redhat.

An Introduction to Apache Spark
Big Data Madison: 29 July 2014

William Benton
@willb
Red Hat, Inc.

About me

- At Red Hat for almost 6 years, working on distributed computing
- Currently contributing to Spark, active Fedora packager and sponsor
- Before Red Hat: concurrency and program analysis research

Forecast

- Background
- Resilient Distributed Datasets
- Spark Libraries
- Community Overview

Background

Processing data in 2005

- MapReduce paper applied very old ideas to distributed data processing
- Hadoop provided open-source MR and distributed FS implementations
- MR allows scale-out on commodity clusters for some real problems

redhat.

Processing data in 2005

DEAN & GHEMAWAT, 2004

- MapReduce paper applied very old ideas to distributed data processing
- Hadoop provided open-source MR and distributed FS implementations
- MR allows scale-out on commodity clusters for some real problems

redhat.

Processing data in 2005

DEAN & GHEMAWAT, 2004 MCCARTHY, 1960

- MapReduce paper applied very old ideas to distributed data processing
- Hadoop provided open-source MR and distributed FS implementations
- MR allows scale-out on commodity clusters for some real problems

Hadoop and MR shortcomings

- MR works well for batch jobs; suffers for iterative or interactive ones
- Special-purpose extensions for machine learning, query, etc.
- MR model is not a natural fit for many programmers or programs

redhat.

Hadoop and MR shortcomings

- MR works well for batch jobs; suffers for iterative or interactive ones
- Special-purpose extensions for machine learning, query, etc.

MAHOUT

- MR model is not a natural fit for many programmers or programs

redhat.

Hadoop and MR shortcomings

- MR works well for batch jobs; suffers for iterative or interactive ones
- Special-purpose extensions for machine learning, query, etc.

MAHOUT

HIVE, PIG

- MR model is not a natural fit for many programmers or programs

Hadoop and MR shortcomings

- MR works well for batch jobs; suffers for iterative or interactive ones
- Special-purpose extensions for machine learning, query, etc.

MAHOUT HIVE, PIG

- MR model is not a natural fit for many programmers or programs

Hadoop and MR shortcomings

- MR works well for batch jobs; suffers for iterative or interactive ones
- Special-purpose extensions for machine learning, query, etc.

MAHOUT HIVE, PIG

- MR model is not a natural fit for many programmers or programs

DATA SCIENTIST TIME: $460/8 HOURS
EC2 C3 LARGE INSTANCE: $0.84/8 HOURS

redhat.

Apache Spark

- Introduced in 2009; donated to Apache in 2013; 1.0 release in 2014
- Based on a fundamental abstraction rather than an execution model
- Supports in-memory computing and a wide range of problems

Spark is general

redhat.

Spark is general

Graph

Spark core

redhat.

