Big Data and APACHE Spark™

Brian Clapper, *bmc@databricks.com*

5 October, 2016

databricks™

Who the *@$! is this guy?

Brian Clapper, bmc@databricks.com

- 32+ years building systems for startups and large enterprises
- Independent consultant and trainer for the last 7 years
- Took a full-time position with Databricks in mid-March
- 2+ years teaching front- and back-end technologies
  - Scala, Java, AngularJS, Ruby, Python, JavaScript, Spark
  - I've taught about 20 Spark classes since June, 2015
- Scala programmer since early 2009 (Scala 2.7)
  - Founder and organizer of Philly Area Scala Enthusiasts (PHASE)
  - Co-organizer of annual Northeast Scala Symposium

First, a few questions

- What’s your programming background?
- Who has done any kind of data processing?

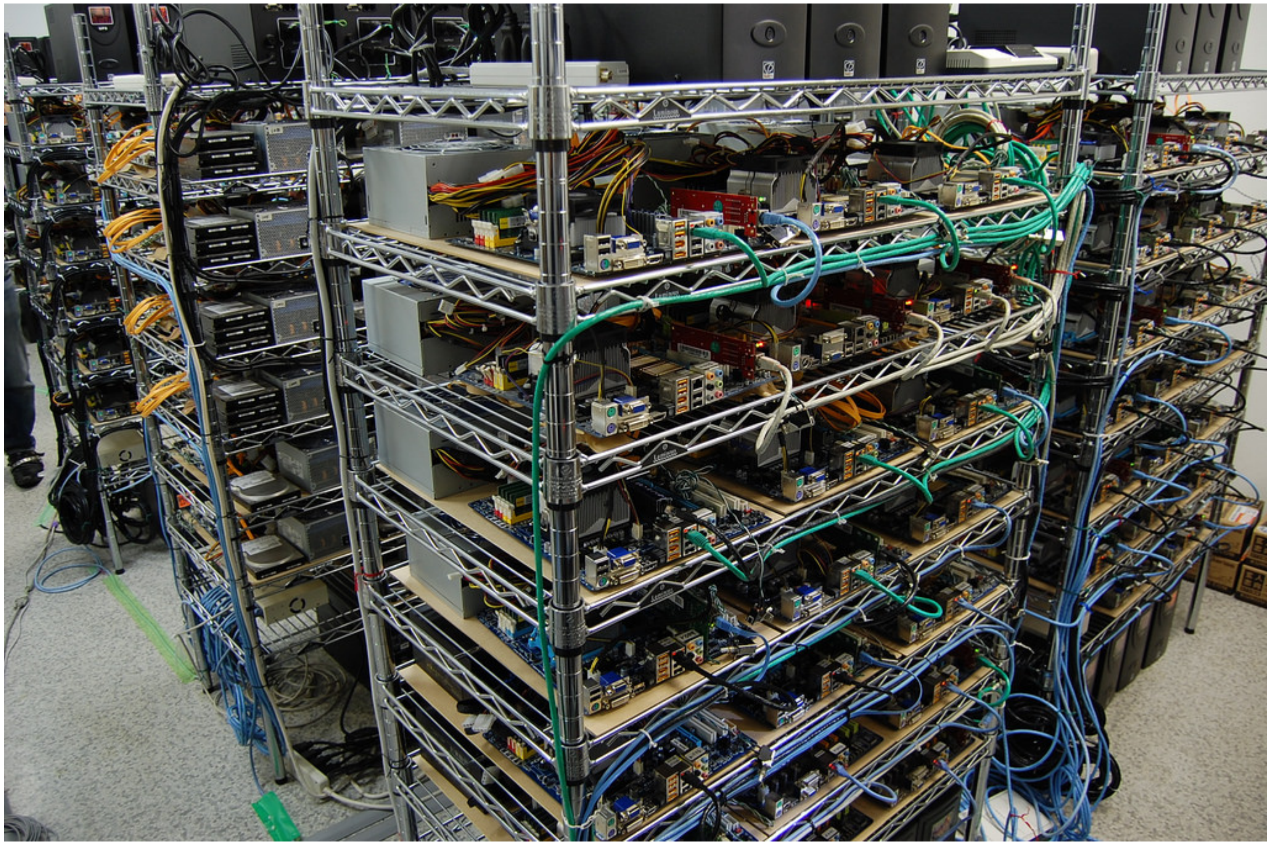
What is Big Data?

- ???
- (I’m waiting for *you* to answer…)

Strategies for Processing Big Data

Buy a *really big* computer.

