amplab
UC BERKELEY

Berkeley Data Analytics Stack: Experience and Lesson Learned

Ion Stoica

UC Berkeley, Databricks, Conviva

Research Philosophy

Follow real problems

Focus on novel usage scenarios

Build real systems
- » Be paranoid about simplicity
- » Very hard to build complex systems in academia

Push for adoption
- » Develop communities
- » Train users

Disclaimer: By no means only way to do research!

A Short History

2006: Start research in cluster computing

- Improve MapReduce scheduler (e.g., Fair Scheduler)

2009: Start building a Data Analytics Stack

- Spring 2009: Mesos
- Summer 2009: Spark
- 2010: Shark
- 2011: SparkStreaming
- 2012: Tachyon
- 2013: MLlib,
- …

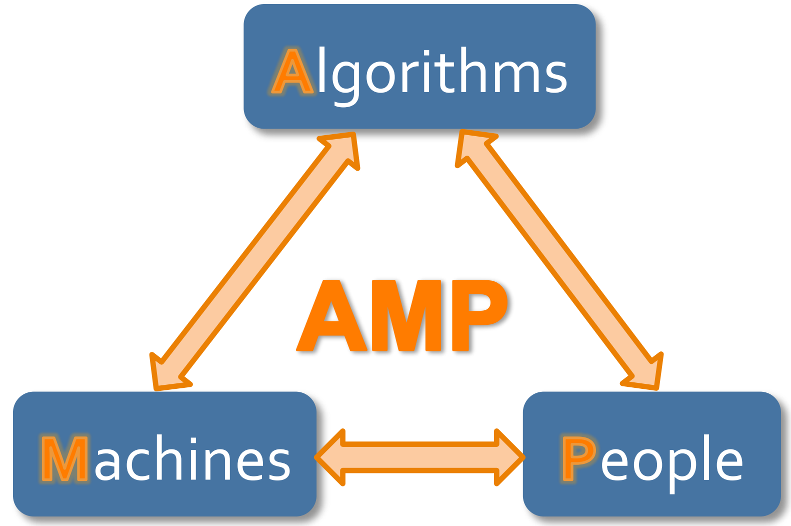
RAD Lab

2005-2010

amplab UC BERKELEY

2011-2017

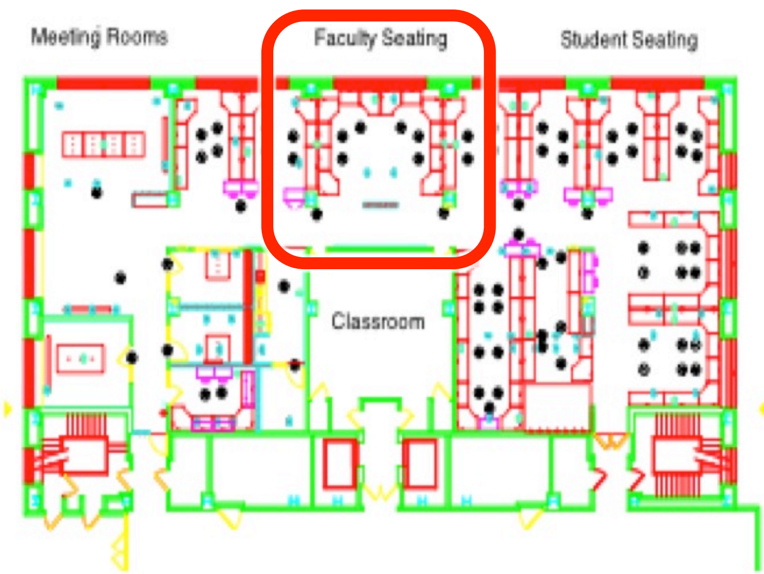
The Berkeley AMPLab



January 2011 – 2017

» 8 faculty
» > 60 students
» 3 software engineer team

Organized for collaboration



3 day retreats
(twice a year)

AMPCamp3
(November, 2014)

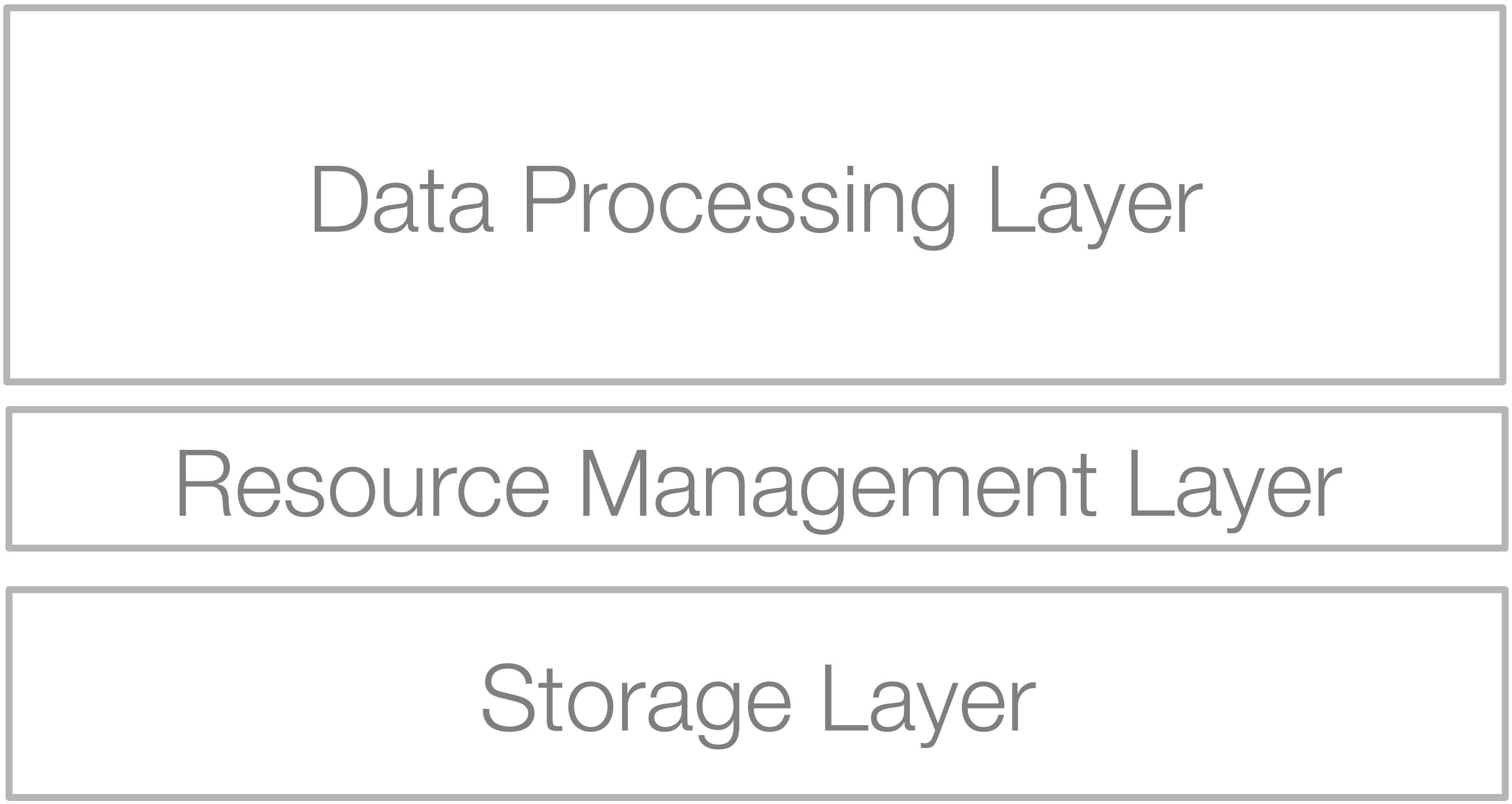
220 campers
(100+ companies)