Spark is general

Spark is general

redhat

Spark is general

redhat

Spark is general

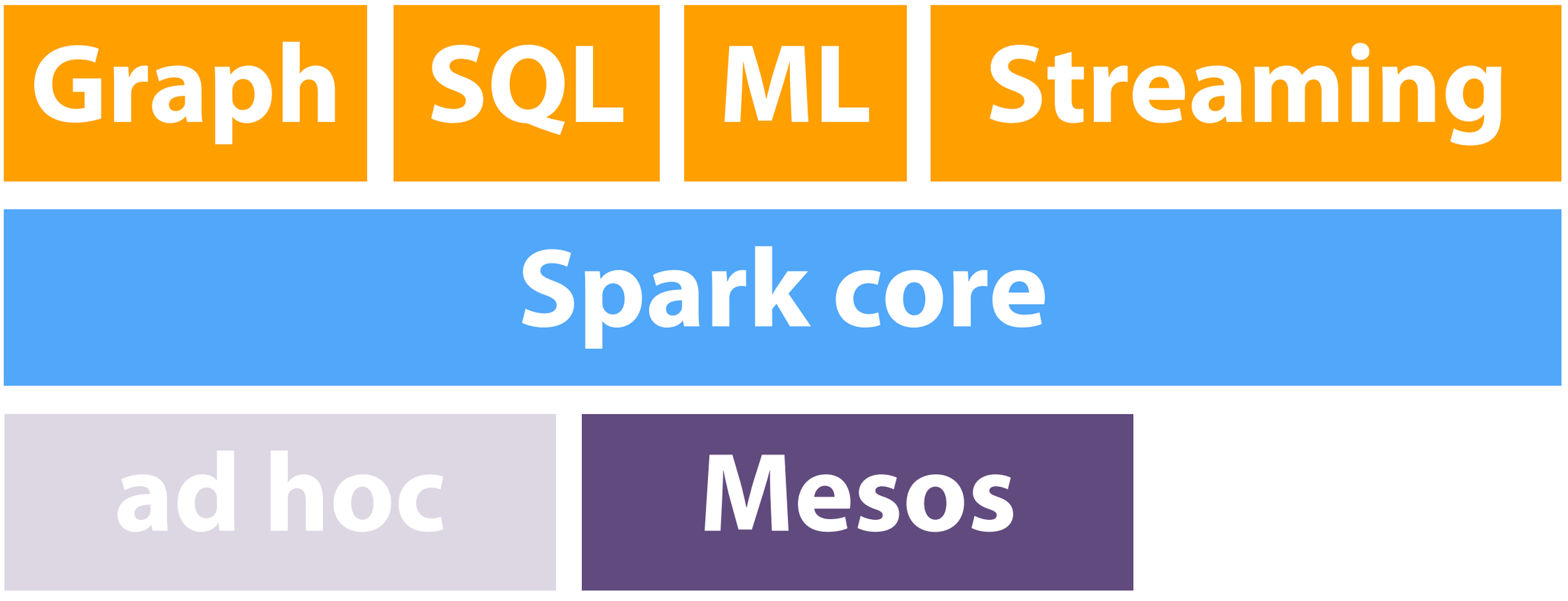
Graph | SQL | ML | Streaming

Spark core

ad hoc

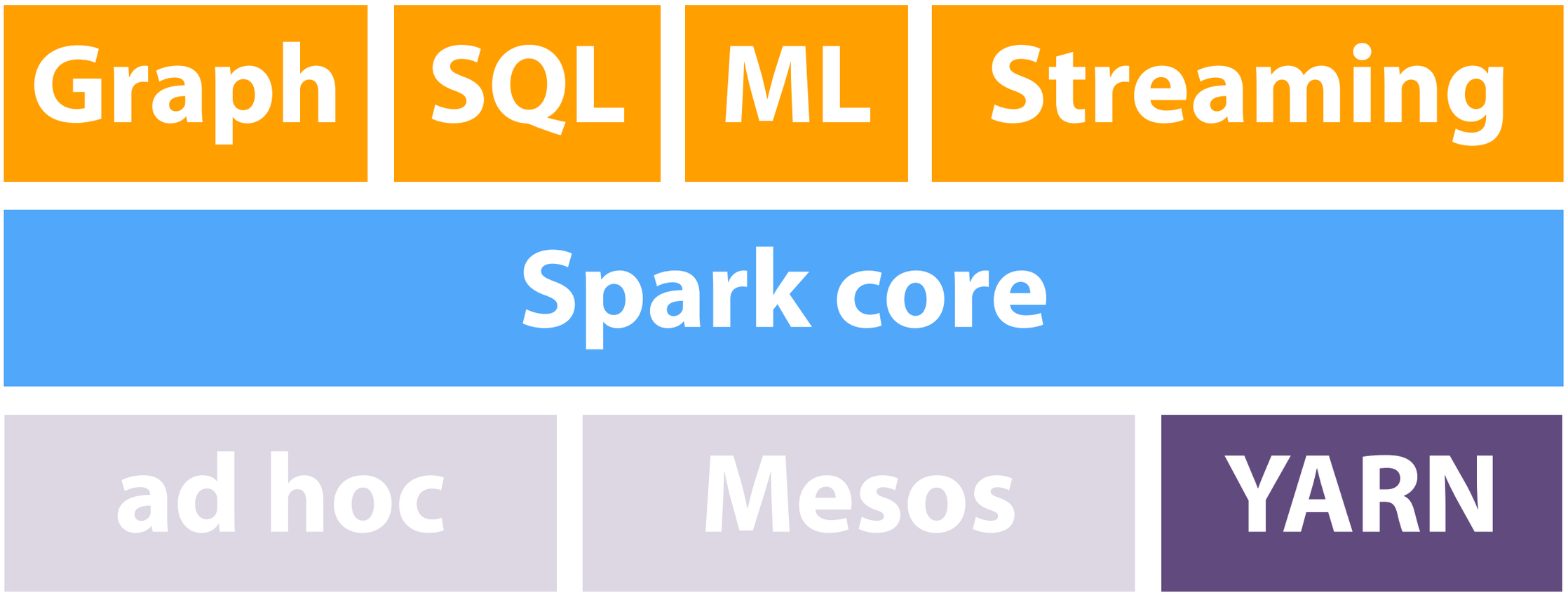
redhat

Spark is general



redhat

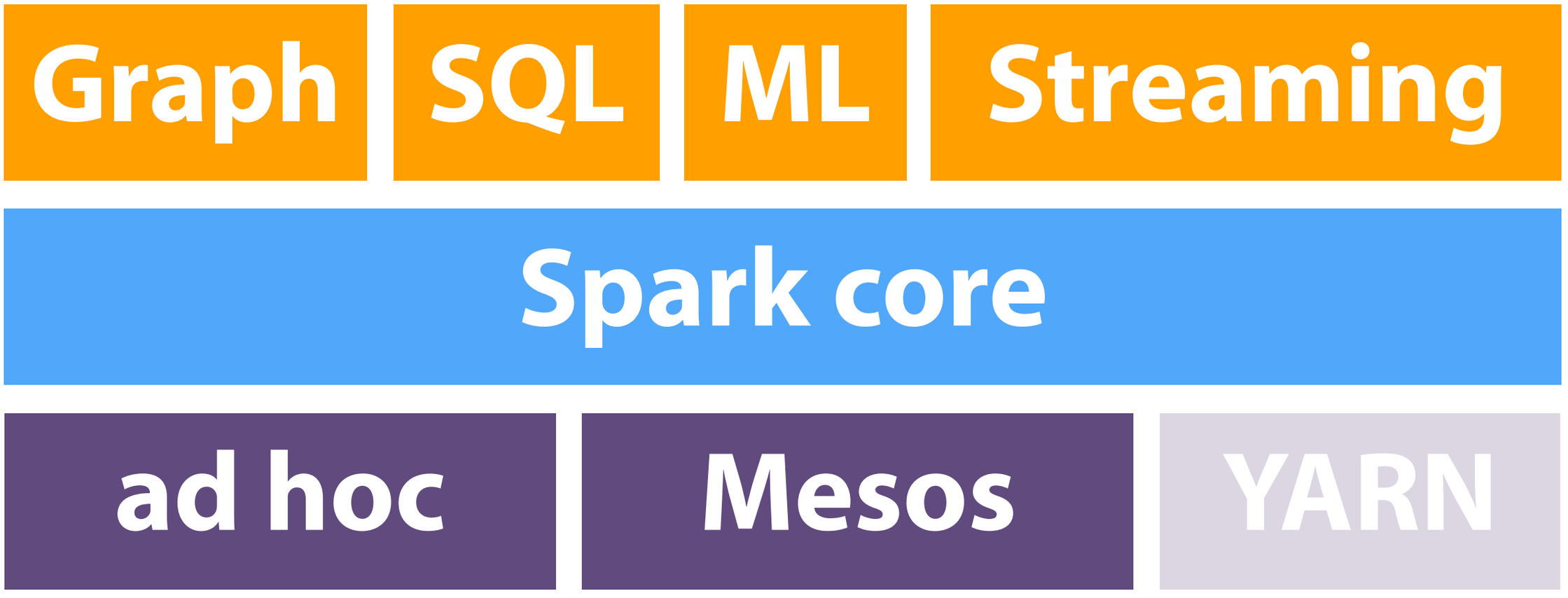
Spark is general



redhat

Spark is general

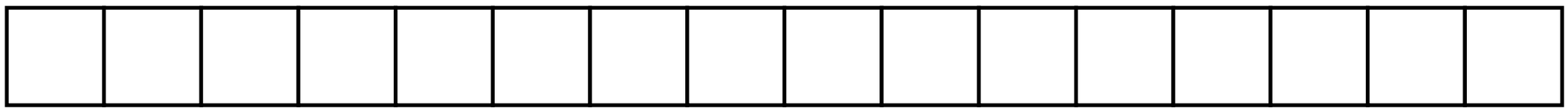
