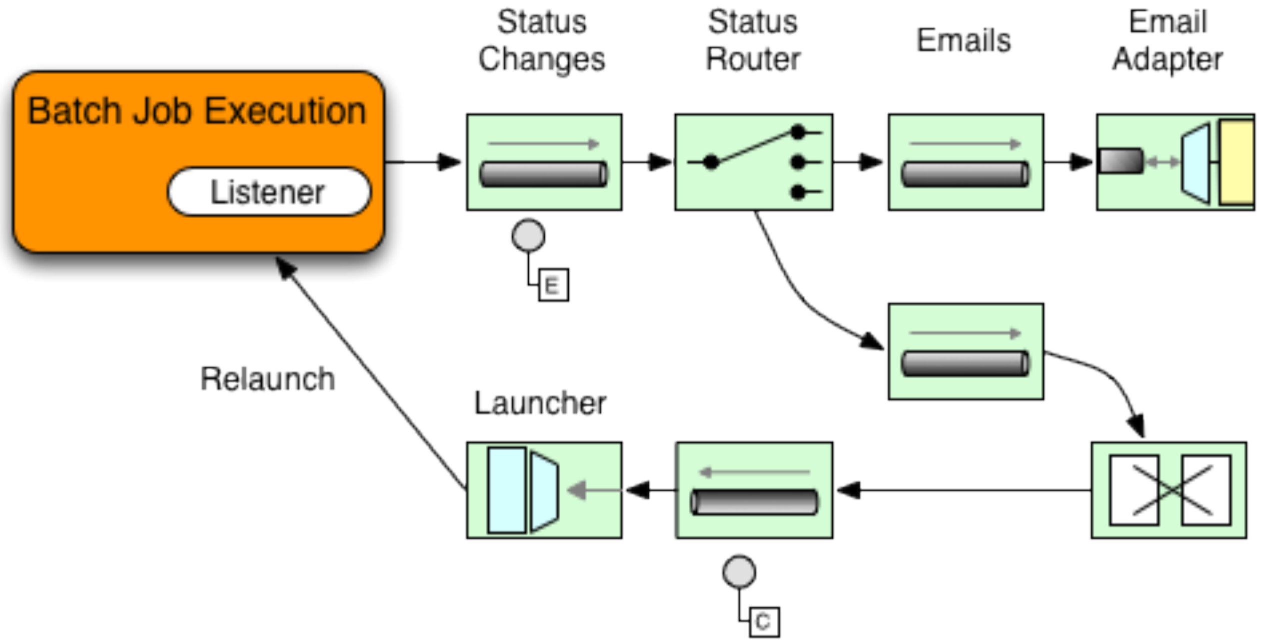
- Provide Date+Number Format
- Define Identifying / Non-Identifying Parameters

JobLaunchRequest

```
<batch-int:job-launching-gateway  request-channel="requestChannel"
                                  reply-channel="replyChannel"
                                  job-launcher="jobLauncher"/>
```

Get feedback with informational messages

- Spring Batch provides support for listeners:
  - StepListener
  - ChunkListener
  - JobExecutionListener



