Nine Decades of Machine Learning

General Assembly
SF, 2014-05-05

Paco Nathan @pacoid
liber118.com/pxn/

Big Data, really?

Another industry writer who's a good friend taunts me that it's almost *always* been about **Medium Data,** except in a few isolated examples

What do you think?

The following talk provide several points to consider, along with way too many links… in case you want a deep-dive in any particular area of interest

Introduction: *Nine Decades of Machine Learning*

An intellectual regime reigned for nearly 200 years in the fields most closely related to quantitative decision making

These aristotelian vices finally began to implode after the end of the Cold War…

(which we're writing a book about!)

Introduction: *Nine Decades of Machine Learning*

This is a story of historical arcs that wielded tremendous influence on business use cases for *cluster computing*, *machine learning*, and *open source*:

- *how machine data used at scale in a social context changed business globally* (Maes => Amazon)
- *how diversity of perspectives gained an upper hand over the statistical cajolery of insurance actuaries* (Breiman => Netflix)
- *how iterative judgements in the midst of changing evidence gained advantage* (Bayes => high ROI)

Some subtle outcomes:

- *use of VMs altered abruptly, compelled by the basic constraints of our power grid*
- *middleware abstractions evolved to handle the complexities of data apps at scale*
- *IoT data rates exceed social networks by orders of magnitude; new skills are required*
- *business people must now leverage advanced math "beyond calculus" for what's ahead*
- *computational thinking + functional programming become vital, but are not difficult*

Context

A Brief History

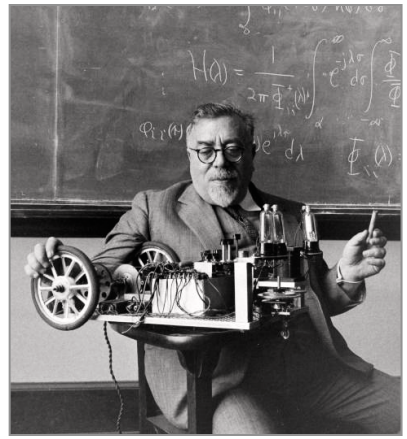
Context: *First Order Cybernetics*

Norbert Wiener helped establish the field of *cybernetics*

Early work focused on filters that could adapt to noisy signals, based on feedback…

He applied this to control systems for radar and anti-aircraft weapons in WWII

Later work focused on the implications of systems incorporating people+machines at large scale…



Norbert Wiener
wikipedia.org

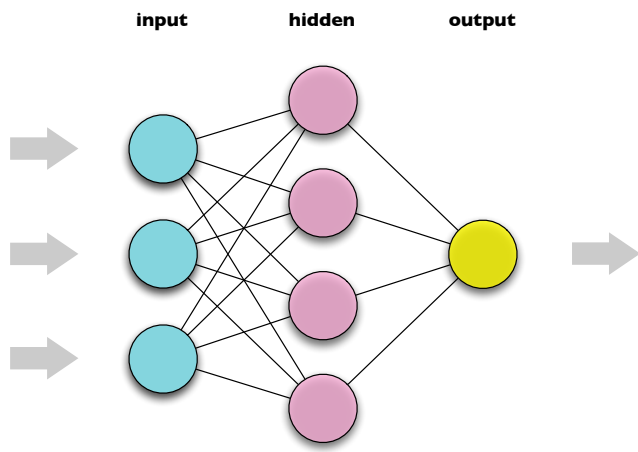
Context: *What The Frog's Eye Sees*

On the theme of signal processing, in the context of biology instead of electronics, **McCulloch & Pitts** studied optic nerves of frogs… recruited to MIT by Weiner

McCulloch & Pitts
wikipedia.org

Networks of artificial neurons could train features – creating an approach that could work with partial or noisy input

What The Frog's Eye Tells The Frog's Brain
Lettvin, Maturana, McCulloch, Pitts
Proc. Inst. Radio Engr. (1959)
http://jerome.lettvin.info/lettvin/Jerome/WhatTheFrogsEyeTellsTheFrogsBrain.pdf



Context: *AM, Eurisko, CYC…*

www.dtic.mil/cgi-bin/GetTRDoc..
www.aaai.org/Papers/AAAI/1983/AAAI83-059.pdf
www.cs.northwestern.edu/~mek802/papers..
newyorker.com/reporting/2009/05/11/090511..

Doug Lenat
wikipedia.org

- mutating short symbolic programs that represented concepts
- identified *pragmatics* and *conventions* as specialized knowledge
- frame language, introducing *metadata*
- vocabulary and grammar extended by learner
- formal definitions for heuristic *search*
- VLSI design, suggested *möbius strip* since 3D rules were relaxed
- banned from Traveller TCS after winning competition twice

Context: *Social Information Filtering for Music Recommendation*

wikipedia.org/wiki/Firefly
businessweek.com/1996/41/b349690.htm
pubs.media.mit.edu/pubs/papers/32paper.ps

- Firefly, an early commercial recommender system
- intent: the volume of data about things is more than any person can digest
- leveraged *similarity* within a network
- an evolution of intelligent agents into web apps
- collect *machine data* about consumer interests
- people communicating with each other and with machines

Pattie Maes
MIT Media Lab

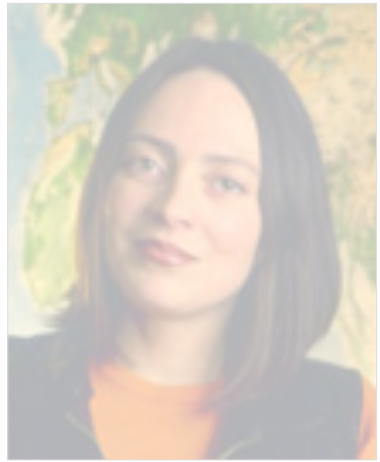
Context: *Social Information Filtering for Music Recommendation*

wikipedia.org/wiki/Firefly
businessweek.com/1996/41/b349690.htm
pubs.media.mit.edu/pubs/papers/32paper.ps

- Firefly,
- intent:
  than ar
- leverag
- an evol
- collect
- people communicating with each other and with machines

This changed everything…

- efforts *before*: some guy coding on a computer, with gov funding
- efforts *after*: intelligent social web apps with 100Ms users



Pattie Maes
MIT Media Lab

Context

3Q 1997 Inflection

Context: *3Q 1997 Inflection Point*

Four independent teams were working toward horizontal scale-out of workflows based on commodity hardware

This effort prepared the way for huge Internet successes during the 1997 holiday season…

AMZN, **EBAY**, Inktomi (**YHOO** Search), then **GOOG**

MapReduce on clusters of commodity hardware and the Apache Hadoop open source stack emerged from this context

Context: *3Q 1997 Inflection Point*

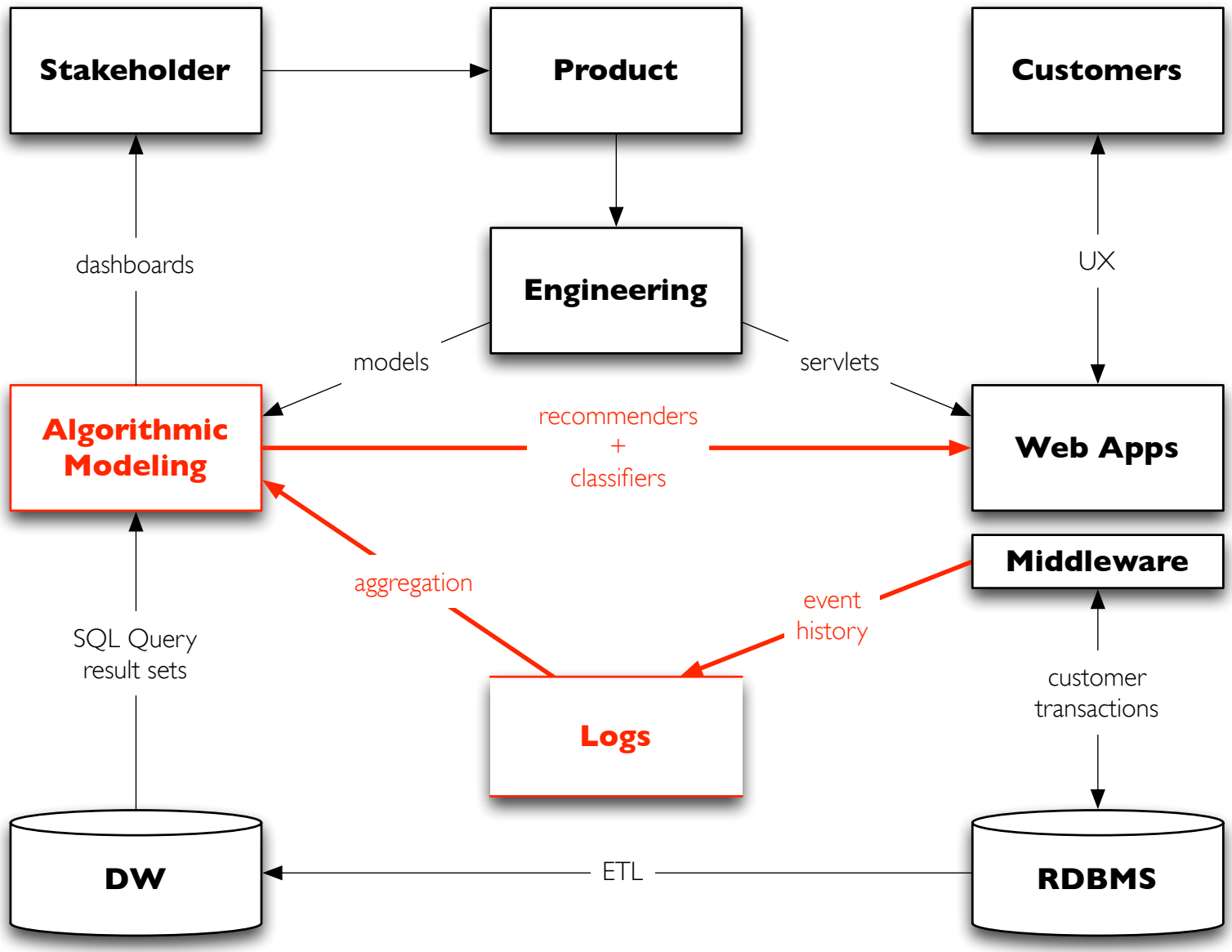
Amazon
"Early Amazon: Splitting the website" – Greg Linden
glinden.blogspot.com/2006/02/early-amazon-splitting-website.html

eBay
"The eBay Architecture" – Randy Shoup, Dan Pritchett
addsimplicity.com/adding_simplicity_an_engi/2006/11/you_scaled_your.html
addsimplicity.com.nyud.net:8080/downloads/eBaySDForum2006-11-29.pdf

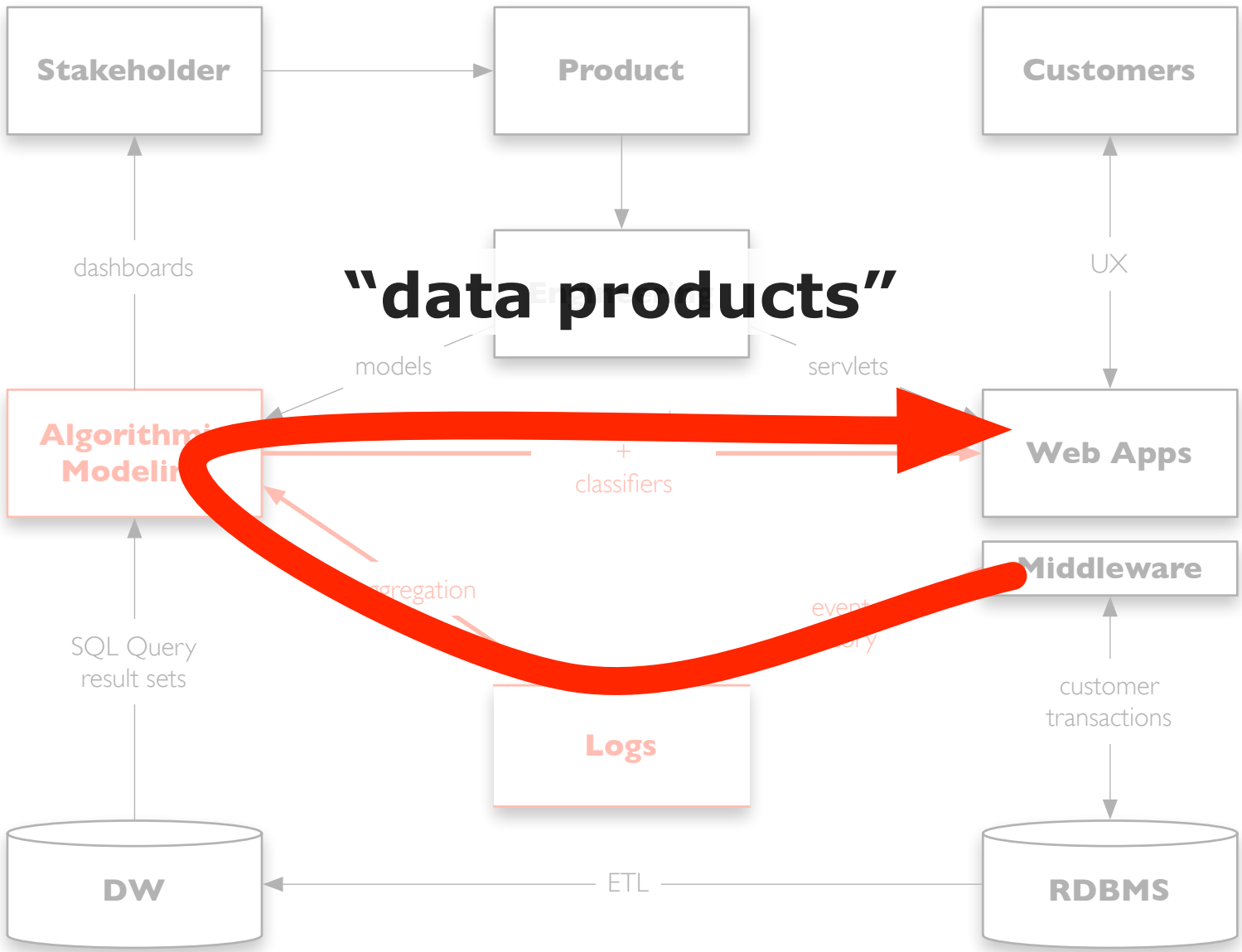
Inktomi (**YHOO** Search)
"Inktomi's Wild Ride" – Erik Brewer (0:05:31 ff)
youtu.be/E91oEn1bnXM

Google
"Underneath the Covers at Google" – Jeff Dean (0:06:54 ff)
youtu.be/qsan-GQaeyk
perspectives.mvdirona.com/2008/06/11/JeffDeanOnGoogleInfrastructure.aspx

Context: *circa 2001, post e-commerce first big successes*



Context: *circa 2001, post e-commerce first big successes*



Context

Algorithmic Modeling

Context: *Lies, Damn Lies, Statistics, Bayesian Statistics*

Leo Breiman chronicled a sea change from *data modeling* (silos, manual process) to the rising use of *algorithmic modeling* circa 2001 (machine data, automation) and *ensembles*, plus the corresponding shift to interdisciplinary teams (Data Science):

Leo Breiman
berkeley.edu

Statistical Modeling: The Two Cultures
Leo Breiman
UC Berkeley (2001)
bit.ly/eUTh9L

A new research community using these tools sprang up. Their goal was predictive accuracy. The community consisted of young computer scientists, physicists and engineers plus a few aging statisticians. They began using the new tools in working on complex prediction problems where it was obvious that data models were not applicable: speech recognition, image recognition, nonlinear time series prediction, handwriting recognition, prediction in financial markets.

Context: *Lies, Damn Lies, Statistics, Bayesian Statistics*

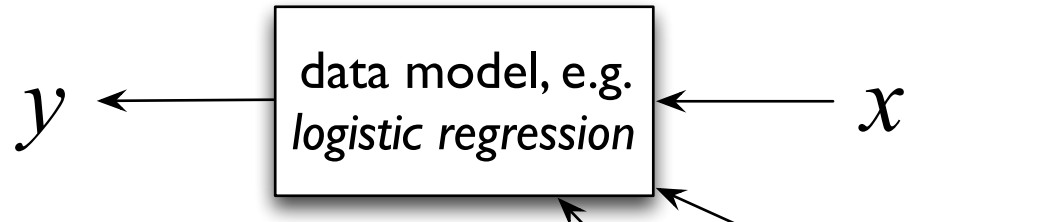
Rashomon, the 1950 Japanese period drama by Akira Kurosawa, symbolizes a long-standing tension in Statistics, one which Mark Twain described ever so succinctly…

wikipedia.org/wiki/Rashomon:

"The film is known for a plot device which involves various characters providing alternative, self-serving and contradictory versions of the same incident."