APIS FOR SCALA, JAVA, PYTHON, AND R
(3RD-PARTY BINDINGS FOR CLOJURE ET AL.)



RDDs

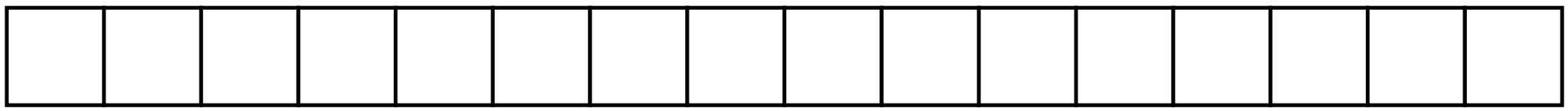
redhat.

Resilient distributed datasets



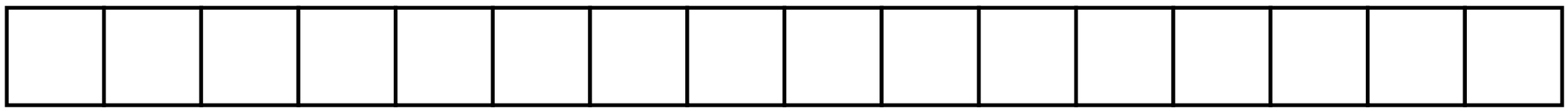
redhat.

Resilient distributed datasets



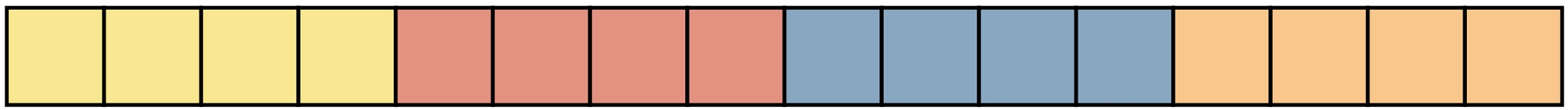
redhat.

Resilient distributed datasets



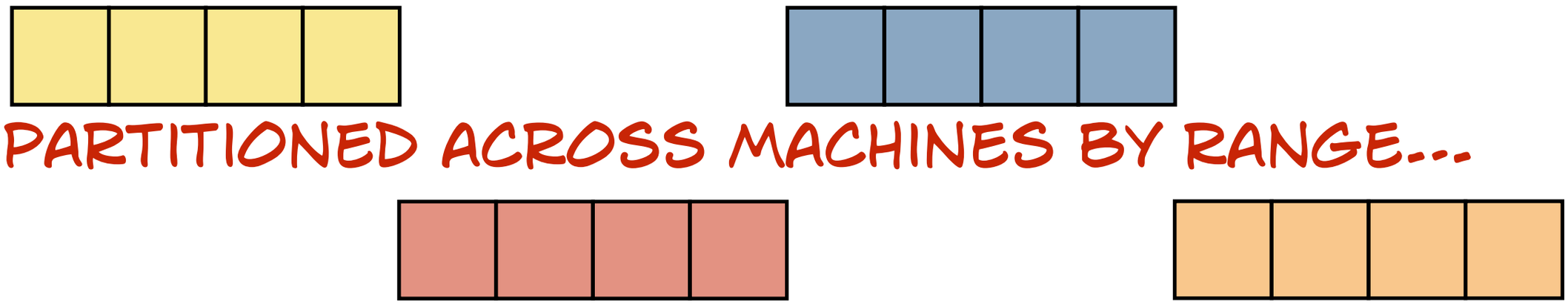
redhat.

