Develop Powerful Big Data Applications Easily with SpringXD

Mark Fisher & Mark Pollack

© 2014 SpringOne 2GX. All rights reserved. Do not distribute without permission.

Speakers

Mark Fisher

- Spring XD – Co Lead
- Spring Integration
- Spring Framework
- Spring AMQP

Mark Pollack

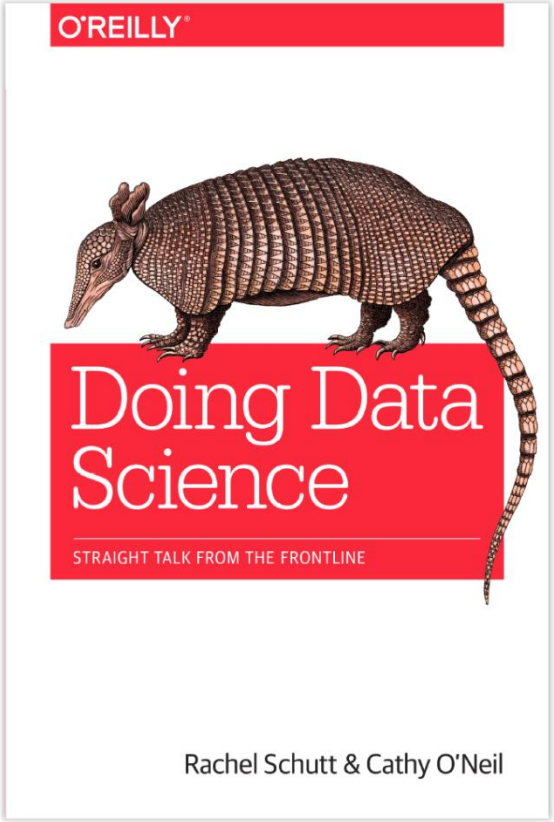
- Spring XD – Co Lead
- Spring Data
- Spring Framework
- Spring .NET

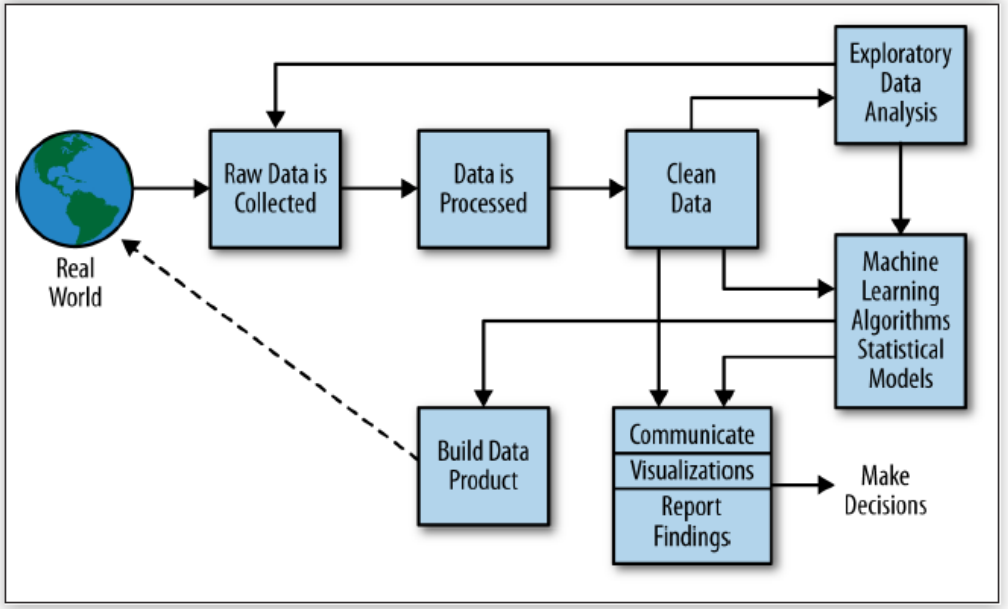
Spring XD

XD = eXtreme Data

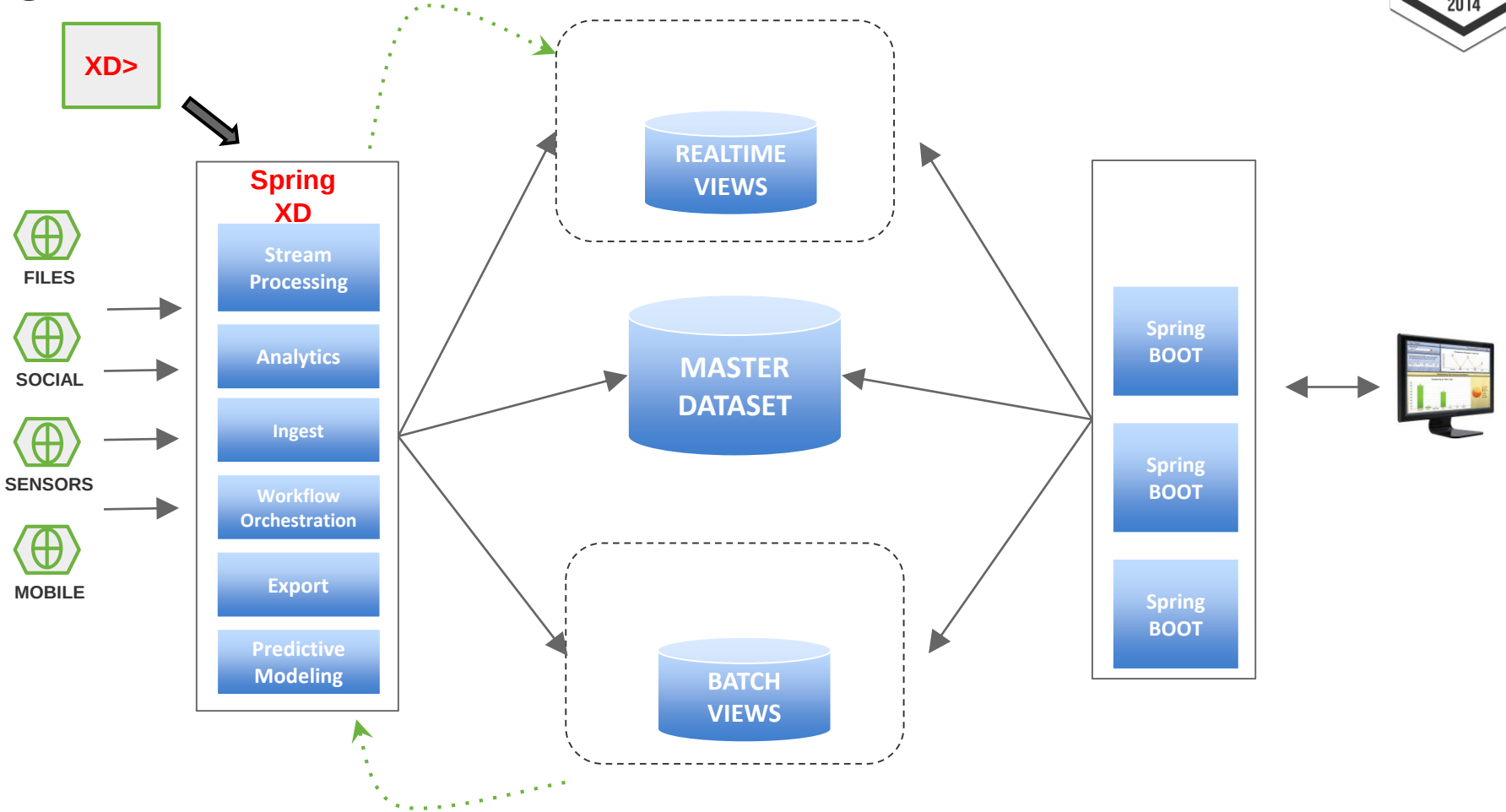
“One stop shop for developing and deploying Big Data Applications”

What is a Big Data Application?

