ADATAO
DATA INTELLIGENCE FOR ALL

R at Scale:

Using Apache Spark & Adatao

Christopher Nguyen, PhD
Co-Founder & CEO

Agenda

1. R + Big Data Science: Problem Statement
2. Big Compute: Solution
3. In-Memory Big-Compute: Why & When
4. Apache Spark & Adatao: Overview & Demo

Christopher Nguyen, PhD

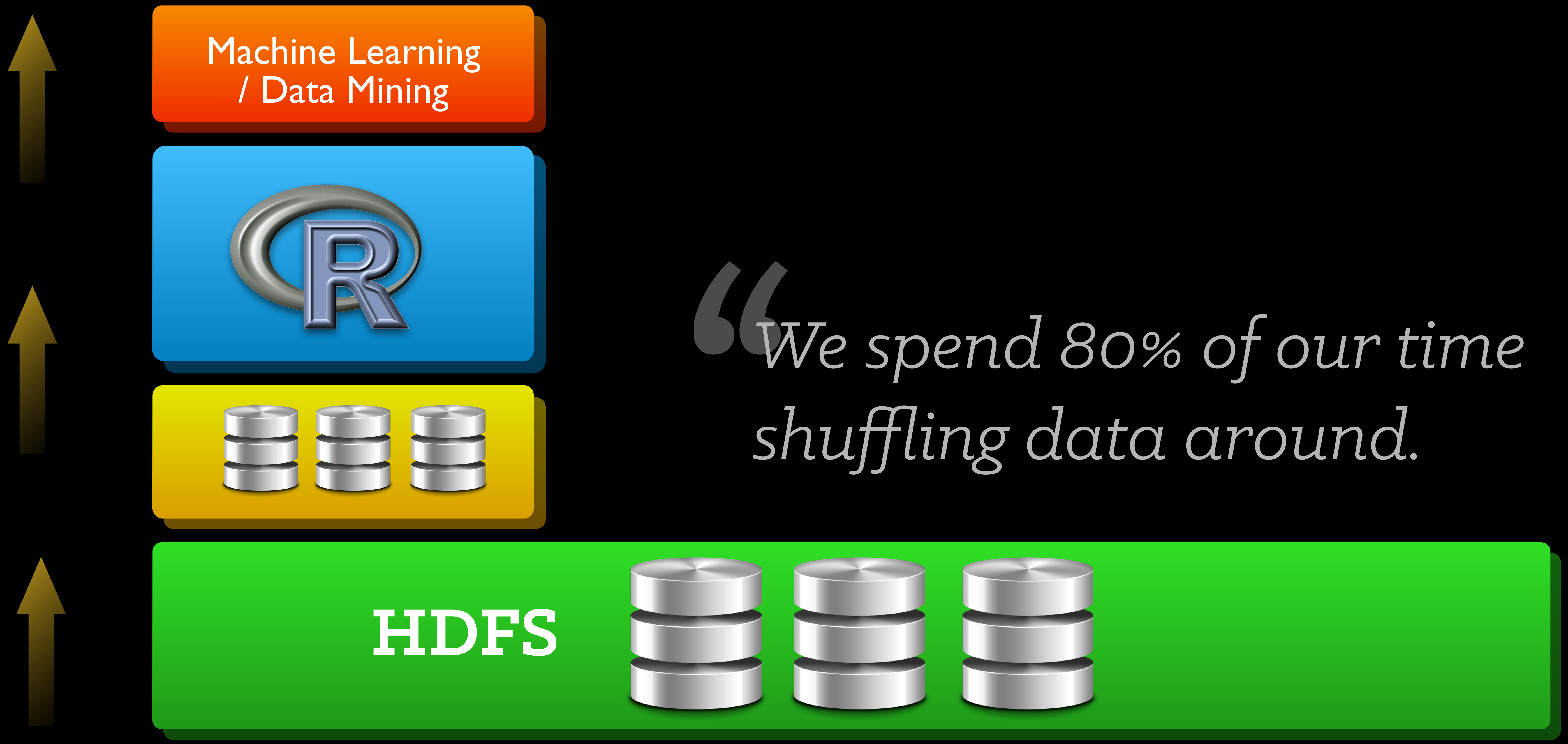
Adatao Inc.

Co-Founder & CEO

- Former **Engineering Director of Google Apps** (Google Founders' Award)
- Former Professor and Co-Founder of the **Computer Engineering program at HKUST**
- **PhD Stanford, BS U.C. Berkeley** *Summa cum Laude*
- **Extensive experience** building technology companies that **solve enterprise challenges**

Conventional approach:

Work on sub-sampled data

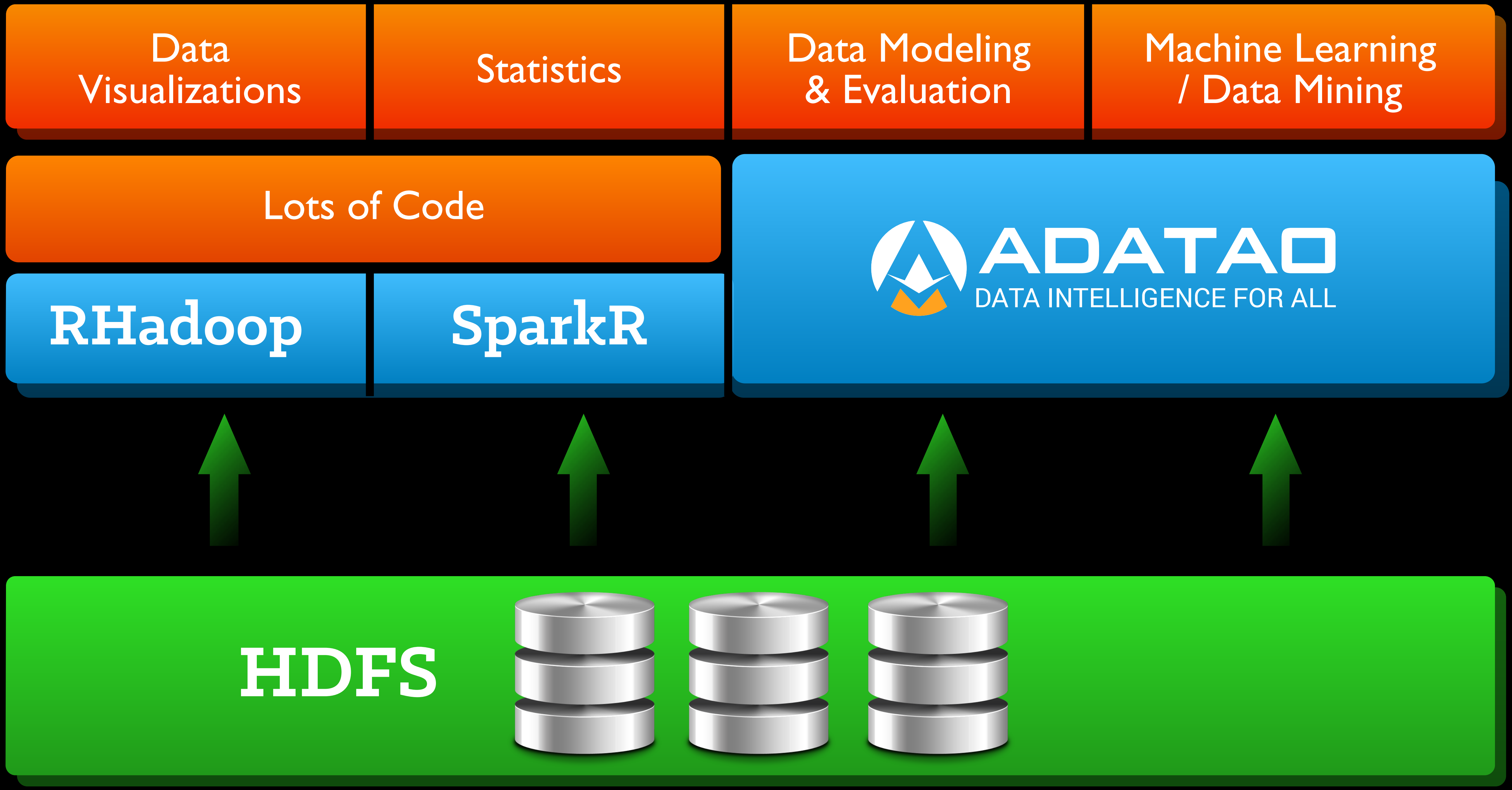


Machine Learning / Data Mining

HDFS

> *“We spend 80% of our time shuffling data around.*

Parallel computing approach:
Work directly on HDFS



Data Visualizations
Statistics
Data Modeling & Evaluation
Machine Learning / Data Mining
Lots of Code
RHadoop
SparkR
ADATAO
DATA INTELLIGENCE FOR ALL
HDFS

ADATAO

DATA INTELLIGENCE FOR ALL

Big Data & Big Compute

Past & Present

How Have We Defined
“Big Data”?