Resilient distributed datasets



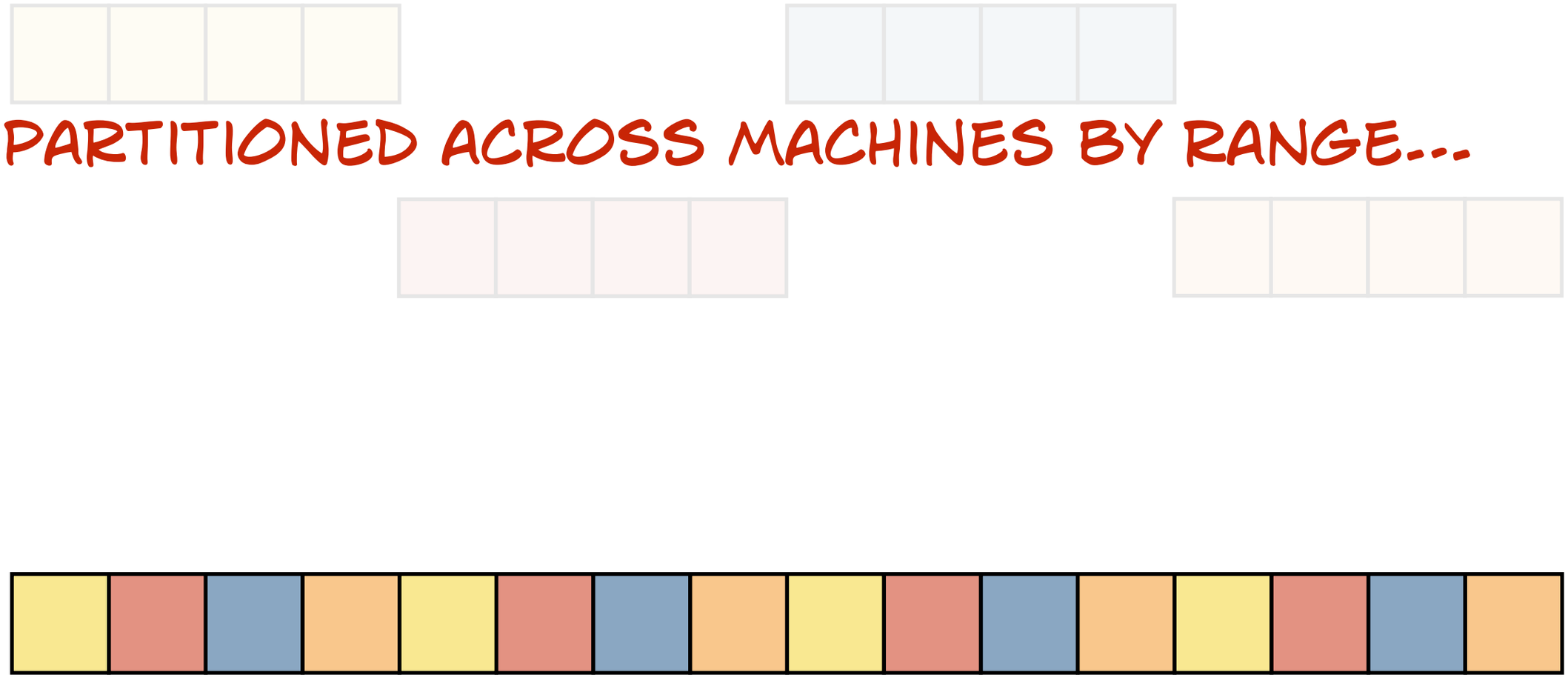
redhat.

Resilient distributed datasets



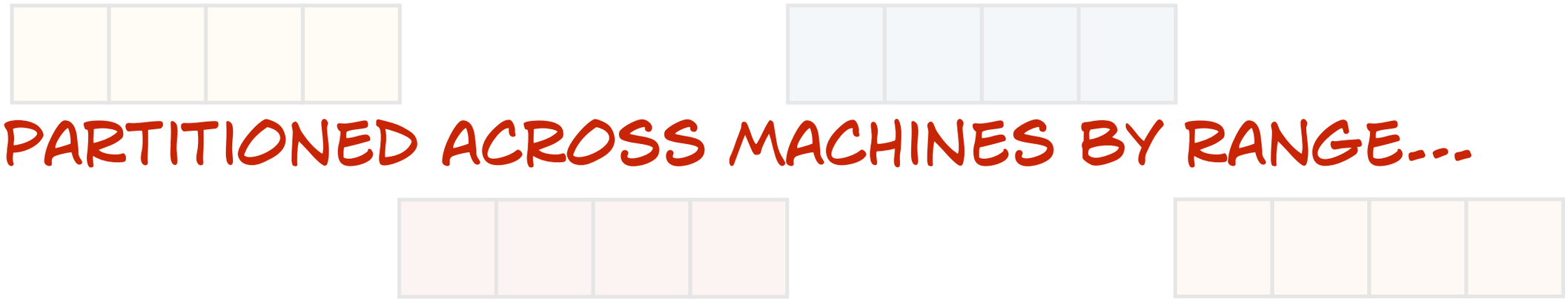
redhat.