Context: *Lies, Damn Lies, Statistics, Bayesian Statistics*

A view of *statistical data modeling*:



$y = f(x, random\ noise, model\ parameters)$

x : *predictor variables*

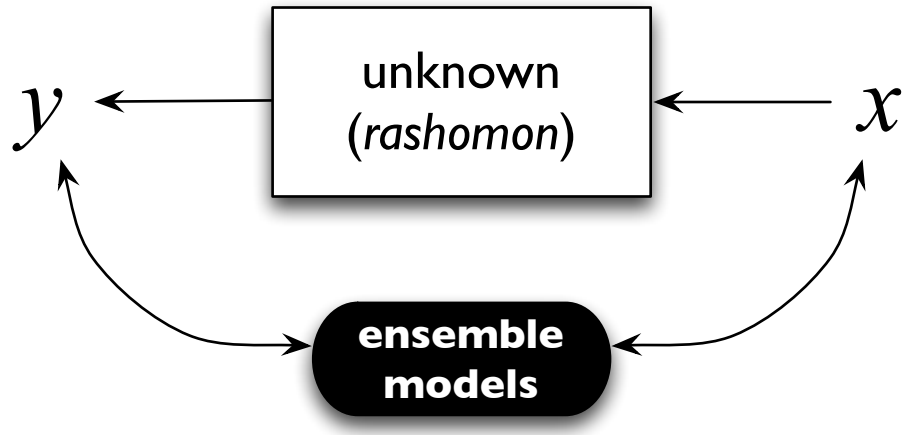
y : *response variables*

optimize for:

goodness-of-fit tests, *residuals*, etc.

Context: *Lies, Damn Lies, Statistics, Bayesian Statistics*

A view of *algorithmic modeling*:



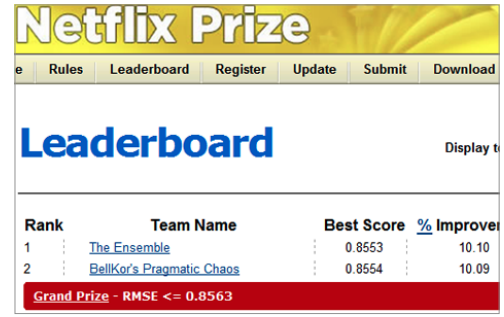
optimize for:

predictive accuracy

Context: *Lies, Damn Lies, Statistics, Bayesian Statistics*

Breiman: "a multiplicity of data models"

BellKor team: 100+ individual models in 2007 Progress Prize



> *while the process of combining models adds complexity, making it more difficult to anticipate or explain predictions, accuracy may increase substantially*

Ensemble Learning: Better Predictions Through Diversity
Todd Holloway
ETech (2008)
abeautifulwww.com/EnsembleLearningETech.pdf

The Story of the Netflix Prize: An Ensemblers Tale
Lester Mackey
Nat'l Academies Seminar (2011)
stanford.edu/~lmackey/papers/

Distributed Systems

Datacenter Computing

Datacenter Computing

Google has been doing *datacenter computing* for years,
to address the complexities of large-scale data workflows:

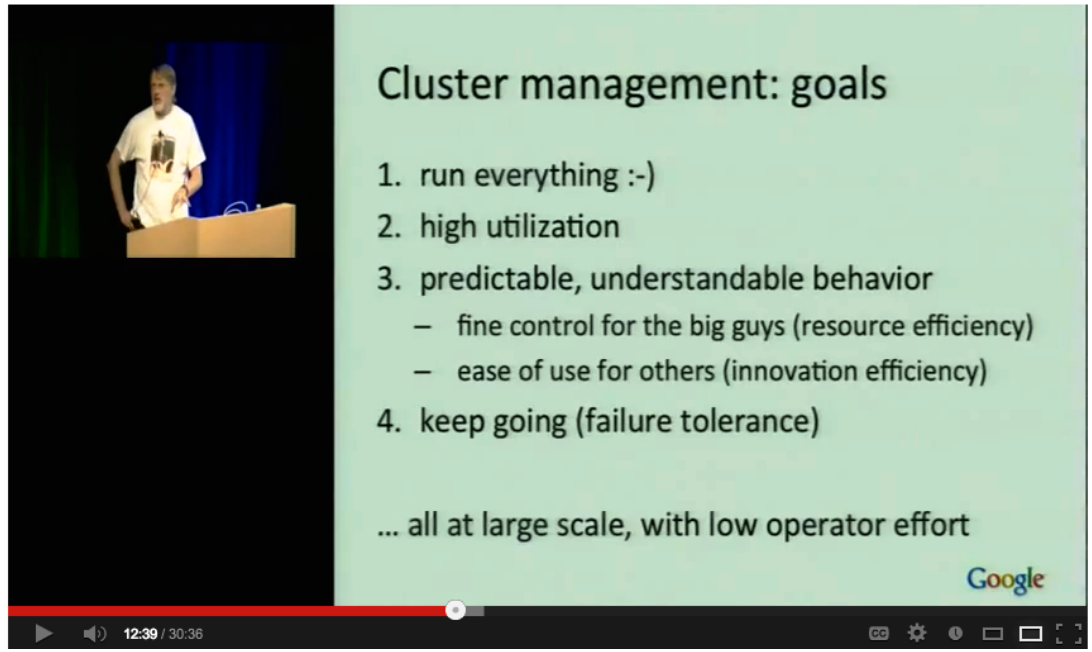
- leveraging the modern kernel: isolation in lieu of VMs
- "most (>80%) jobs are batch jobs, but the majority of resources (55–80%) are allocated to service jobs"
- mixed workloads, multi-tenancy
- relatively high utilization rates
- JVM? not so much…
- reality: scheduling batch is simple; scheduling services is hard/expensive

“Return of the Borg”

Return of the Borg: How Twitter Rebuilt Google’s Secret Weapon
Cade Metz
wired.com/wiredenterprise/2013/03/google-borg-twitter-mesos

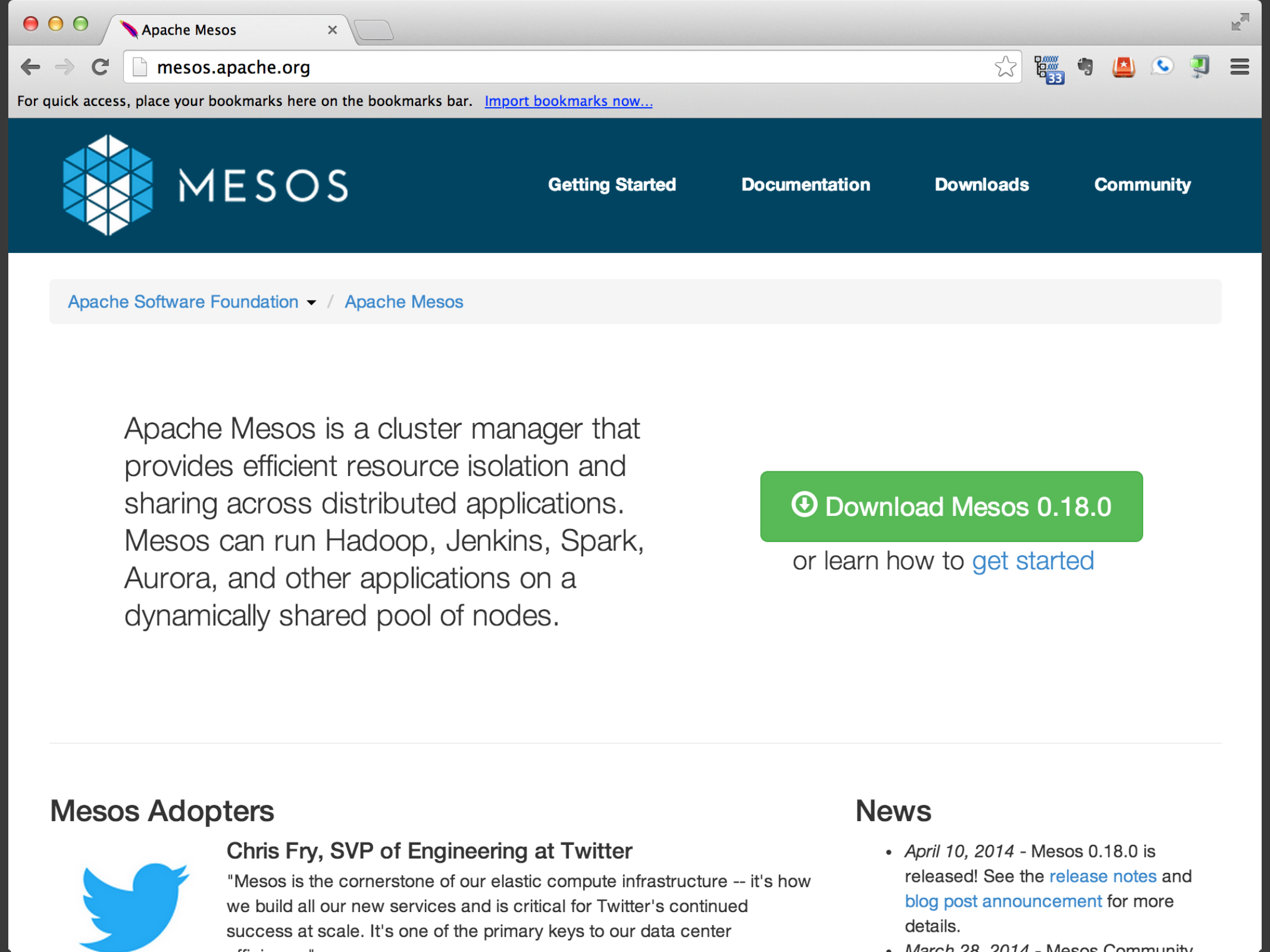
Omega: flexible, scalable schedulers for large compute clusters
Malte Schwarzkopf, Andy Konwinski, Michael Abd-El-Malek, John Wilkes
eurosys2013.tudos.org/wp-content/uploads/2013/paper/Schwarzkopf.pdf

2011 GAFS Omega
John Wilkes, et al.
youtu.be/0ZFMIO98Jkc



Google describes the business case...

Taming Latency Variability
Jeff Dean
plus.google.com/u/0/+ResearchatGoogle/posts/C1dPhQhcDRv

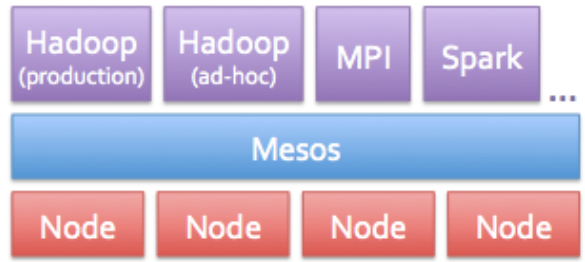

Apache Mesos
mesos.apache.org
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
MESOS
Getting Started
Documentation
Downloads
Community
Apache Software Foundation / Apache Mesos
Apache Mesos is a cluster manager that provides efficient resource isolation and sharing across distributed applications. Mesos can run Hadoop, Jenkins, Spark, Aurora, and other applications on a dynamically shared pool of nodes.
Download Mesos 0.18.0
or learn how to get started
Mesos Adopters
Chris Fry, SVP of Engineering at Twitter
"Mesos is the cornerstone of our elastic compute infrastructure -- it's how we build all our new services and is critical for Twitter's continued success at scale. It's one of the primary keys to our data center
News
April 10, 2014 - Mesos 0.18.0 is released! See the release notes and blog post announcement for more details.

Apache Mesos: open source datacenter computing

a common substrate for cluster computing

mesos.apache.org

heterogenous assets in your datacenter or cloud
made available as a homogenous set of resources



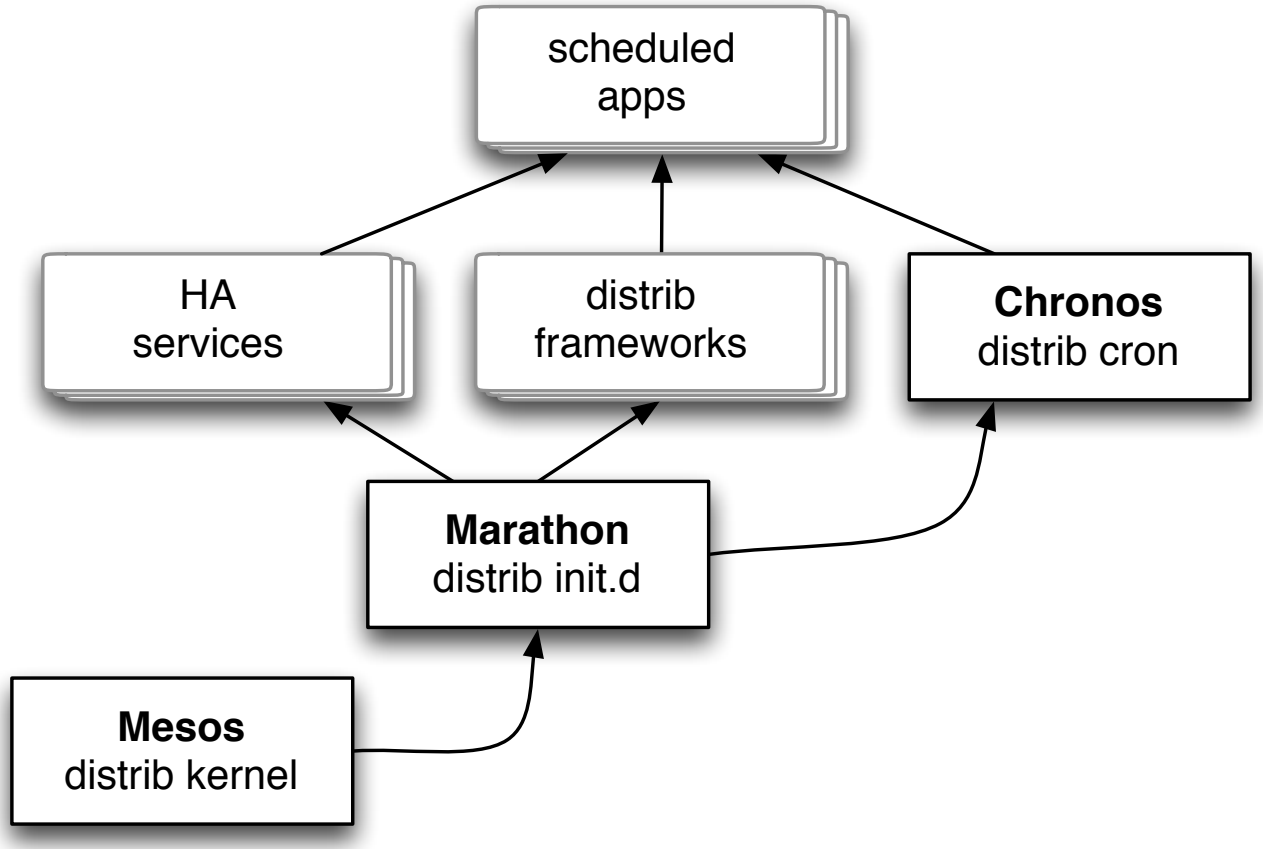
- top-level Apache project
- scalability to 10,000s of nodes
- obviates the need for virtual machines
- isolation (pluggable) for CPU, RAM, I/O, FS, etc.
- fault-tolerant leader election based on Zookeeper
- APIs in **C++**, **Java/Scala**, **Python**, **Go**, **Erlang**, **Haskell**
- web UI for inspecting cluster state
- available for Linux, OpenSolaris, Mac OSX