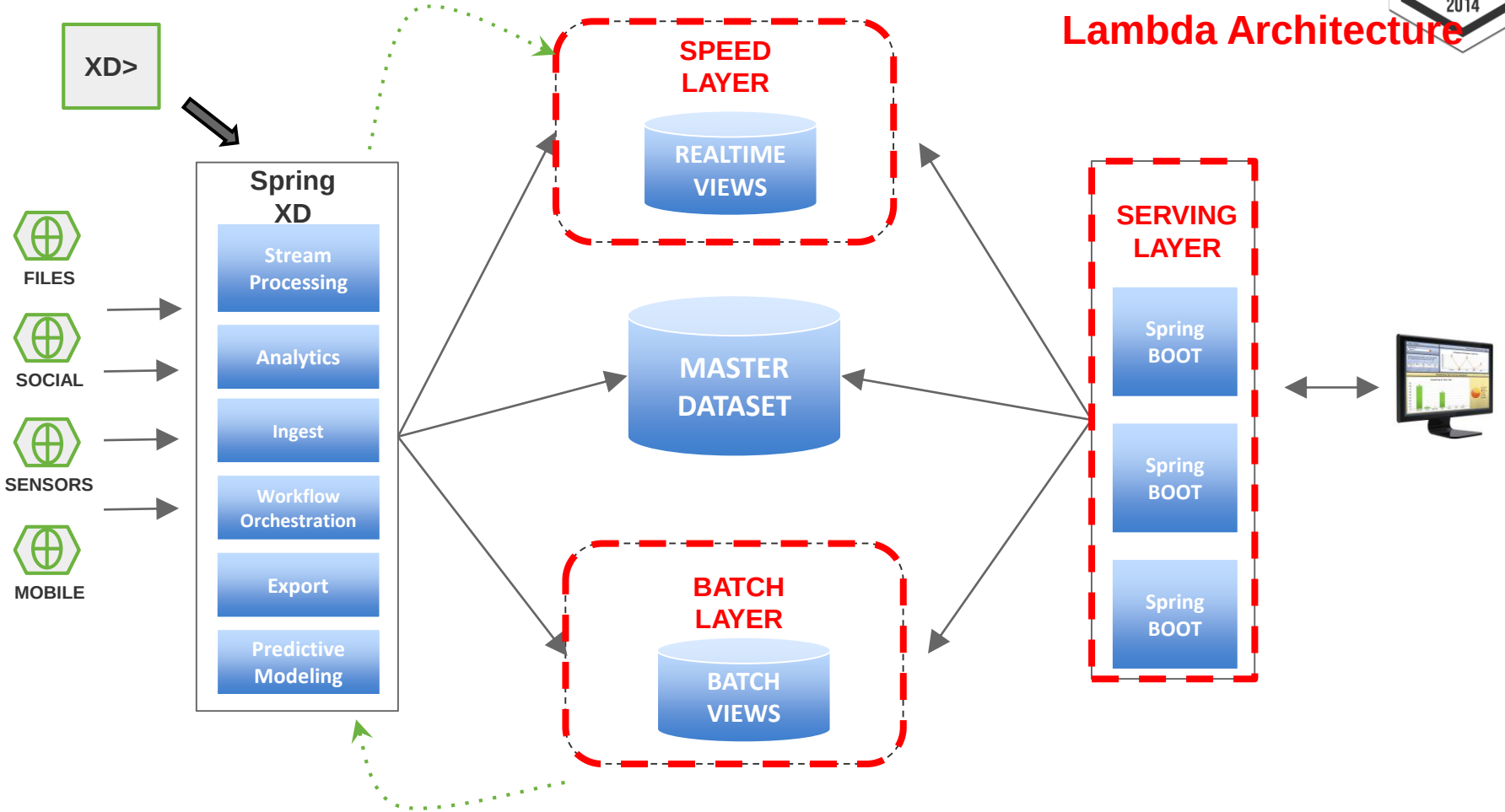


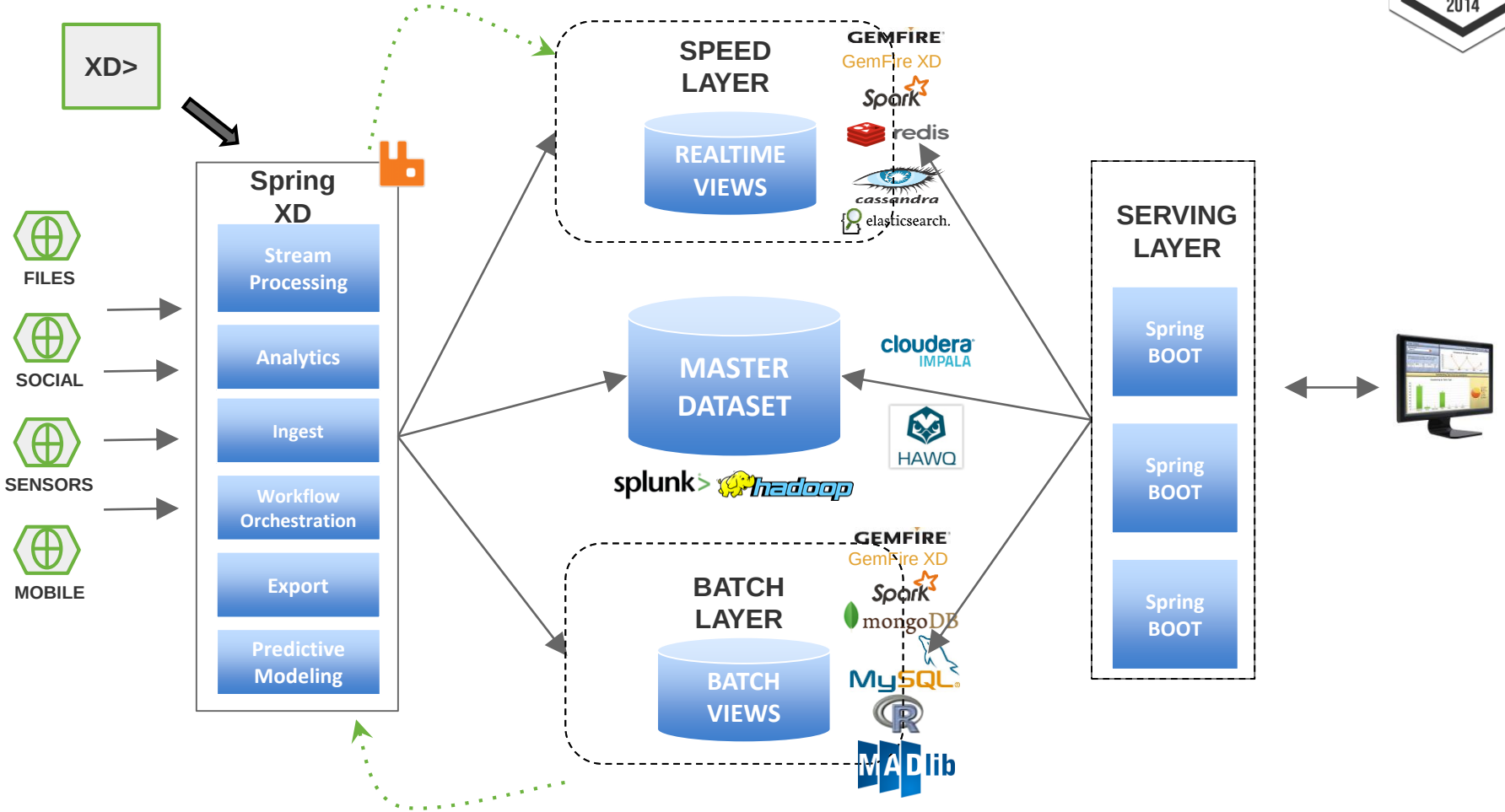


Big Data Architecture

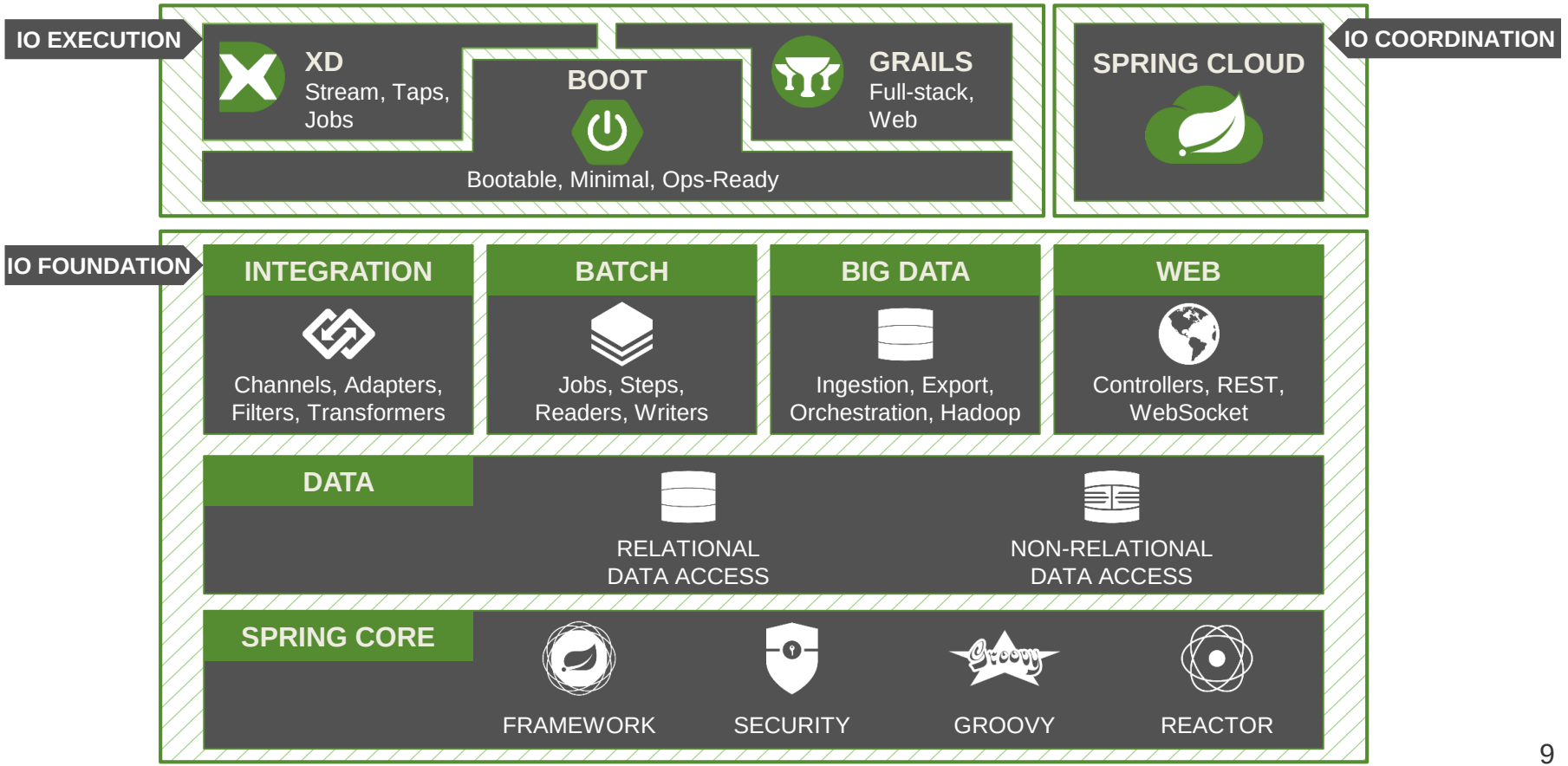


Lambda Architecture

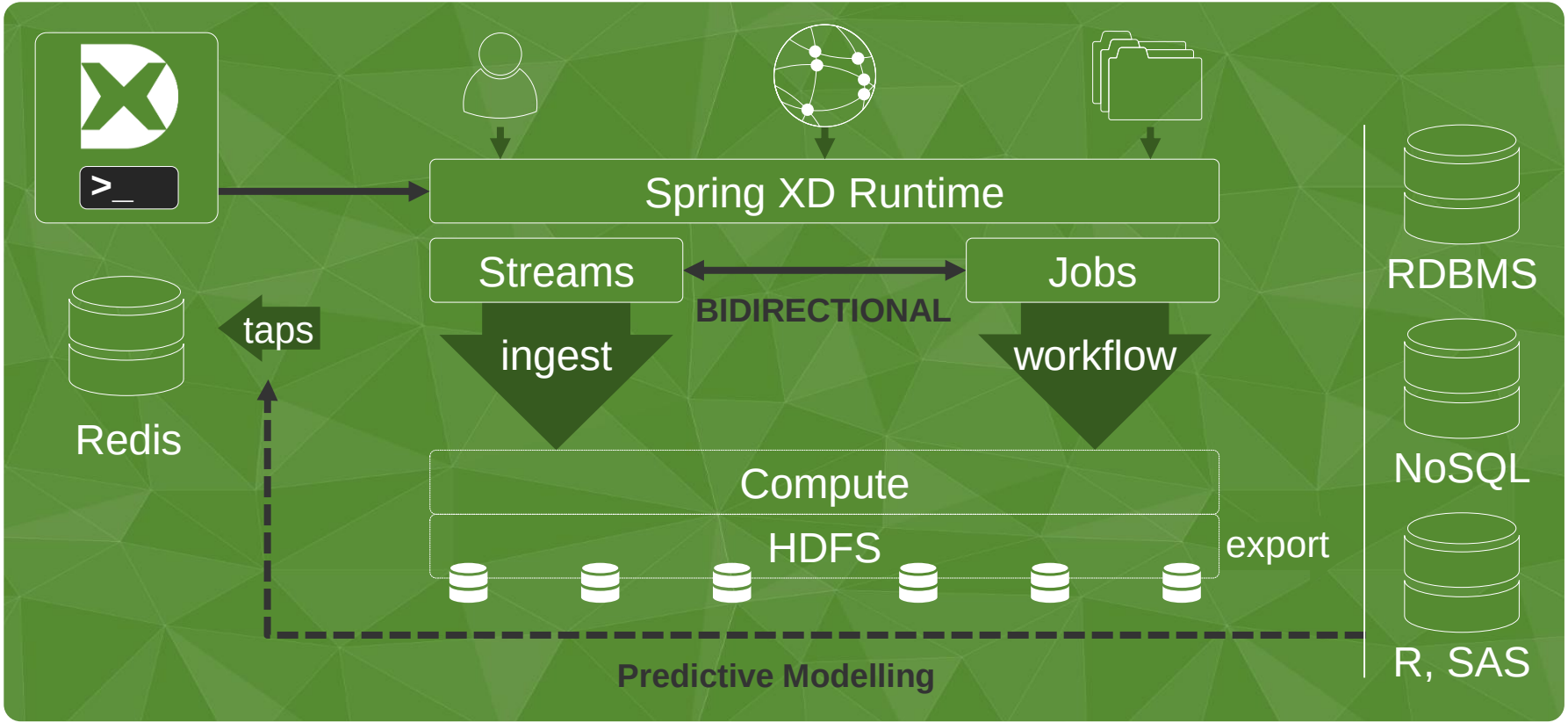



XD>
Spring XD
Stream Processing
Analytics
Ingest
Workflow Orchestration
Export
Predictive Modeling
FILES
SOCIAL
SENSORS
MOBILE
SPEED LAYER
REALTIME VIEWS
GEMFIRE
GemFire XD
Spark
redis
cassandra
elasticsearch.
MASTER DATASET
cloudera IMPALA
HAWQ
splunk>
hadoop
BATCH LAYER
BATCH VIEWS
GEMFIRE
GemFire XD
Spark
mongoDB
MySQL
R
MADlib
SERVING LAYER
Spring BOOT
Spring BOOT
Spring BOOT

Spring IO Platform



Spring XD: Unified Platform for Big Data



Streams

HTTP
Tail
File
Mail
Twitter
Gemfire
Syslog
TCP
UDP
JMS
RabbitMQ