Resilient distributed datasets

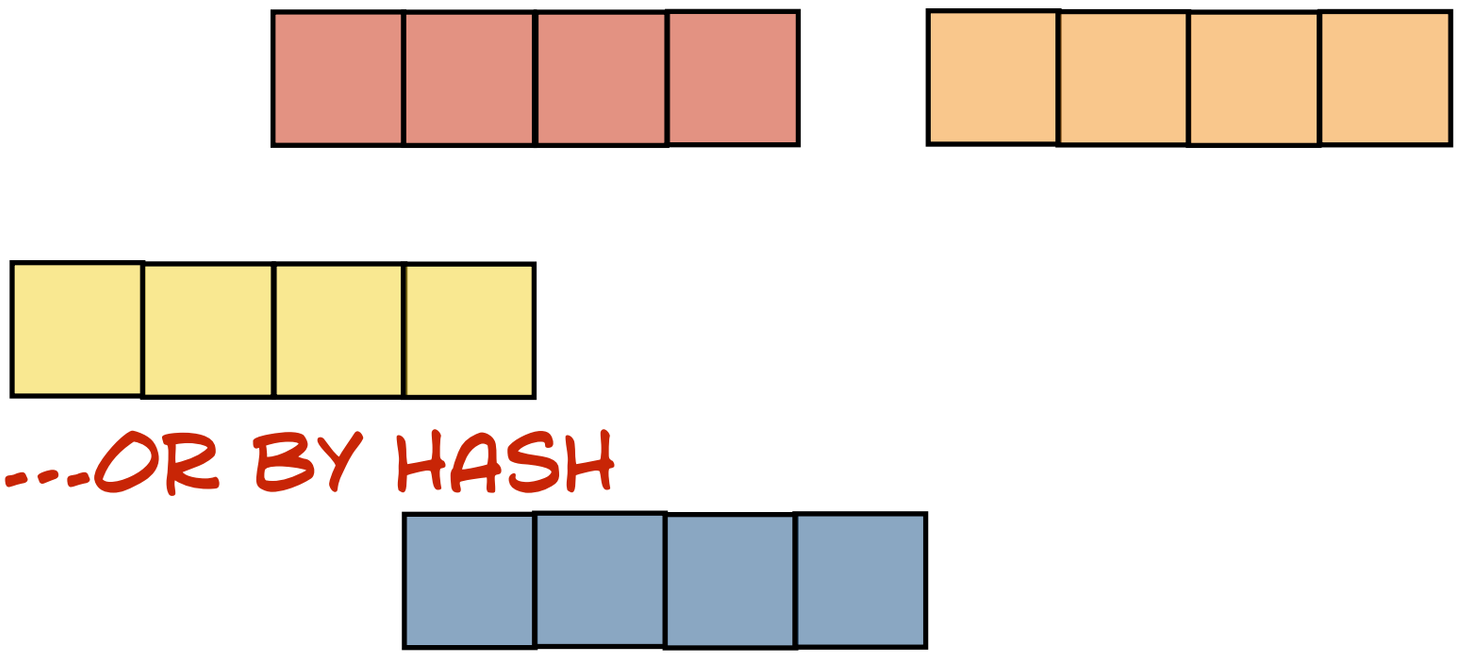


redhat.

Resilient distributed datasets



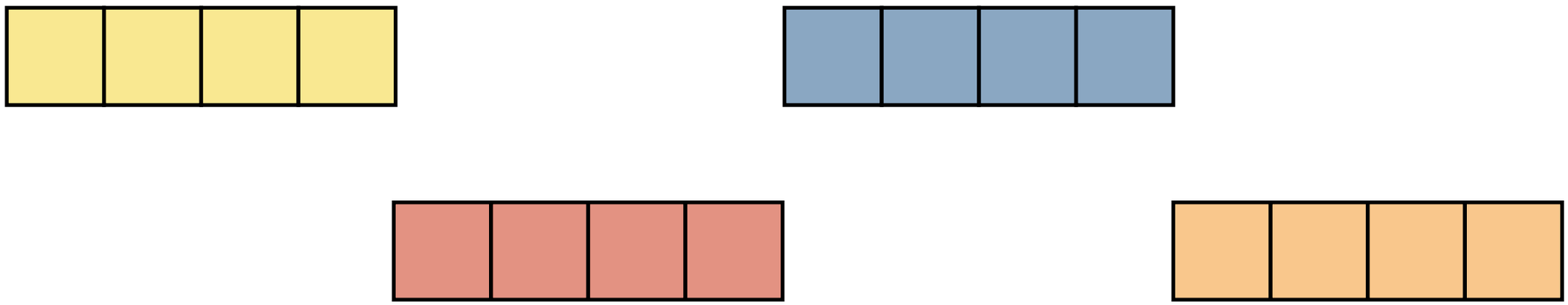
PARTITIONED ACROSS MACHINES BY RANGE...



...OR BY HASH

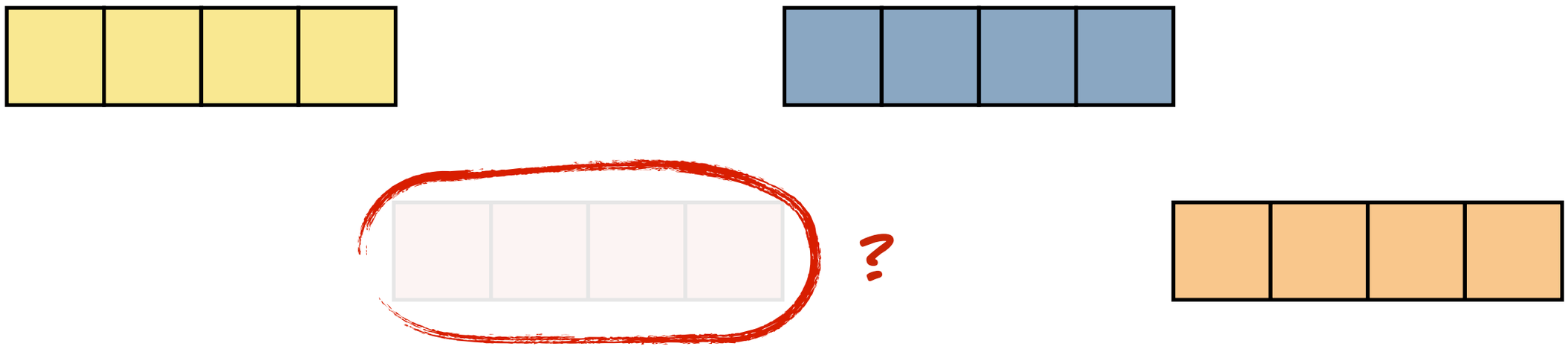
redhat.

Resilient distributed datasets



redhat.