Strategies for Processing Big Data



Buy many *smaller* computers.

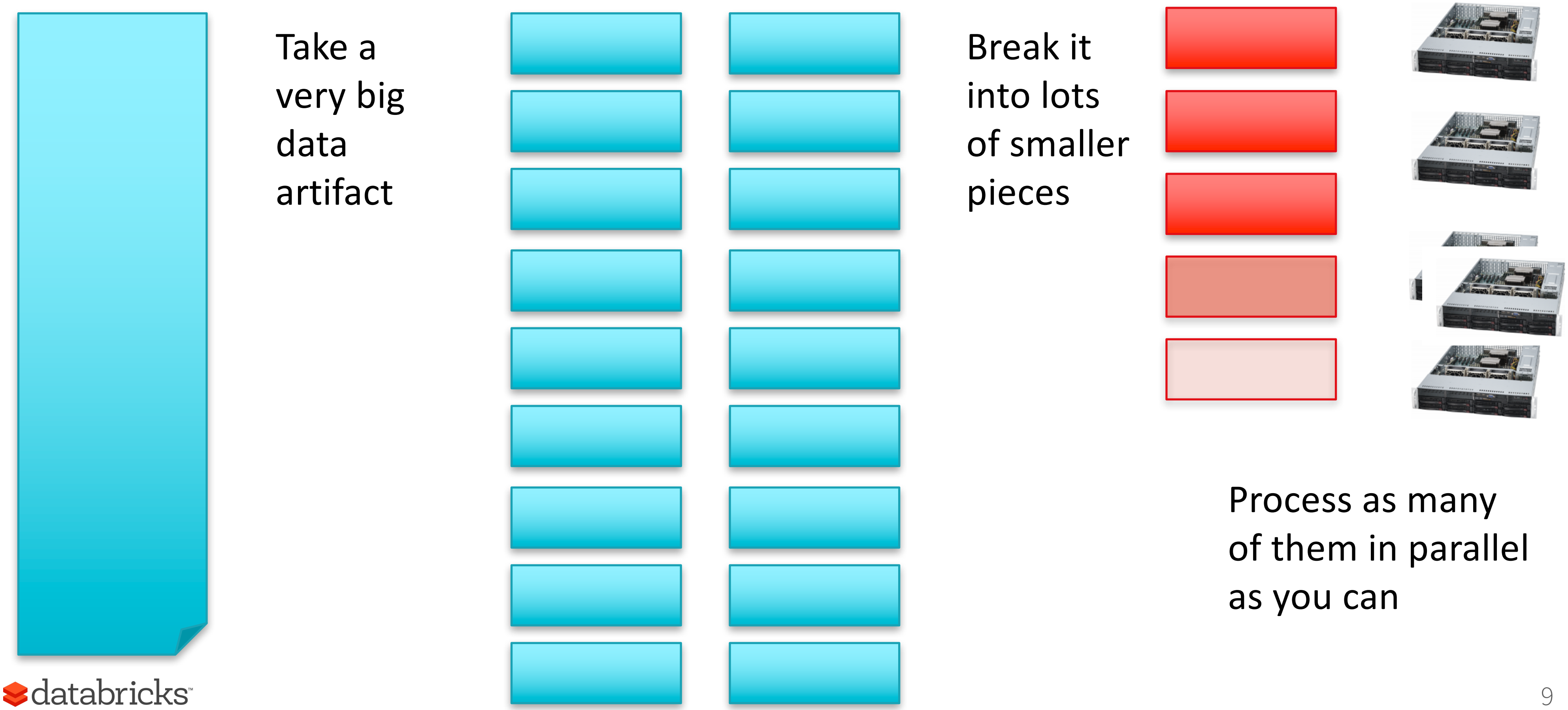
Strategies for Processing Big Data

Rent someone else's spare capacity.

Apache Spark

- Will work on any of the above
- Designed for *distributed* processing (scale out, not up)
- Optimized to prefer memory over disk
- Built with *functional programming* concepts in mind
  - Immutability, in particular
- Easier to code than Hadoop Map/Reduce
  - IMHO

Concepts



Take a very big data artifact

Break it into lots of smaller pieces

Process as many of them in parallel as you can

What are things we do with big data?

- ETL
  - Converting to more useful formats
  - Feeding data warehouses and data lakes
- Data analysis
- Machine learning
  - Recommendation algorithms
  - Predictive algorithms (e.g., logistic regression)
  - Sifting algorithms (e.g., K-means clustering)
  - and more
- Can you think of other things?

Apache Spark

- Let me scribble on the white board for awhile
- … then, I have a couple examples.