The Berkeley AMPLab

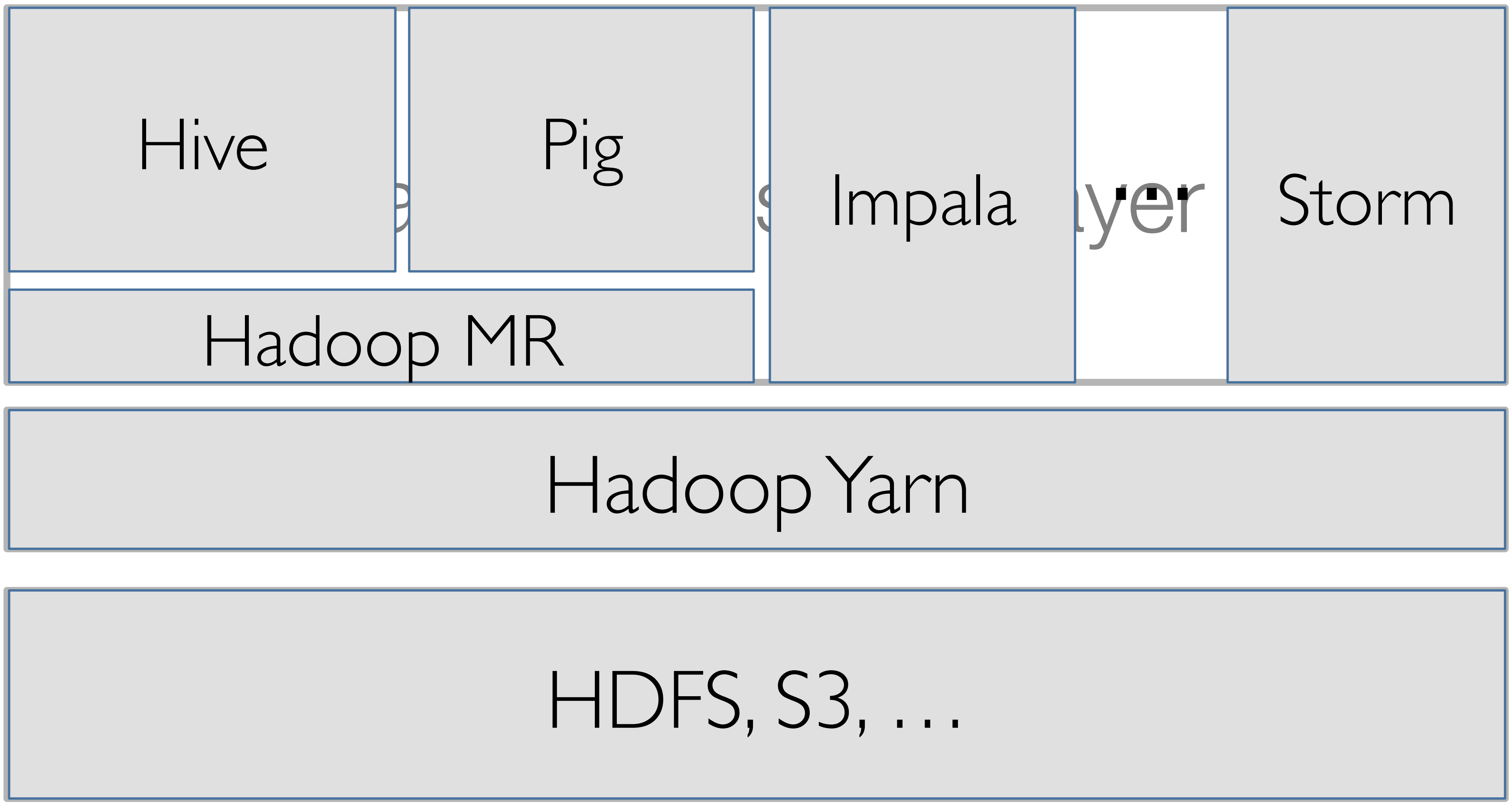
Governmental and industrial funding:

Goal: Next generation of open source data analytics stack for industry & academia:
Berkeley Data Analytics Stack (BDAS)