Apache Mesos: One Large Pool of Resources

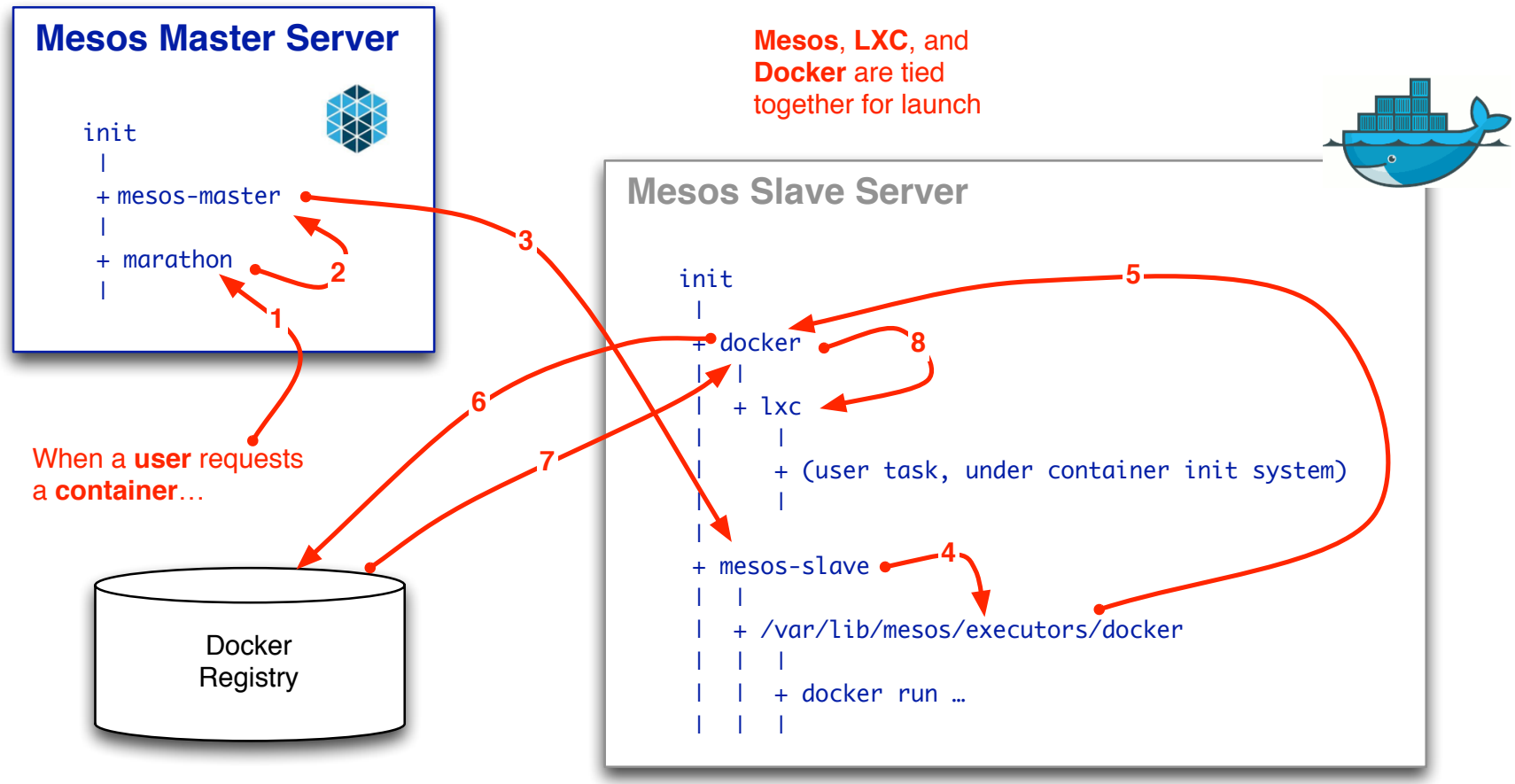
"We wanted people to be able to program for the datacenter just like they program for their laptop."

Ben Hindman

Apache Mesos: Much Like Booting Linux



Example: Docker on Mesos

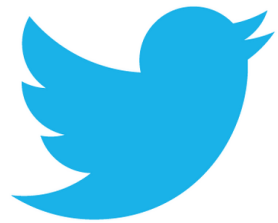


mesosphere.io/2013/09/26/docker-on-mesos/

Production Deployments (public)

Case Study: Twitter (bare metal / on premise)

"Mesos is the cornerstone of our elastic compute infrastructure – it's how we build all our new services and is critical for Twitter's continued success at scale. It's one of the primary keys to our data center efficiency."



Chris Fry, SVP Engineering

blog.twitter.com/2013/mesos-graduates-from-apache-incubation

wired.com/gadgetlab/2013/11/qa-with-chris-fry/

- key services run in production: analytics, typeahead, ads
- Twitter engineers rely on Mesos to build all new services
- instead of thinking about static machines, engineers think about resources like CPU, memory and disk
- allows services to scale and leverage a shared pool of servers across datacenters efficiently
- reduces the time between prototyping and launching

Case Study: Airbnb (fungible cloud infrastructure)

"We think we might be pushing data science in the field of travel more so than anyone has ever done before… a smaller number of engineers can have higher impact through automation on Mesos."

Mike Curtis, VP Engineering
gigaom.com/2013/07/29/airbnb-is-engineering-itself-into-a-data...

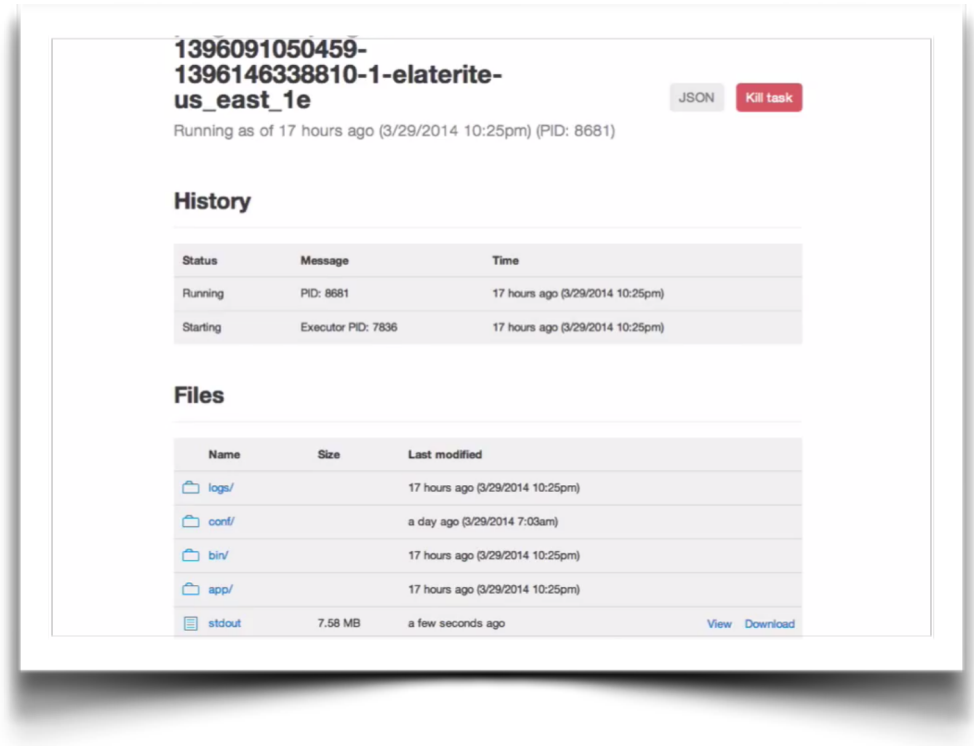
- improves resource management and efficiency
- helps advance engineering strategy of building small teams that can move fast
- key to letting engineers make the most of AWS-based infrastructure beyond just Hadoop
- allowed company to migrate off Elastic MapReduce
- enables use of Hadoop along with Chronos, Spark, Storm, etc.

Case Study: HubSpot (cluster management)

Tom Petr
youtu.be/ROn14csiikw

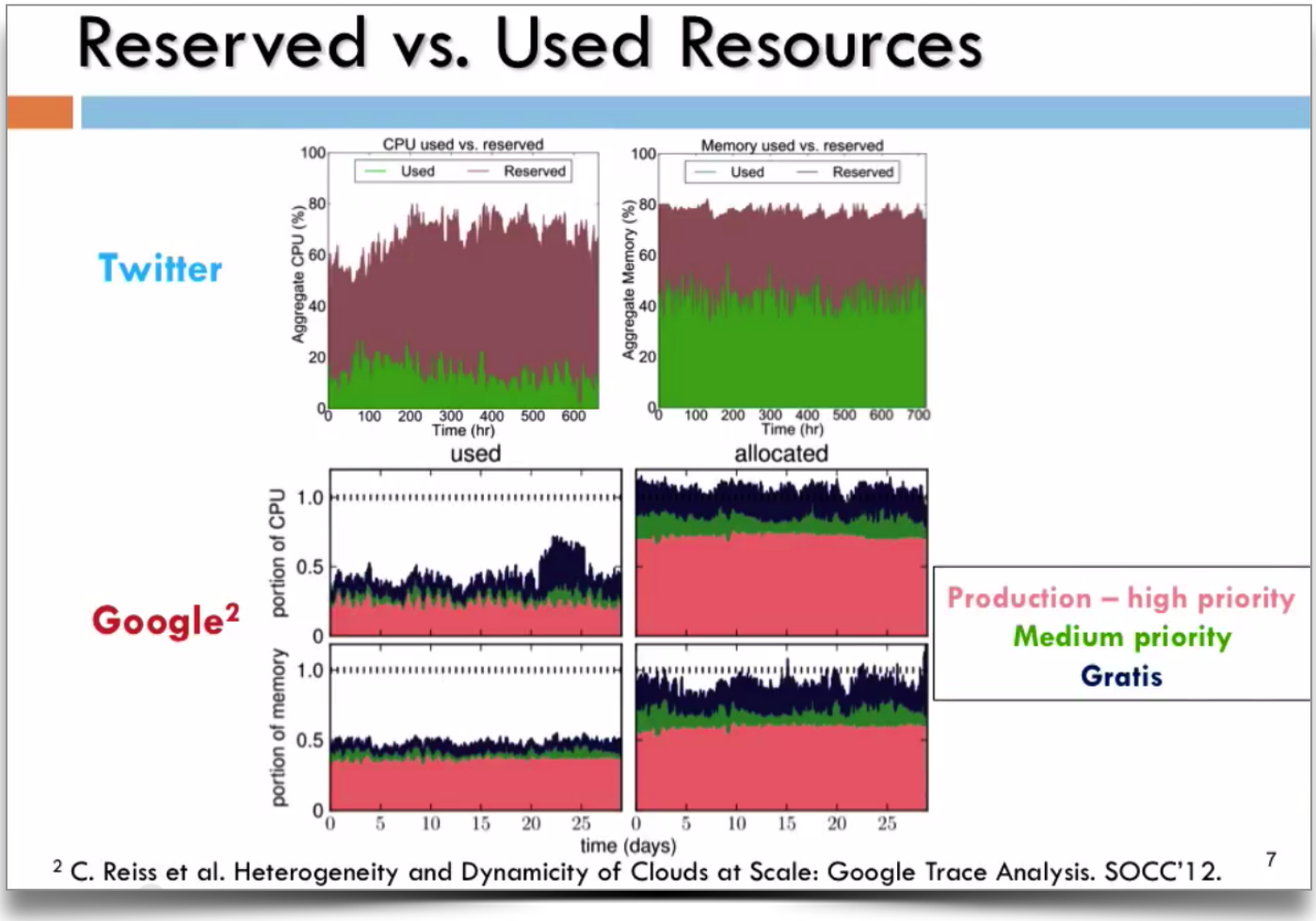
mesosphere.io/resources/mesos-case-study-hubspot/

- 500 deployable objects; 100 deploys/day to production; 90 engineers; 3 devops on Mesos cluster
- "Our QA cluster is now a fixed $10K/month — that used to fluctuate"



Quasar+Mesos @ Stanford, Twitter, etc....

Improving Resource Efficiency with Apache Mesos
Christina Delimitrou
youtu.be/YpmElyi94AA



O'REILLY®

Data

Search

Posts by Topic: Big Data Data Science Health Data Data Journalism Hadoop Visit oreilly.com

Apache Mesos: Open Source Datacenter Computing

Print

Listen

Read Later

by Paco Nathan | @pacoid | +Paco Nathan | Comment | January 8, 2014

Tweet 89 g+1 8 Like 45 Share 42

Virtual machines (VMs) have enjoyed a long history, from IBM's CP–40 in the late 1960s on through the rise of VMware in the late 1990s. Widespread VM use nearly became synonymous with "cloud computing" by the late 2000s: public clouds, private clouds, hybrid clouds, etc. One firm, however, bucked the trend: Google.

Google's datacenter computing leverages isolation in lieu of VMs. Public disclosure is limited, but the Omega paper from EuroSys 2013 provides a good overview. See also two YouTube videos: John Wilkes in 2011 GAFS Omega and Jeff Dean in Taming Latency Variability… For the business case, see an earlier Data blog post, that discusses how multi-tenancy and efficient utilization translates into improved ROI.

Datacenter Computing with Apache Mesos

slideshare.net/pacoid/datacenter-computing-with-apache-mesos

vimeo.com/79016209

strataconf.com/strata2014/public/schedule/detail/31869

Data Workflows

Abstraction Layers

Data Workflows for Machine Learning

Machine Learning in production apps has become less about algorithms (even though that work is quite fun and vital)

Performing real work is more about:

- socializing a problem within an organization
- feature engineering ("Beyond Product Mgrs")
- tournaments in CI/CD environments
- operationalizing high-ROI apps at scale
- etc.

Let's crawl out on a limb and state that leveraging great frameworks to build data workflows is more important than chasing after diminishing returns on highly nuanced algorithms

Data Workflows for Machine Learning

Middleware has been evolving for Big Data, and there are some great examples – we'll review several…

Process has been evolving too, right along with the use cases

Popular frameworks typically provide some Machine Learning capabilities within their core components, or at least among their major use cases

Their requirements for scale, robustness, cost trade-offs, interdisciplinary teams, etc., serve as guides in general

Data Workflows for Machine Learning

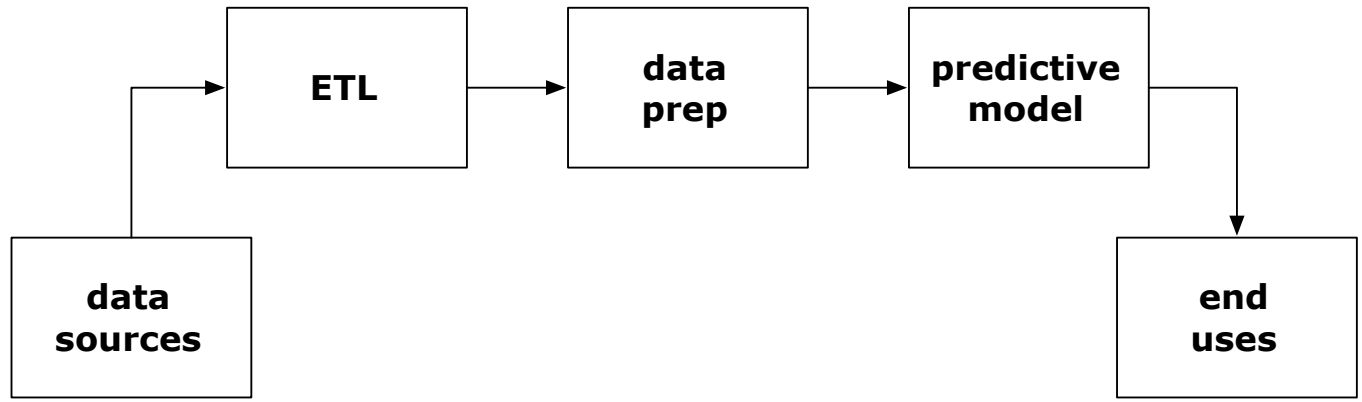
Caveat Auditor: not claiming to be expert in each framework and environment described in this talk – expert with a few of them perhaps, but more to the point: embroiled in many use cases

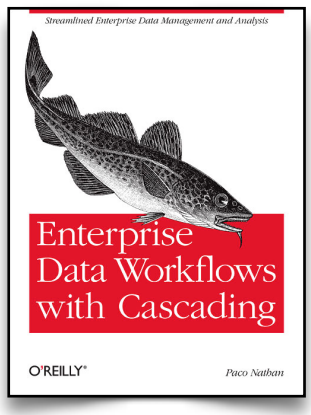
This talk attempts to define a "scorecard" for evaluating important ML data workflow features: what's needed for use cases, compare and contrast of what's available, plus some indication of which frameworks are likely to be best for a given scenario.