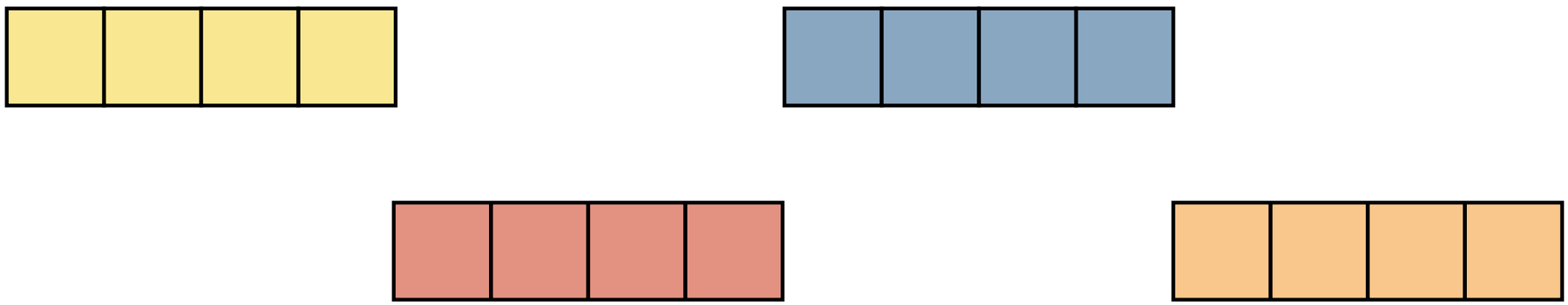
Resilient distributed datasets



FAILURES MEAN PARTITIONS CAN DISAPPEAR...

redhat.

Resilient distributed datasets



FAILURES MEAN PARTITIONS CAN DISAPPEAR...

...BUT THEY CAN BE RECONSTRUCTED!

redhat.

RDDs are partitioned, immutable, lazy collections

redhat.

RDDs are partitioned, immutable, lazy collections

TRANSFORMATIONS CREATE NEW RDDS
THAT ENCODE A DEPENDENCY DAG

ACTIONS RESULT IN EXECUTING CLUSTER
JOBS & RETURN VALUES TO THE DRIVER

redhat.

RDDs (more formally)

- A set of partitions
- Lineage information
- A function to compute partitions from parent partitions
- A partitioning strategy
- Preferred locations for partitions

RDDs (more formally)

- A set of partitions
- Lineage information
- A function to compute partitions from parent partitions
- A partitioning strategy
- Preferred locations for partitions

(THESE ARE OPTIONAL)

redhat.

Creating RDDs

- **From a collection:** `parallelize()`
- **…a local or remote file:** `textFile()`
- **…or HDFS:** `hadoopFile(); sequenceFile();objectFile()`
- **(These all act lazily)**

RDD[T] transformations

- `map(f: T=>U): RDD[U]`
- `flatMap(f: T=>Seq[U]): RDD[U]`
- `filter(f: T=>Boolean): RDD[T]`
- `distinct(): RDD[T]`
- `keyBy(f: T=>K): RDD[(K, T)]`

redhat.

RDD[(K,V)] transformations

- `sortByKey(): RDD[(K,V)]`
- `groupByKey(): RDD[(K,Seq[V])]`
- `reduceByKey(f: (V,V)=>V): RDD[(K,V)]`
- `join(other: RDD[(K,W)]): RDD[(K,(V,W))]`

redhat.

RDD[(K,V)] transformations

- `sortByKey(): RDD[(K,V)]`
- `groupByKey(): RDD[(K,Seq[V])]`
- `reduceByKey(f: (V,V)=>V): RDD[(K,V)]`
- `join(other: RDD[(K,W)]): RDD[(K,(V,W))]`

...AND MANY OTHERS INCLUDING CARTESIAN PRODUCT, COGROUP, SET OPERATIONS, ETC.

Other RDD transformations

- **Explicitly** `repartition` **and shuffle or** `coalesce` **to fewer partitions**
- **Provide hints to** `cache` **an intermediate RDD in memory or** `persist` **it to memory and/or disk**

(REMEMBER: ALL TRANSFORMATIONS ARE LAZY)

redhat.

RDD[T] actions

- `collect(): Array[T]`
- `count(): Long`
- `reduce(f: (T,T)=>T): T`
- `saveAsTextFile(path)`
- `saveAsSequenceFile(path)`

RDD[T] actions

- `collect(): Array[T]`
- `count(): Long`
- `reduce(f: (T,T)=>T): T`
- `saveAsTextFile(path)`
- `saveAsSequenceFile(path)`

...AND MANY OTHERS INCLUDING FOREACH, TAKE, SAMPLING, ETC.

redhat.

RDD[T] actions

(REMEMBER: ALL ACTIONS ARE EAGER)

- `collect(): Array[T]`
- `count(): Long`
- `reduce(f: (T,T)=>T): T`
- `saveAsTextFile(path)`
- `saveAsSequenceFile(path)`

...AND MANY OTHERS INCLUDING FOREACH, TAKE, SAMPLING, ETC.