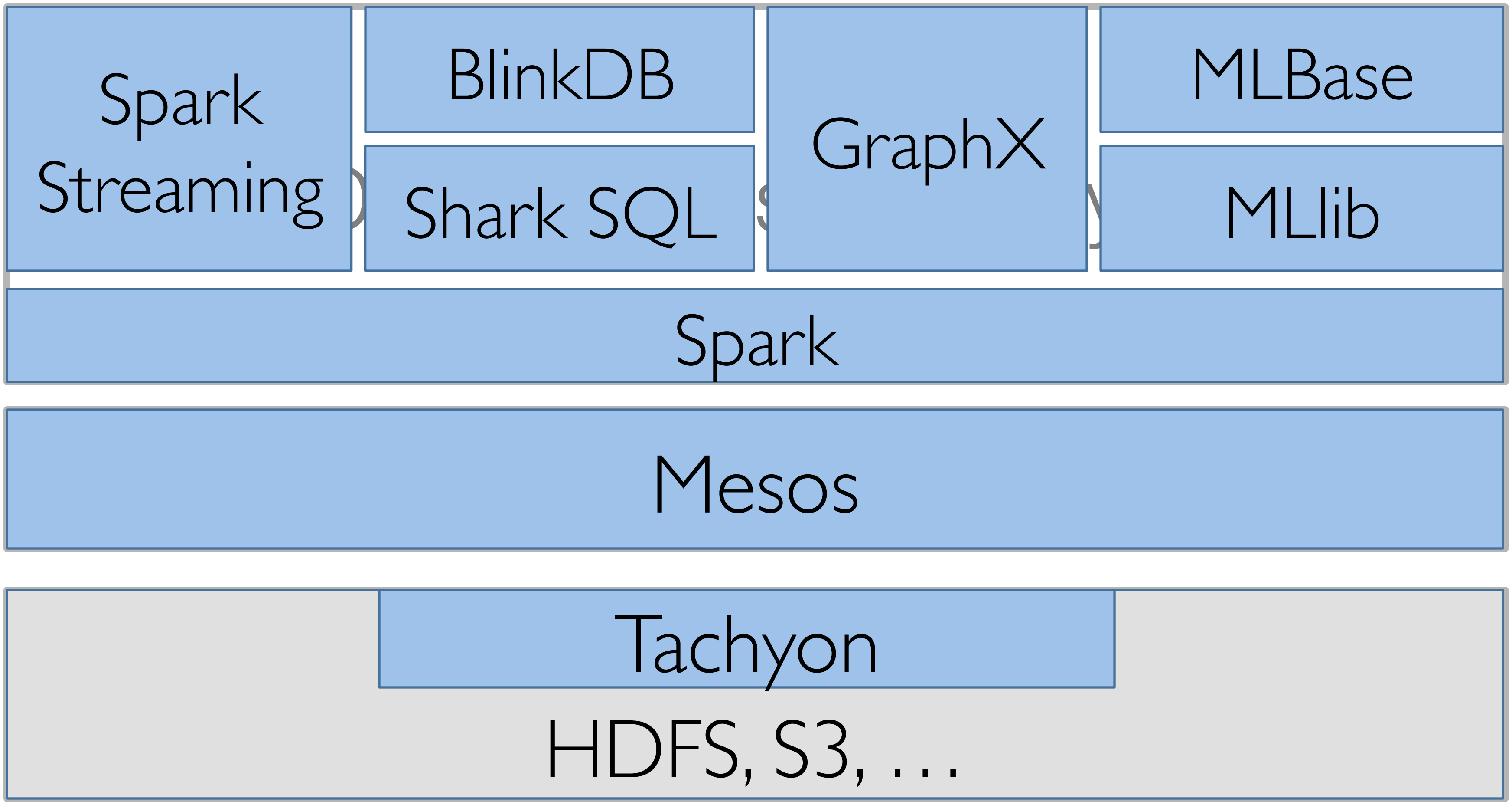
Data Processing Stack



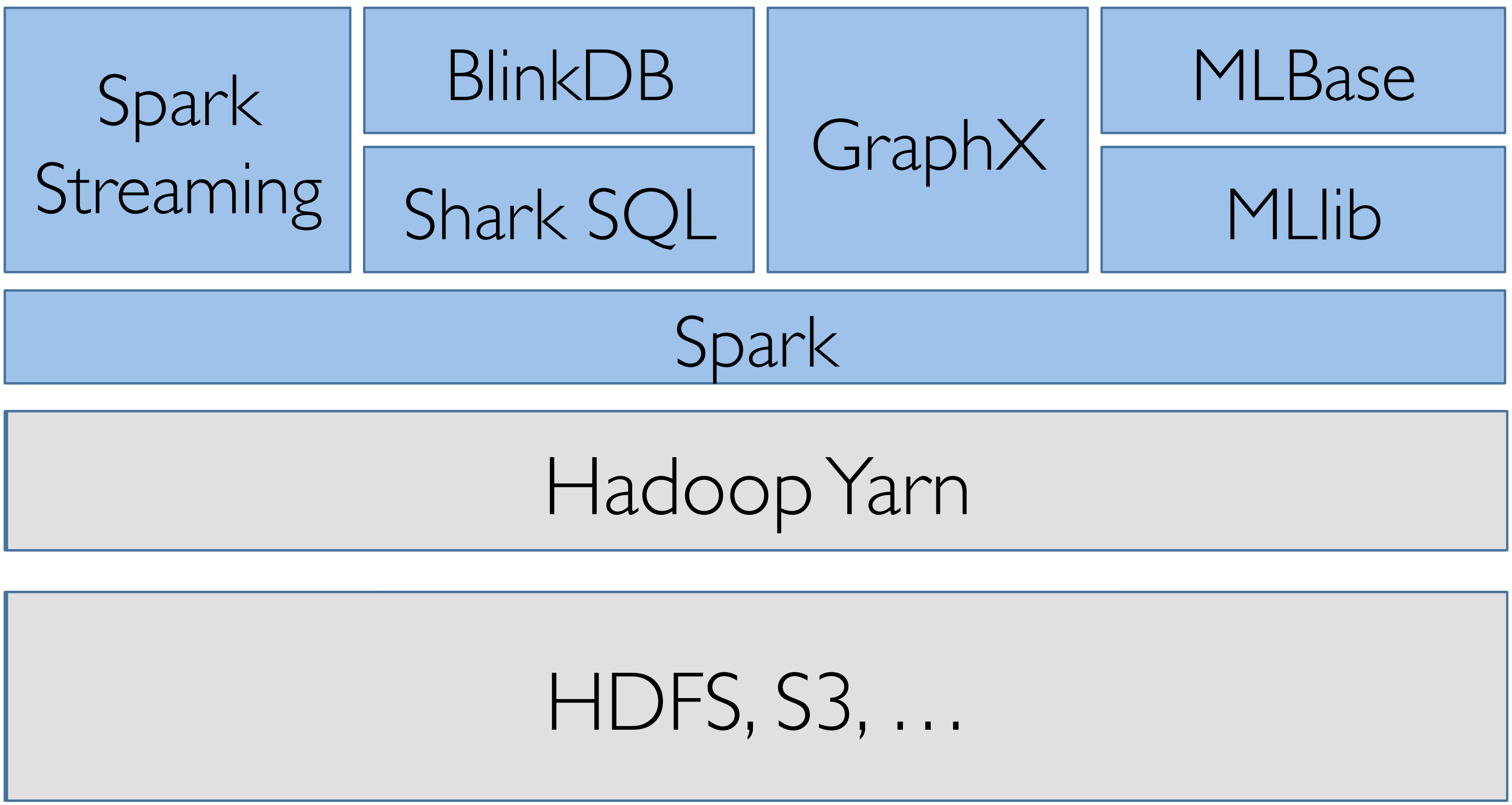
Hadoop Stack



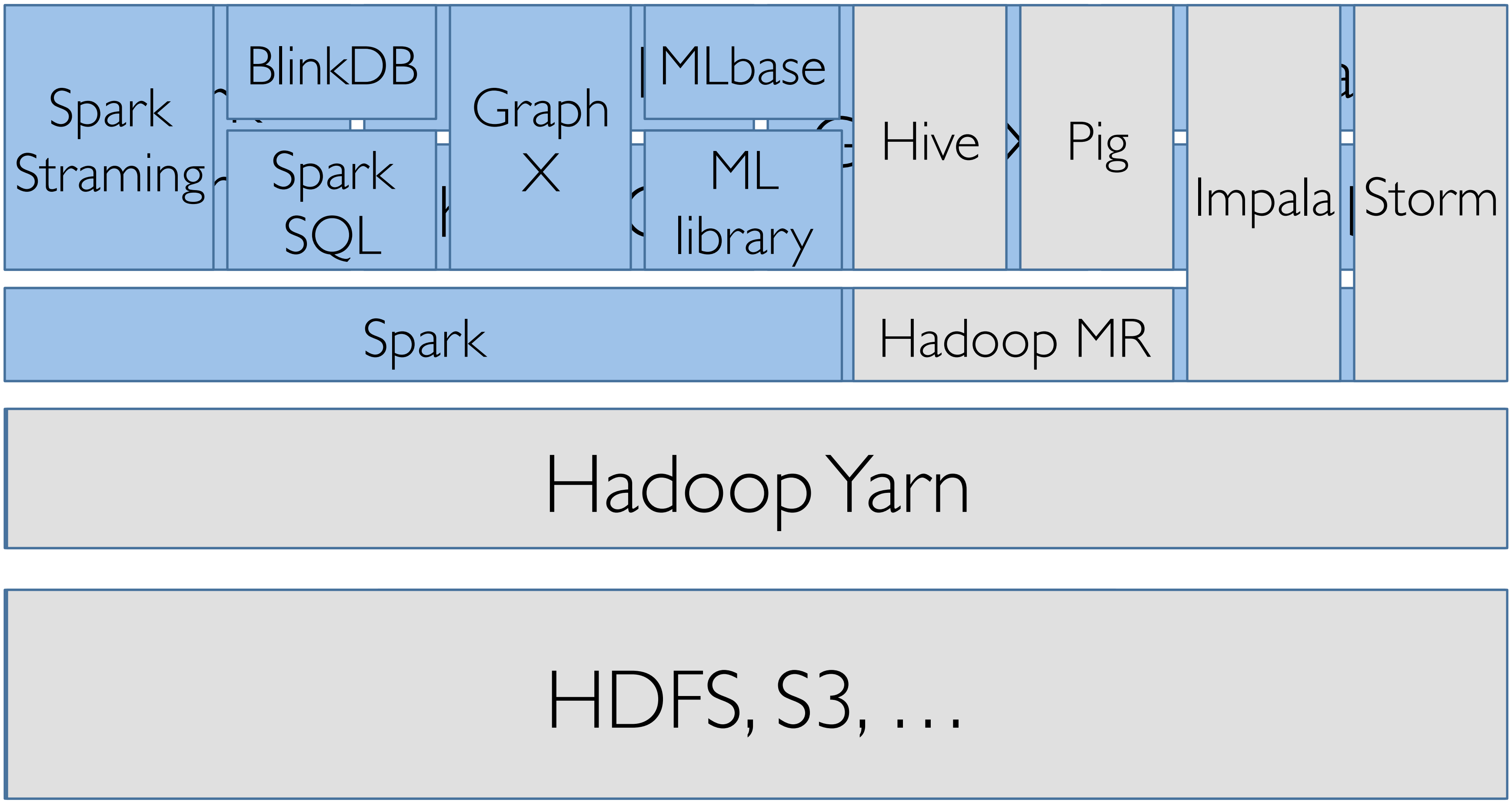
BDAS Stack


Spark Streaming
BlinkDB
Shark SQL
GraphX
MLBase
MLlib
Spark
Mesos
Tachyon
HDFS, S3, …

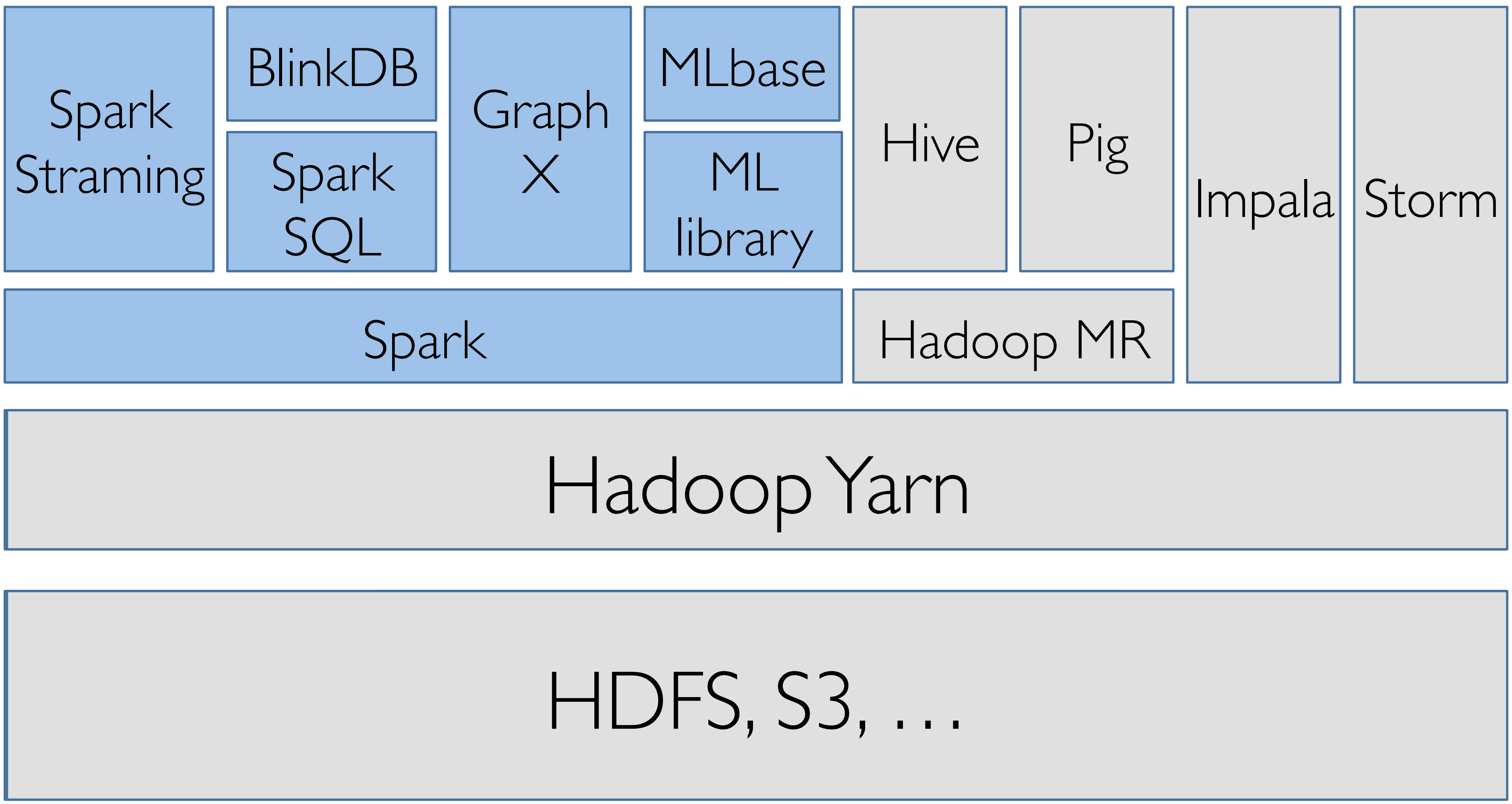
How do BDAS & Hadoop fit together?