Seriously, this is a work in progress…

Abstraction Layers: Definitions

Middleware has been evolving for Big Data and Machine Learning… the following design pattern – a DAG – shows up in many places, through many frameworks:



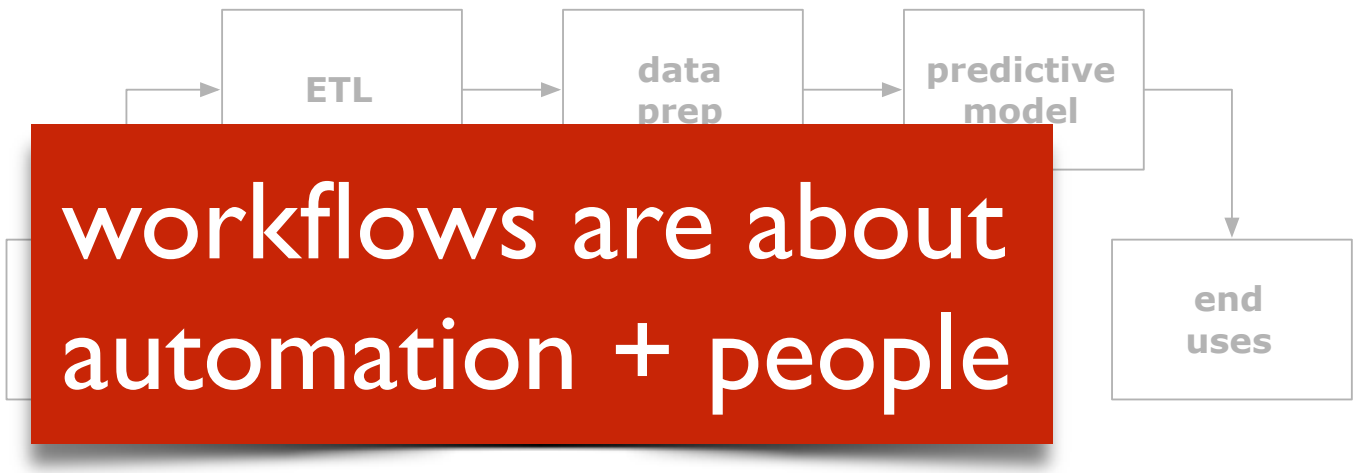


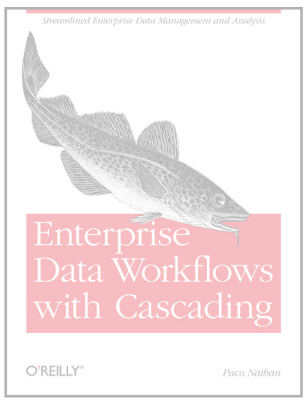
Enterprise Data Workflows with Cascading
O'Reilly (2013)

shop.oreilly.com/product/0636920028536.do

Abstraction Layers: Definitions

Middleware has been evolving for Big Data and Machine Learning… the following design pattern – a DAG – shows up in many places, through many frameworks:

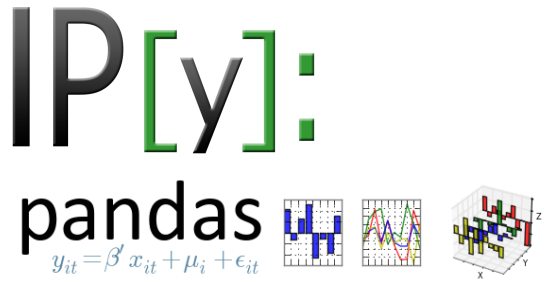


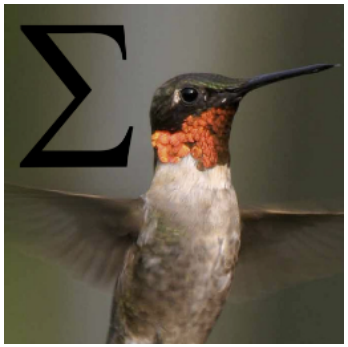


Enterprise Data Workflows with Cascading
O'Reilly (2013)

shop.oreilly.com/product/0636920028536.do


KNIME


IP[y]:
pandas
$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$


Σ


actian


open data
Augustus

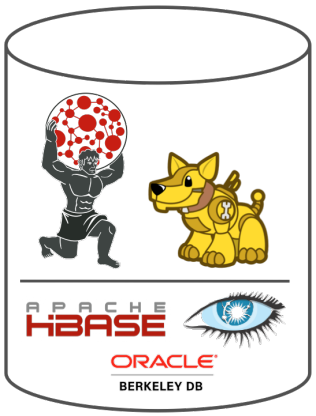

Scalding


RStudio


{m}


Cascalog


Spark


APACHE
HBASE
ORACLE
BERKELEY DB


julia


cascading


ML base


ADAPA®

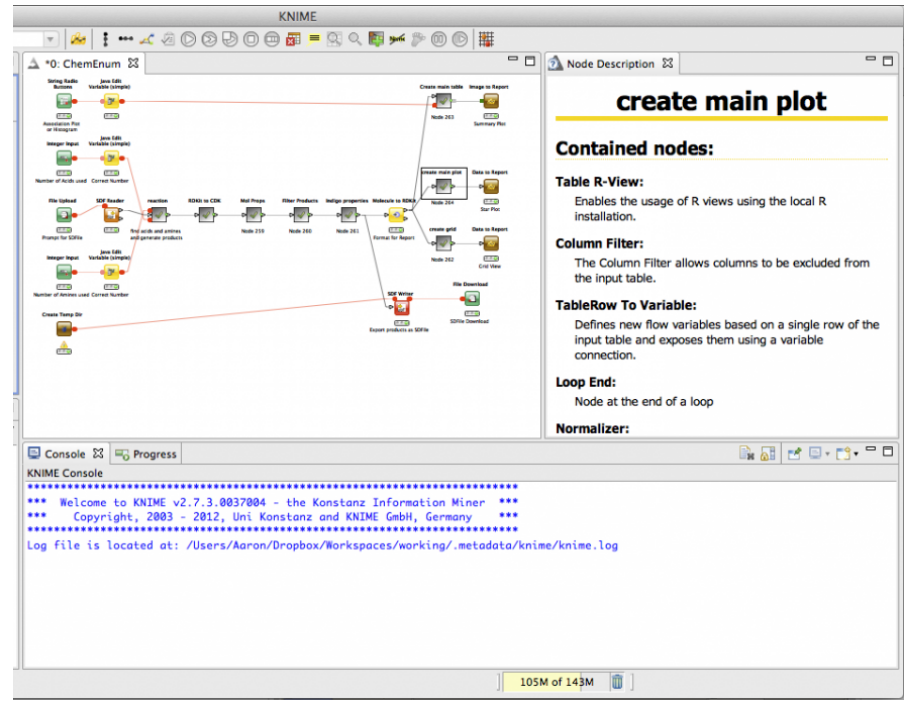

mahout

Example: KNIME

"a user-friendly graphical workbench for the entire analysis process: data access, data transformation, initial investigation, powerful predictive analytics, visualisation and reporting."

knime.org

- large number of integrations (over 1000 modules)
- ranked #1 in customer satisfaction among open source analytics frameworks
- visual editing of reusable modules
- leverage prior work in R, Perl, etc.
- Eclipse integration
- easily extended for new integrations



Example: Python stack

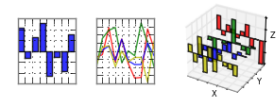
Python has much to offer – ranging across an organization, not just for the analytics staff

- ipython.org
- pandas.pydata.org
- scikit-learn.org
- numpy.org
- scipy.org
- code.google.com/p/augustus
- continuum.io
- nltk.org
- matplotlib.org

IP[y]:

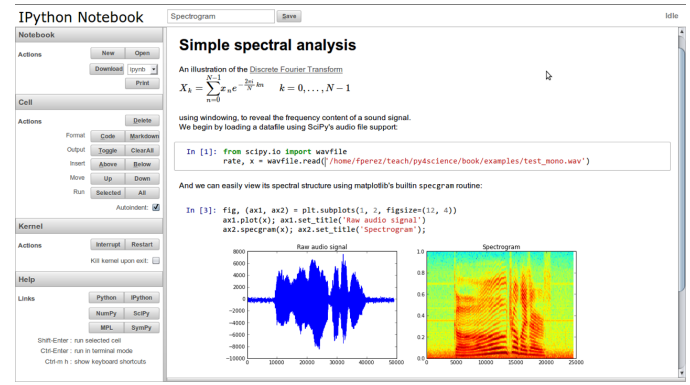
pandas

$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$



open data

Augustus



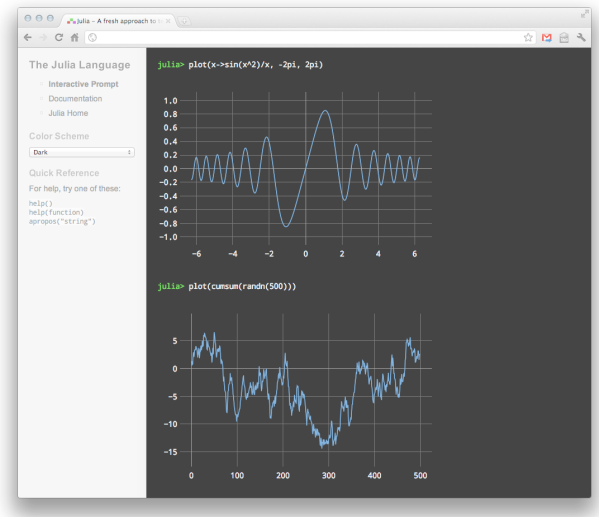
Example: Julia

"a high-level, high-performance dynamic programming language for technical computing, with syntax that is familiar to users of other technical computing environments"

julialang.org

- significantly faster than most alternatives
- built to leverage parallelism, cloud computing
- still relatively new — one to watch!

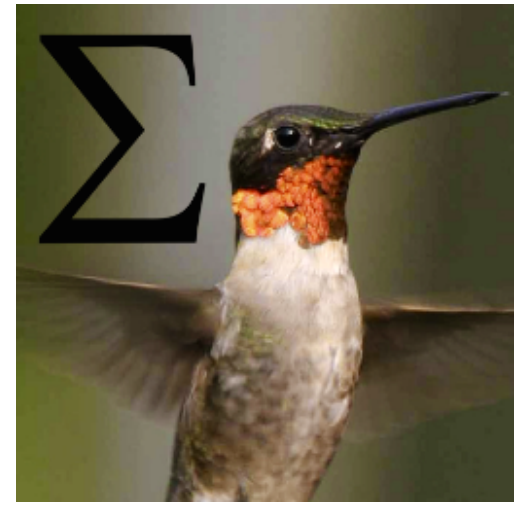
```
importall Base

type BubbleSort <: Sort.Algorithm end

function sort!(v::AbstractVector, lo::Int, hi::Int, ::BubbleSort, o::Sort.Ordering)
    while true
        clean = true
        for i = lo:hi-1
            if Sort.lt(o, v[i+1], v[i])
                v[i+1], v[i] = v[i], v[i+1]
                clean = false
            end
        end
        clean && break
    end
    return v
end
```



Example: Summingbird

"a library that lets you write streaming MapReduce programs that look like native Scala or Java collection transformations and execute them on a number of well-known distributed MapReduce platforms like Storm and Scalding."

- switch between Storm, Scalding (Hadoop)
- Spark support is in progress
- leverage Algebird, Storehaus, Matrix API, etc.

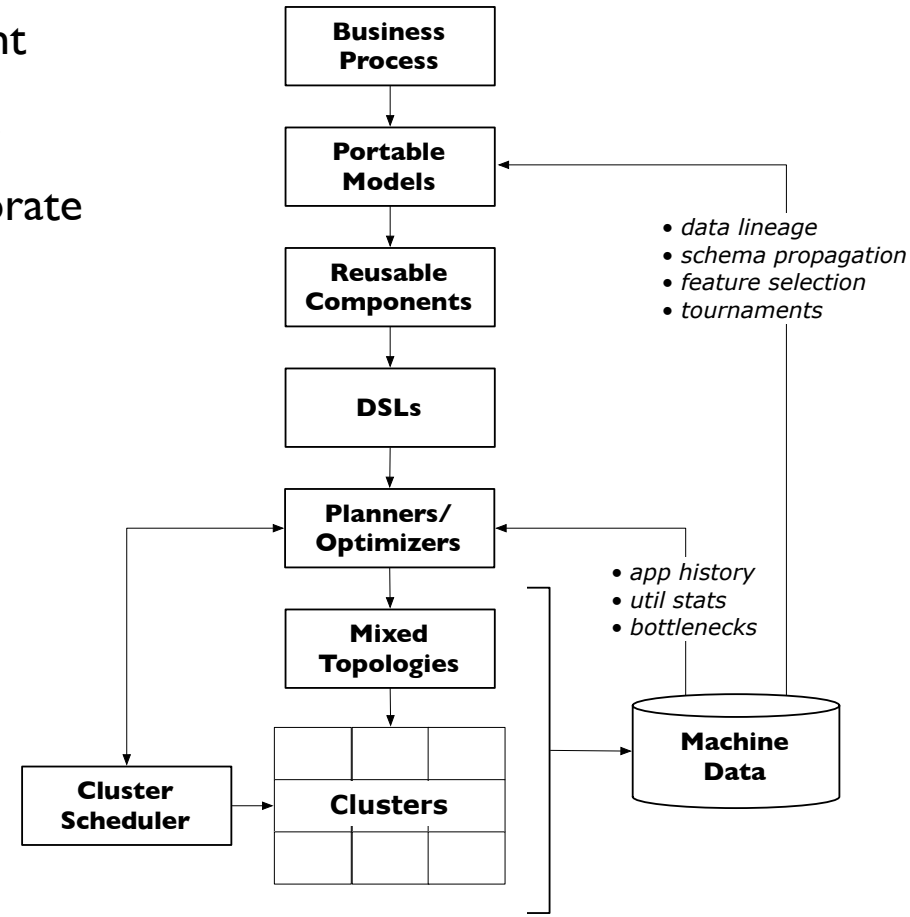


github.com/twitter/summingbird

```
def wordCount[P <: Platform[P]]
  (source: Producer[P, String], store: P#Store[String, Long]) =
    source.flatMap { sentence =>
      toWords(sentence).map(_ -> 1L)
    }.sumByKey(store)
```

Nine Points: a wish list

1. includes people, defines oversight for exceptional data
2. separation of concerns, allows for literate programming
3. multiple abstraction layers for metadata, feedback, and optimization
4. testing: model evaluation, TDD, app deployment
5. future-proof system integration, scale-out, ops
6. visualizing workflows allows people to collaborate through code
7. abstract algebra and functional programming containerize business process
8. blend results from different time scales: batch plus low-latency
9. optimize learners in context, to make model selection potentially a compiler problem



Nine Points: a tentative scorecard

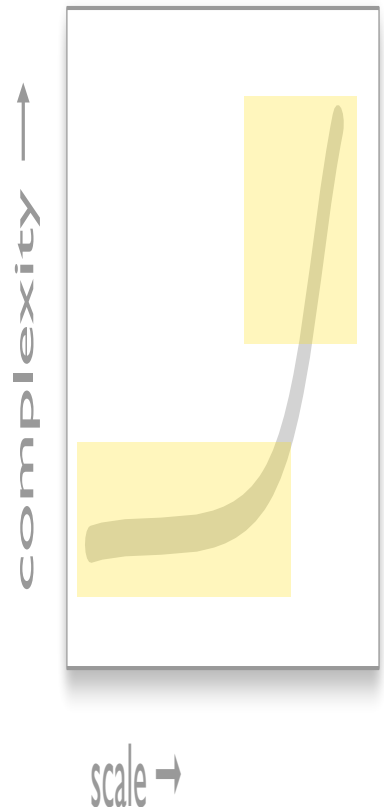
	Spark, MLbase	Oryx	Summing bird	Cascalog	Cascading	KNIME	Py Data	R Markdown	MBrace
includes people, exceptional data									
separation of concerns									
multiple abstraction layers									
testing in depth									
future-proof system integration									
visualize to collab									
can haz monoids									
blends batch + "real-time"									
optimize learners in context									
can haz PMML		✔			✔	✔	✔	✔	