MQTT
Trigger
Reactor TCP/UDP

Filter
Transformer
Object-to-JSON
JSON-to-Tuple
Splitter
Aggregator
HTTP Client
Groovy Scripts
Java Code
JPMML Evaluator

File
HDFS
JDBC
TCP
Log
Mail
RabbitMQ
Gemfire
Splunk
MQTT
Dynamic Router
Counters

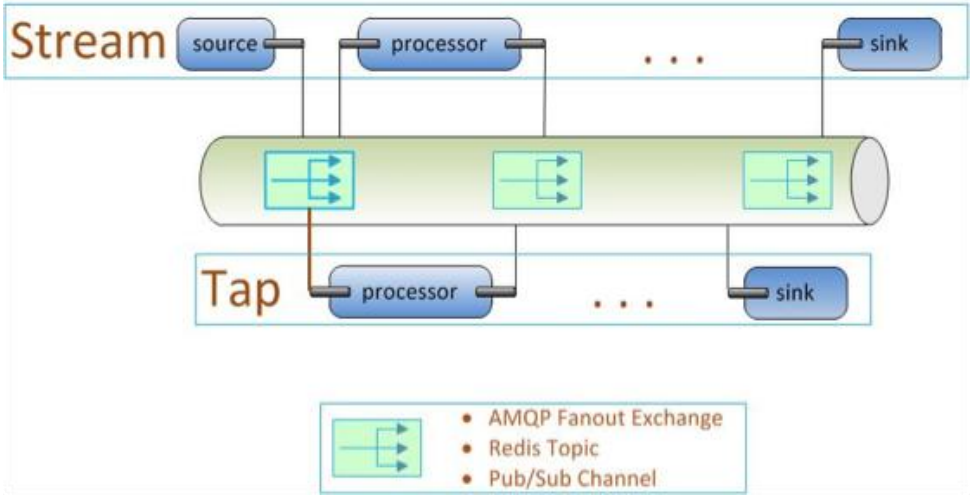
Demo:

Spring XD - Streams

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Taps

- “Listen” to data on another stream



Analytics

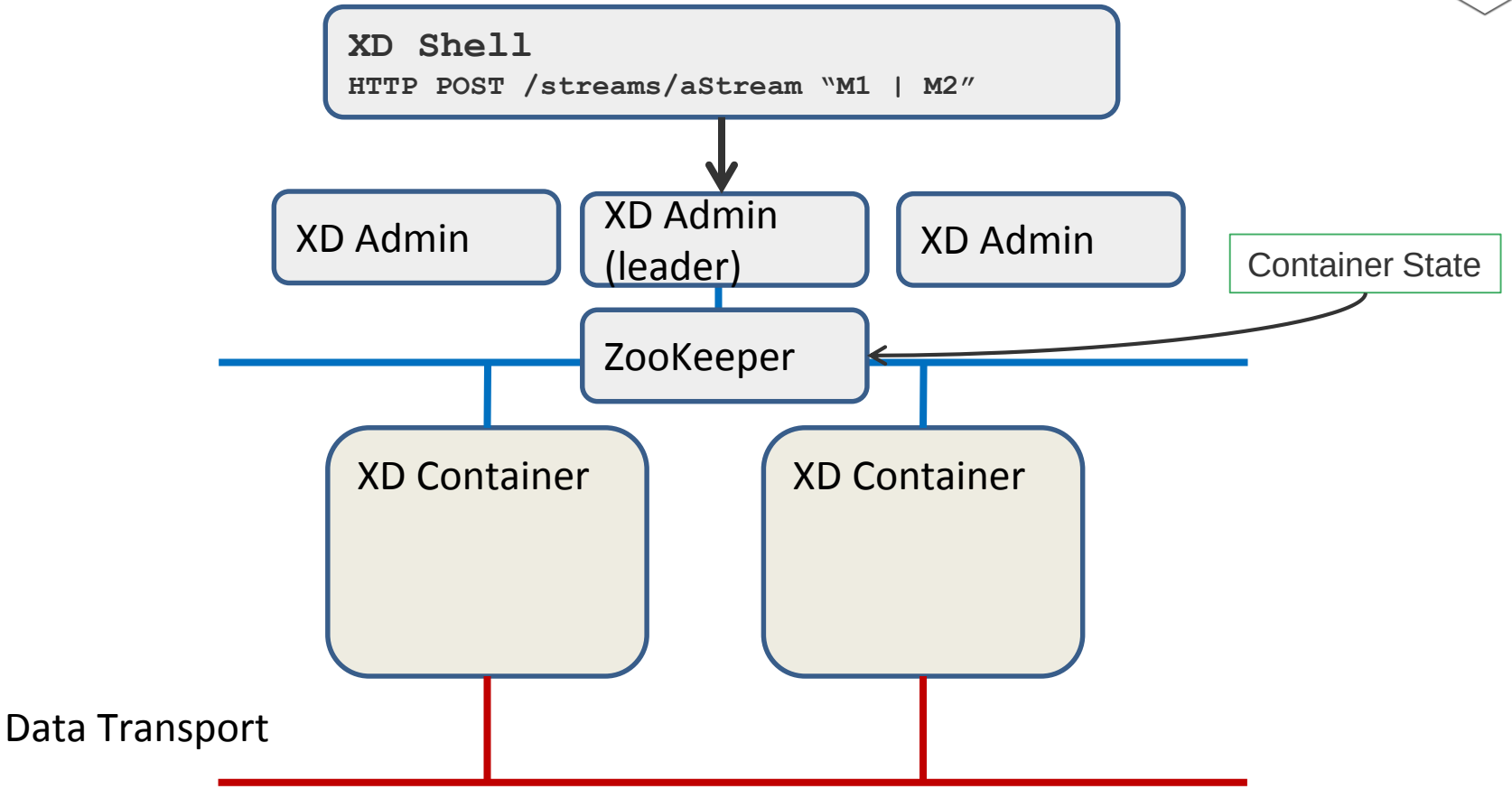
- Counters and Gauges
  - Simple & Field Value Counter
    - *How many tweets for #java*
  - Aggregate Counter
    - *How many tweets for #java in the week/day/hour*
  - Gauge & Rich Gauge
    - *How many requests per minute*?
- Abstract API. Implemented in
  - In-Memory
  - Redis