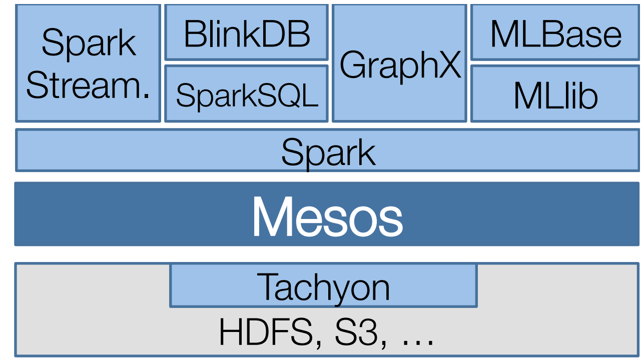
How do BDAS & Hadoop fit together?



How do BDAS & Hadoop fit together?



Apache Mesos

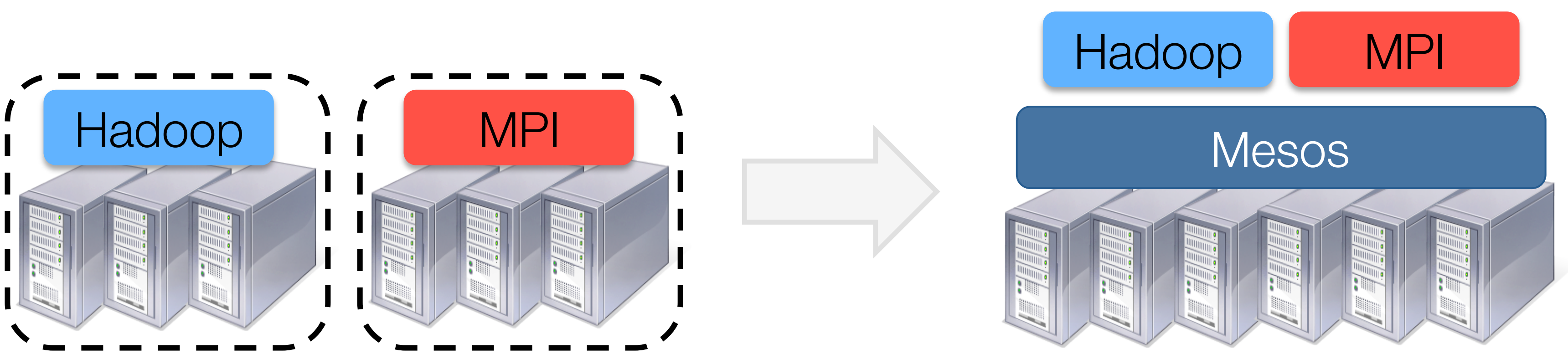


Problem: per-framework cluster

» Inefficient resource usage
» Hard to experiment, upgrade
» Hard to share data

Solution: common resource sharing layer

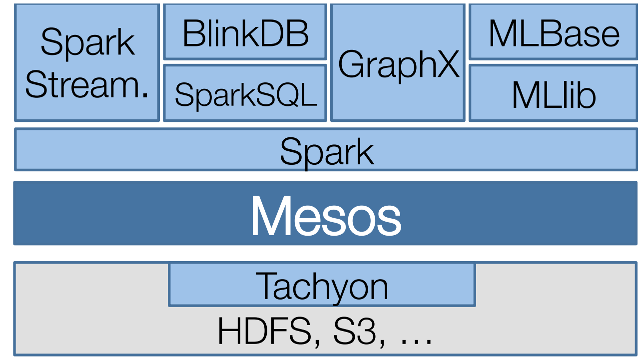
» Abstracts ("virtualizes") resources to frameworks
» Enable diverse frameworks to share cluster



Uniprograming

Multiprograming

Apache Mesos

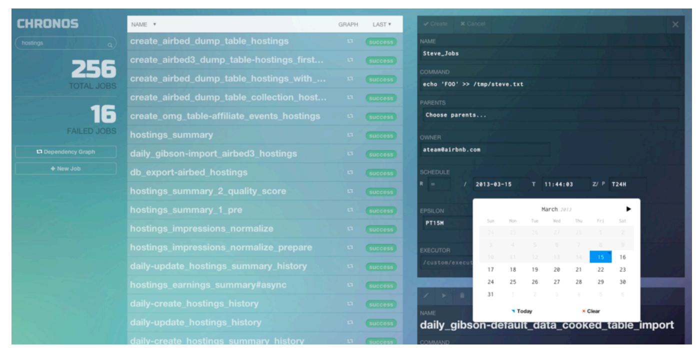


Open Source: 2010 (first release: 10,000 LoC)

Apache Project: 2012

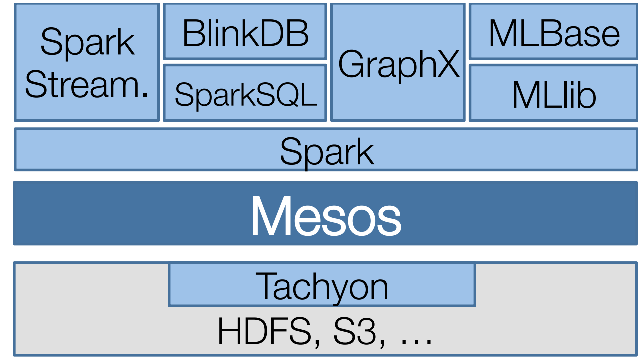
Used in production at Twitter for past 2.5 years
- » +10,000 machines
- » +500 engineers using it

Third party Mesos schedulers
- » AirBnB's Chronos
- » Twitter's Aurora



Mesospehere: startup to commercialize Mesos

Mesos Meetups



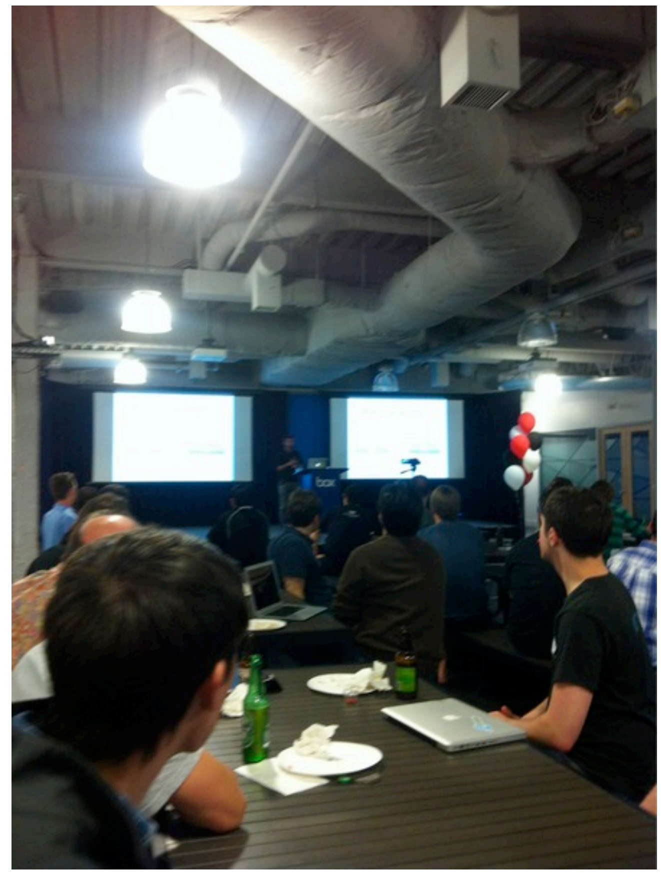
Sept 2012: started Bay Area Spark Meetup