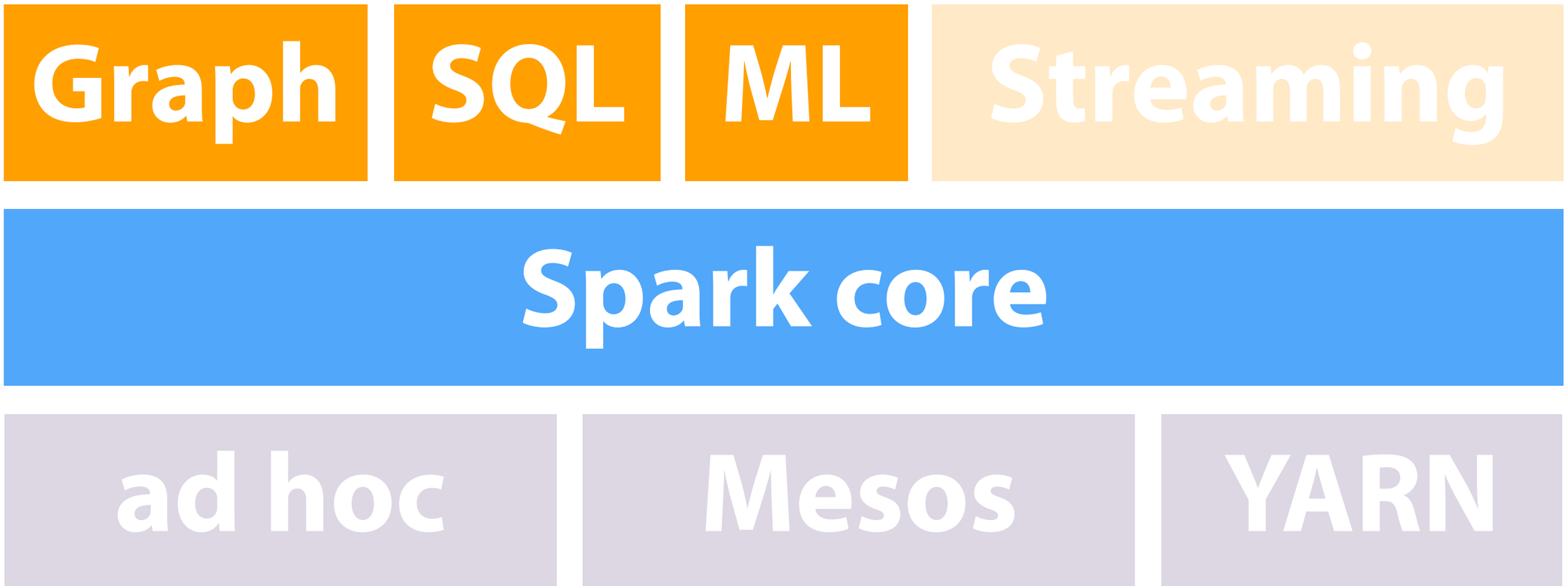
redhat.

Example: word count in Spark

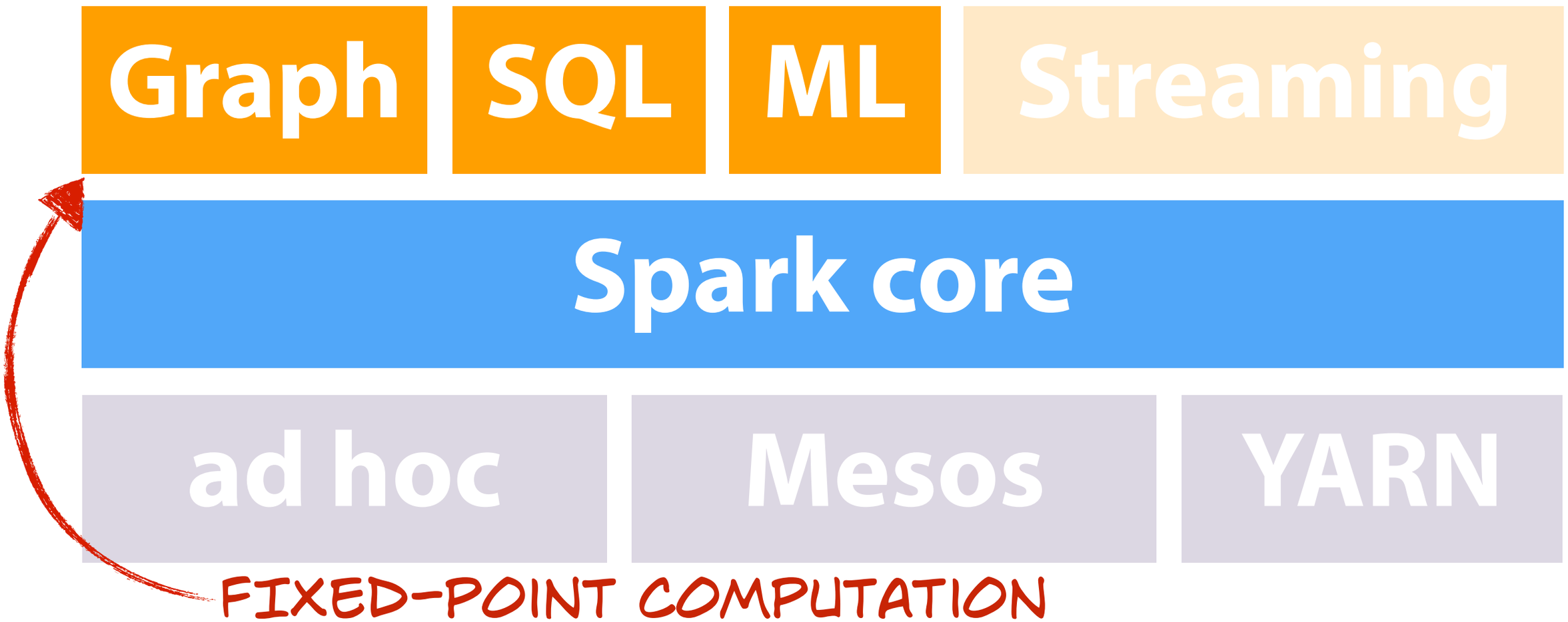
```
val file = spark.textFile("hdfs://...")

val counts = file.flatMap(line =>
                  line.split(" "))
                 .map(word => (word, 1))
                 .reduceByKey(_ + _)

counts.saveAsTextFile("hdfs://...")
```

Libraries

Spark libraries



Spark libraries



redhat.