Demo:

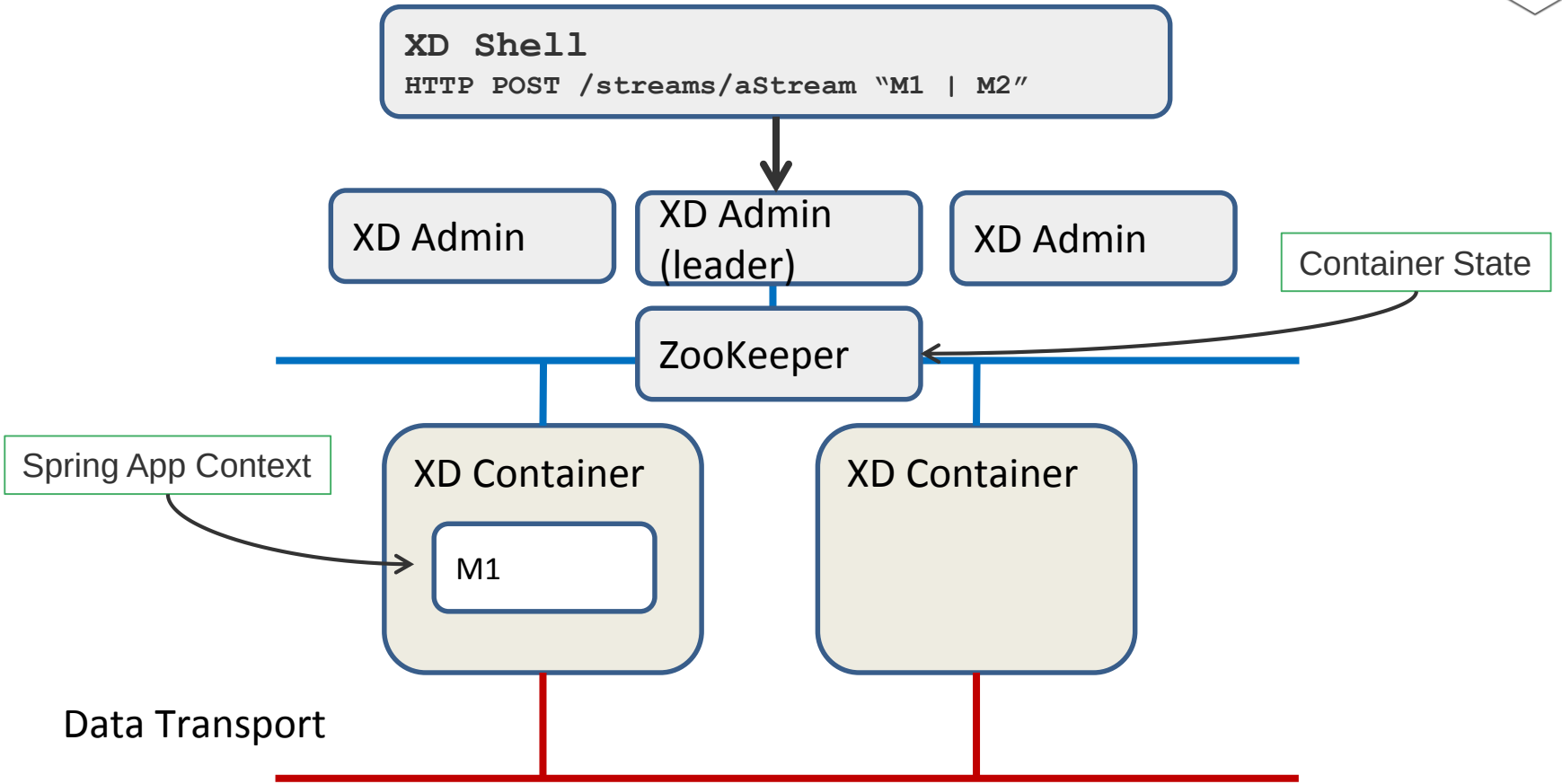
Spring XD - Taps

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

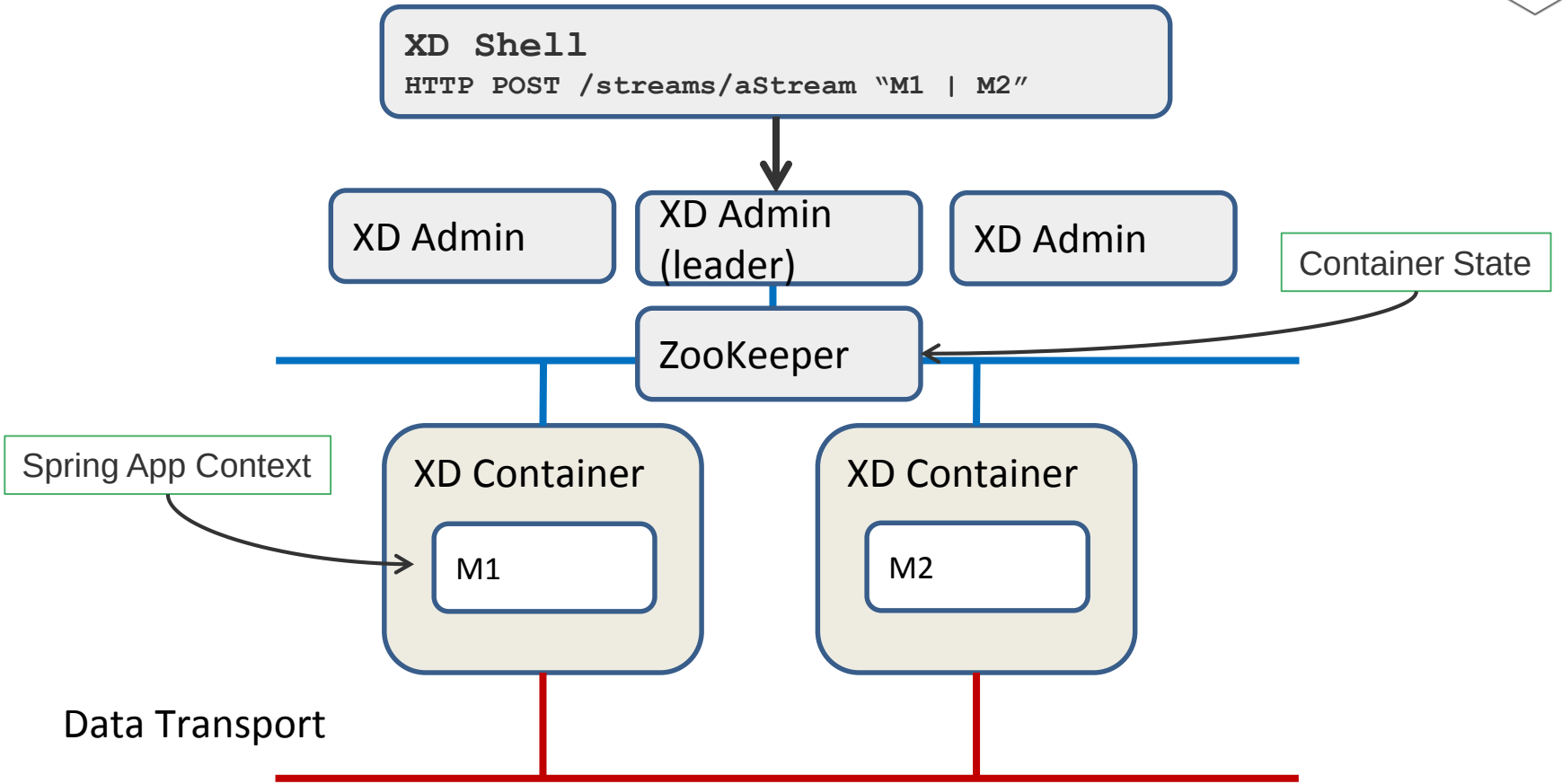
Spring XD Runtime



Spring XD Runtime



Spring XD Runtime



Deployment Manifest

Deployment Manifest

- The stream/job definition defines the logical view of processing
- The deployment manifest defines the physical view of processing
- Important properties relate to module count and data partitioning

```
xd:>stream create test1 --definition
    "http | transform --expression=payload.toUpperCase() | log”

xd:>stream deploy --name test1 --properties "module.transform.count=3"
```

Deployment Manifest – Data Partitioning

```
stream create words --definition "http |
                            splitter --expression=payload.split(' ') | log"

stream deploy words --properties
module.splitter.producer.partitionKeyExpression=payload,module.log.count=2

http post --data
    "How much wood would a woodchuck chuck if a woodchuck could chuck wood"
```