Get feedback with informational messages

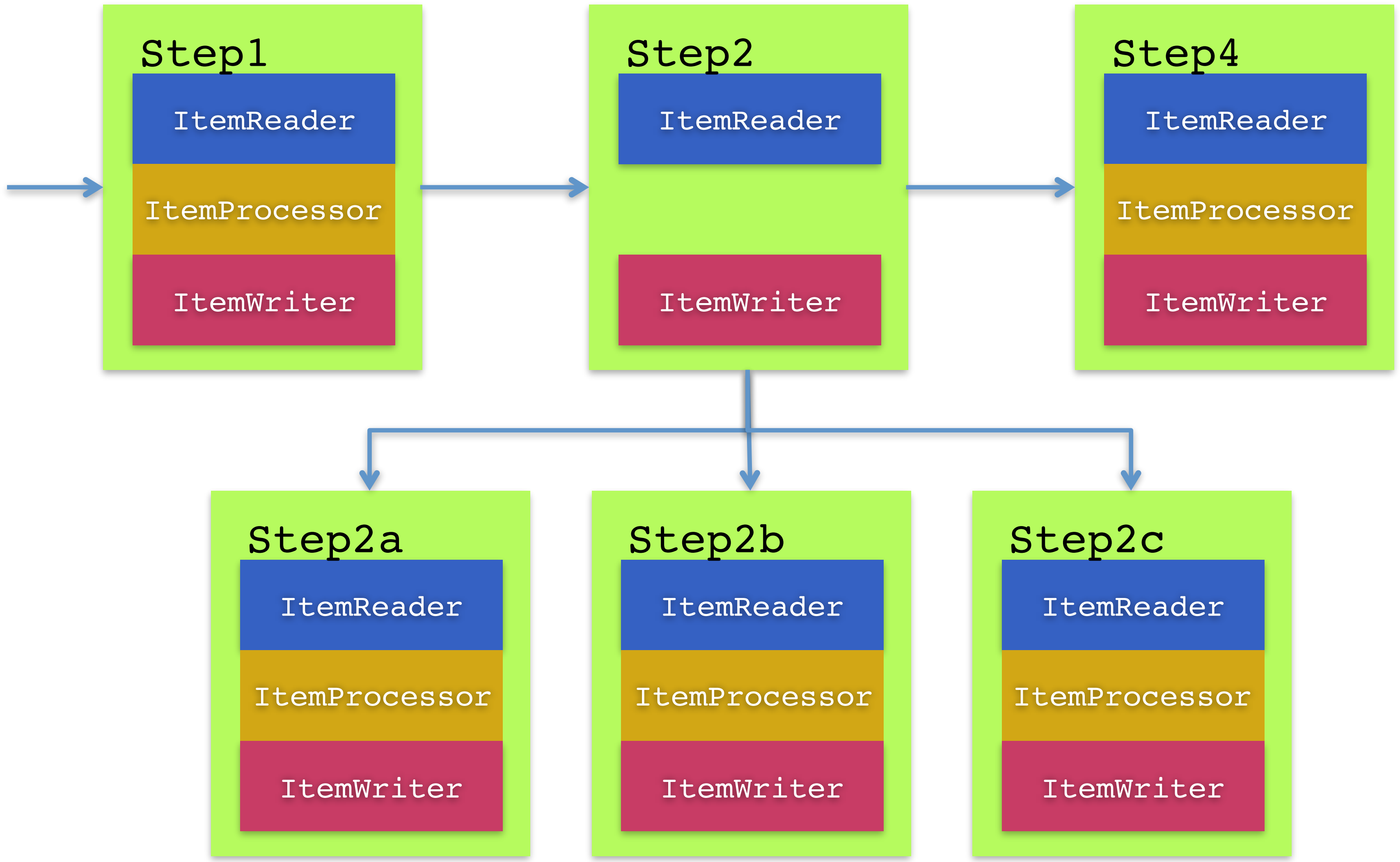
```
<batch:job id="importPayments">
    ...
    <batch:listeners>
        <batch:listener ref="notificationExecutionsListener"/>
    </batch:listeners>
</batch:job>

<int:gateway id="notificationExecutionsListener"
    service-interface="o.s.batch.core.JobExecutionListener"
    default-request-channel="jobExecutions"/>
```