Abstraction Layers and Learning Curves

commercial use of distributed systems may surface issues of managing complexity

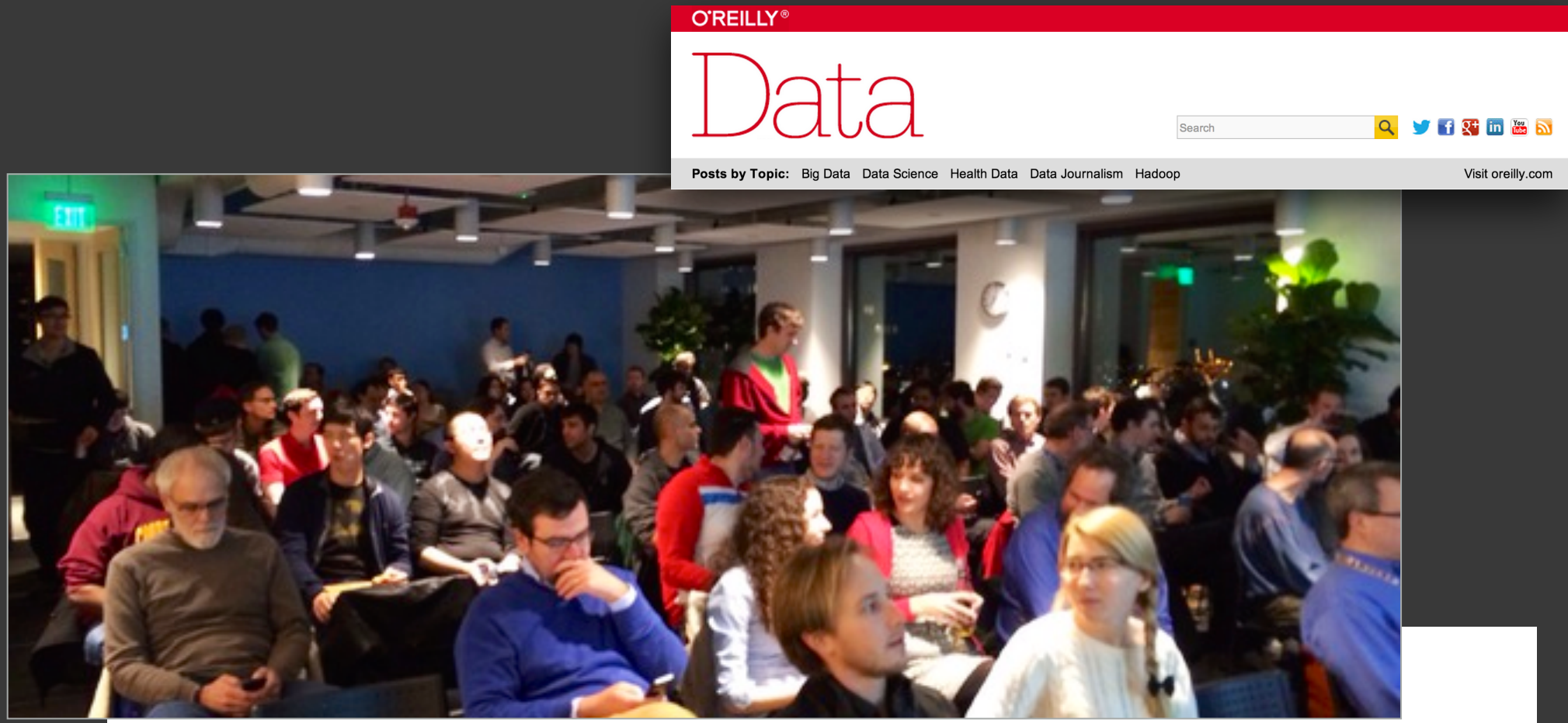
budget for learning curve to mitigate team risks

some orgs practice engineering "conservatism": highly structured process and strictly codified practices… people learn a few things well, then mitigate subsequent learning curve costs?



Notes from the Mystery Machine Bus
Steve Yegge
goo.gl/SeRZa

Throw Your Life a Curve
Whitney Johnson
blogs.hbr.org/johnson/2012/09/throw-your-life-a-curve.html



Data Workflows for Machine Learning

slideshare.net/pacoid/data-workflows-for-machine-learning-33341183

vimeo.com/91794551

oscon.com/oscon2014/public/schedule/detail/34913

Data Workflows

PMML

PMML: an industry standard

- an established XML standard for predictive model markup
- organized by Data Mining Group (DMG), since 1997 **http://dmg.org/**
- members: *IBM*, *SAS*, *Visa*, *FICO*, *Equifax*, *NASA*, *Microstrategy*, *Microsoft*, etc.
- PMML concepts for *metadata*, *ensembles*, etc., translate directly into workflow abstractions

"PMML is the leading standard for statistical and data mining models and supported by over 20 vendors and organizations. With PMML, it is easy to develop a model on one system using one application and deploy the model on another system using another application."

wikipedia.org/wiki/Predictive_Model_Markup_Language

PMML: vendor coverage

PMML: model coverage

- Association Rules: *AssociationModel* element
- Cluster Models: *ClusteringModel* element
- Decision Trees: *TreeModel* element
- Naïve Bayes Classifiers: *NaiveBayesModel* element
- Neural Networks: *NeuralNetwork* element
- Regression: *RegressionModel* and *GeneralRegressionModel* elements
- Rulesets: *RuleSetModel* element
- Sequences: *SequenceModel* element
- Support Vector Machines: *SupportVectorMachineModel* element
- Text Models: *TextModel* element
- Time Series: *TimeSeriesModel* element

ibm.com/developerworks/industry/library/ind-PMML2/

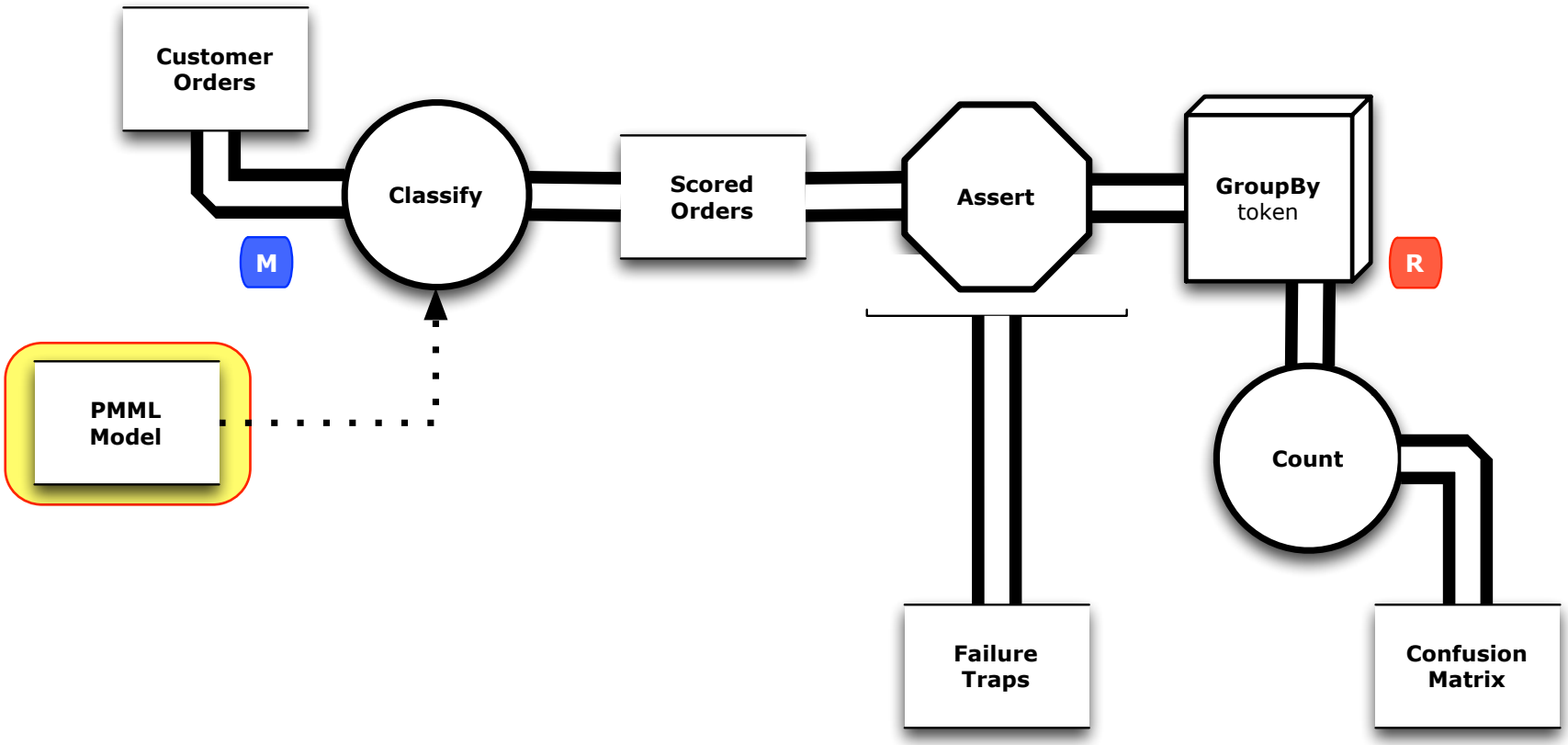
PMML: create a model in R

```
## train a RandomForest model

f <- as.formula("as.factor(label) ~ .")
fit <- randomForest(f, data_train, ntree=50)

## test the model on the holdout test set

print(fit$importance)
print(fit)

predicted <- predict(fit, data)
data$predicted <- predicted
confuse <- table(pred = predicted, true = data[,1])
print(confuse)

## export predicted labels to TSV

write.table(data, file=paste(dat_folder, "sample.tsv", sep="/"),
  quote=FALSE, sep="\t", row.names=FALSE)

## export RF model to PMML

saveXML(pmml(fit), file=paste(dat_folder, "sample.rf.xml", sep="/"))
```

PMML: serialize model as XML


```
<?xml version="1.0"?>
<PMML version="4.0" xmlns="http://www.dmg.org/PMML-4_0"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="http://www.dmg.org/PMML-4_0
 http://www.dmg.org/v4-0/pmml-4-0.xsd">
 <Header copyright="Copyright (c)2012 Concurrent, Inc." description="Random Forest Tree Model">
  <Extension name="user" value="ceteri" extender="Rattle/PMML"/>
  <Application name="Rattle/PMML" version="1.2.30"/>
  <Timestamp>2012-10-22 19:39:28</Timestamp>
 </Header>
 <DataDictionary numberOfFields="4">
  <DataField name="label" optype="categorical" dataType="string">
   <Value value="0"/>
   <Value value="1"/>
  </DataField>
  <DataField name="var0" optype="continuous" dataType="double"/>
  <DataField name="var1" optype="continuous" dataType="double"/>
  <DataField name="var2" optype="continuous" dataType="double"/>
 </DataDictionary>
 <MiningModel modelName="randomForest_Model" functionName="classification">
  <MiningSchema>
   <MiningField name="label" usageType="predicted"/>
   <MiningField name="var0" usageType="active"/>
   <MiningField name="var1" usageType="active"/>
   <MiningField name="var2" usageType="active"/>
  </MiningSchema>
  <Segmentation multipleModelMethod="majorityVote">
   <Segment id="1">
    <True/>
    <TreeModel modelName="randomForest_Model" functionName="classification" algorithmName="randomForest"
splitCharacteristic="binarySplit">
     <MiningSchema>
      <MiningField name="label" usageType="predicted"/>
      <MiningField name="var0" usageType="active"/>
      <MiningField name="var1" usageType="active"/>
      <MiningField name="var2" usageType="active"/>
     </MiningSchema>
...
```

PMML: score model in Pattern



github.com/ceteri/pattern

PMML: score model in Pattern

```
public static void main( String[] args ) throws RuntimeException {
  String inputPath = args[ 0 ];
  String classifyPath = args[ 1 ];
  // set up the config properties
  Properties properties = new Properties();
  AppProps.setApplicationJarClass( properties, Main.class );
  HadoopFlowConnector flowConnector = new HadoopFlowConnector( properties );
   // create source and sink taps
  Tap inputTap = new Hfs( new TextDelimited( true, "\t" ), inputPath );
  Tap classifyTap = new Hfs( new TextDelimited( true, "\t" ), classifyPath );
   // handle command line options
  OptionParser optParser = new OptionParser();
  optParser.accepts( "pmml" ).withRequiredArg();
  OptionSet options = optParser.parse( args );

  // connect the taps, pipes, etc., into a flow
  FlowDef flowDef = FlowDef.flowDef().setName( "classify" )
    .addSource( "input", inputTap )
    .addSink( "classify", classifyTap );

  if( options.hasArgument( "pmml" ) ) {
    String pmmlPath = (String) options.valuesOf( "pmml" ).get( 0 );
    PMMLPlanner pmmlPlanner = new PMMLPlanner()
      .setPMMLInput( new File( pmmlPath ) )
      .retainOnlyActiveIncomingFields()
      .setDefaultPredictedField( new Fields( "predict", Double.class ) );
   flowDef.addAssemblyPlanner( pmmlPlanner );
    }

  // write a DOT file and run the flow
  Flow classifyFlow = flowConnector.connect( flowDef );
  classifyFlow.writeDOT( "dot/classify.dot" );
  classifyFlow.complete();
  }
```

Abstract Algebra

Functional Programming for Big Data

Just Enough Math: *The Business Case for Beyond Calculus*

Advanced math for business use cases, to leverage open source frameworks for Big Data

The kind of math that Twitter and others firms employ for their revenue apps – readily accessible even if you didn't take several years of Calculus

Each morsel of advanced math paired with a concrete business use case, based on an example start-up firm… along with < 30 lines of Python that you can run from your browser

Just Enough Math: *Abstract Algebra*

Unfortunately, math papers tend to be tough to read… Here's an example of a *good* one:

"Introduction to Semigroups and Monoids"
Peter L. Clark @UGA
http://math.uga.edu/~pete/semigroup.pdf

A **group** is a monoid M in which each element has an inverse.[3]

Exercise 2.2: a) Show that a monoid M is a group iff: for each $x \in M$, the maps

$$x\bullet : M \to M,\ y \mapsto xy, \quad \bullet x : M \to M,\ y \mapsto yx$$

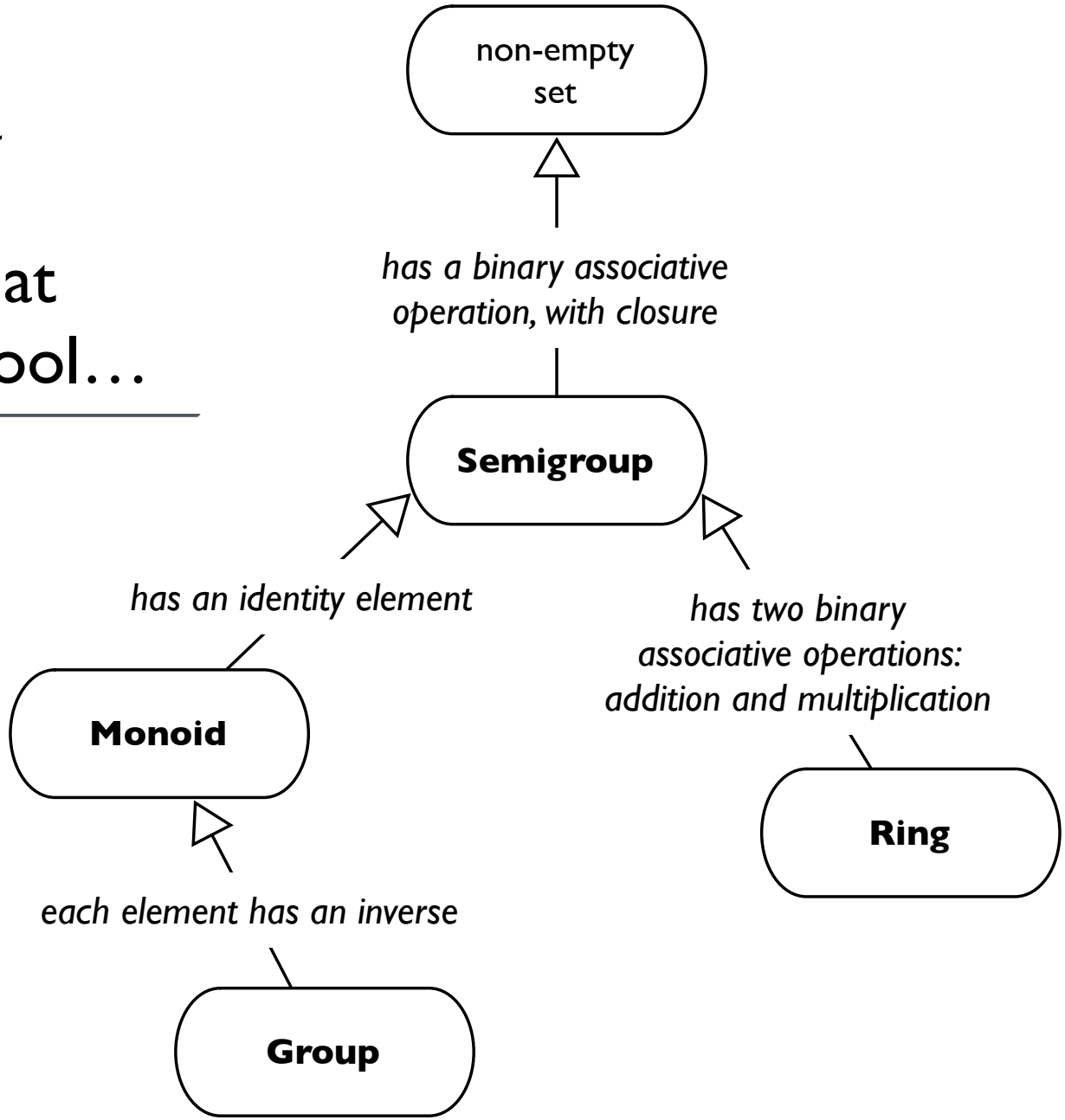
are both bijections.
b) A nontrivial group has no absorbing element.
c) For any monoid M, neither M^e nor M^a is a group.