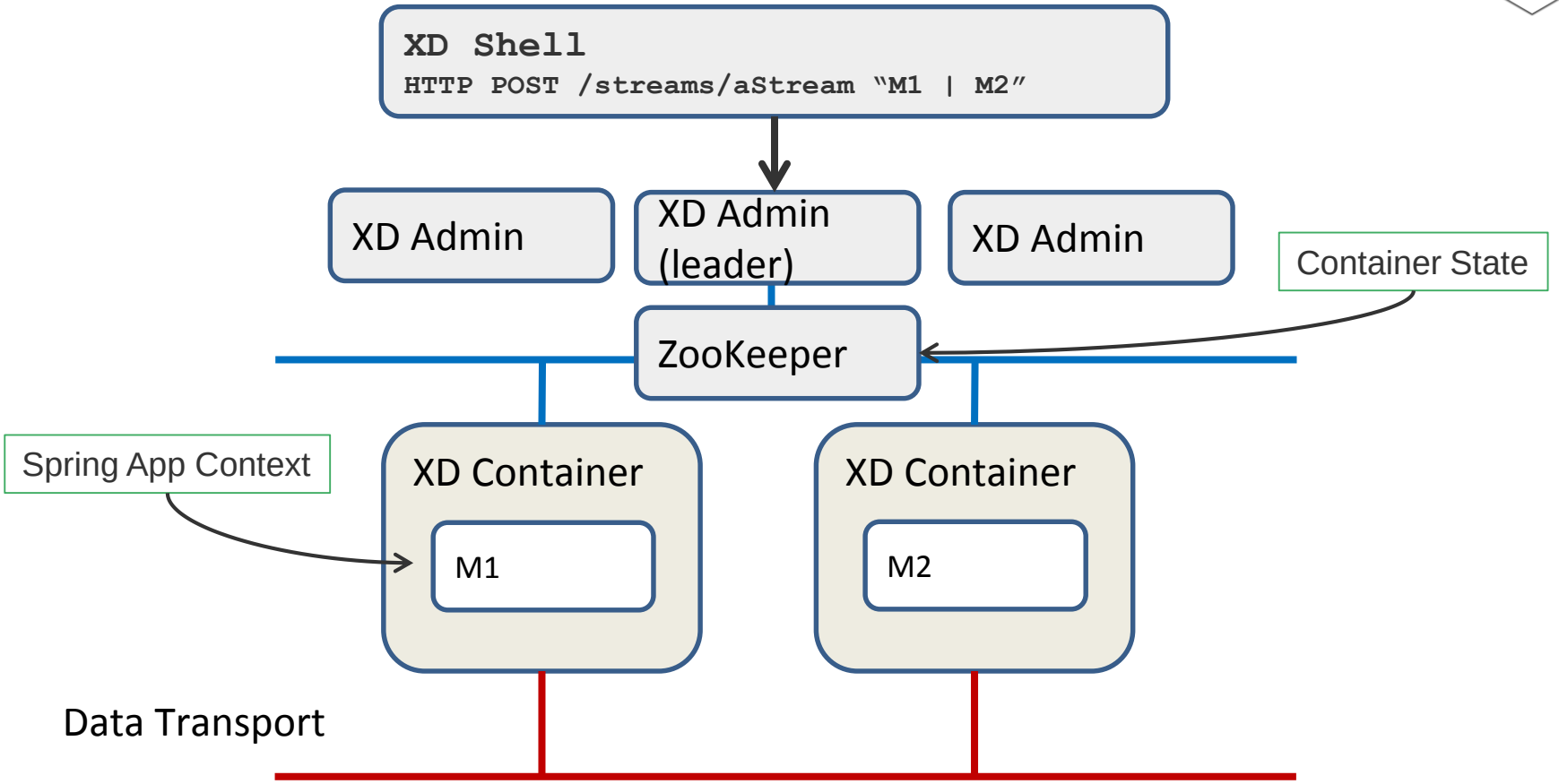
Demo:

Spring XD - Partitioning

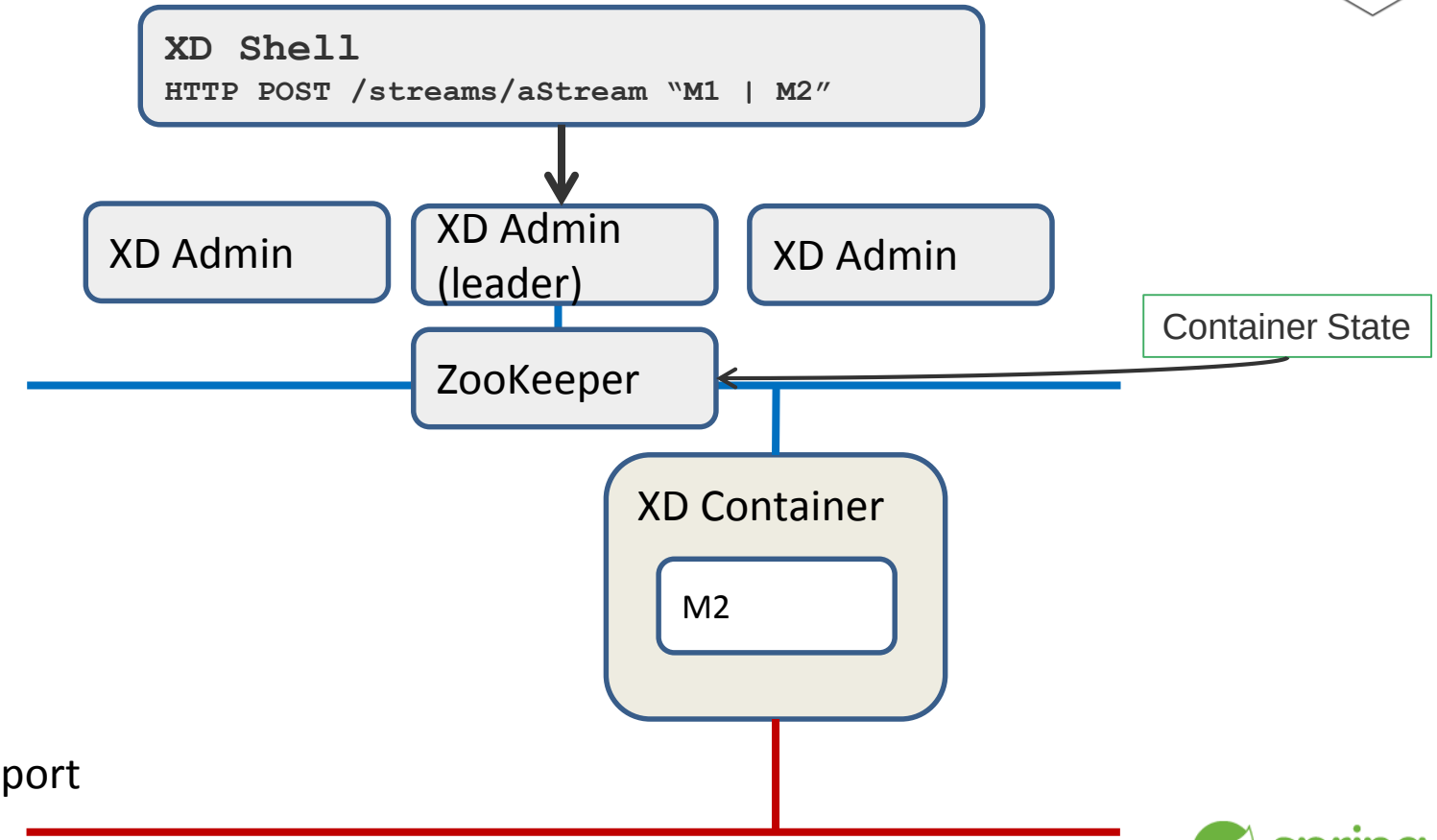
Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Distributed, Fault Tolerant Runtime