Externalizing batch process execution

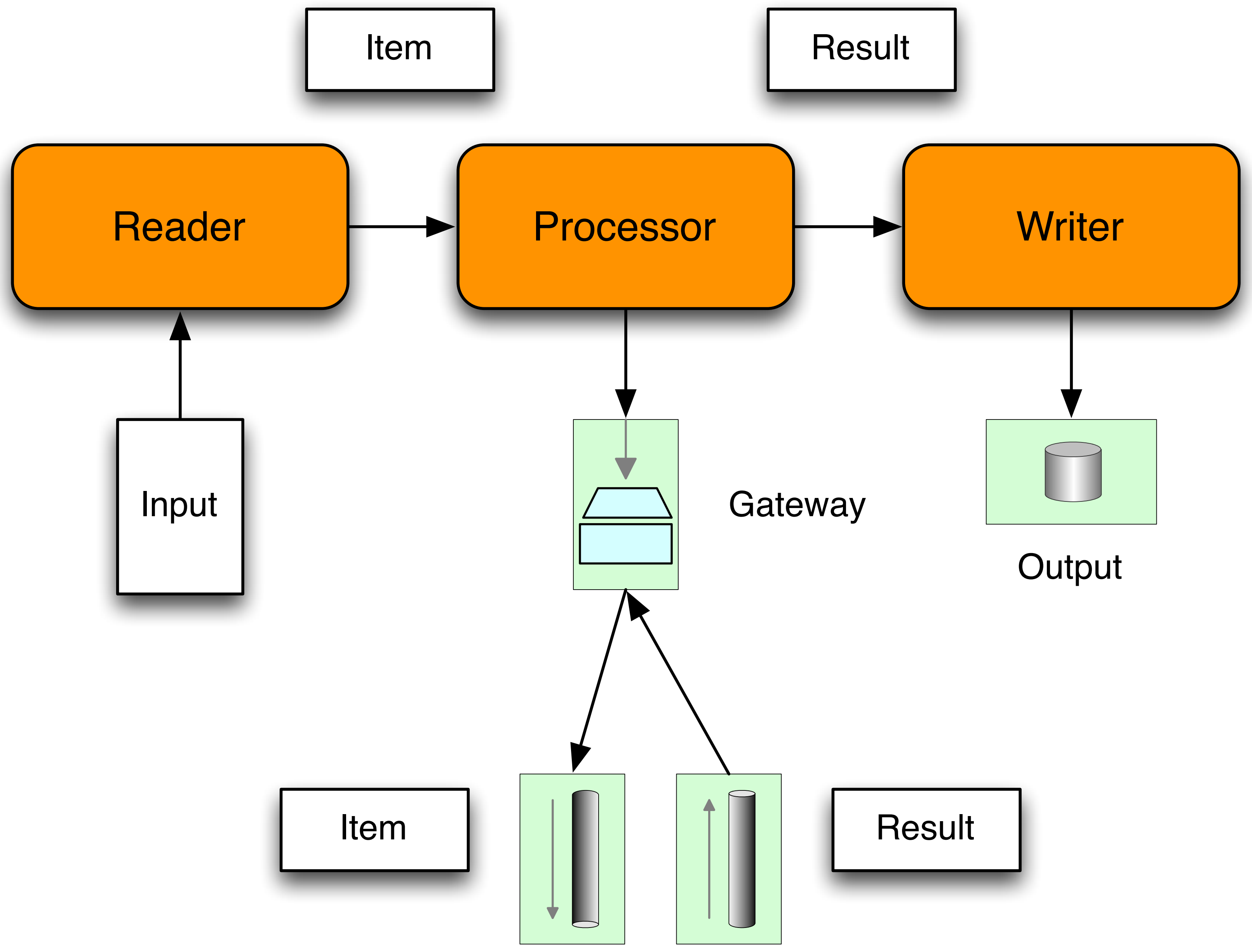
- Use Spring Integration inside of Batch jobs
  - e.g. ItemProcessor + ItemWriter
- Offload complex processing
- Asynchronous processing support:
  - AsyncItemProcessor
  - AsyncItemWriter
- Externalize chunk processing using ChunkMessageChannelItemWriter