Spark libraries

QUERY EXECUTION

Graph | SQL | ML | Streaming

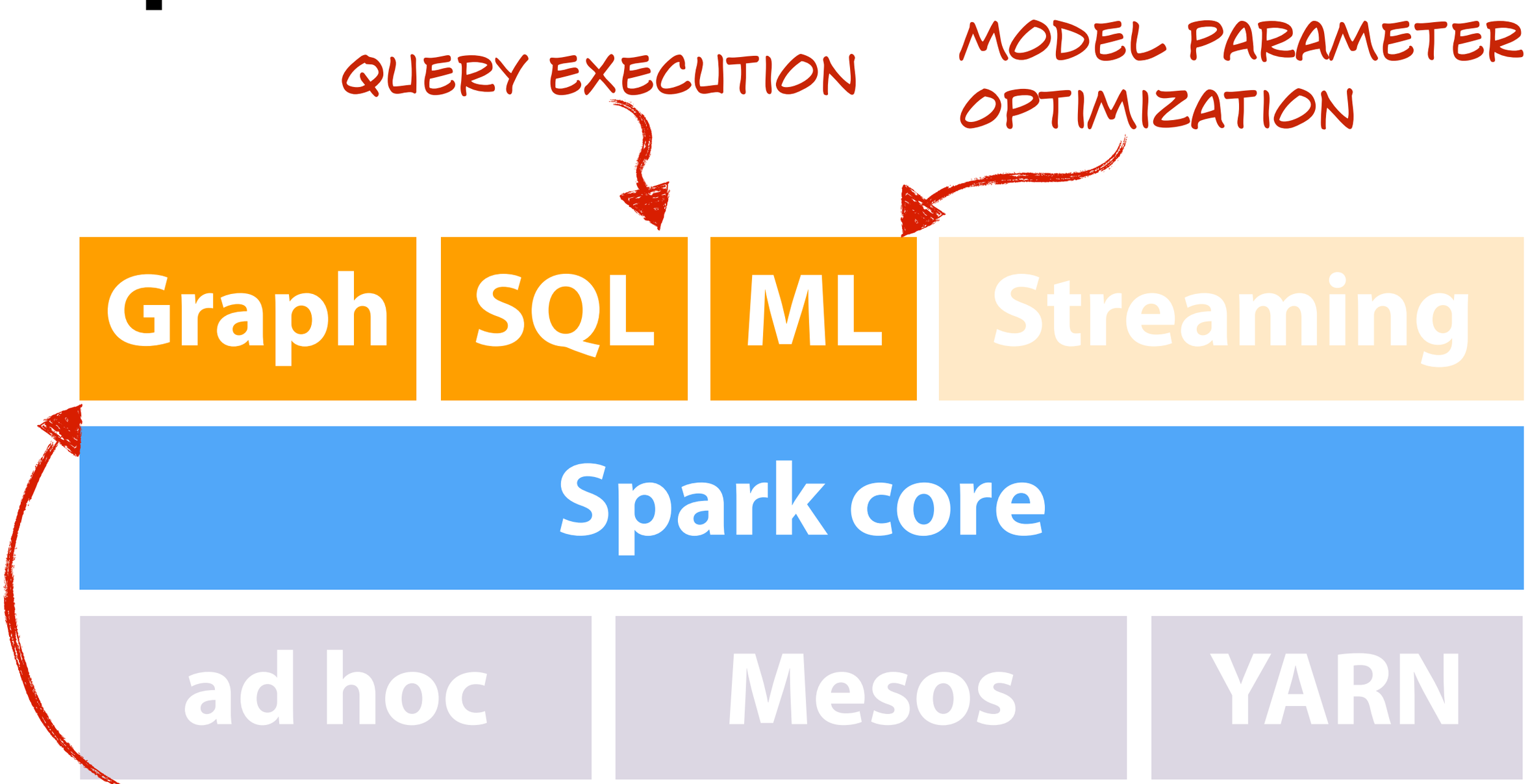
Spark core

ad hoc | Mesos | YARN

FIXED-POINT COMPUTATION

redhat

Spark libraries



redhat

Spark libraries

QUERY EXECUTION

MODEL PARAMETER OPTIMIZATION

Graph | SQL | ML | Streaming

Spark core

ad hoc | Mesos | YARN

FIXED-POINT COMPUTATION

redhat.

Spark MLlib

- Implementations of classic learning algorithms on data in RDDs
- Regression, classification, clustering, recommendation, filtering, etc.
- High performance due to caching, in-memory execution

redhat.

Spark SQL features

- SchemaRDD type
- Catalyst: a relational algebra analysis/optimization framework
- SQL and HiveQL implementations; Hive warehouse support
- LINQ-style embedded query DSL

redhat.

Spark SQL example

```
case class Trackpoint(lat: Double,
  lon: Double, ts: Long) {}
```

```
// assume points is an RDD of Trackpoints
points.registerAsTable("points")

val results =
  sql("""select * from points
         where ts > max(ts) - 600""")
```

Spark Streaming

- **Goal: use the same abstraction for streaming as for batch or interactive**
- ***Discretized stream* abstraction: streams as sequences of RDDs**

Community & Applications

redhat.

Developer community

- https://github.com/apache/spark
- First open-source release in 2010
- 121k lines of code
- 300+ contributors all-time; 80+ in the last month
- Active and friendly mailing list

User community

- Spark Summit: established in 2013; growing and expanding
- Lots of cool applications (BI, medical, geospatial, security, fun)
- Many learning resources

redhat.

User community

MORE THAN 2X AS BIG IN '14

- Spark Summit: established in 2013; growing and expanding
- Lots of cool applications (BI, medical, geospatial, security, fun)
- Many learning resources

redhat.

User community

MORE THAN 2X AS BIG IN '14

- Spark Summit: established in 2013; growing and expanding

SPARK SUMMIT EAST: NYC 2015

- Lots of cool applications (BI, medical, geospatial, security, fun)
- Many learning resources

http://spark-summit.org/2014/agenda

Thanks!

willb@redhat.com
http://chapeau.freevariable.com