» Now +800 members

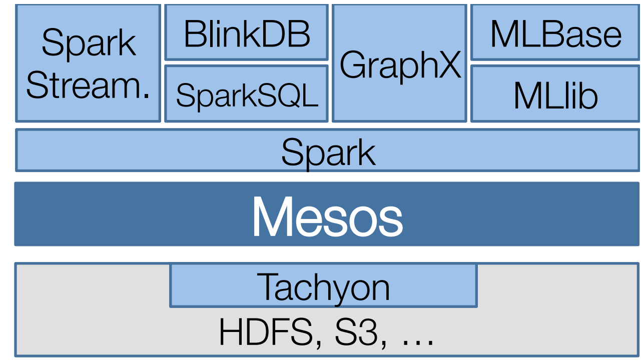
Other user groups:

» +700 members

» New York, Atlanta, Seattle, Los Angeles, Paris (France), Amsterdam (Netherlands), London (UK)

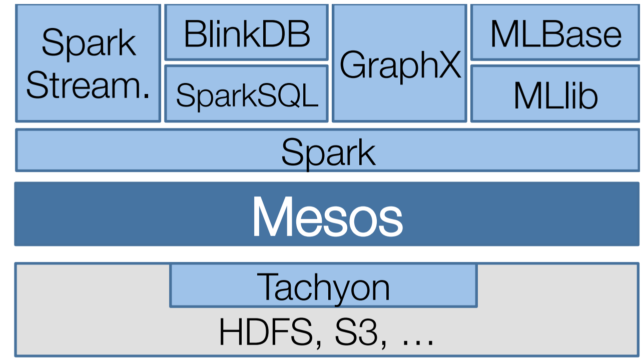


Monthly Contributors

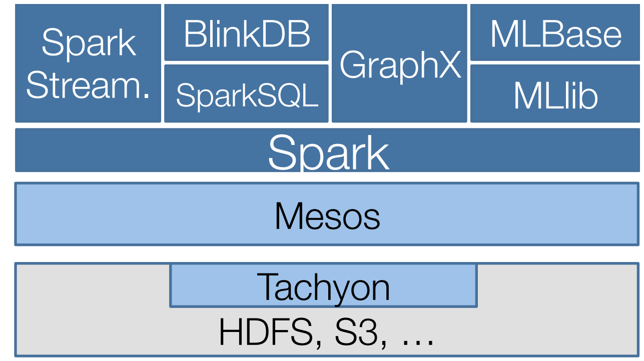


65 contributors for last 12 months

Selected Users



Apache Spark



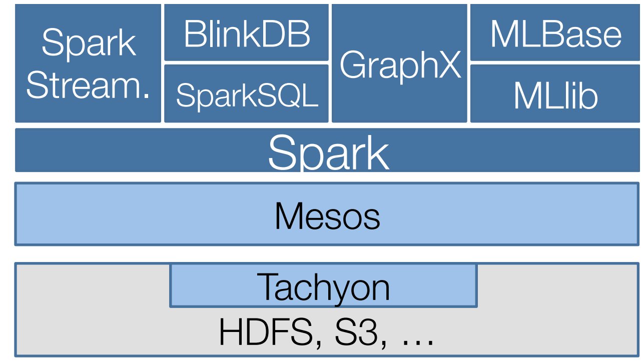
Problem: Need to support workloads beyond batch (MapReduce)

» Interactive, streaming, iterative (ML), graph processing

Motivating use cases:

» Iterative computations (ML researchers in RADLab)
» Interactive queries (Conviva, Facebook)

Apache Spark



Distributed Execution Engine

- » Fault-tolerant, efficient in-memory storage
- » Low-latency large-scale task scheduler
- » Powerful prog. model and APIs: Python, Java, Scala

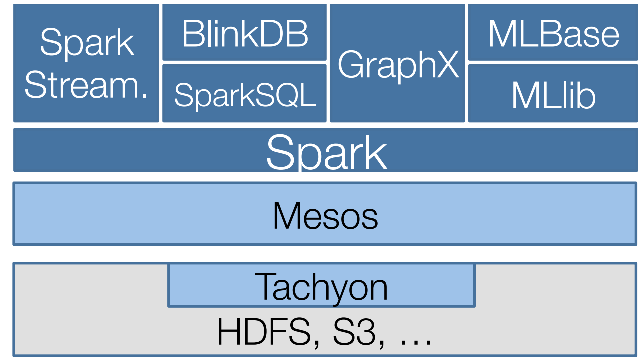
Fast: up to 100x faster than Hadoop MR

- » Can run sub-second jobs on hundreds of nodes

Easy to use: 2-5x less code than Hadoop MR

General: support interactive & iterative apps

Apache Spark



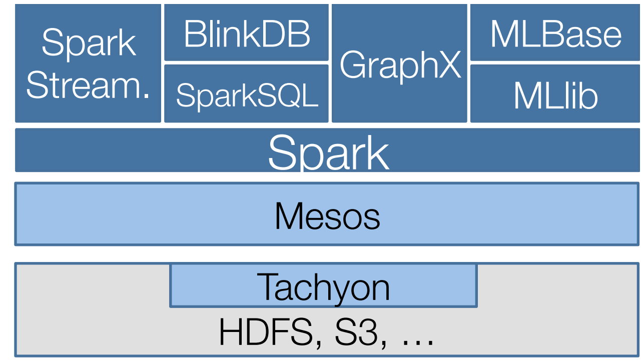
Open Source: end of 2010 (<3,000 LoC, Scala)

Apache Project: 2013

Over time has grown to include key components (*everyone being motivated by Spark use in prod.*)

» Shark (2010) → SparkSLQ (2014)
» SparkStreaming (2011)
» MLlib (2013)
» GraphX (2014)