Exercise 2.3: Show that any group G is isomorphic to its opposite group M^{op}.

The subclass of groups is in many ways simpler and better behaved than the class of all monoids. In this section we explore the following theme: suppose M is a monoid which is not a group: what can we do about it?

Just Enough Math: *Abstract Algebra*

Instead, here's a cheat-sheet we really wished that we'd had in school…



Just Enough Math: *Functional Programming*

John Backus wanted to increase the *parallelism* of programs, moving away from *Von Neumann architecture*

He used lambda calculus to build a relatively "pure" form of *functional programming*

John Backus
acm.org

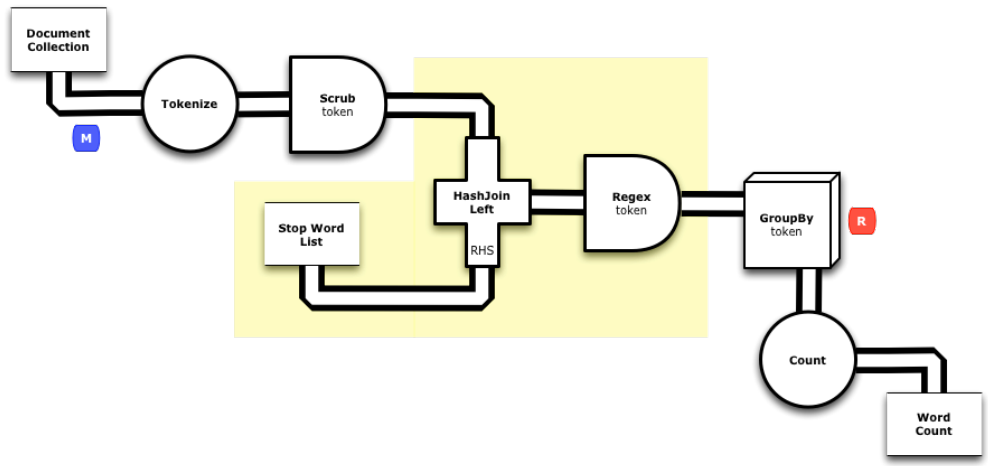
"Can Programming Be Liberated from the von Neumann Style? A Functional Style and Its Algebra of Programs"
ACM Turing Award (1977)
stanford.edu/class/cs242/readings/backus.pdf

λ

Just Enough Math: *Functional Programming*

Chris Wensel created a Java API called *Cascading* (2007) as a way of building *Enterprise data workflows* atop Hadoop

Rather than code directly in MapReduce, developers use some aspects of functional programming inside Java to define data workflows

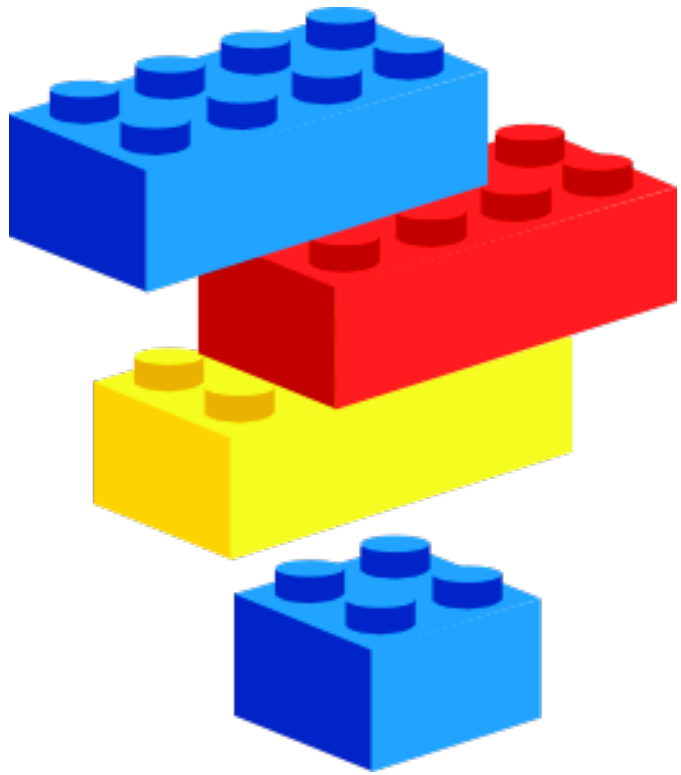


Just Enough Math: *Functional Programming*

Workflows definitions in Cascading use *function composition* to build pipelines, much the same as in Algebra 2…

$$v = g(y) = g(f(x))$$
$$= 2(x + 3) + 1$$

key point: *intermediate values* flowing through those pipelines are well-defined and fit together nicely based on the math, using a computable *schema*



Add ALL the Things:
Abstract Algebra Meets Analytics

infoq.com/presentations/abstract-algebra-analytics

Avi Bryant, *Strange Loop* (2013)

Avi Bryant
@avibryant

- *grouping doesn't matter (associativity)*
- *ordering doesn't matter (commutativity)*
- *zeros get ignored*

In other words, while partitioning data at scale is quite difficult, you can let the math allow your code to be flexible at scale

Just Enough Math: *Performance Bottlenecks*

Algebra for Analytics

speakerdeck.com/johnynek/algebra-for-analytics

Oscar Boykin, *Strata SC* (2014)

- *"Associativity allows parallelism in reducing"* by letting you put the () where you want
- *"Lack of associativity increases latency exponentially"*

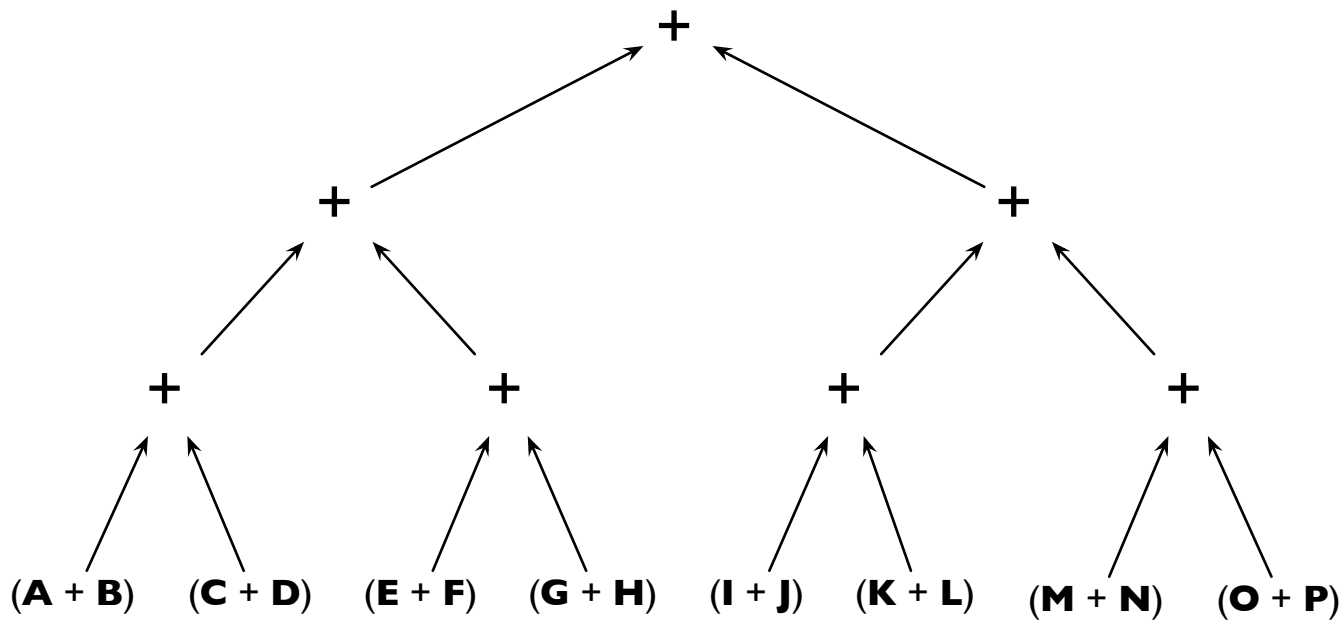
Oscar Boykin
@posco

A Big Idea!

Algebra for Analytics
Oscar Boykin, *Strata SC* (2014)

A + B + C + D + E + F + G + H + I + J + K + L + M + N + O + P

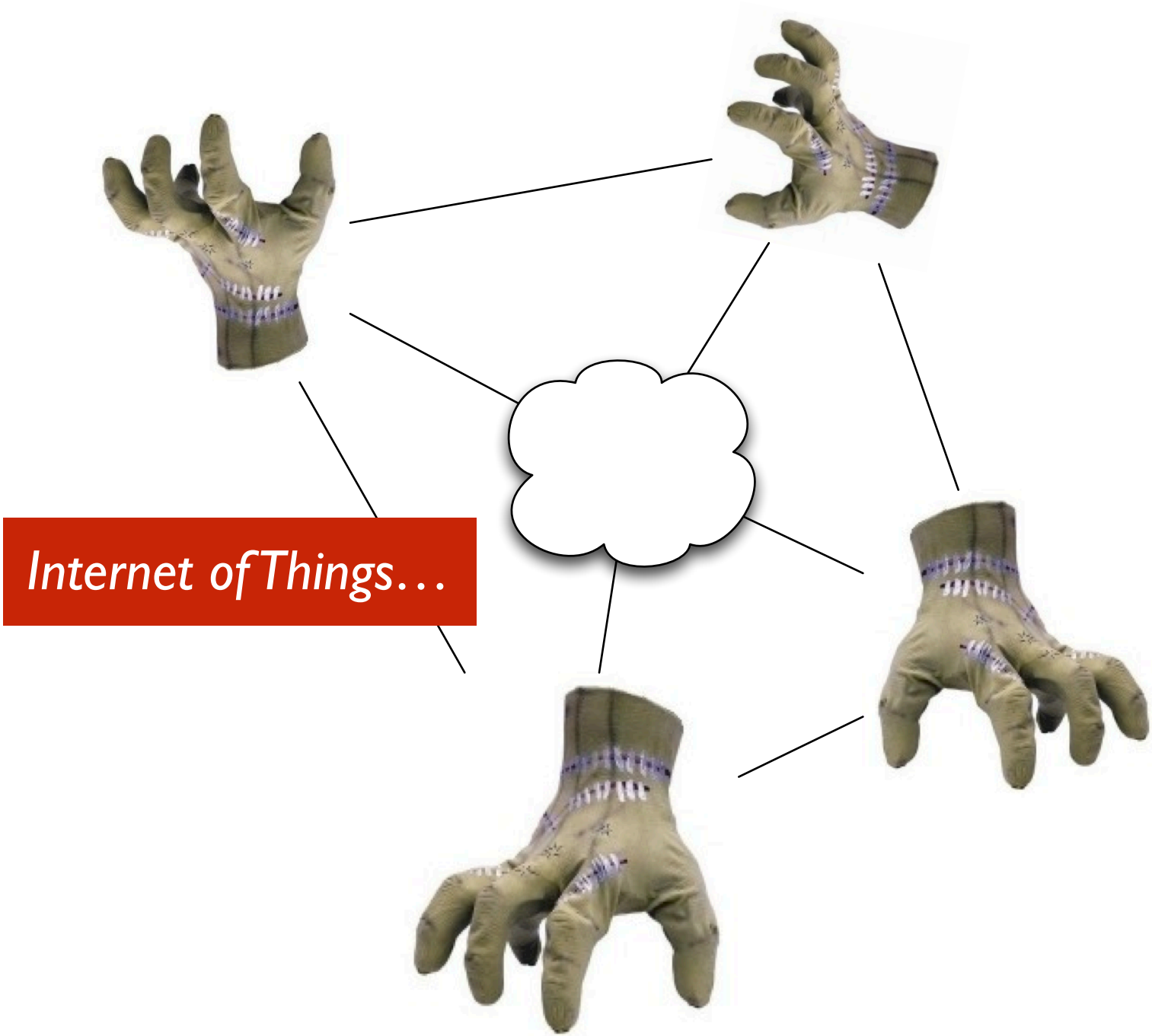
(A + B)
+ C
+ D
+ E
+ F
+ G
+ H
+ I
+ J
+ K
+ L
+ M
+ N
+ O
+ P



latency = (N - 1) = 15

latency = log2(N) = 4

Just Enough Math: *Because IoT !*



Just Enough Math: *Because IoT !*

Big Data? We're just getting started

- ~12 exabytes/day, jet turbines on commercial flights
- Google self-driving cars, ~1 Gb/s per vehicle
- one-meter resolution satellites

consider the implications of data from Jawbone, Nike, etc.

technologyreview.com/...

O'REILLY®

Data

Search

Posts by Topic: Big Data Data Science Health Data Data Journalism Hadoop Visit oreilly.com

Ag+Data

Print

Listen

"Do you want to become a farmer?!" In a sense, yes.

by Paco Nathan | @pacoid | +Paco Nathan | Comments: 2 | April 21, 2014

Tweet 57 +1 6 Like 14 Share 61

Two years ago an informal group met for drinks in downtown Palo Alto: a mix of grad students, investors, and data science experts in Silicon Valley. In the back and forth of our conversation, we took turns describing planned projects. At the time, prominent VC firms were racing headlong into health care ventures. Much of our group seemed pointed in that direction.

In my turn, I mentioned one word: *Agriculture*.

That drew laughter, "You want to become a farmer?!"

In a sense, yes.

Whitepaper: Agricultural Systems + Data Outlook 1Q 2014

strata.oreilly.com/2014/04/agdata.html

goo.gl/OK8RFf

Bayesian Stats

Bayes Theorem

Just Enough Math: *Bayes Theorem*

Statistics is a broadly defined term and also relatively new: before the late 19th century it mostly meant gathering measurements, as in "vital statistics"

Statistics used in predictive modeling comes largely from the 20th century, and emerged contemporarily with cybernetics, lambda calculus, etc.

Oddly, 19th c. *Frequentist* notions of statistics prevailed over an arguably more robust but less accepted 18th c. idea … until **Big Data**

Just Enough Math: *Bayes Theorem*

Thomas Bayes was a mathematician and minister in Britain during the 1700s, elected as a Fellow into the Royal Society in 1742 (for defending Newton)

He postulated about conditional probability in:

An Essay towards solving a Problem in the Doctrine of Chances
Thomas Bayes
Phil. Trans. (1763)
rstl.royalsocietypublishing.org/content/53/370

www.stat.ucla.edu/history/essay.pdf

Given the number of times ion which an unknown event has happende and failed: Required the chance that the probability of its happening in a single trial lies somewhere between any two degrees of probability that can be named.