Remote Chunking

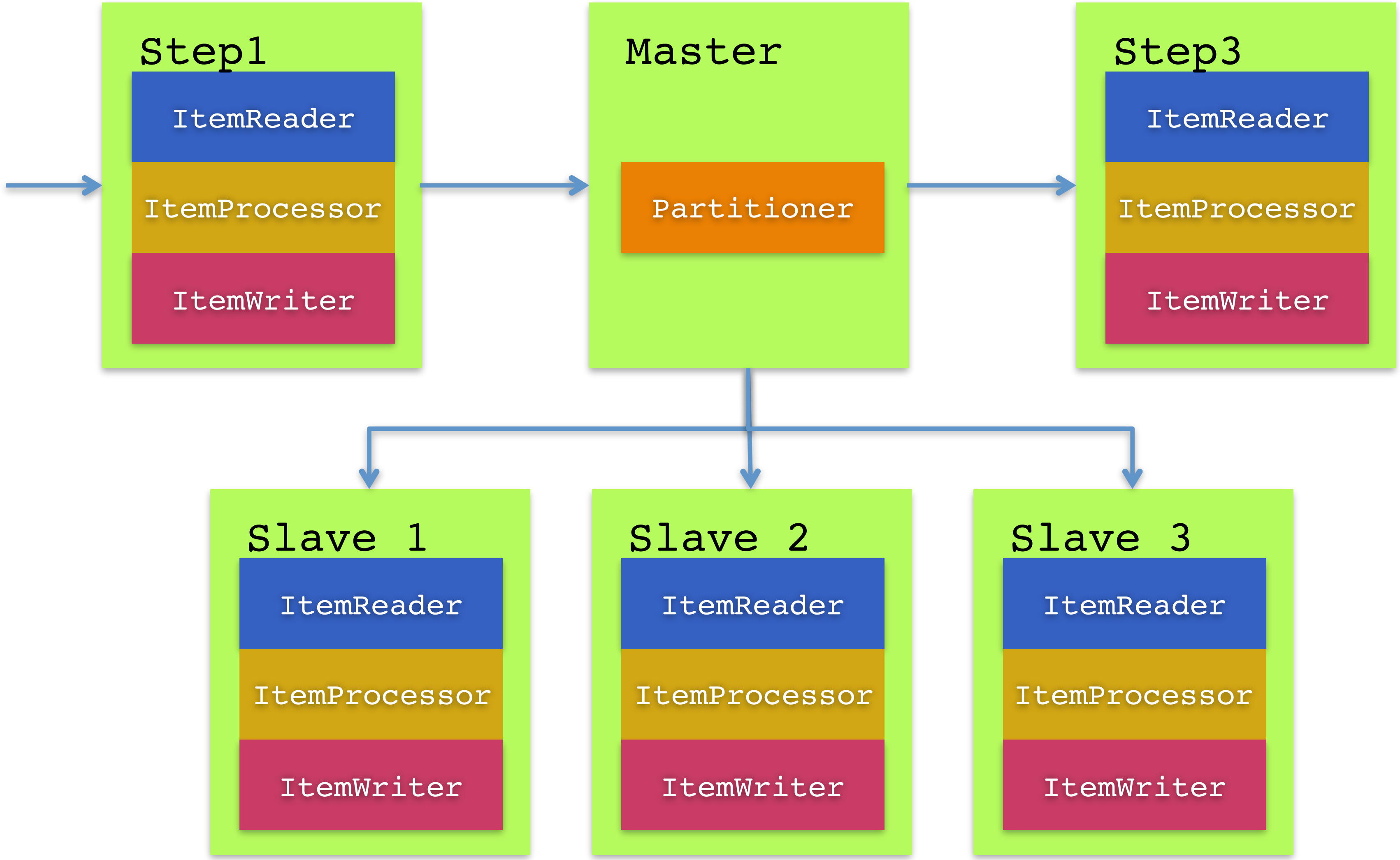


Asynchronous Processors

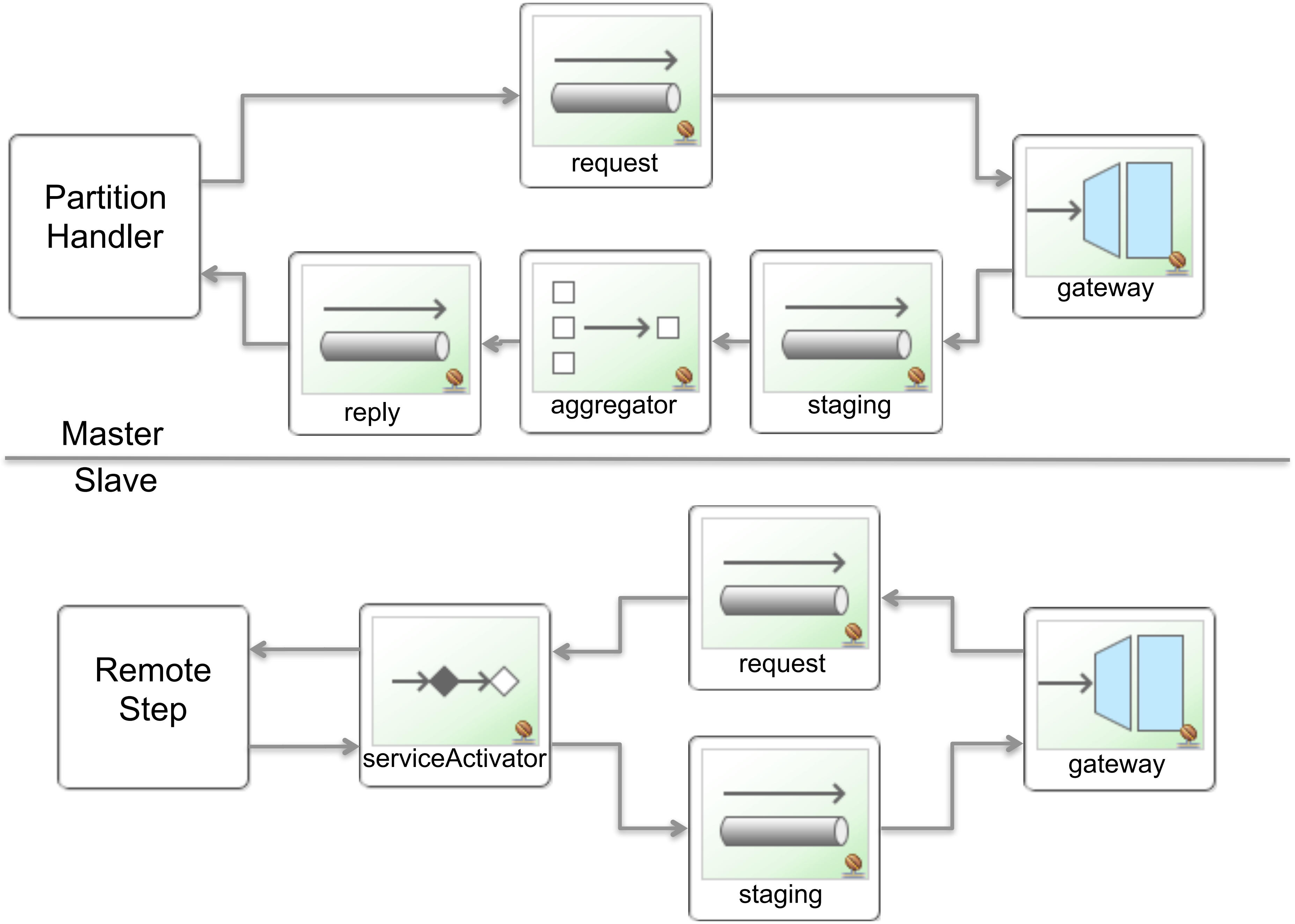
- AsyncItemWriter
- AsyncItemProcessor