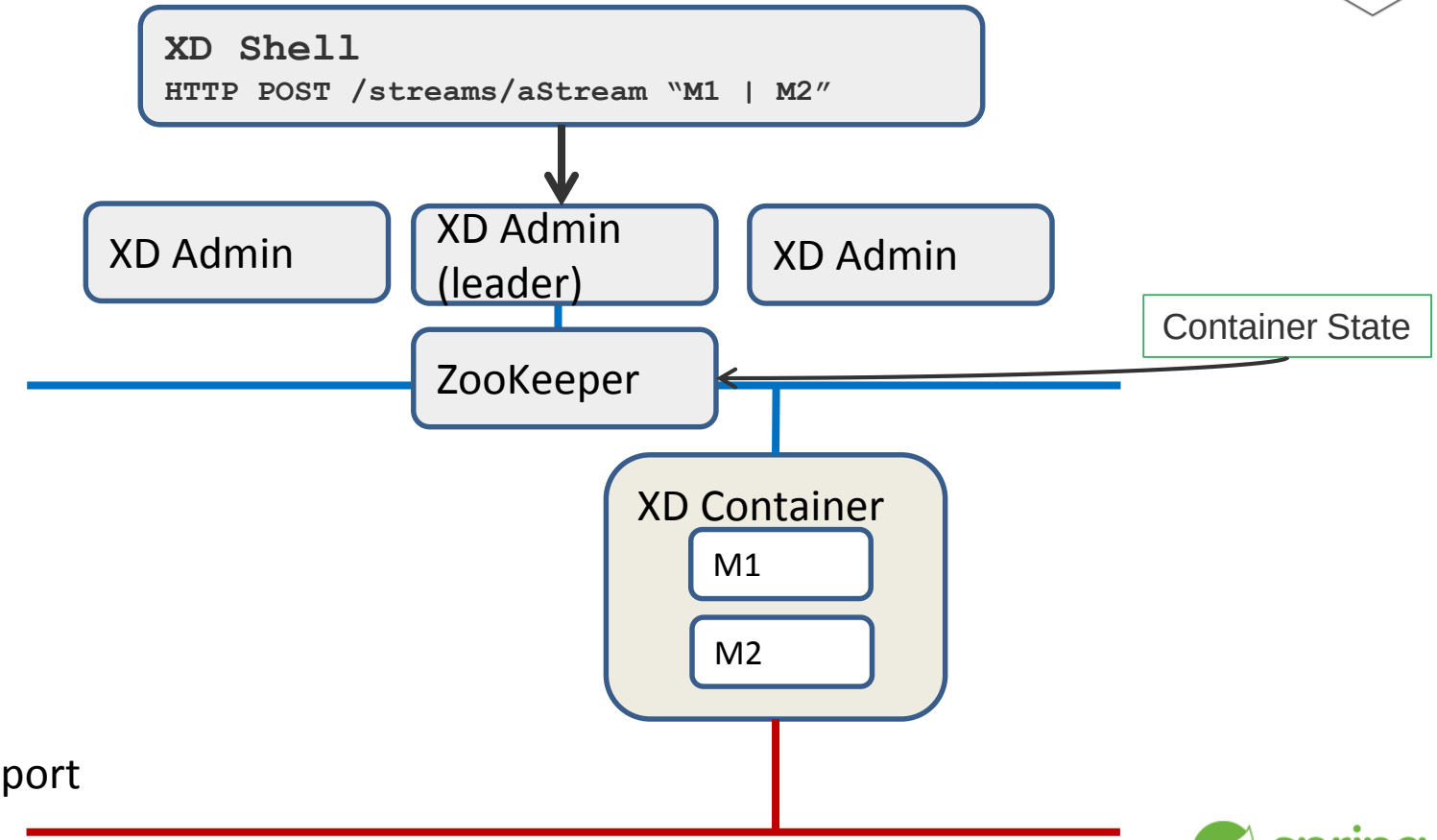
Spring XD – Runtime – Fault Tolerance



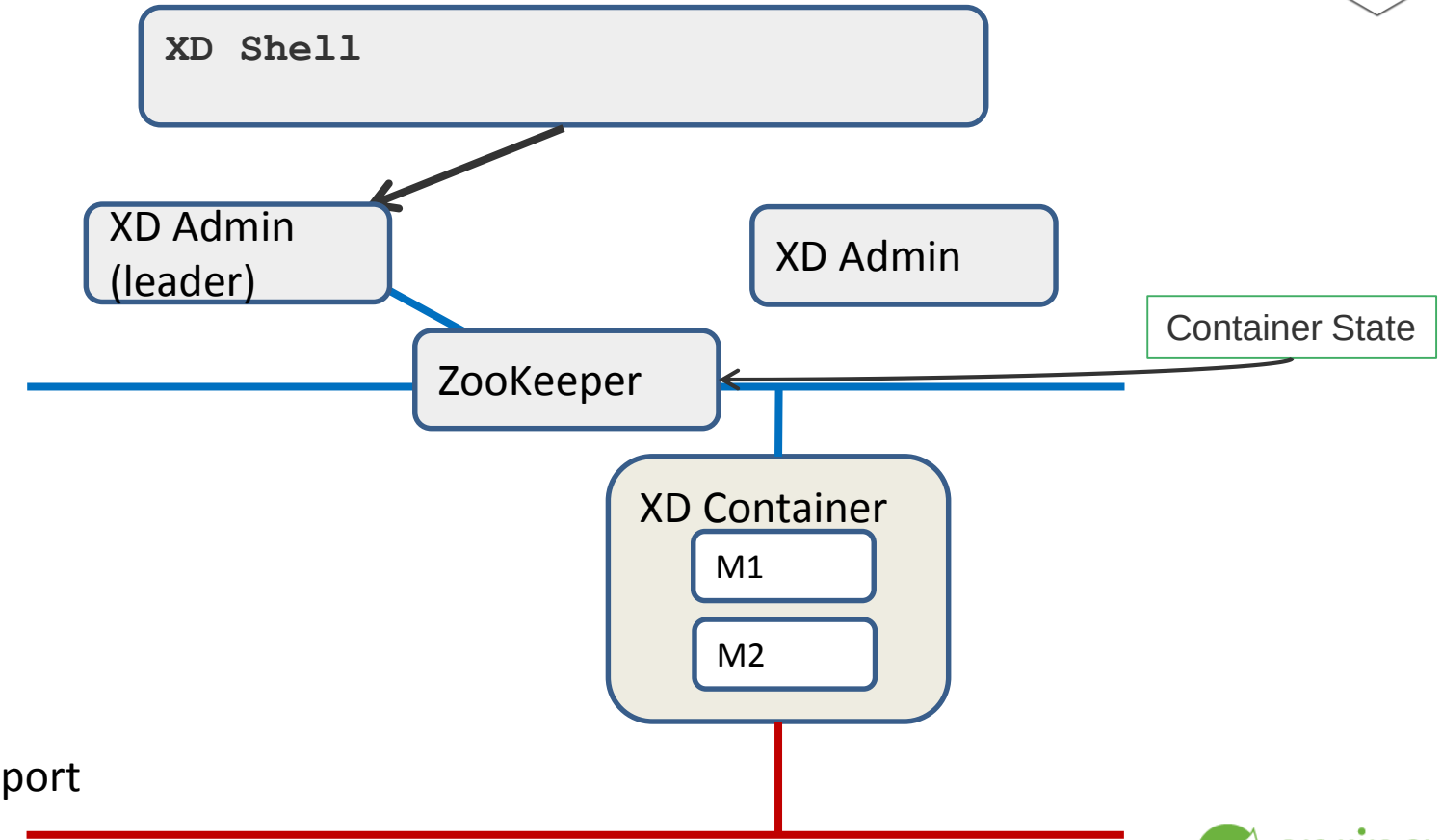
Spring XD – Runtime – Fault Tolerance



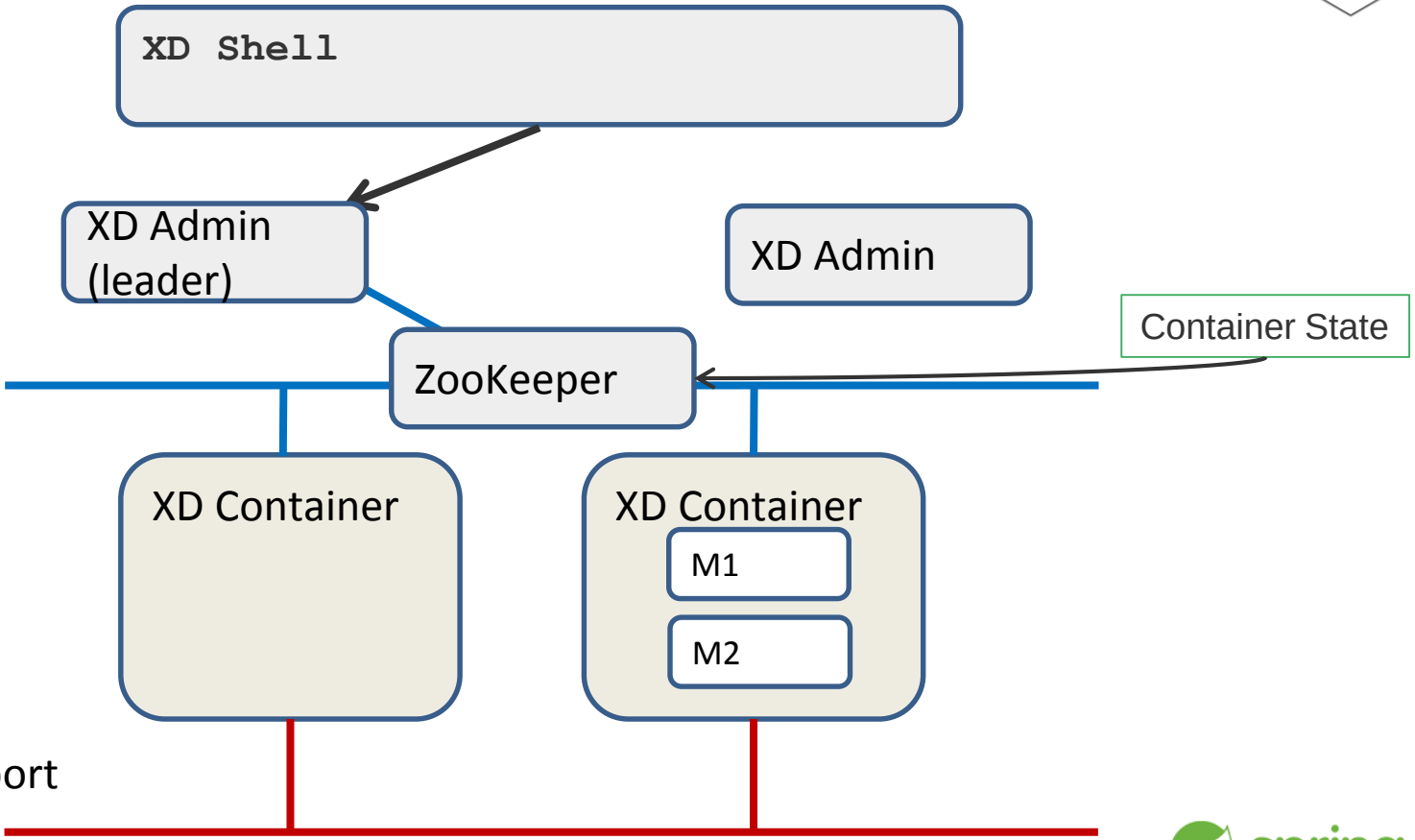
Spring XD – Runtime – Fault Tolerance



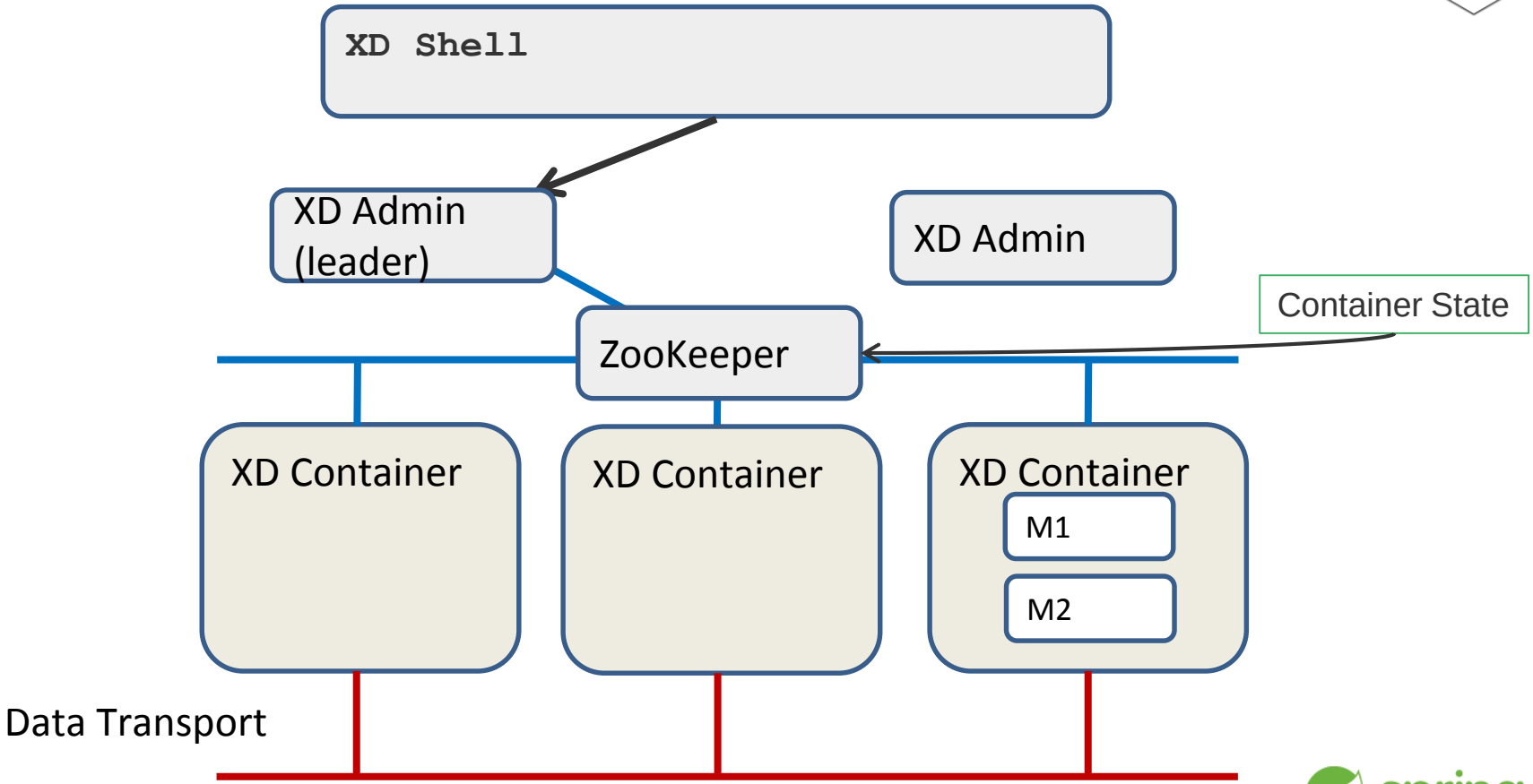
Spring XD – Runtime – Fault Tolerance



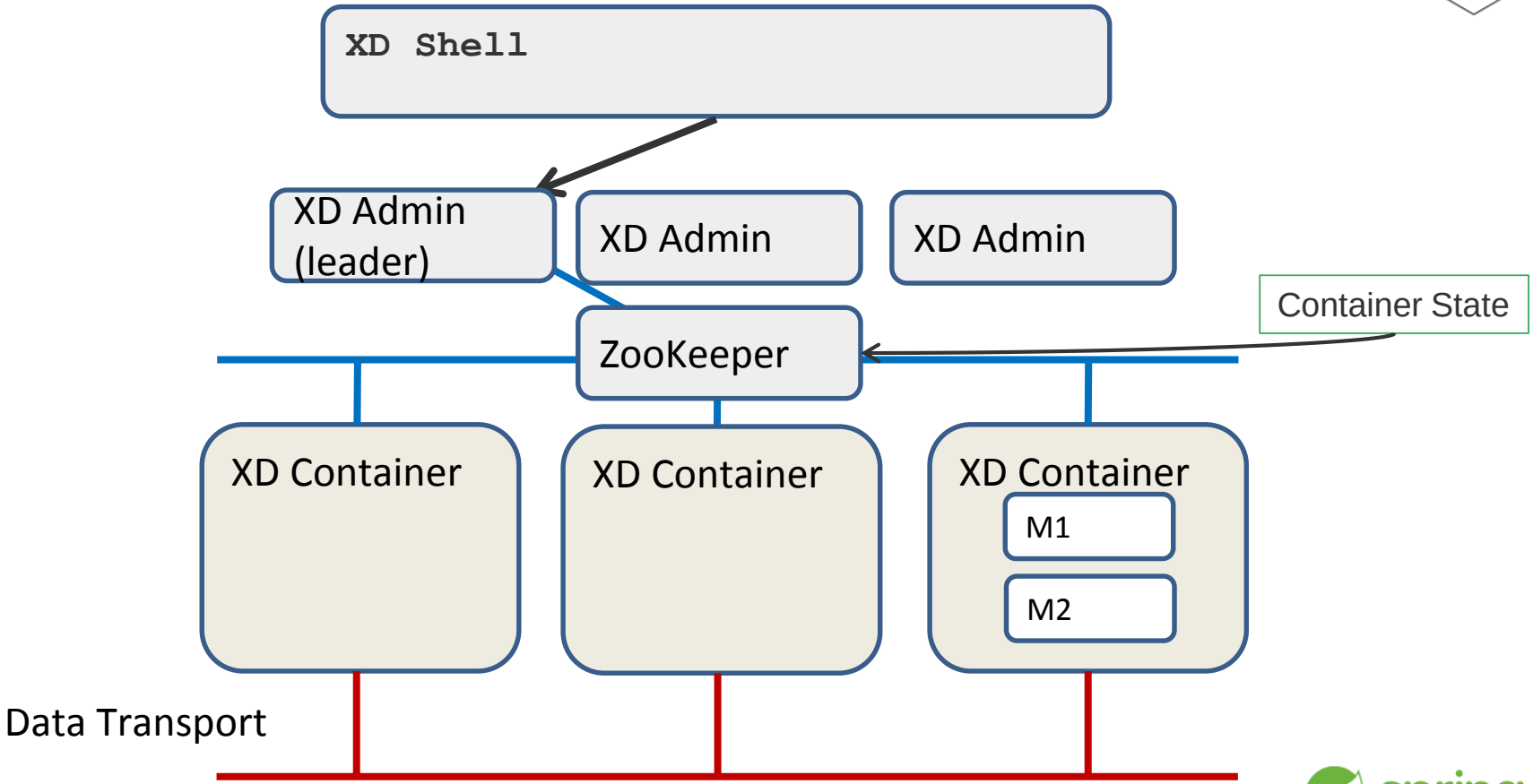
Spring XD – Runtime – Fault Tolerance



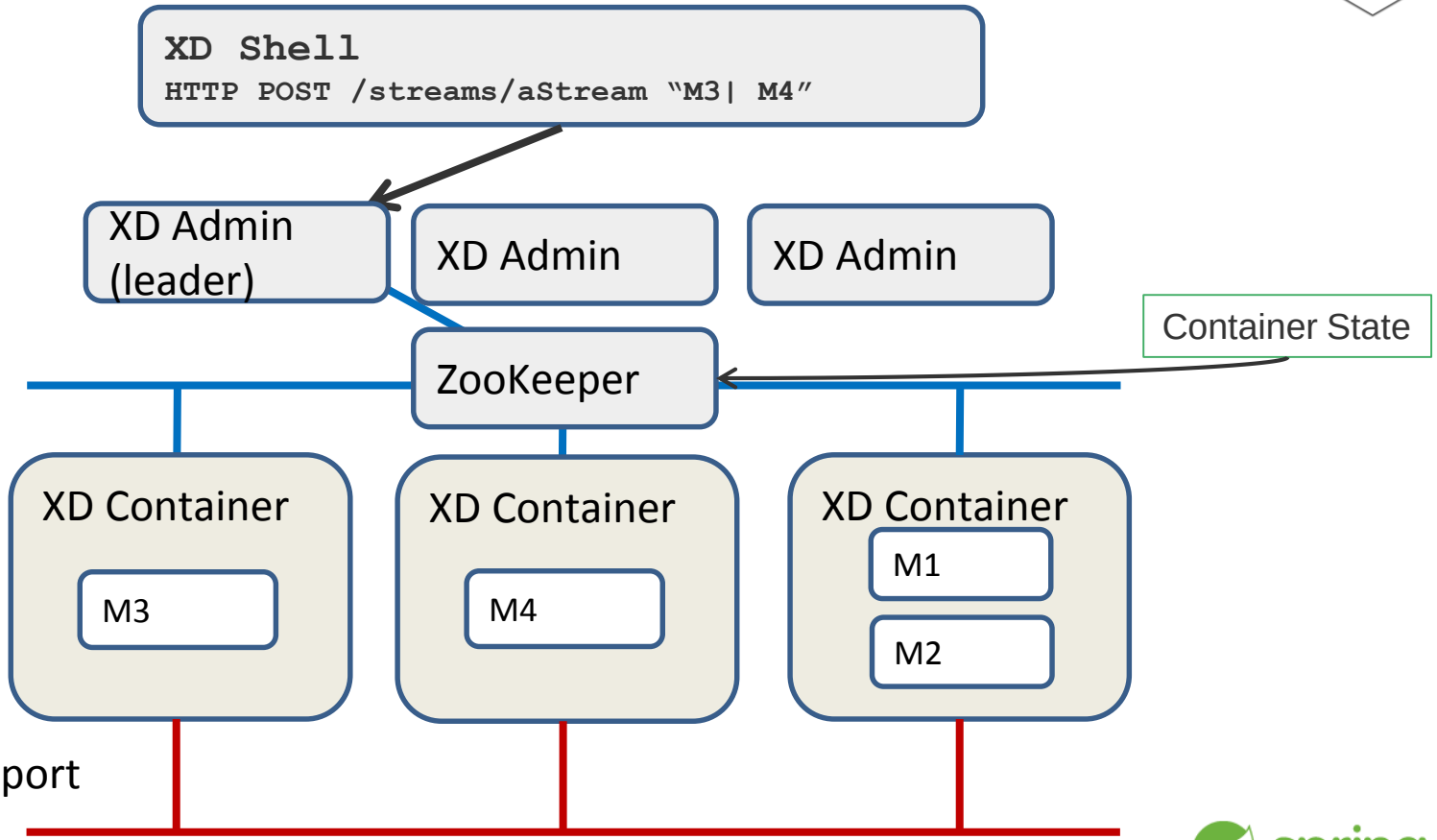
Spring XD – Runtime – Fault Tolerance



Spring XD – Runtime – Fault Tolerance



Spring XD – Runtime – Fault Tolerance

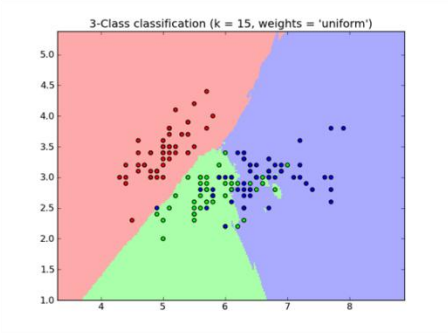


Predictive Models

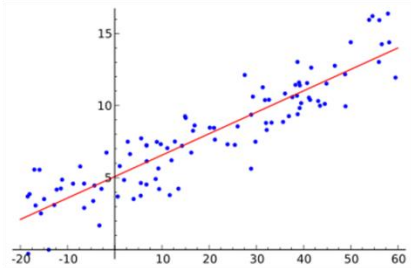
Predictive Models

Data → Algorithm → Model → New Data → Prediction