Just Enough Math: *Bayes Theorem*

Bayes Theorem lets us update the probability of hypothesis H in light of evidence in data D

The Posterior (current decision)	The Evidence (the data)	The Prior (past decisions)

$$P(H|D) = \frac{P(D|H)}{P(D)} \cdot P(H)$$

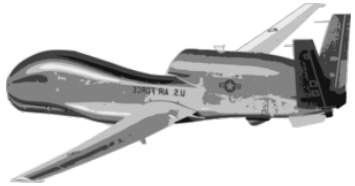
posterior $P(H|D)$ – we want to calculate the probability that hypothesis H is true given the evidence in data D

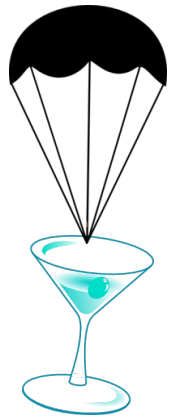
likelihood $P(D|H)$ – the probability of obtaining the evidence in data D if the hypothesis H were true

normalizing constant $P(D)$ – the marginal probability of the evidence in data D under any hypothesis

prior $P(H)$ – the probability of hypothesis H being true prior to gathering the evidence in data D

Just Enough Math: *Bayes Theorem*





Foobartendr.io gathers customer feedback about **Top Bartendrs** using upvote and downvote buttons

bartendr	*up vote*	*down vote*	*frequentist estimate*	*bayesian estimate*
Deepali	45	5	`45/(5+45) = 0.90`	`46/(6+46) = 0.88`
Kirill	2	0	`2/(0+2) = 1.00`	`3/(1+3) = 0.75`

Bayesian point estimate with a *Beta*(1, 1) prior adds 1 upvote and 1 downvote as *shrinkage* to normalize each bartendr score

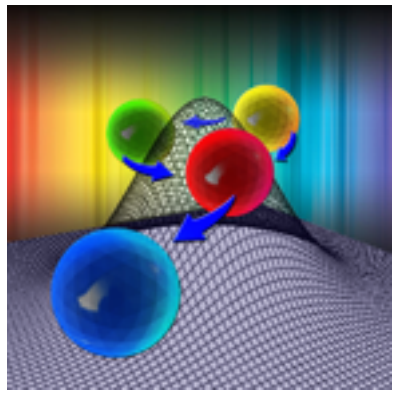
Just Enough Math: *Bayes Theorem*

Making judgements in isolation is a luxury enjoyed by insurance actuaries, not business executives and entrepreneurs. "Stats 101" that you probably took catered to the former, not the later.

Bayes Theorem is about developing credence that combines the benefits of new evidence (your Big Data) with the history of prior decisions (your business context), while providing mechanisms for judgements to be refined, iterating on crucial decisions.

Just Enough Math: *Probabilistic Programming*

Considering the notion of data workflows in the context of Bayesian approaches, algorithmic modeling, ensembles, etc., check out these resources (great authors) regarding *probabilistic programming*:



PPAML @ darpa.mil

Probabilistic Programming: Why, What, How, When
Beau Cronin
Strata SC (2014)
speakerdeck.com/beaucronin/probabilistic-programming-strata-santa-clara-2014

Why Probabilistic Programming Matters
Rob Zinkov
Convex Optimized (2012-06-27)
zinkov.com/posts/2012-06-27-why-prob-programming-matters/

Linear Algebra

Eigensomethingorother

Just Enough Math: *Linear Systems*

Recall from *system of equations*, if we have **N** variables and **N** equations, we can solve for the variables

Linear algebra make this a bit simpler, with the problem can be stated as one equation $\mathbf{Ax} = \mathbf{y}$, where matrix $\mathbf{A}$ and vector $\mathbf{y}$ are:

$$\mathbf{A} = \begin{pmatrix} 3 & 9 \\ 4 & 8 \end{pmatrix}$$

$$\mathbf{y} = \begin{bmatrix} 5 \\ 12 \end{bmatrix}$$

Just Enough Math: *Eigensomethingorother*

Suppose we have a square matrix $\mathbf{A}$,
a non-zero vector $\mathbf{x}$, and some scalar λ,
such that:

$$\mathbf{Ax} = \lambda\mathbf{x}$$

The vector $\mathbf{x}$ is called an *eigenvector* of $\mathbf{A}$

The scalar λ is called an *eigenvalue* of $\mathbf{A}$

Multiple solutions may exist for λ and $\mathbf{x}$,
but we'll get to that later

Just Enough Math: *Eigensomethingorother*

Example use case – *face recognition*

decompose images into "building blocks" of facial features

Low-dimensional procedure for the characterization of human faces
Sirovich, Kirby (1987)
opticsinfobase.org

Just Enough Math: *Eigensomethingorother*

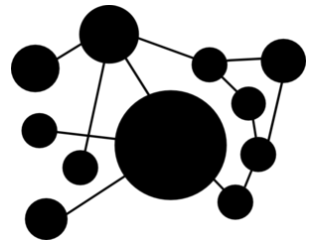
Example use case – *graphs*

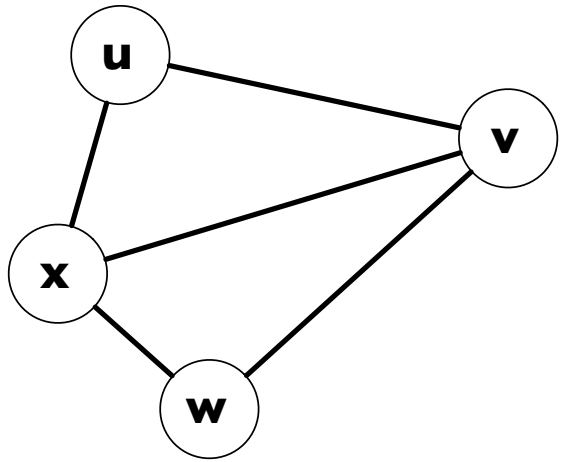
analyze the structure of large graphs

gephi.org

Just Enough Math: *Algebraic Graph Theory*

Suppose we have a graph as shown below:



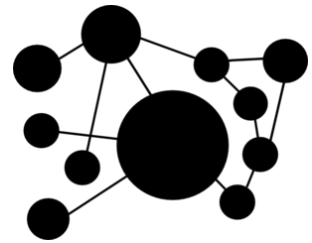


We call **x** a *vertex* (sometimes called a *node*)

An *edge* (sometimes called an *arc*) is any line connecting two vertices

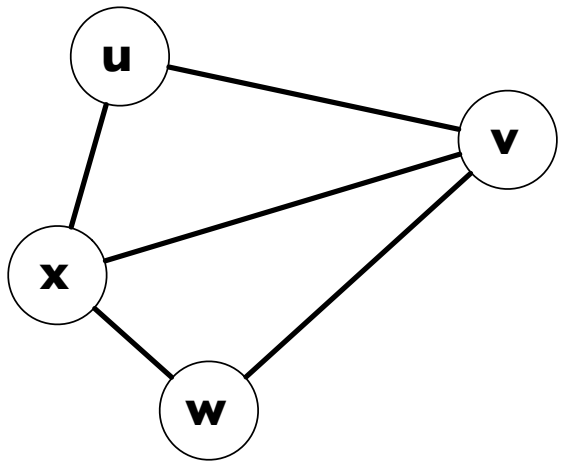
Real-world data is full of graphs!

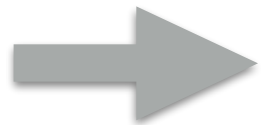
Just Enough Math: *Algebraic Graph Theory*



We can represent this kind of graph as an *adjacency matrix*:

- label the rows and columns based on the vertices
- entries get a 1 if an edge connects the corresponding vertices, or 0 otherwise

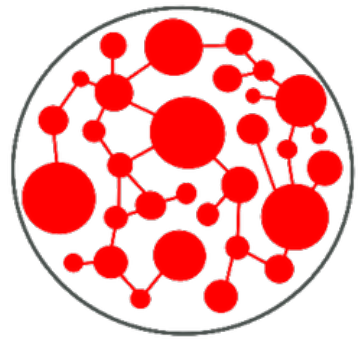




	u	v	w	x
u	0	1	0	1
v	1	0	1	1
w	0	1	0	1
x	1	1	1	0

Just Enough Math: *Graph Computing*

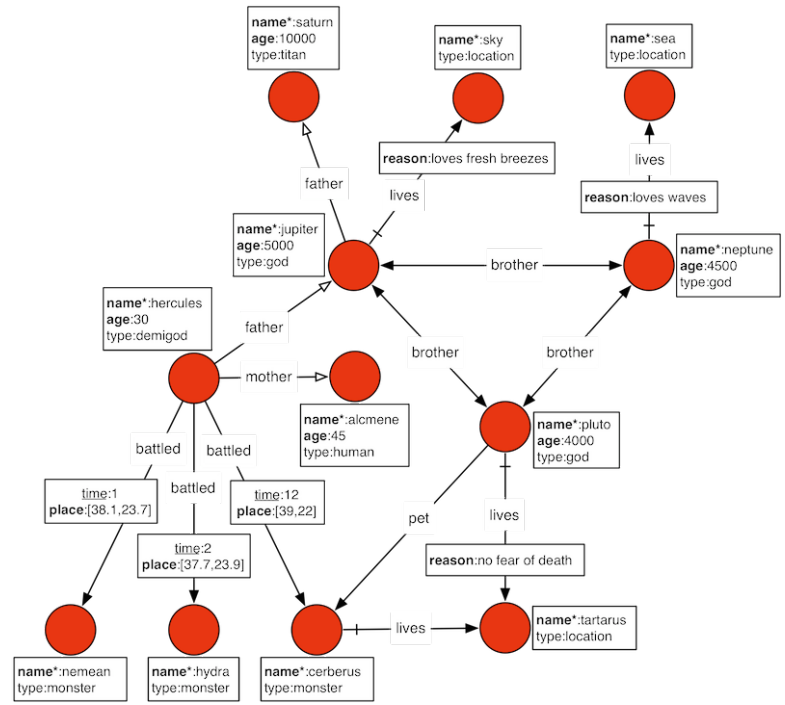
Matthias Bröcheler, Marko Rodriguez
@ThinkAurelius are authors of *Titan*,
a robust, feature rich, open source graph
query engine running atop Cassandra,
HBase, BerkeleyDB, etc.



Check out their excellent
intros to graph theory in
practice at scale…

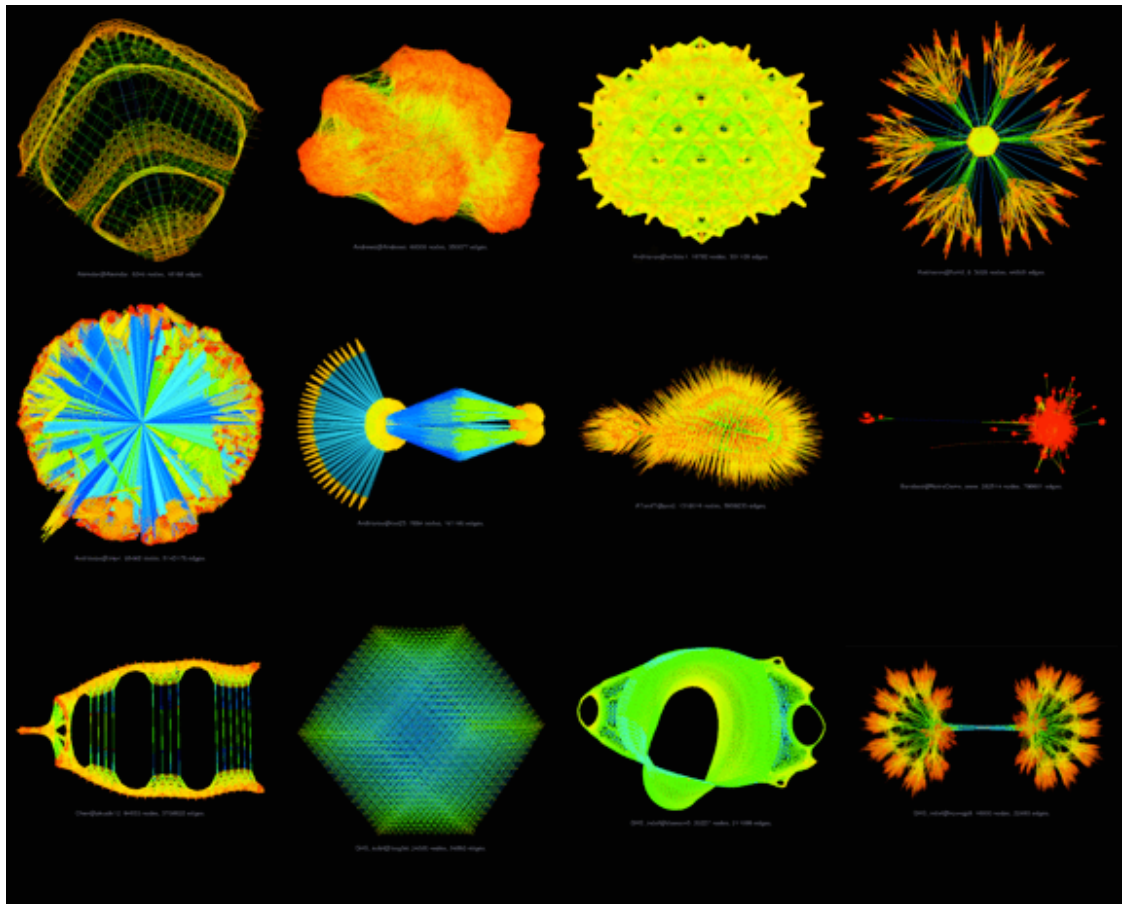
markorodriguez.com/2013/01/09/on-graph-computing/

github.com/thinkaurelius/titan/wiki/Getting-Started



Sparse Matrix Collection… for when you **really** need a wide variety of sparse matrix examples, e.g., to evaluate new ML algorithms

University of Florida Sparse Matrix Collection
cise.ufl.edu/research/sparse/matrices/



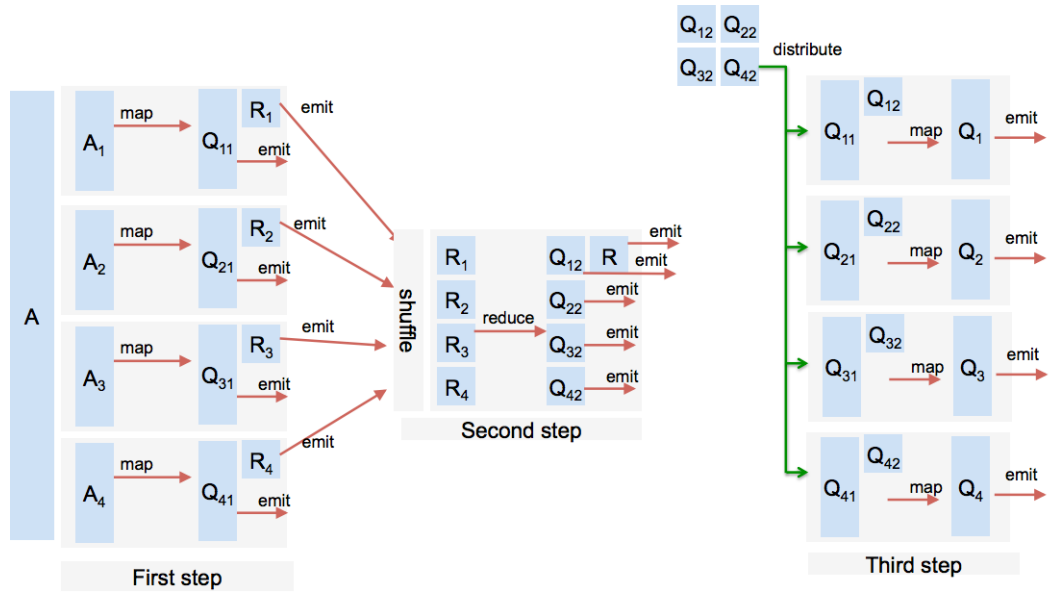
Just Enough Math: *Matrix Factorization*

MapReduce Streaming TSQR

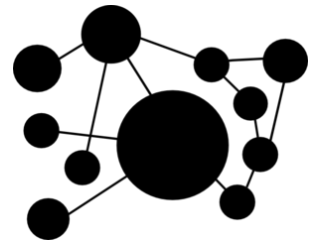
github.com/arbenson/mrtsqr

Hadoop streaming for QR factorization, mostly written in Python with some C++

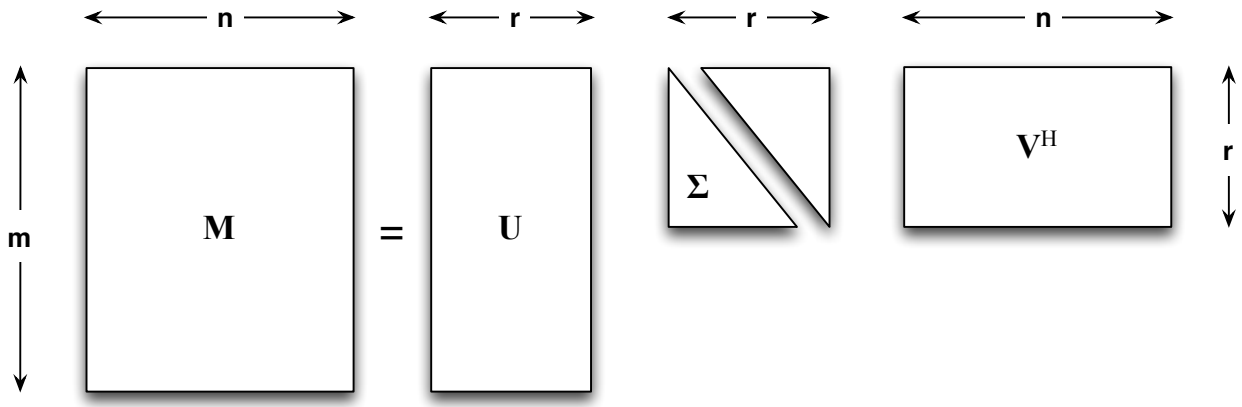
Direct QR factorizations for tall-and-skinny matrices in MapReduce architectures
Austin Benson, David Gleich, James Demmel (2013)
arxiv.org/abs/1301.1071