Old Definition

✗ **Big Data has**
Problems

Huge Volume

High Velocity

Great Variety

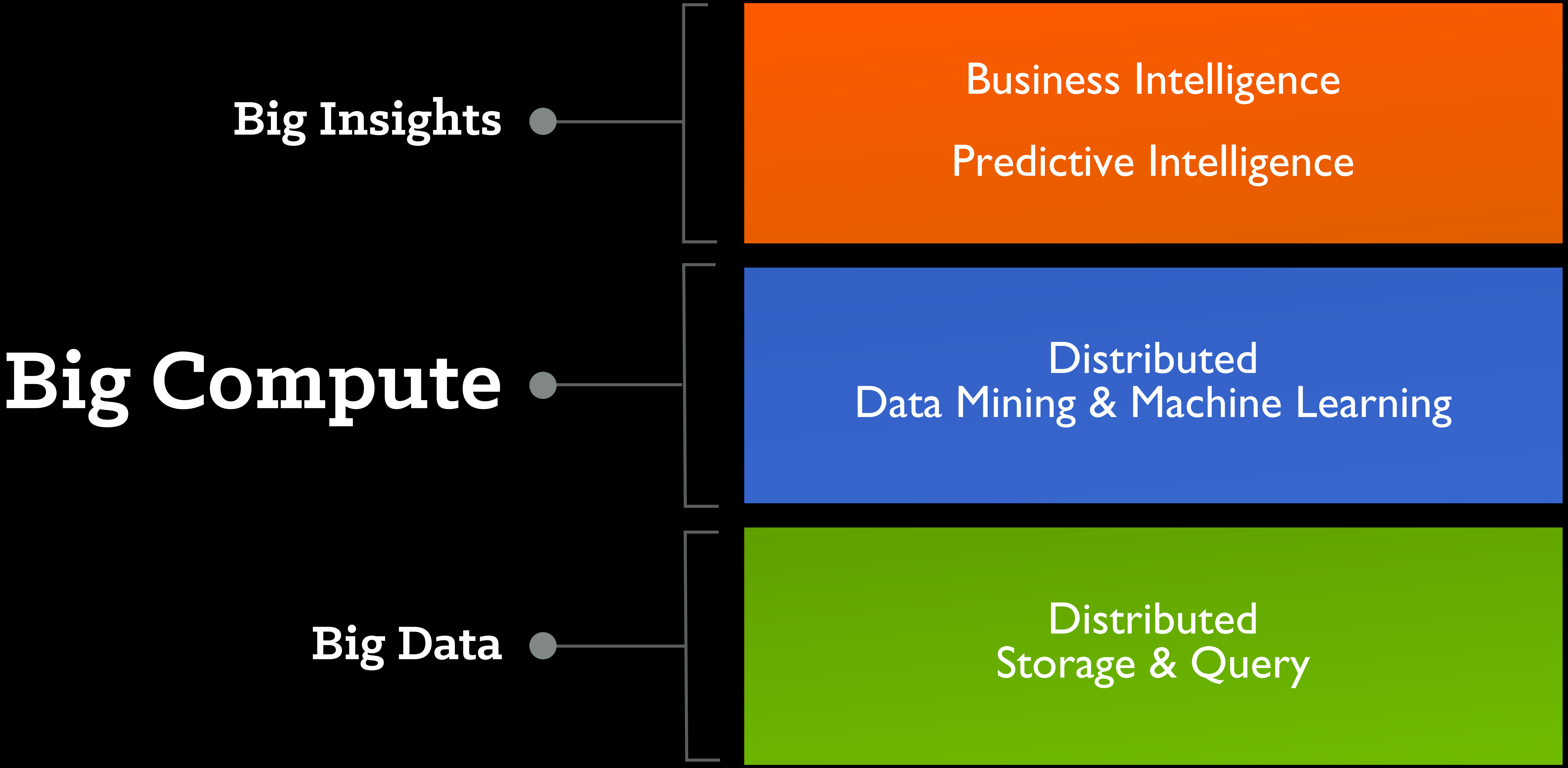
→

New Definition

✓ **BIG DATA + BIG COMPUTE**
=Opportunitie$

(Machine) Learn from Data

“Big Compute” Defined



Big Insights

Business Intelligence

Predictive Intelligence

Big Compute

Distributed
Data Mining & Machine Learning

Big Data

Distributed
Storage & Query

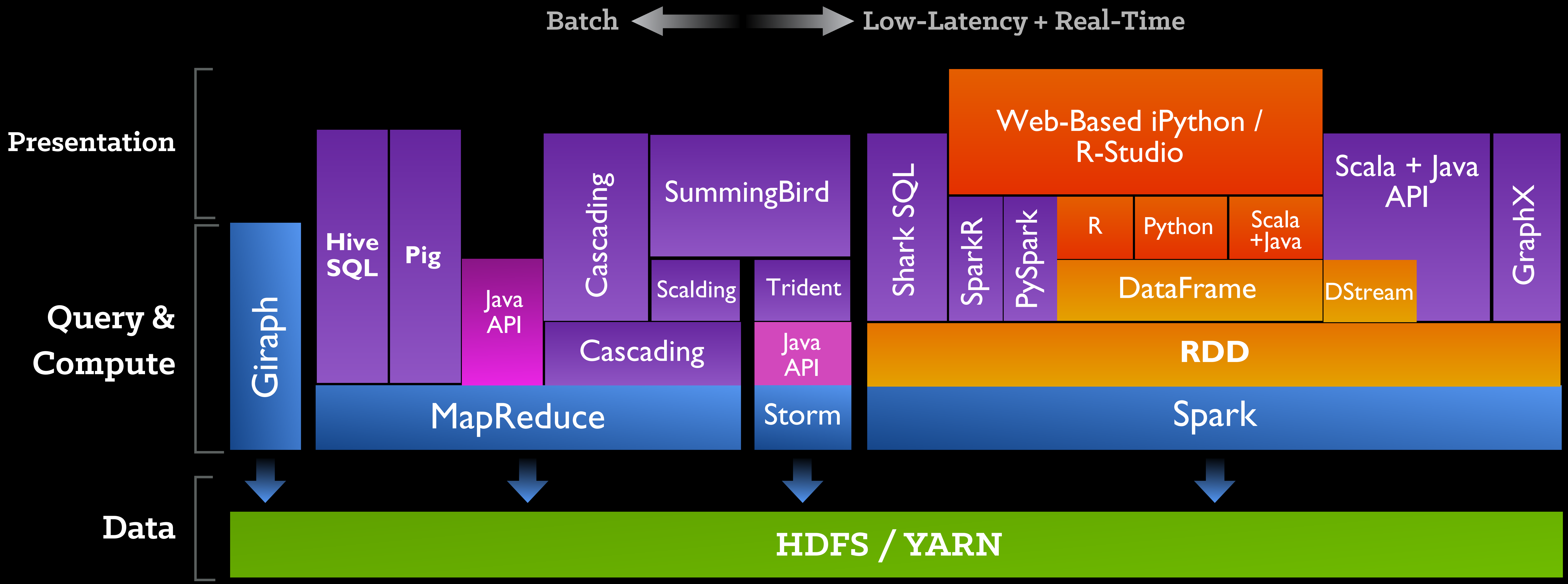
What's Been Missing
In the Big-Data Stack?

	Small-Data Stack	Big-Data Stack
Presentation	Visualization	???
Application	Business Applications	???
Storage	RDBMS	HDFS