On its way to become the platform of choice for developing Big Data applications



Spark Meetups

Jan 2012: started Bay Area Spark Meetup

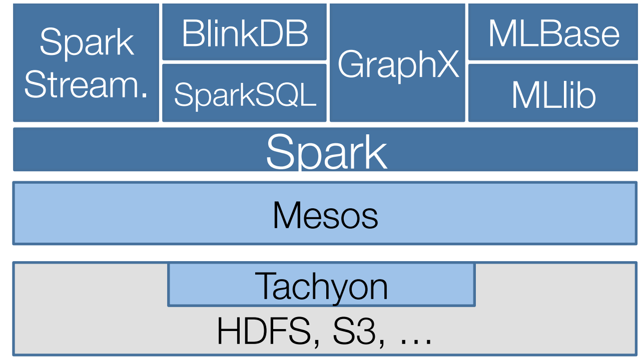
» +3100 members

Now

» 33 cities

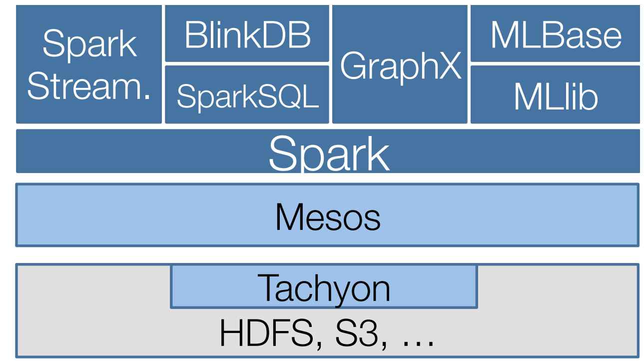
» 13 countries

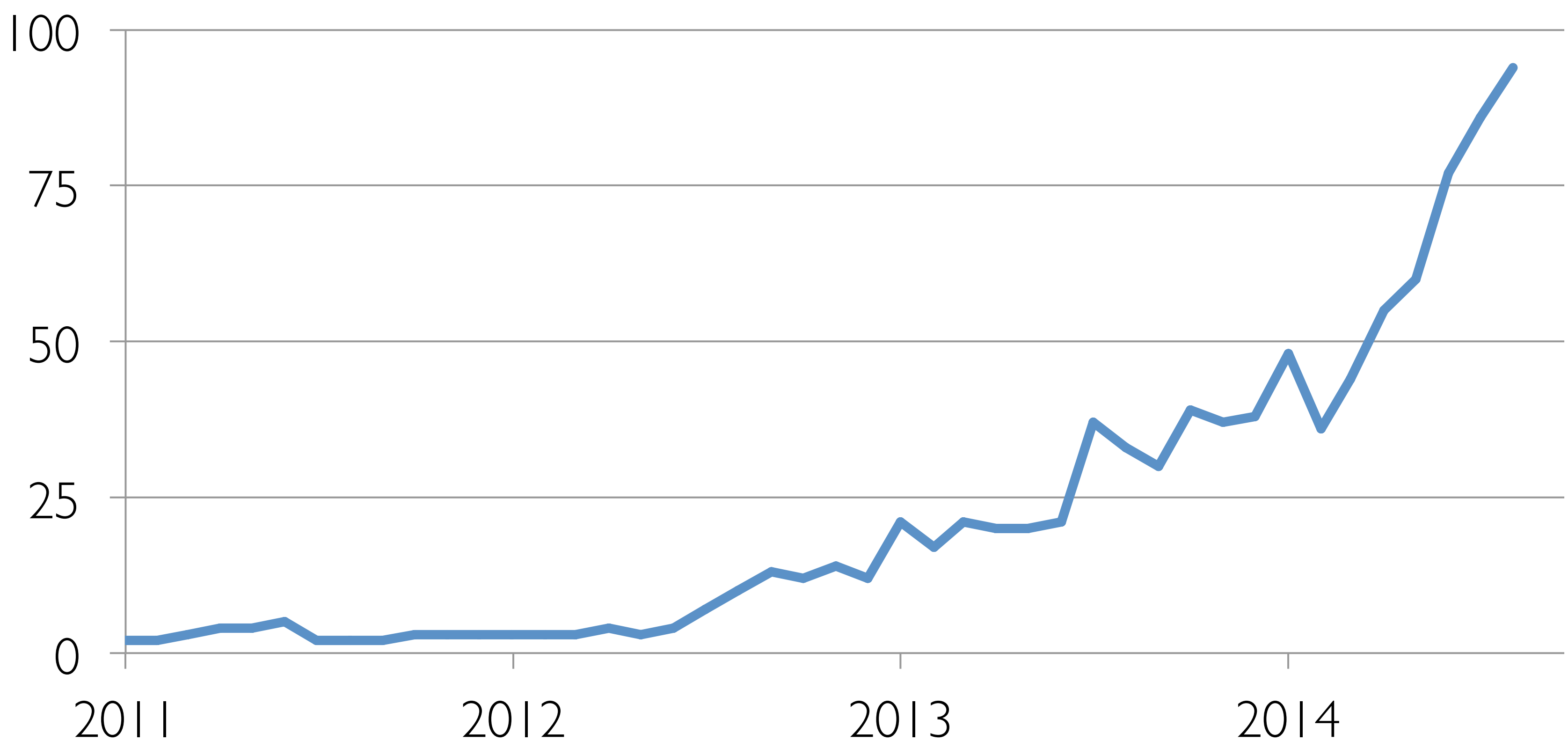
Meetups Around the World



Groups	Members	Interested	Cities	Countries
40	10,900	795	33	13

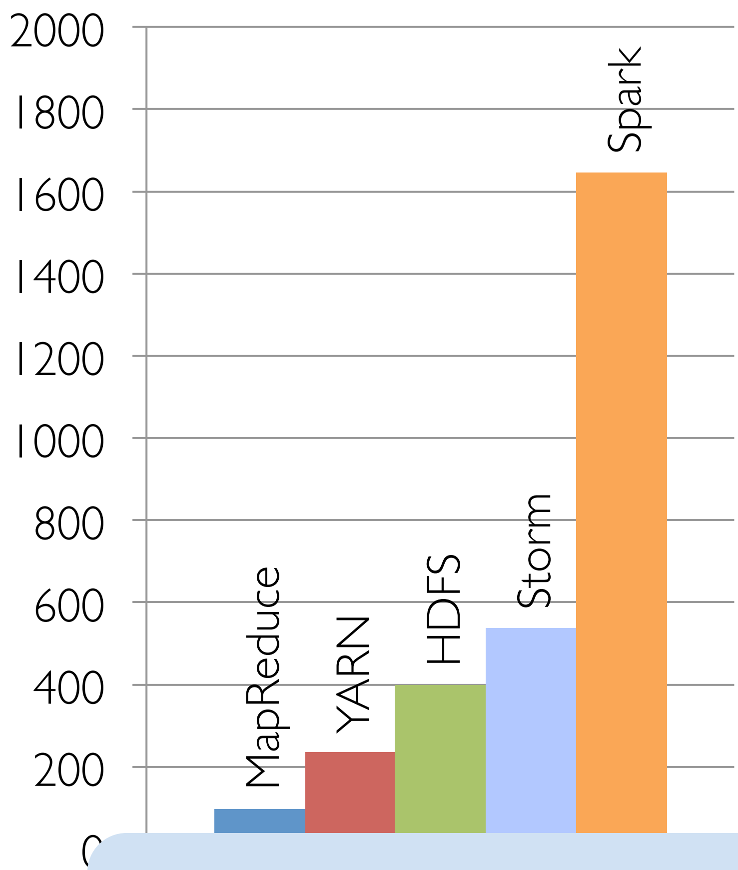
Monthly Contributors

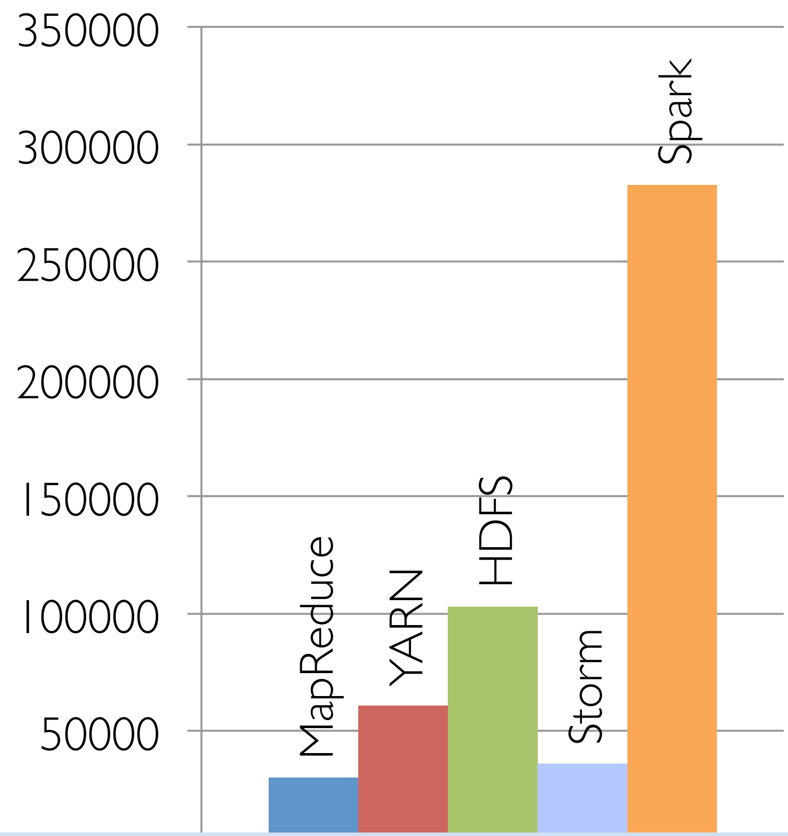




371 contributors for last 12 months

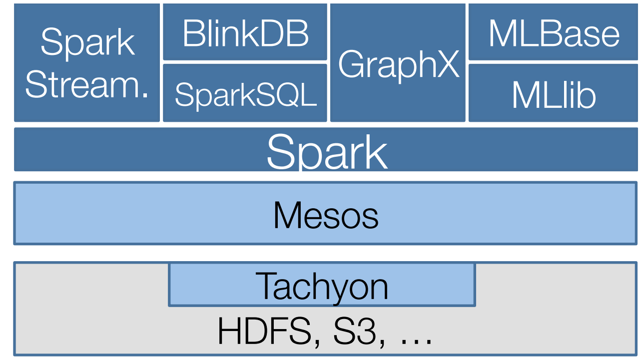
Compared to Other Projects





2-3x more activity than: Hadoop, Storm, MongoDB, NumPy, D3, Julia, …

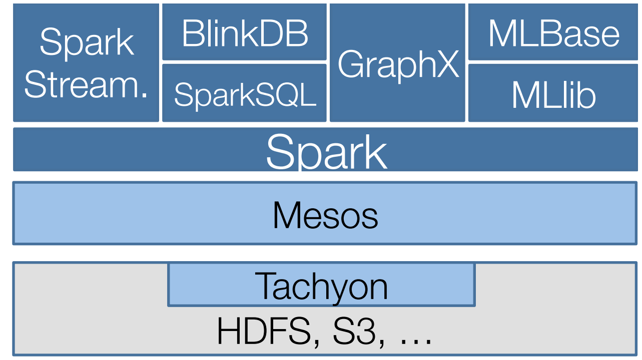
Wide Adoption



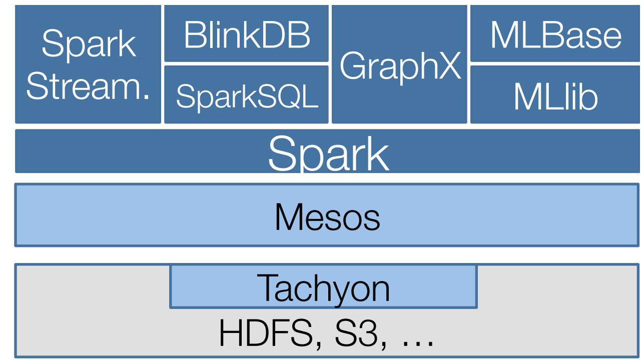
All major Hadoop distributions include Spark