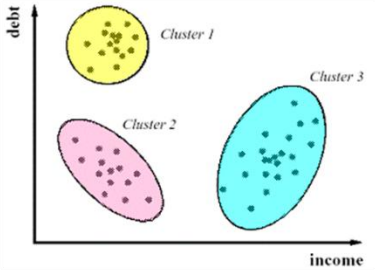
Classification



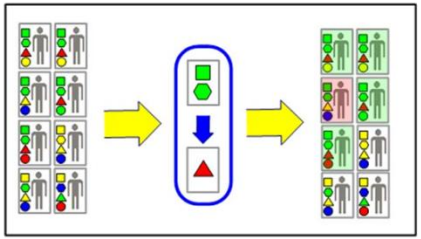
Regression



Clustering



Associations



Concepts

- Model
  - Parameterized algorithm
- Model Building
  - Derive a parameterized algorithm from the data
  - Slow process. Done offline, as a batch process, due to amount of data involved
- Model Scoring
  - Use the model to predict new information
  - Fast process. Can be done as part of stream processing

PMML

- Predictive Model Markup Language
- XML interchange format for analytical models
- From the Data Mining Group http://www.dmg.org
- Processing + models
- Supported by statistics and data minig tools
  - R/Rattle, SAS Enterprise Miner, SPSS, Weka