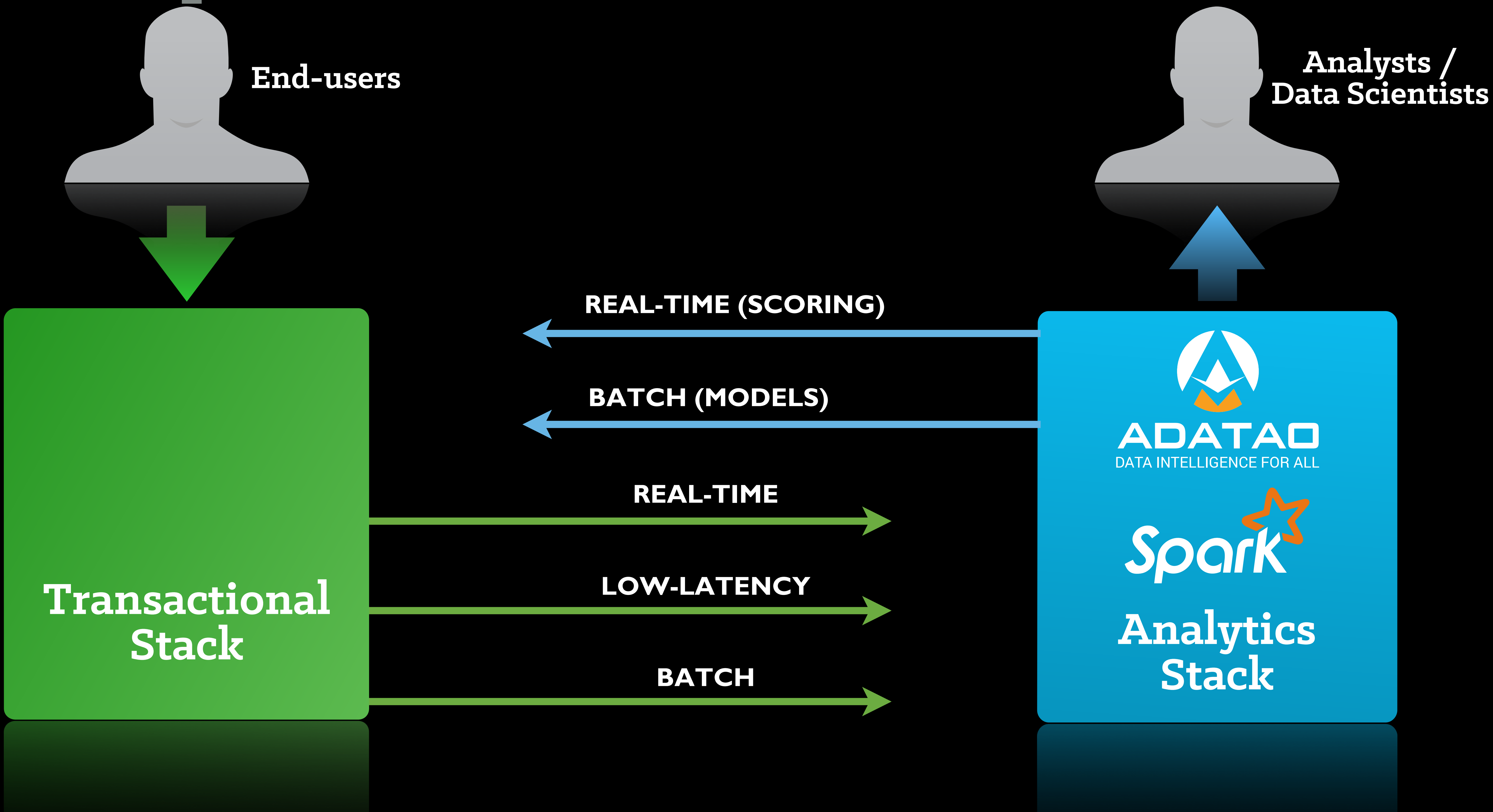
Alphabet Soup

Key

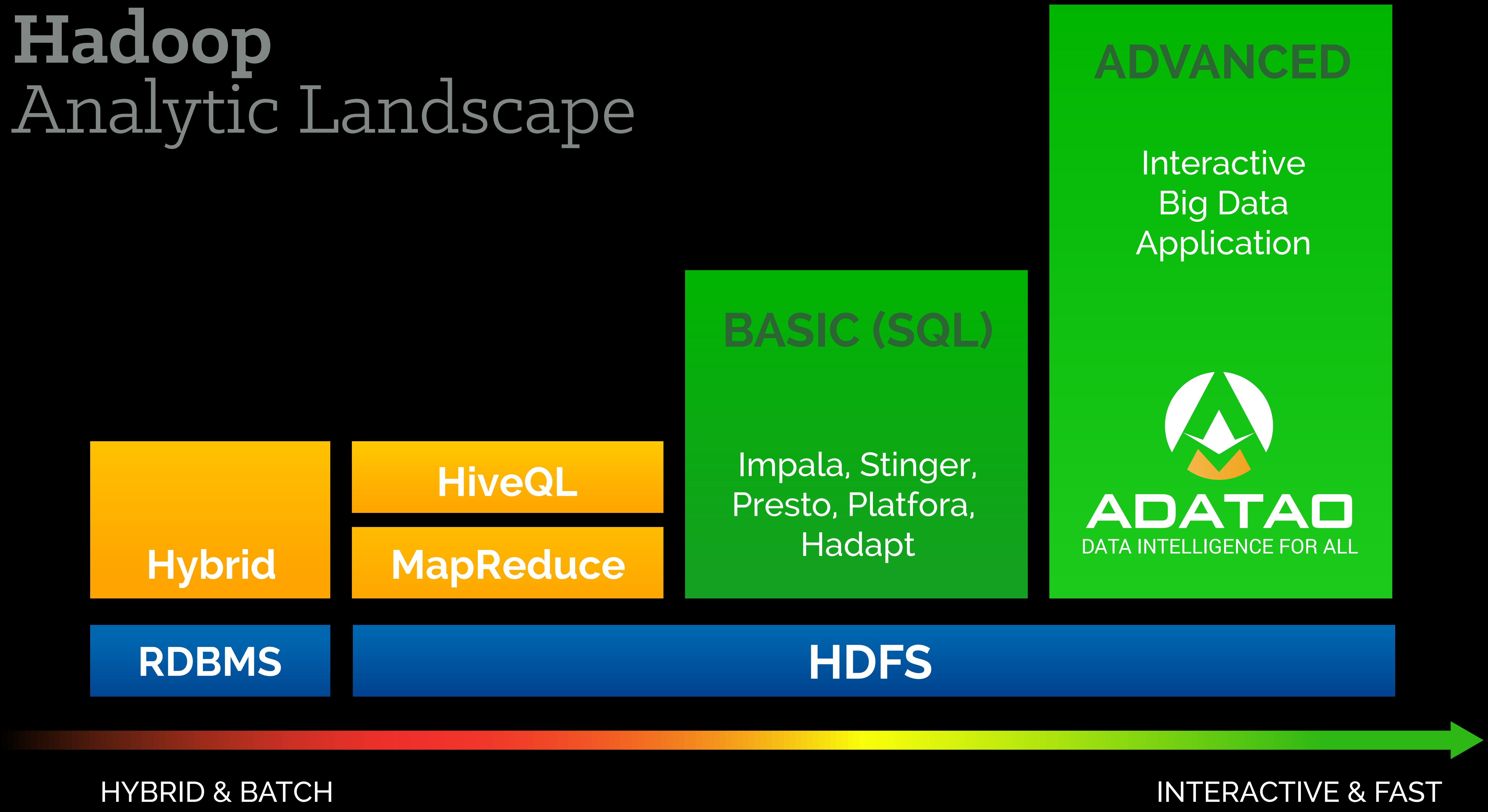
- 🟥 Proprietary (deep) language integration [Adatao]
- 🟧 Persistent In-Memory Data Structures
- 🟪 High-Level Primitives (group, cogroup, join…)
- 🟪 Low-Level Primitives (map, reduce, shuffle…)
- 🟦 Execution Engine



Key components in the Enterprise Data Flow



Hadoop
Analytic Landscape



ADVANCED

Interactive
Big Data
Application

ADATAO
DATA INTELLIGENCE FOR ALL

BASIC (SQL)

Impala, Stinger,
Presto, Platfora,
Hadapt

HiveQL

MapReduce

Hybrid

RDBMS

HDFS

HYBRID & BATCH

INTERACTIVE & FAST

ADATAO

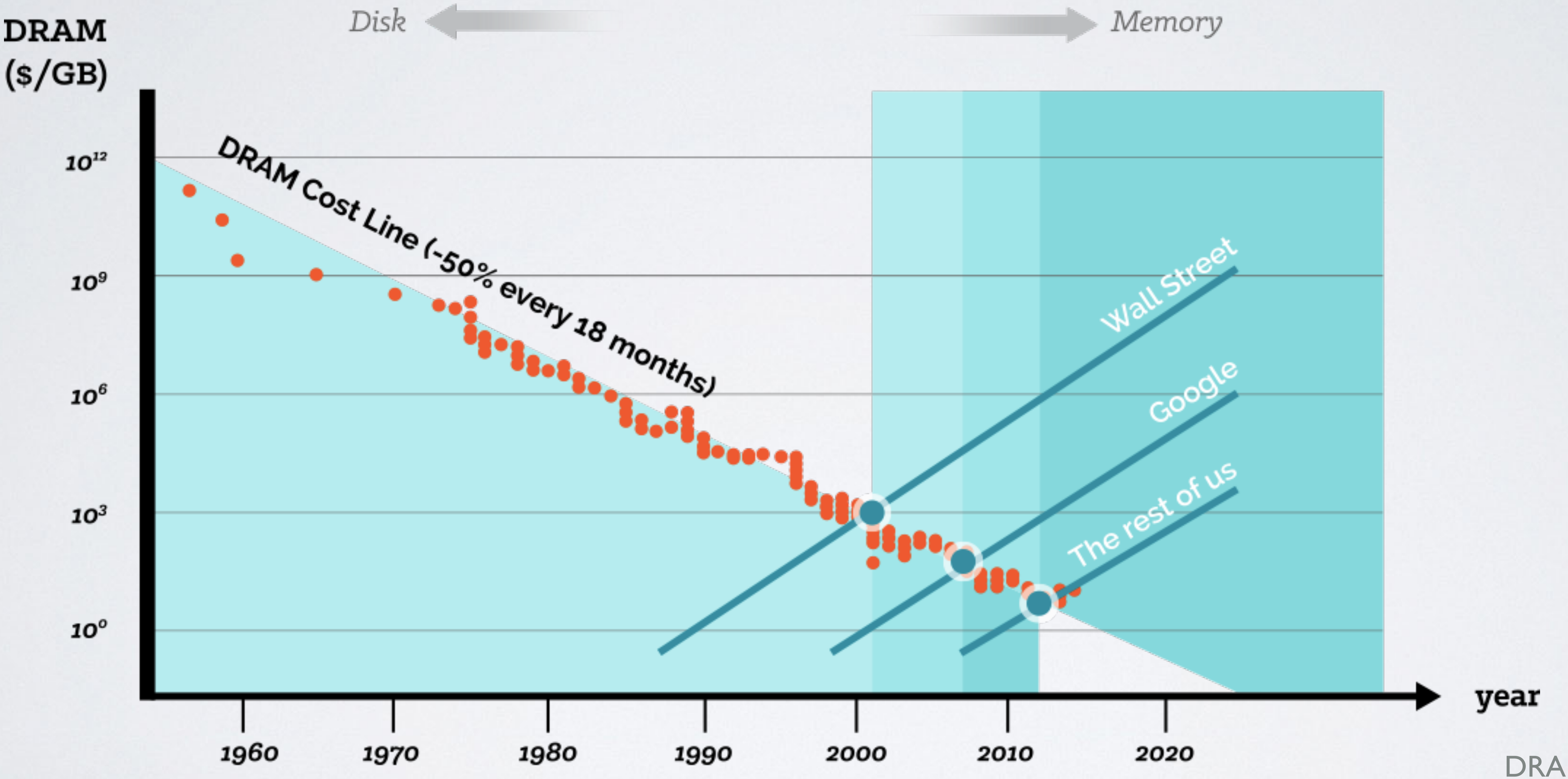
DATA INTELLIGENCE FOR ALL

In-Memory Big Compute

Why? When?