Beyond Hadoop

Selected Users



Events



December 2013

Talks from 22 organizations

450 attendees

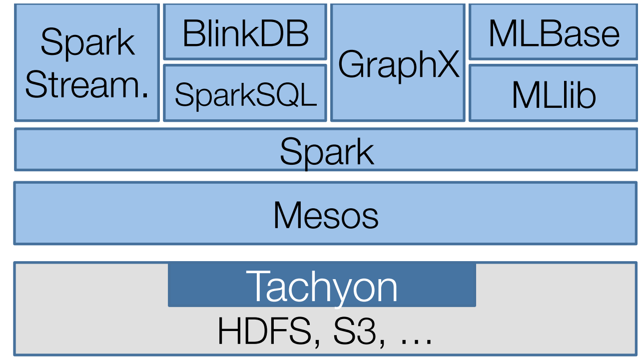
June 2014

Talks from 50 organizations

1100 attendees

spark-summit.org

Tachyon



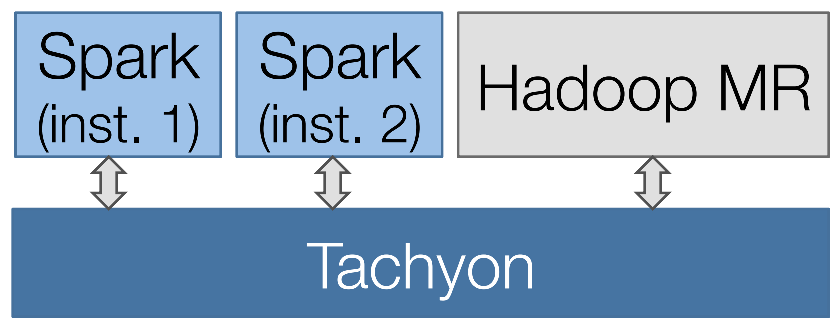
Problem:

» Different Spark contexts cannot share in-mem data

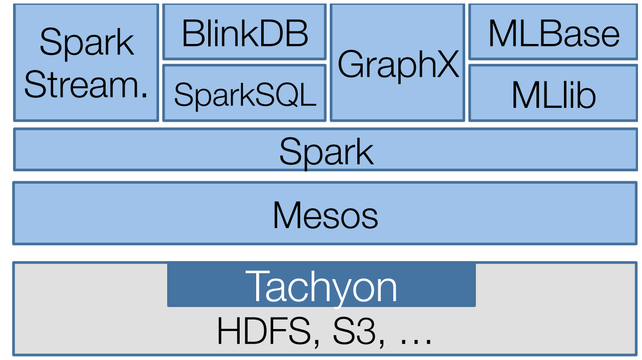
Solution:

» Flexible API, including HDFS API

» Allow multiple frameworks (including Hadoop) to share in-memory data



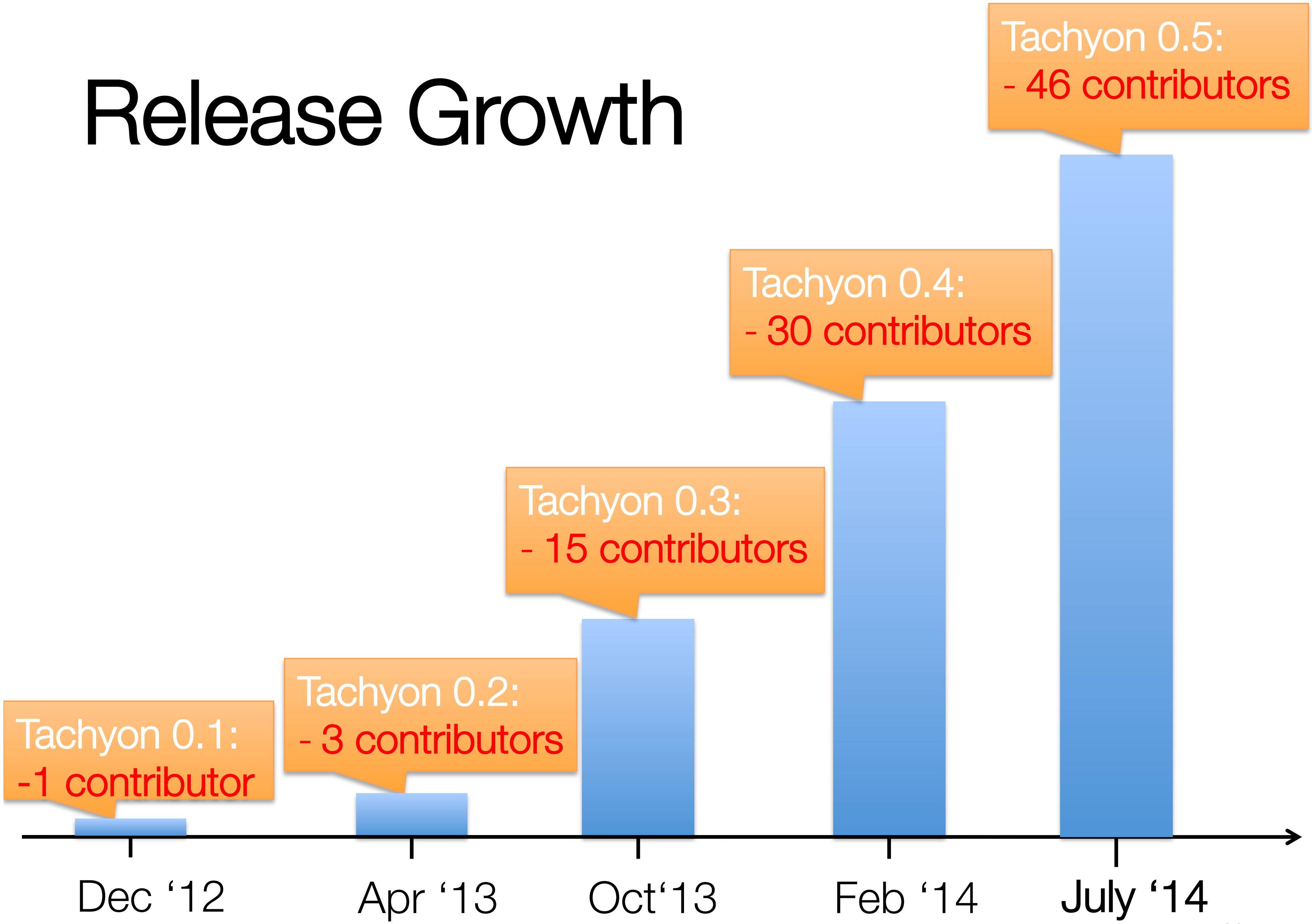
Tachyon



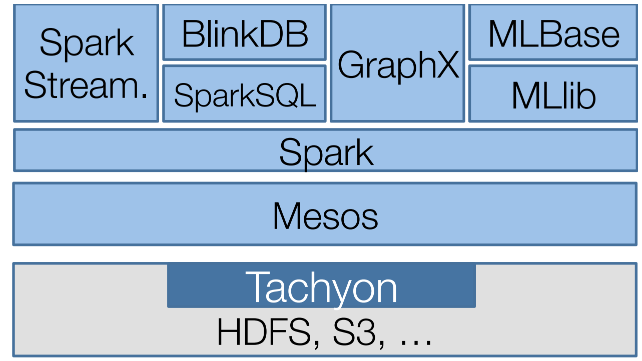
Open Source: Dec 2012 (<10,000 LoC)