Remote Partitioning



Remote Partitioning



Demo

Payment Import

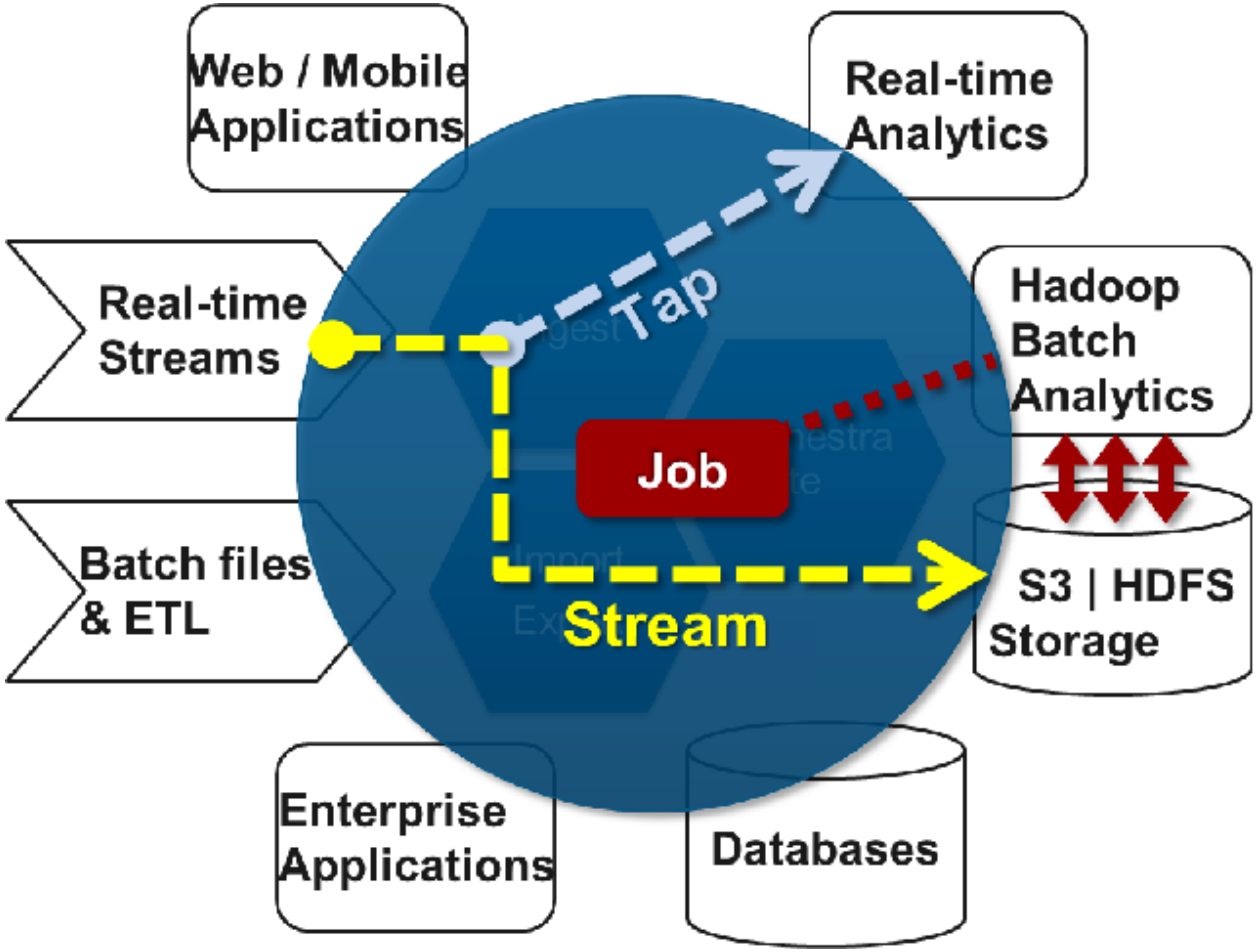
https://github.com/ghillert/spring-batch-integration-sample