Big-Compute

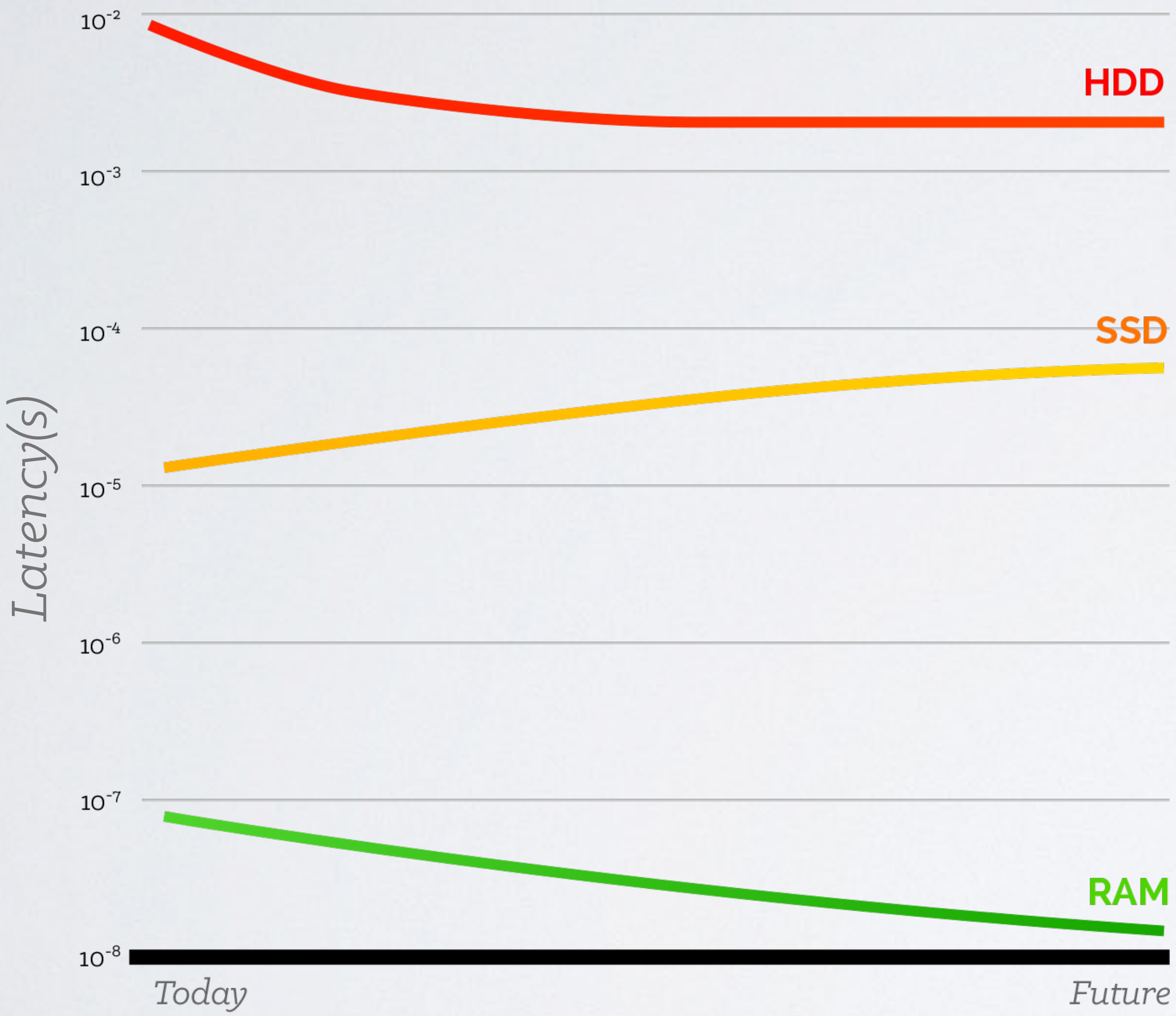
Value vs Cost Cross-Over Points



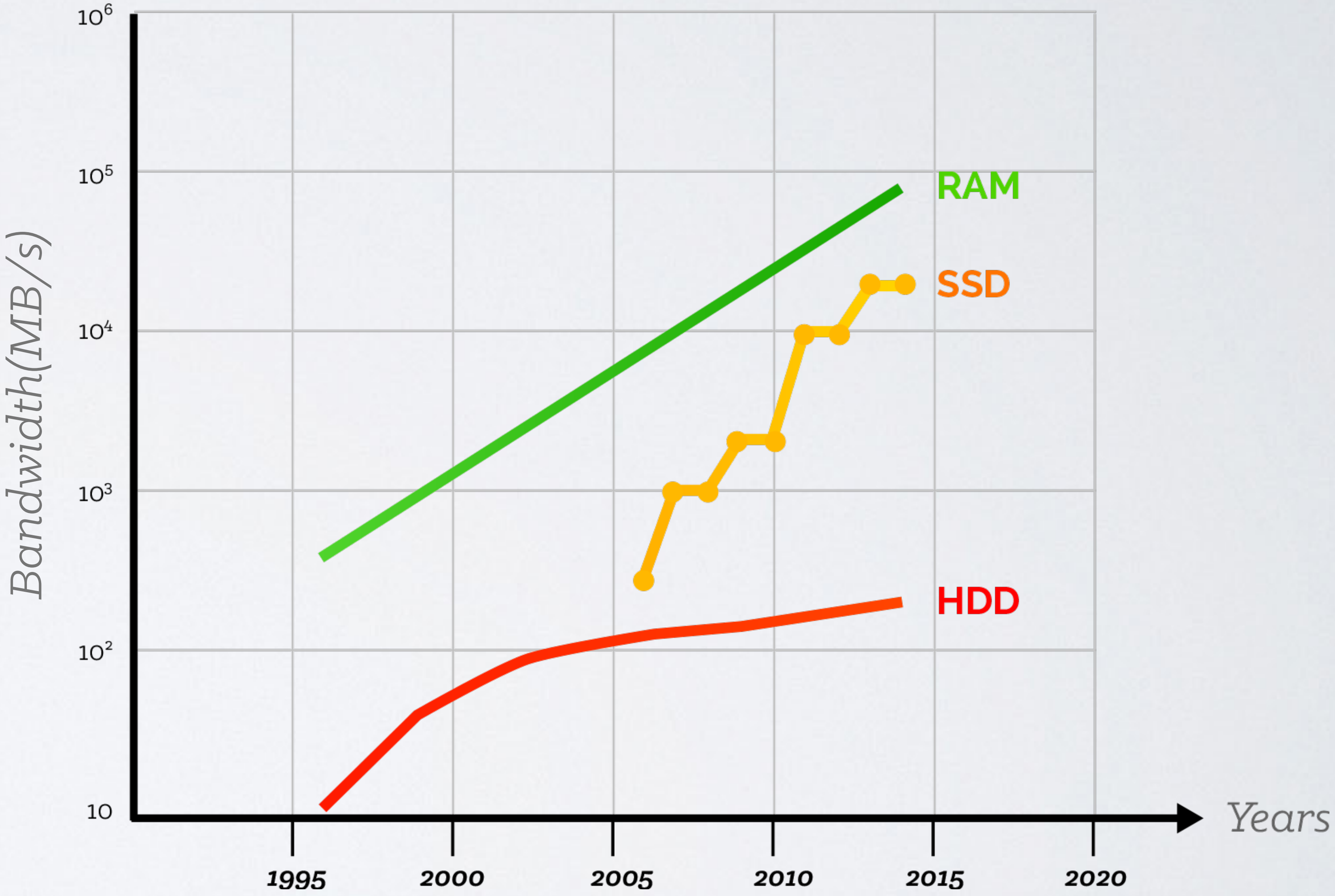
DRAM Source: jcmit.com

The Future Increasingly Favors RAM

Latency Trends



Bandwidth Trends



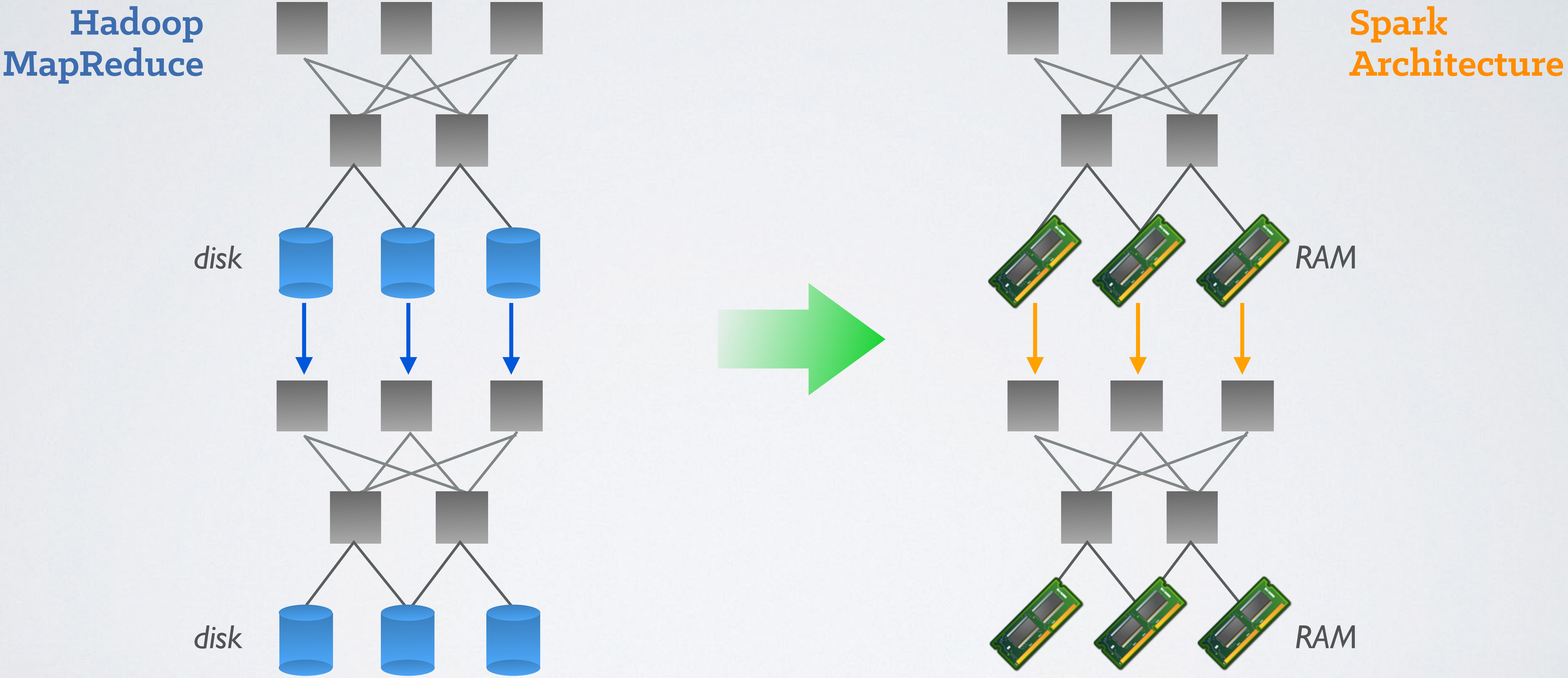
Unified Big Compute