- Java Evaluator API
  - JPMML-Evaluator project
  - Provides model scoring

springone 2GX DALLAS 2014

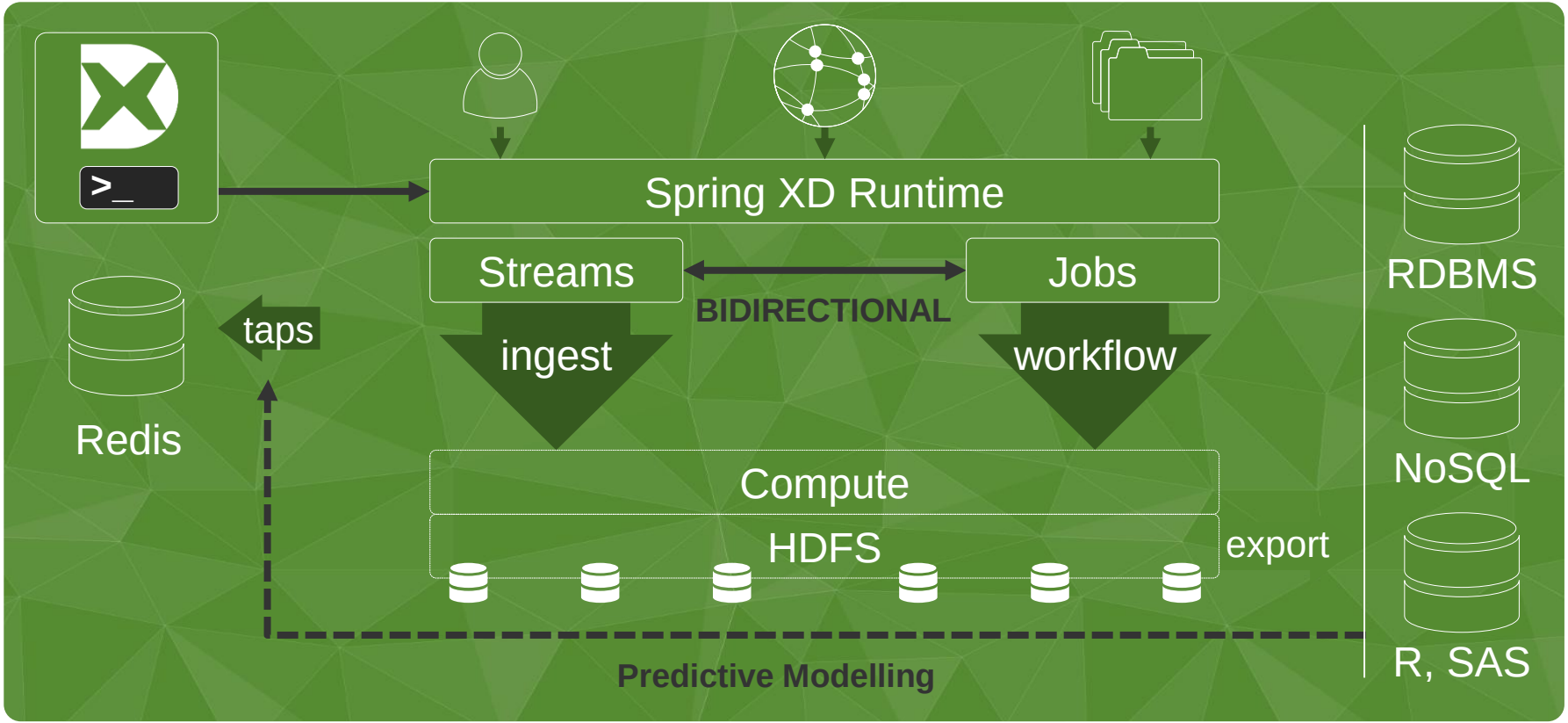
spring

Demo:

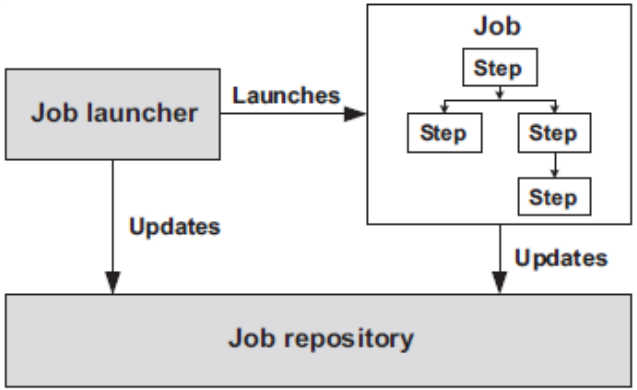
Spring XD – Predictive Models

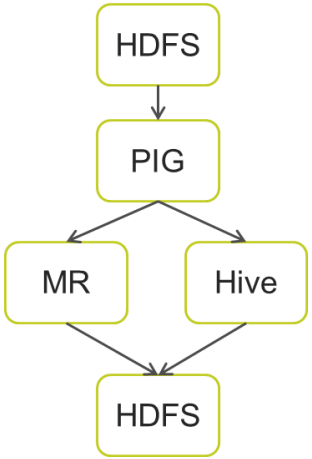
Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Spring XD: Unified Platform for Big Data



Jobs





CSV to JDBC

FTP to HDFS

JDBC to HDFS

HDFS to JDBC

HDFS to MongoDB

Learn More…

- Project: http://projects.spring.io/spring-xd/
- GitHub: https://github.com/spring-projects/spring-xd/
- Issues: https://jira.springsource.org/browse/XD
- Wiki: https://github.com/spring-projects/spring-xd/wiki
- Samples: https://github.com/spring-projects/spring-xd-samples