Spring XD

http://www.springsource.org/spring-xd

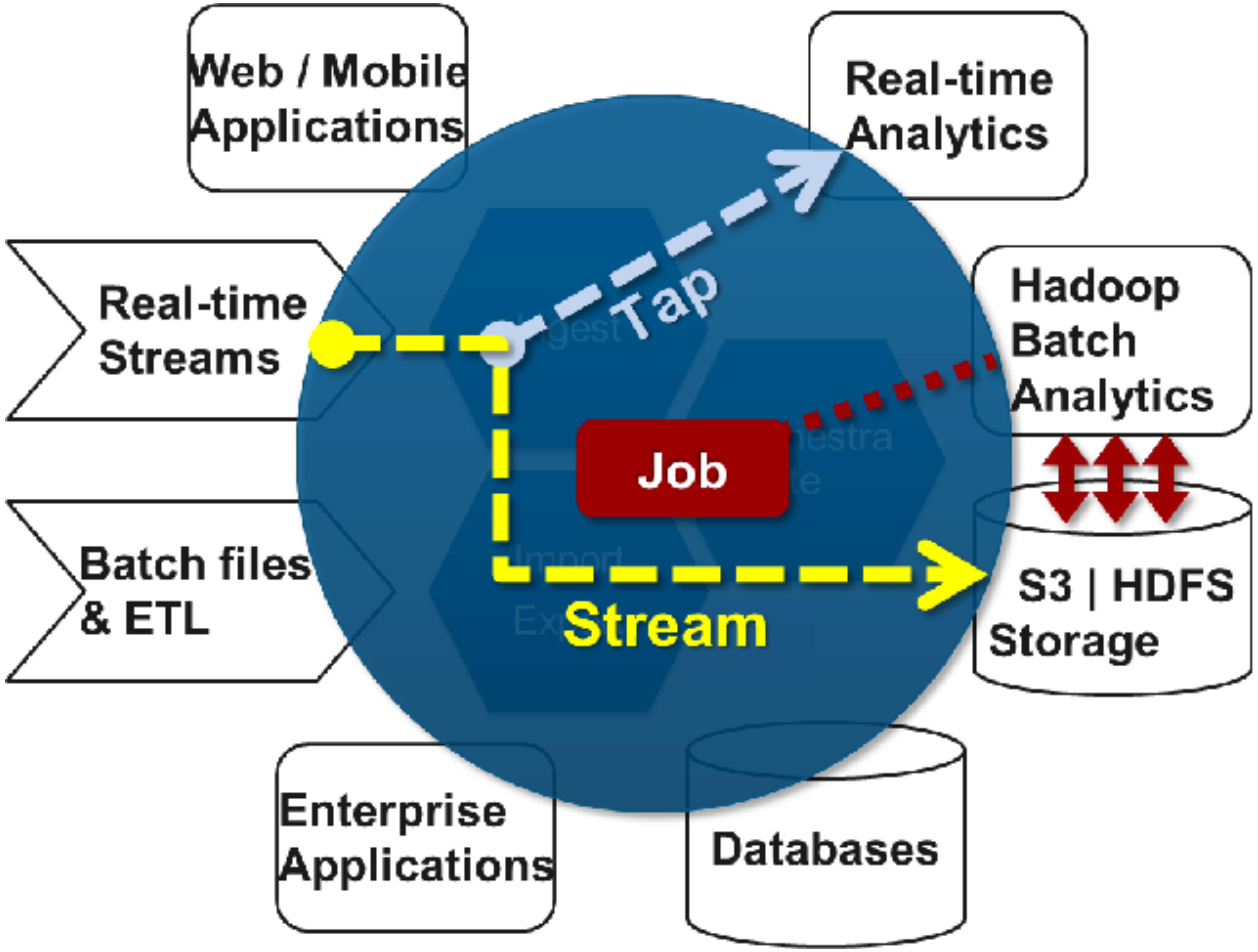
Tackling Big Data Complexity

- Unified agile experience for
- Data Ingestion
- Real-time Analytics
- Workflow Orchestration
- Data Export



Tackling Big Data Complexity cont.