ADATAO

DATA INTELLIGENCE FOR ALL

Spark

Comparison:
Hadoop MapReduce vs. Spark Architecture



Hadoop MapReduce

disk

disk

Spark Architecture

RAM

RAM

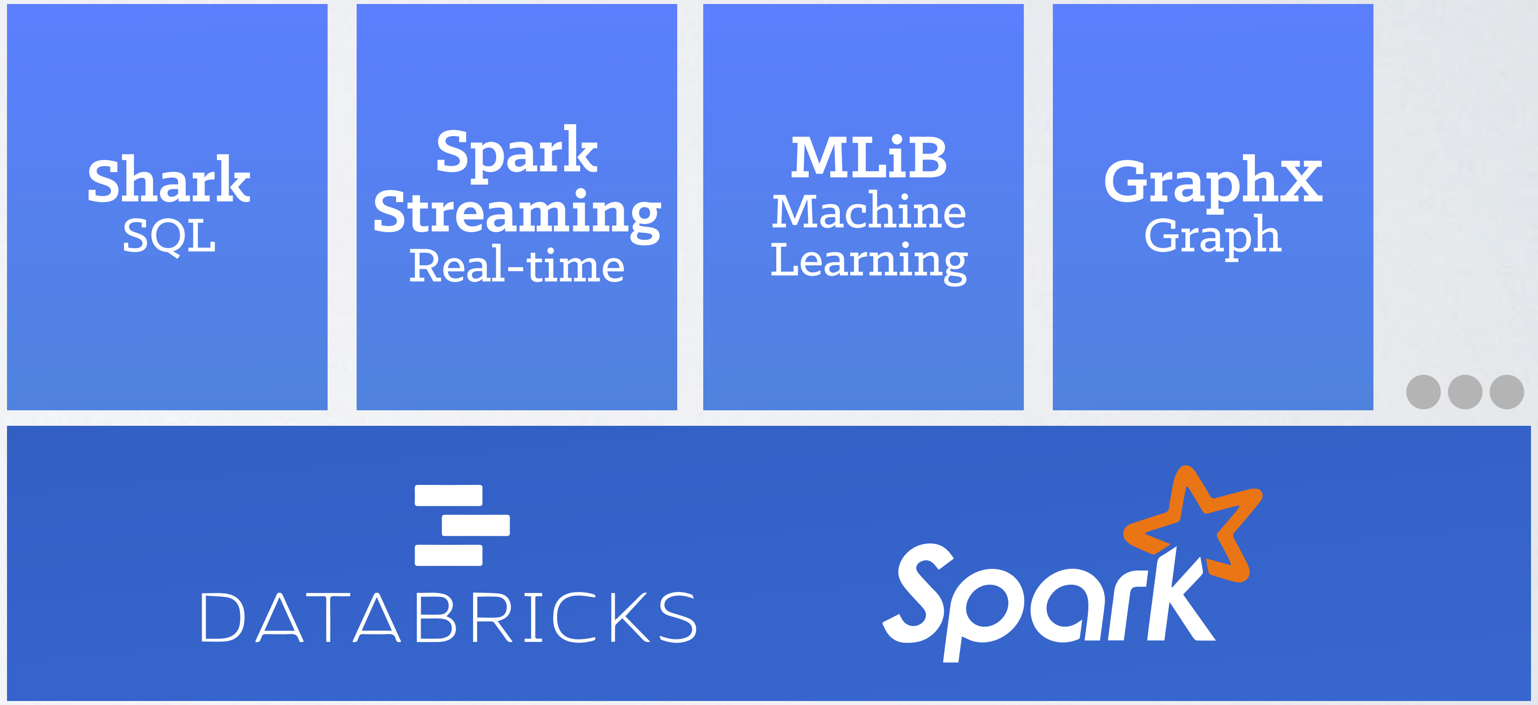
Apache Spark:
Big-Compute Engine

A Compute Engine for Hadoop Data that is:

Fast
Up to **100x** Faster than MapReduce

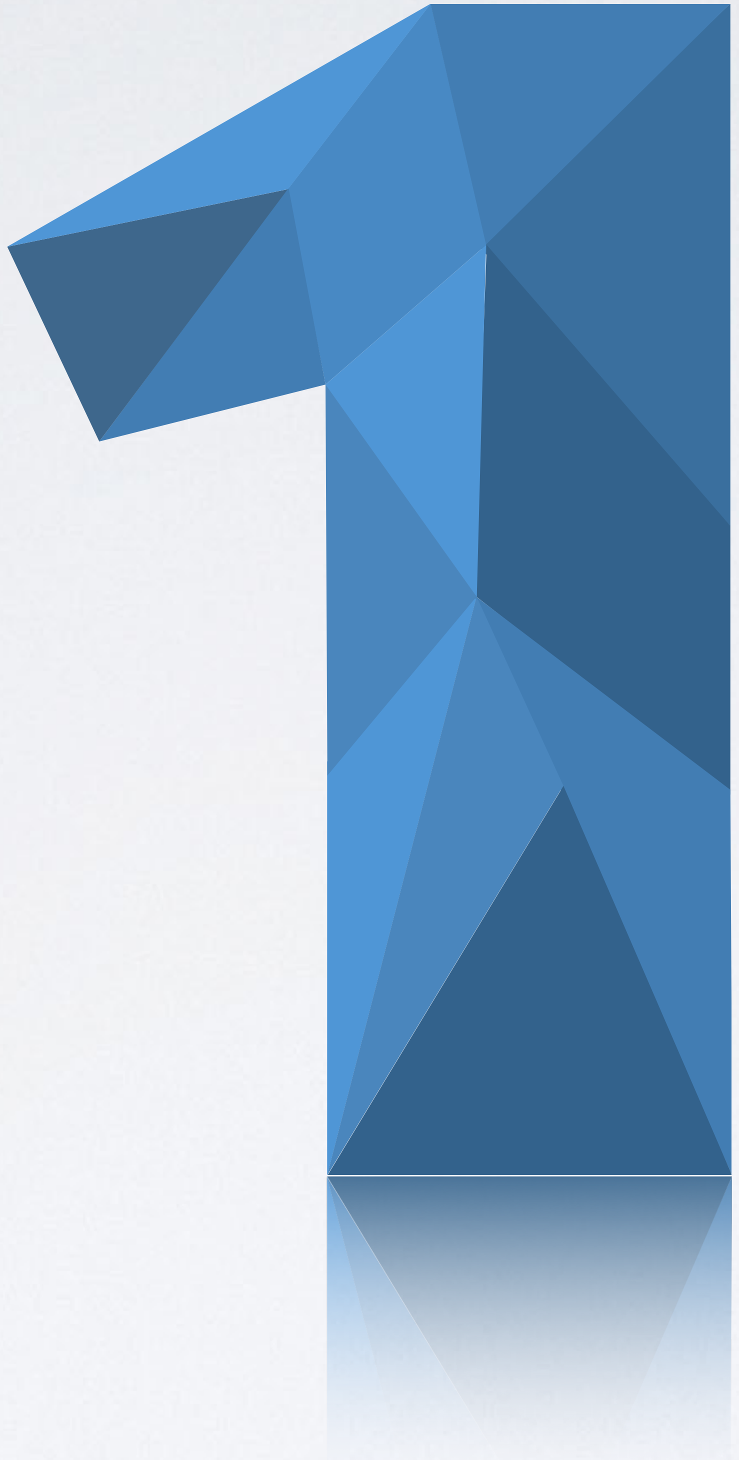
Sophisticated
Can run today’s **most advanced algorithms**