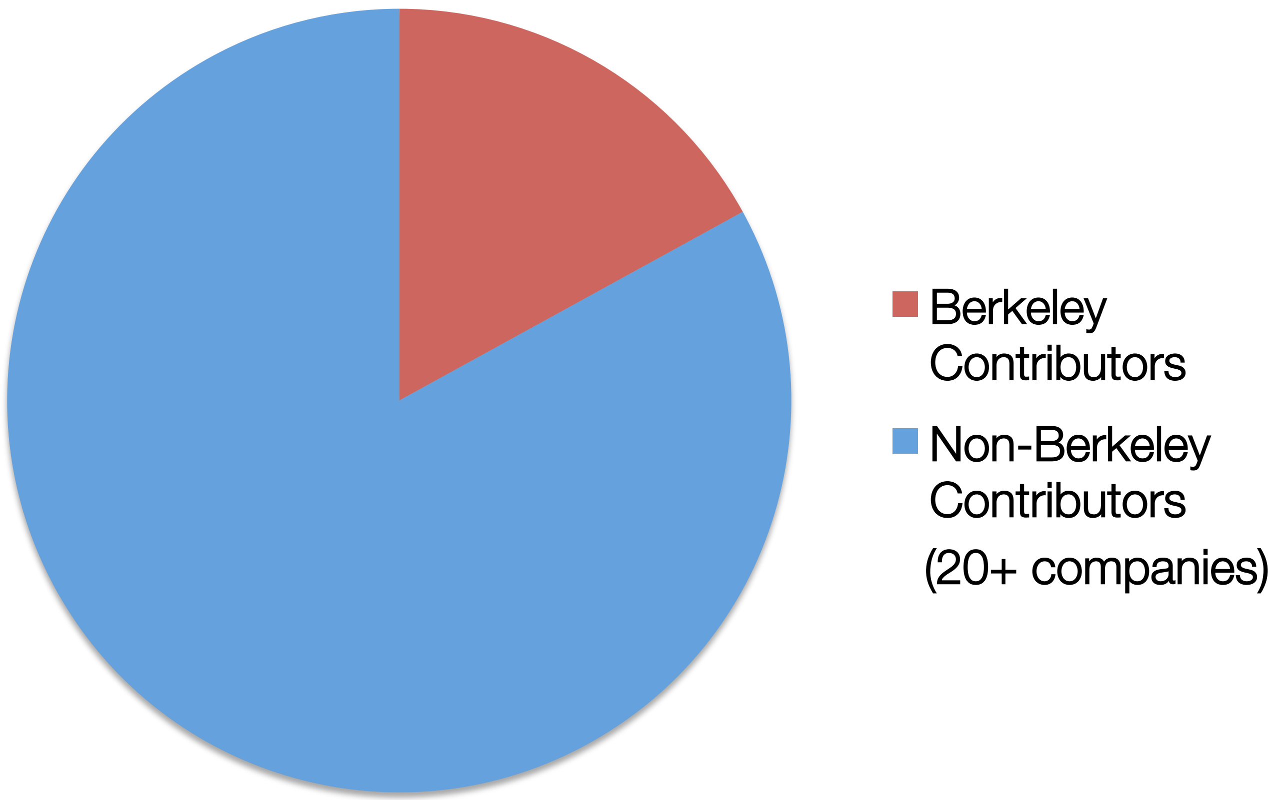
Becoming narrow waist for storage in Big Data space

Release Growth

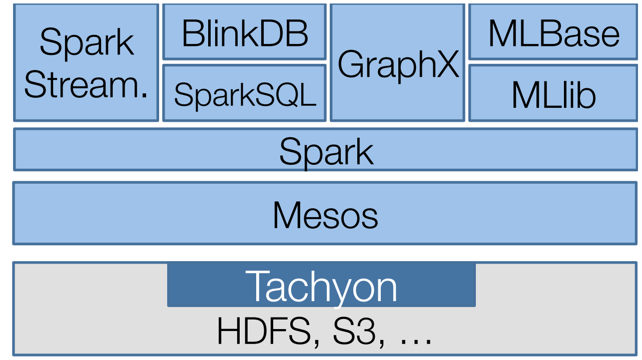


Open Community



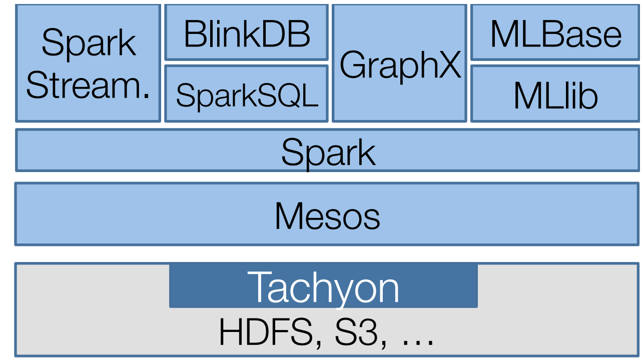


Selected Users

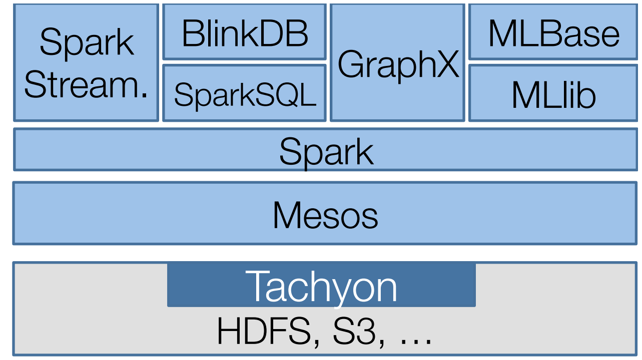


YAHOO!
intel
NOKIA
CONVIVA
Alibaba.com Global trade starts here.™
redhat
databricks
cloudera
ClearStory DATA Now You See It™
Palantir Technologies
Pivotal™
Atigeo™

Multiple File System Choices



Reaching Tipping Point



Pivotal

EMC²

The Future Architecture of a Data Lake: In-memory Data Exchange Platform Using Tachyon and Apache Spark

OCTOBER 14, 2014 | **NEWS** | BY PAUL M. DAVIS

GIGAOM

Pivotal and EMC are betting on Spark cousin Tachyon as in-memory file system

by Derrick Harris OCT. 14, 2014 - 11:47 AM PDT

ZDNet

Pivotal bets on Tachyon as next in-memory file system

database TRENDS AND APPLICATIONS

Pivotal Expands on Data Lake Vision with Embrace of Project Tachyon

Oct 14, 2014

Training: Integral Part of Success

Aug 2012: AMP Camp training workshop

- 150 in-person, 3000 online
- Now a regular event (Strata NY, training +450 people)

This year alone

- +1,800 trained people

Not Only Industrial Impact…

10s of papers at top conferences

» SOSP, SIGCOMM, SIGMOD, NIPS, VLDB, OSDI, NSDI, …

6 Best Paper Awards

» SIGCOMM, NSDI, EuroSys (2), ICML, ICDE

Great crop of students

» Last two years: MIT (3), Stanford (1), MSR, …

Open new research directions

» Resource allocation / microeconomy (DRF)

» Machine learning (Bootstrap Diagnosis)

And Even Saving Lives!

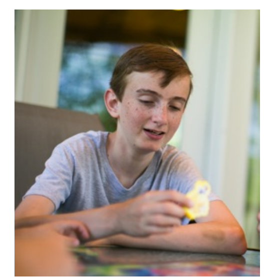
Scalable Nucleotide Alignment (SNAP)

» 3x-10x faster than state of art with same accuracy

ADAM Pipeline

» In use at the Broad Institute, Duke, Harvard, USCS

» 10x-50x faster than state of art



Already saving lives!

» See “SNAP Helps Discover Infection” AMPLab blog 6/4/14 and M. Wilson, …, & C. Chiu, “Actionable Diagnosis of Neruoleptospirosis by Next-Generation Sequencing,” New England Journal of Medicine, 6/4/14

Research Philosophy

Follow real problems

Focus on novel usage scenarios

Build real systems

- » Be paranoid about simplicity
- » Very hard to build complex systems in academia

Push for adoption

- » Develop communities
- » Train users

Two Types of Research Proj.

New systems

- Inspired from people using ours/existing systems
- E.g., Spark, Shark/SparkSQL, MLlib, SparkStreaming, Tachyon, …

New algorithms, techniques, optimizations

- Workload traces from large clusters (e.g., Facebook, Conviva)
- E.g., LATE, Sparrow, PACMan, Scarlett, …

Challenge: Public Clouds

Hugely convenient and powerful

- » E.g., we won Terabyte sort benchmark this year using 206 AWS instances
  - 3x faster, 10x fewer machines than last year (Yahoo!)
- » Whatever you deploy on AWS/Azure/GC can be used by anyone
  - Large pool of users (beyond academia)
  - Easy to train
- » Large public data sets already available http://aws.amazon.com/datasets/

Why Use Experimental Testbeds?

Control and visibility

- » Bare-metal servers
  - Some clouds do provide this: Rackspace, DigitalOcean
- » SDN networks, RDMAs, …

Re-configurability, heterogeneity

Free!

Enable end-to-end / cross-layer optimizations

What about Data and Apps?

Ideally, unique data not found on other clouds

Example:

- Fine grained logs/traces of cloud usage (public clouds cannot provide this)
- Scientific data (?)

Applications

- Ability to run existing systems/apps (need to maintain them!)
- New education apps (?)

Conclusions

Have right expectations, key to success!

Be aware that:

» Public clouds cover a big range of needs for system research

» Insights for new use cases unlikely to come from these testbeds

Focus on what is unique:

» Cross-layer optimization exploiting access to network (SDN, RDMA), storage, bare-bone servers

» Make available unique data sets (e.g., fine grained logs)