- Built on existing assets
  - Spring Integration
  - Spring Batch
  - Spring Data
    - Redis, GemFire, Hadoop
- XD = 'eXtreme Data’



Data Ingestion Streams

- DSL based on Unix pipes and filters syntax

```
http | file
```

- Modules are parameterizable

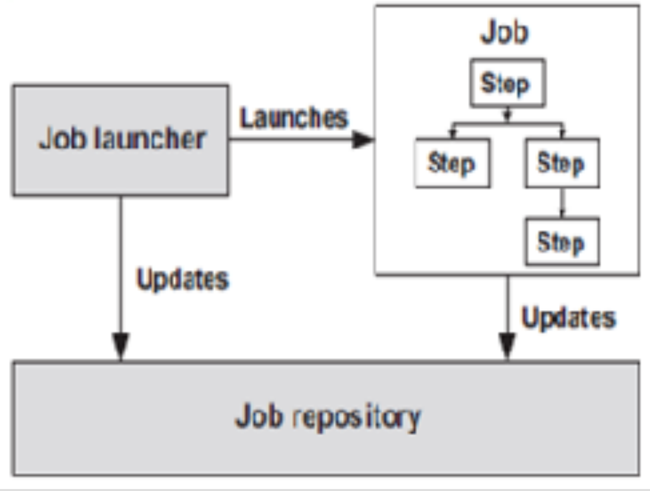
```
twittersearch --query=spring | file --dir=/spring
```

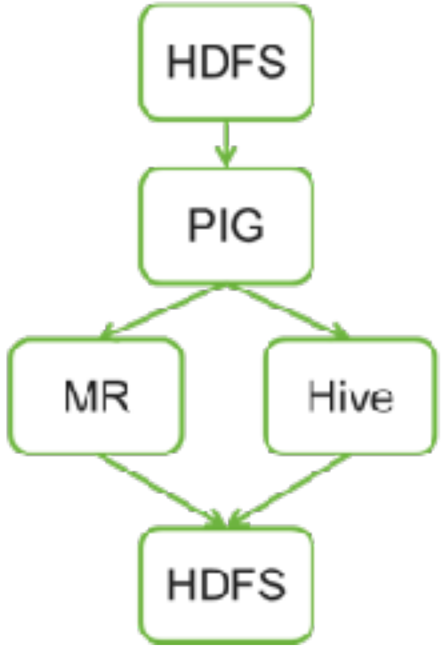
- Simple logic can be added via expressions or scripts

```
http |
filter --expression=”payload?.customerCode matches ‘GOLD[0-9]+’”
| hdfs
```

Hadoop workflow managed by Spring Batch

- Reuse Batch infrastructure and features to manage Hadoop workflows
  - Job state management, launching, monitoring, restart/retry policies, etc.
- Step can be any Hadoop job type or HDFS script
- Can mix and match with other Batch readers/writers
  - (e.g. JDBC for import/export use-cases)