Fully Open Source
One of most active projects in Big Data



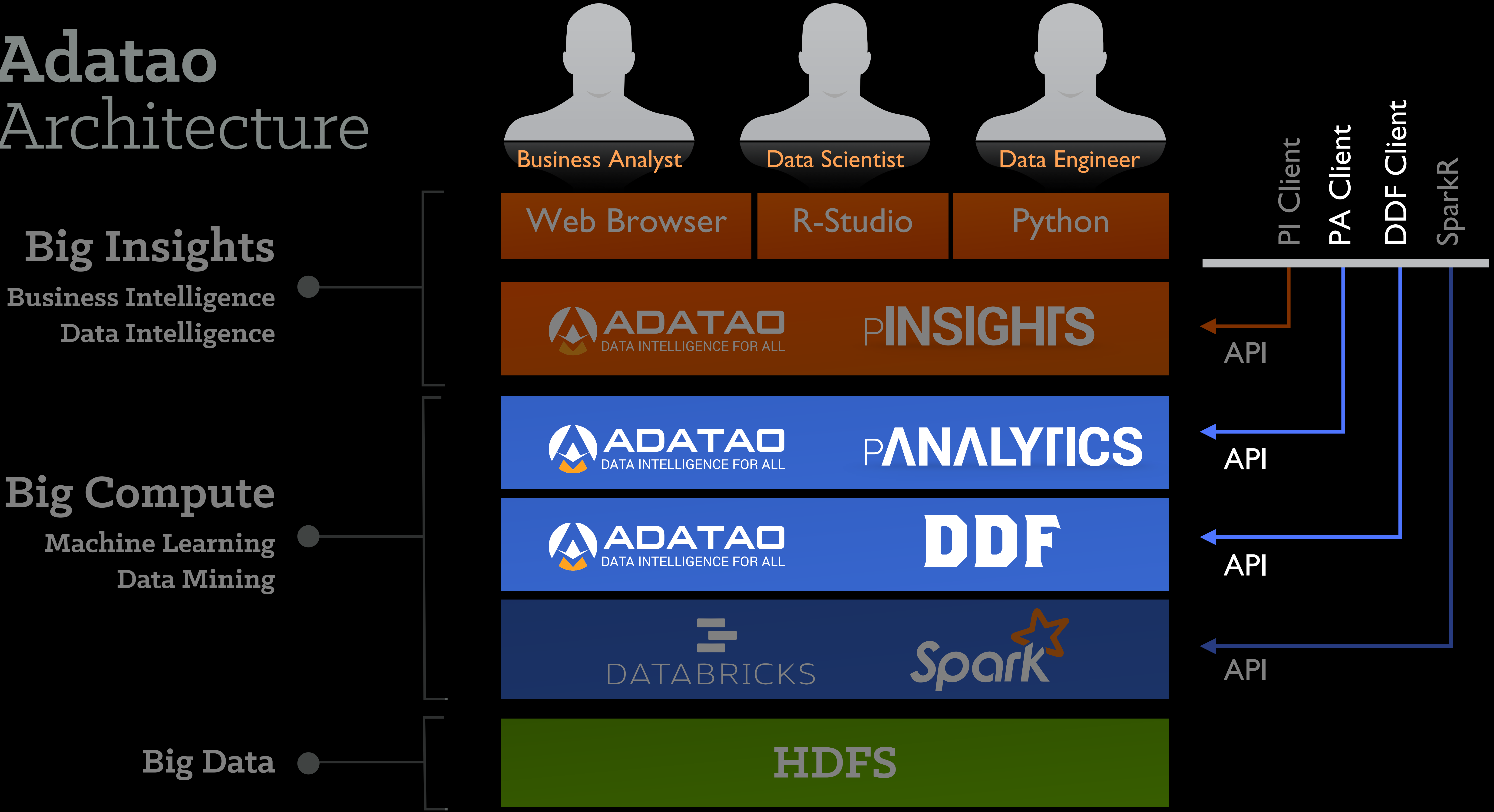
ADATAO

DATA INTELLIGENCE FOR ALL



Unified
Workbench *for*
Collaborative Data
Intelligence

Adatao Architecture



Big Insights

Business Intelligence
Data Intelligence

Big Compute

Machine Learning
Data Mining

Big Data

Business Analyst

Data Scientist

Data Engineer

Web Browser

R-Studio

Python

ADATAO DATA INTELLIGENCE FOR ALL pINSIGHTS

ADATAO DATA INTELLIGENCE FOR ALL pANALYTICS

ADATAO DATA INTELLIGENCE FOR ALL DDF

DATABRICKS Spark

HDFS

PI Client

PA Client

DDF Client

SparkR

API

API

API

API

RHadoop

```
library(rmr2)
library(rhdfs)
hdfs.init()
from.dfs(mapreduce(
  input = ’/tmp/airline.csv’,
  input.format = make.input.format(”csv”, sep = ”,”),
  map = function(., data) {
    # filter out non-numeric values (header and NA)
    filter = !is.na(data[,15])
    data = data[filter,]
    # emit composite key (airline|year|month) and delay
    keyval(
    data[,c(9,1,2)],
    data[,15, drop = FALSE])
  },
  reduce = function(k,delays) {
    keyval(k, mean(delays[,1]))
  }
))
```

SparkR

```
library(SparkR)
sc <- sparkR.init()
airlineRDD <- textFile(sc, ”/tmp/airline.csv”)

map.func <- function(line) {
  data <- unlist(strsplit(line, ”,”))
  if (data[15] != ”NA”) { list(data[c(9,1,2)],c(as.integer(data[15]), 1L)) }
}

avg.arrdelay <- lapply(
  reduceByKey(lapply(airlineRDD, map.func),”+”,2L),
  function(row) { list(row[[1]], row[[2]][1]/row[[2]][2]) }
)
```

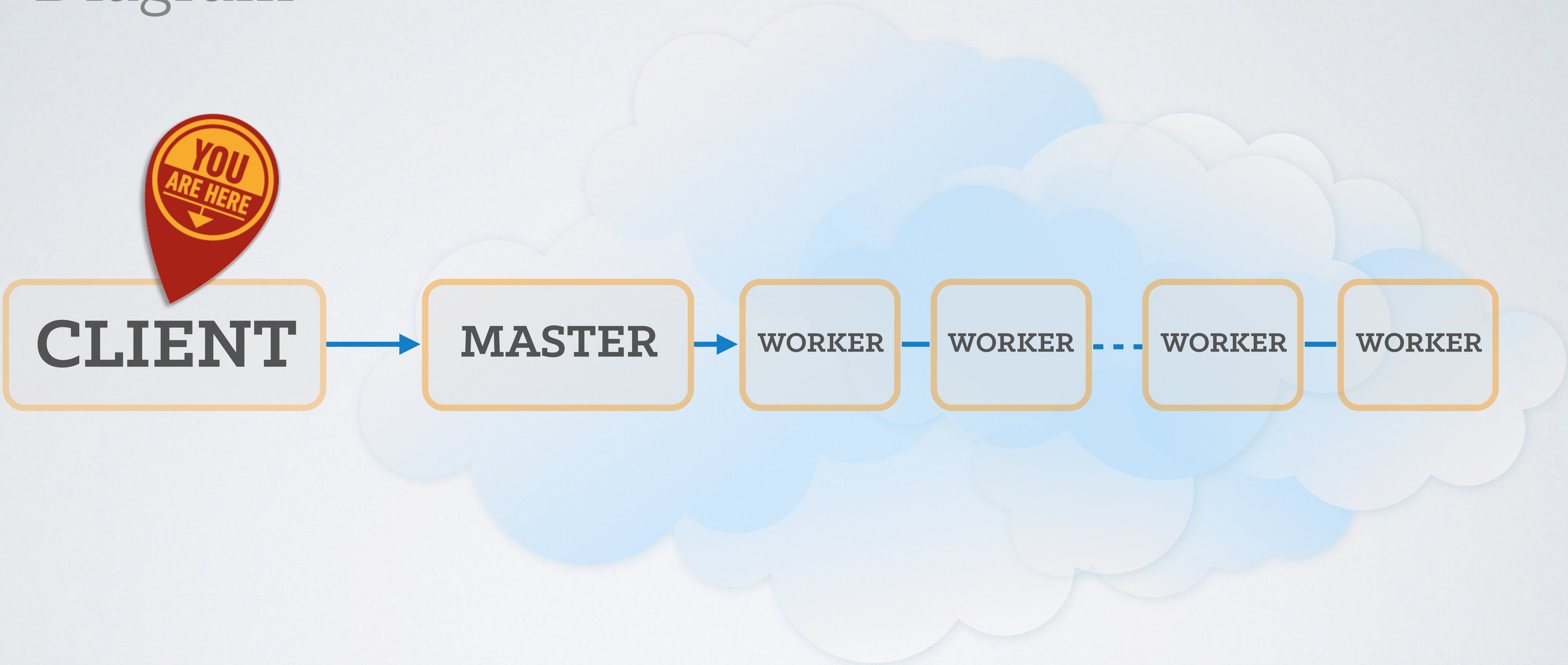
ADATAO
DATA INTELLIGENCE FOR ALL

```
df <- adatao.sql2ddf(‘select * from airline’)
avg.arrdelay <- adatao. aggregate (arrdelay ~ uniquecarrier + year + month, df, FUN=mean)
```

Feature Comparison

	RHadoop	SparkR	Adatao
Support Hive Tables	✗	✗	✓
Support HDFS	✓	✓	✓
Ability to Write MapReduce in R	✓	✓	✓
Native R Idioms	✗	✗	✓
DataFrame Abstraction	✗	✗	✓
Data Extraction	✗	✗	✓
Data Transformation	Raw	Raw	Idiomatic
Data Exploration	✗	✗	✓
Speed	✗	✓	✓✓

Demo Deployment
Diagram



Adatao Benefits

☑ Stop Moving Data Around
Data Science Directly on Hadoop Datasets

☑ Focus on Analysis, not MapReduce
High-Level Programmable API (DDF)

☑ Model Terabytes in Seconds
Powerful, Fast, Interactive Data Science

☑ Native R Data.frame Experience
Table-like Abstraction on Top of Big Data

☑ Zero-Effort Model Deployment
Transactional & Analytic Support in One Stack

☑ Easily Visualize & Collaborate
Beautiful Charting, Dashboarding & RT Collaboration

ADATAO
DATA INTELLIGENCE FOR ALL

Adatao Demo

ADATAO

DATA INTELLIGENCE FOR ALL

To learn more about
Adatao & DDF
contact us, or come to our
Spark Summit talk

www.adatao.com