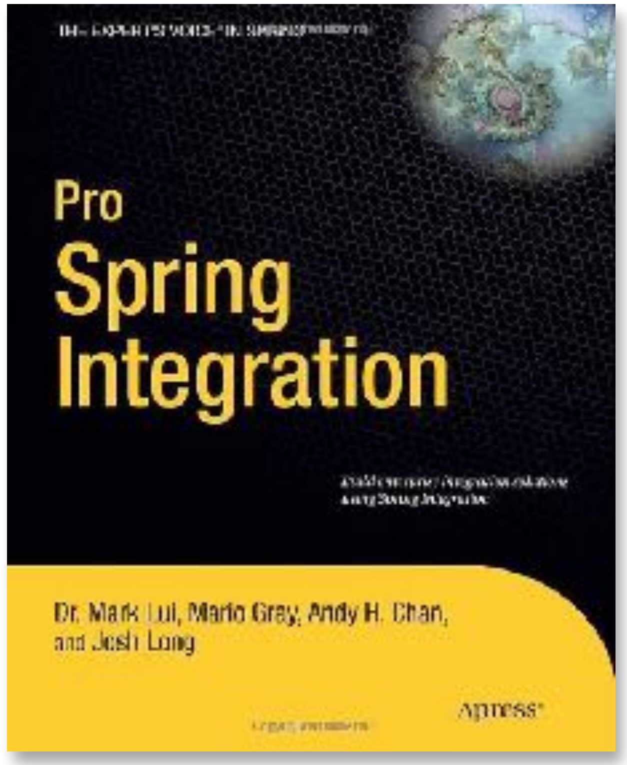
Demo

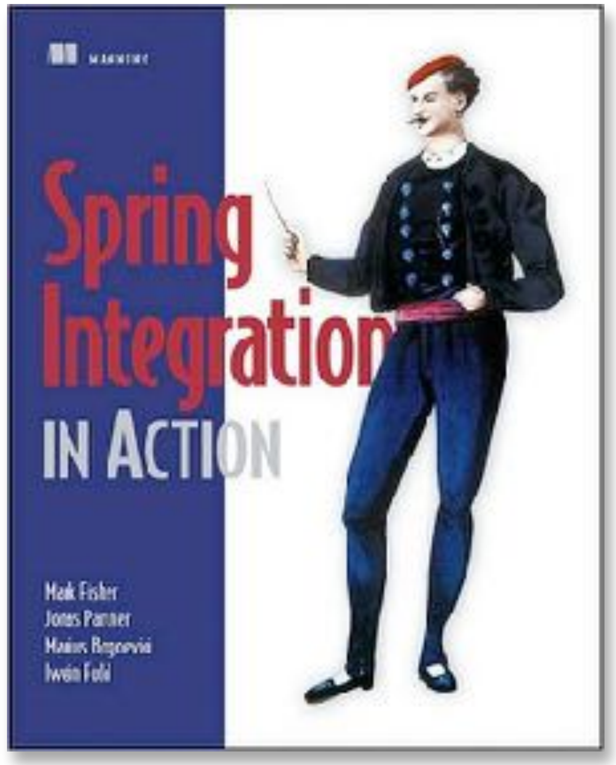
Spring XD
Batch Word-count Sample

https://github.com/SpringSource/spring-xd-samples

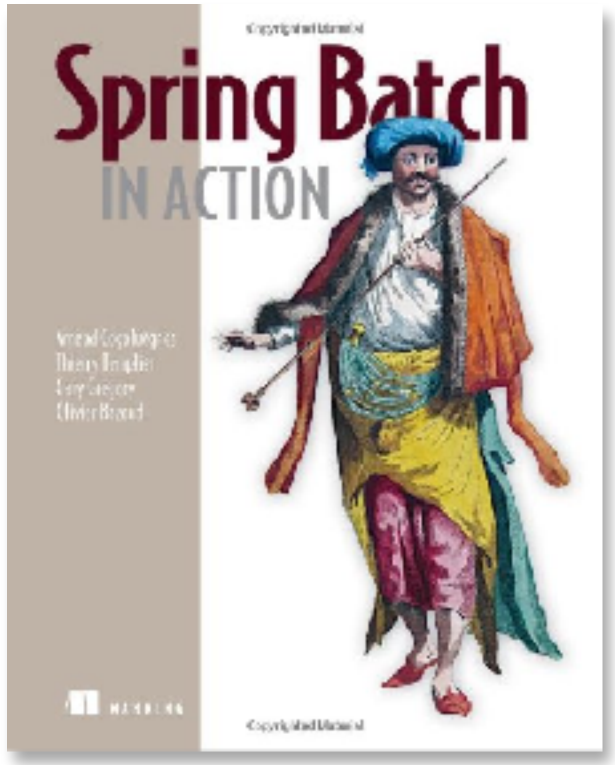
Books

Enterprise Application Patterns and Messaging
Just Spring Integration
O'REILLY
Madhusudhan Konda
Foreword by Josh Long


THE EXPERT'S VOICE® IN SPRING
Pro
Spring
Integration
Dr. Mark Lui, Mario Gray, Andy H. Chan, and Josh Long
Apress®


MANNING
Spring
Integration
IN ACTION
Mark Fisher
Jonas Partner
Marius Bogoevici
Iwein Fuld

THE EXPERT'S VOICE® IN SPRING
Pro
Spring Batch
BATCH PROCESSING WITH THE SPRING BATCH FRAMEWORK
Michael T. Minella
Apress®


Spring Batch
IN ACTION
MANNING

Learn More. Stay Connected.

Questions?

Thank You!!

Twitter: twitter.com/springframework
YouTube: youtube.com/user/SpringSourceDev
Google +: plus.google.com/+springframework
LinkedIn: springsource.org/linkedin
Facebook: facebook.com/groups/springsource