Just Enough Math: *Matrix Factorization*

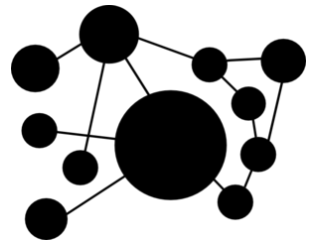


Next, consider a matrix factorization called *Singular Value Decomposition*, based on the form $\mathbf{M} = \mathbf{U\Sigma V}^{\mathrm{H}}$

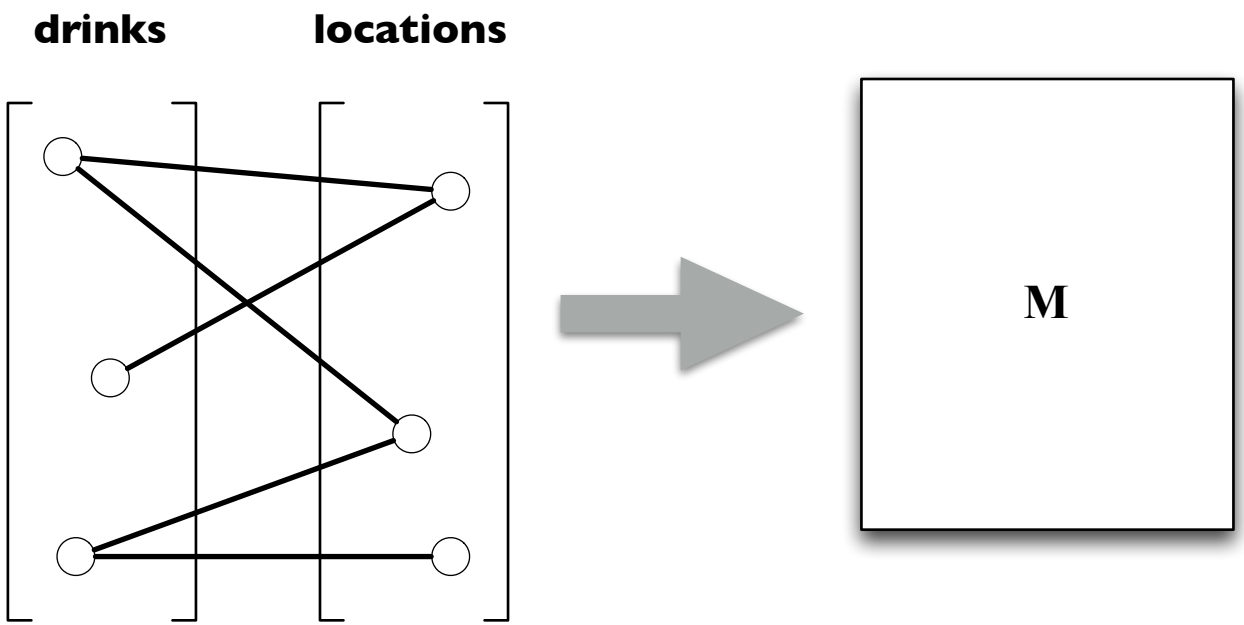


The idea is to reduce a high dimensional, highly variable set of data points down to a lower dimensional space, exposing the structure in the original data more clearly

Just Enough Math: *Matrix Factorization*



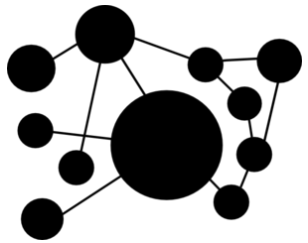
For now, it's more interesting to consider rows and columns of $\mathbf{M}$ representing the set elements of a *bipartite graph*:



Say we put drinks on one side, and order locations on the other side…

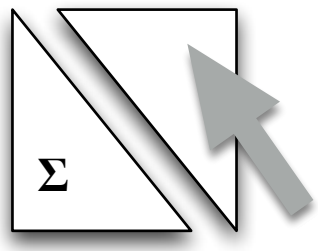
Just Enough Math: *Matrix Factorization*

We identify where the most variation is, then approximate the original data using less dimensions



In other words, we can take a ginormous graph (represented as a matrix), one that is too large to store conveniently or use for recommendations…

Then prune the smallest singular values of Σ and the SVD scales down – massively reduced data, with essential structure preserved

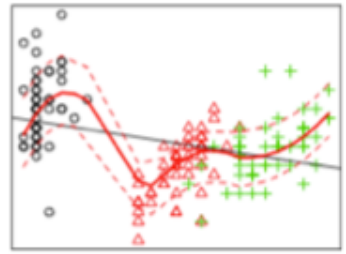


A Big Idea!

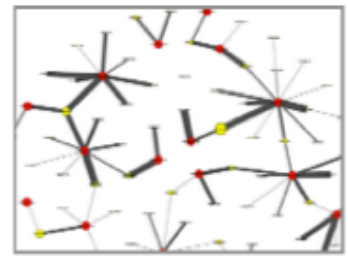
a. represent some data set as a *graph*
b. discover (or validate) insights using a *graph query* engine, e.g., Titan
c. transform the graph to a *sparse matrix*
d. run *QR factorization* on a cluster (Hadoop? Spark?) to build an *SVD*
e. adjust *singular values* on the Σ diagonal to reduce the graph size dramatically
f. expose the *structure* in the original data more clearly

A Big Idea!

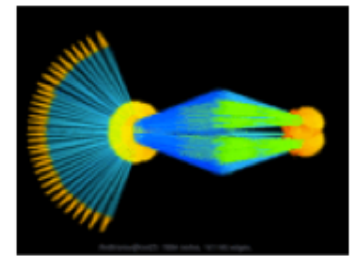
starting with real-world data ⇒



leverage graph queries for representation ⇒



convert to sparse matrix, leverage FP and adv math ⇒



achieve high-ROI parallelism at scale, mostly about optimization ⇒

Sidebar: *Pedro Domingos, generalist*

The categorization of *machine learning* algorithms as a subset of **optimization**:

- **representation**: classifier represented in some formal language that computers can handle
- **evaluation**: evaluation function needed to distinguish good classifiers from bad ones
- **optimization**: searching among classifiers in the language for the highest-scoring one

Pedro Domingos
washington.edu

A Few Useful Things to Know about Machine Learning
Pedro Domingos
U Washington (2012)
homes.cs.washington.edu/~pedrod/papers/cacm12.pdf

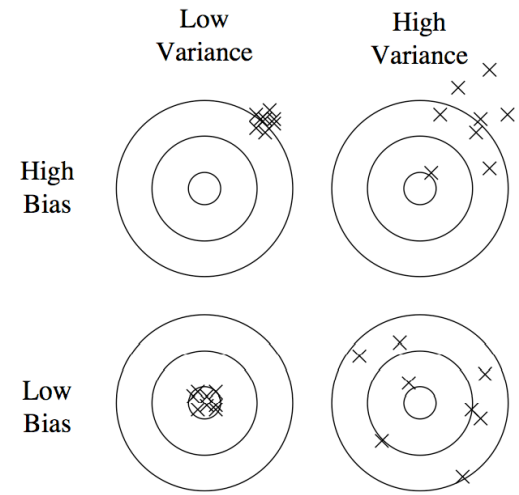
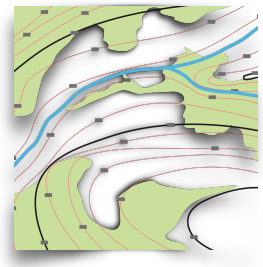


Figure 1: Bias and variance in dart-throwing.

Optimization

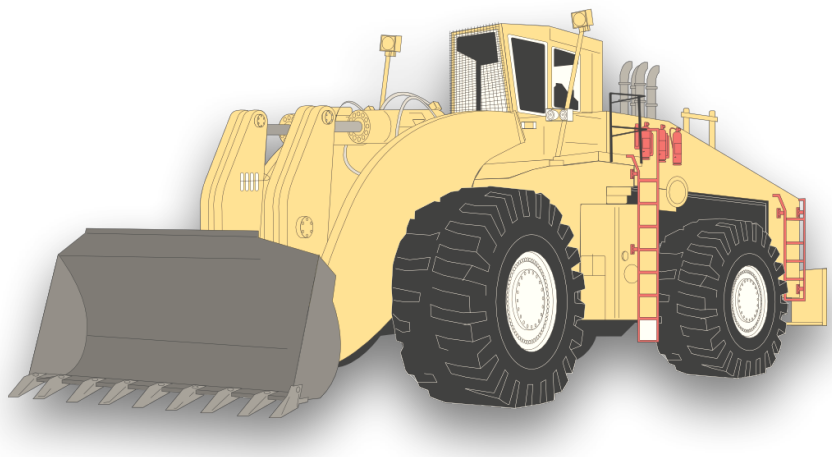
Outlook Ahead

Just Enough Math: *Optimization*

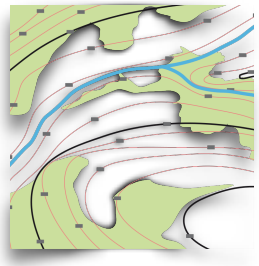


When the *Heavy Machinery and Vehicles* sector begins to invest in machine learning, cluster computing, etc., at R&D levels comparable with Twitter, they won't be building a social network…

They will be optimizing business lines worth billions in durable goods and industrial plant, optimizing for *supply chain*, *pricing strategies*, *maintenance schedules*, etc.



Just Enough Math: *Optimization*



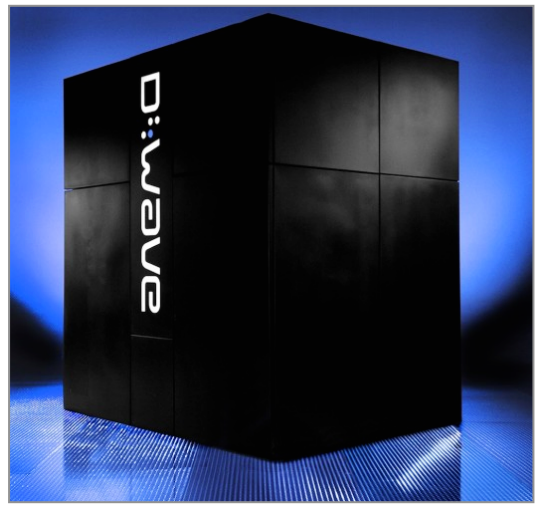
Pulling cluster traces from servers in just about any datacenter, those resources likely get spent on:

- **moving data** between different servers
  (what others think your Data Scientists do)
- **cleaning data** prior to use, perhaps manually
  (what your Data Scientists really do)
- something akin to **stochastic gradient descent**
  (what your Data Scientists think they do)
- highly-available **request/response** services
  (what your Data Scientists should think about doing)

A Big Idea!

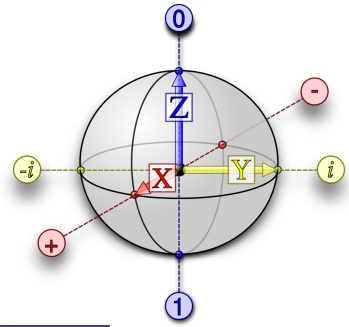
One advantage of *quantum algorithms* is to run large vector quantization problems in constant time… Rework high-ROI apps to leverage lots of ML and large clusters, then slash datacenter costs exponentially – a potentially dramatic game-changer

Fast quantum algorithm for numerical gradient estimation
Stephen P. Jordan
Phys. Rev. Lett. 95, 050501 (2005)
arxiv.org/abs/quant-ph/0405146



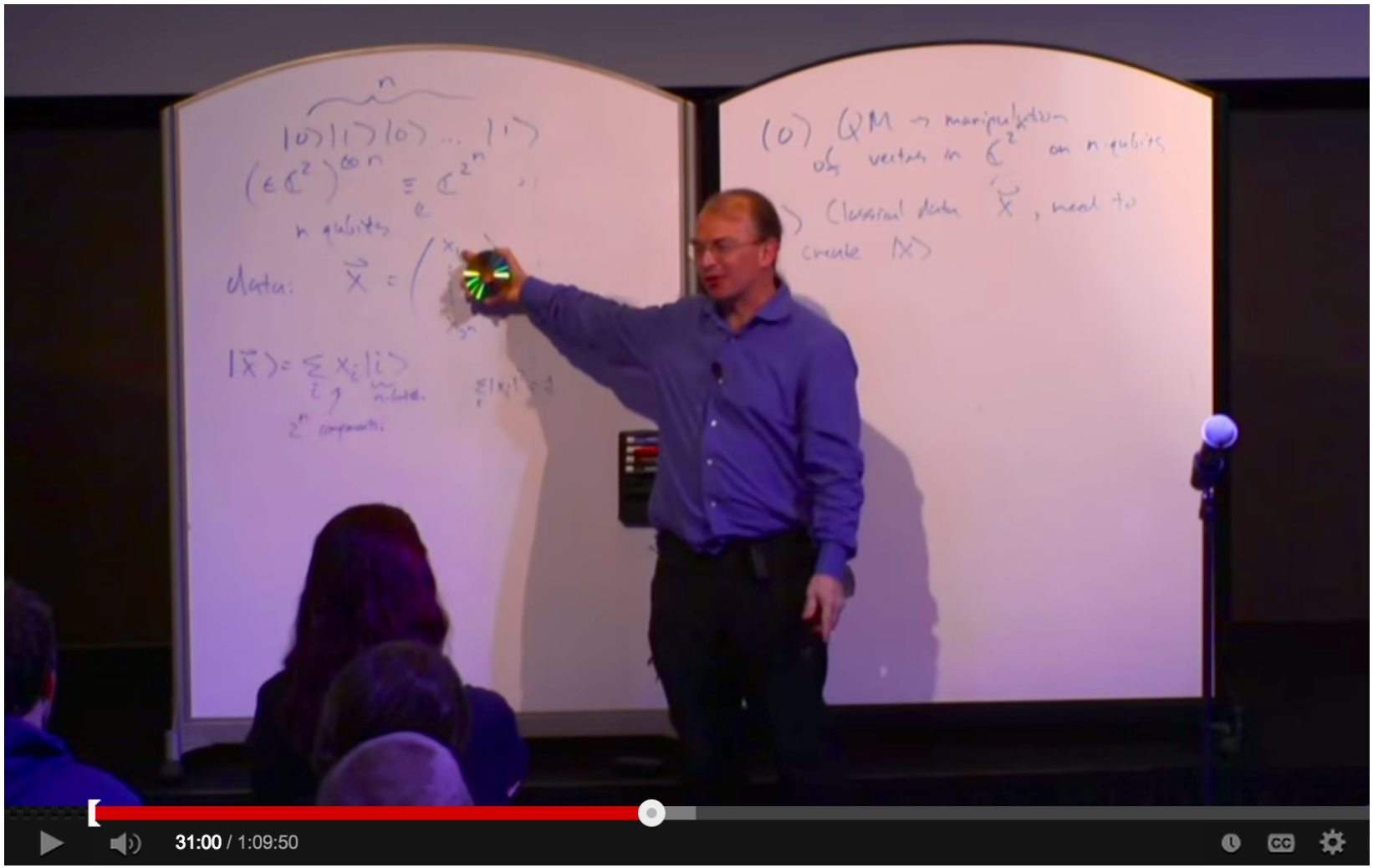
dwavesys.com

Just Enough Math: *Quantum Machine Learning*

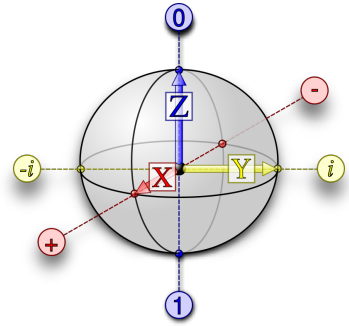


Seth Lloyd @MIT

youtu.be/wkBPp9UovVU



Just Enough Math: *Google Plays Minecraft*



How likely is that scenario, how far off? *Google Quantum AI team* built a Minecraft mod called *qCraft* intend for identifying non-linear thinkers adept at quantum

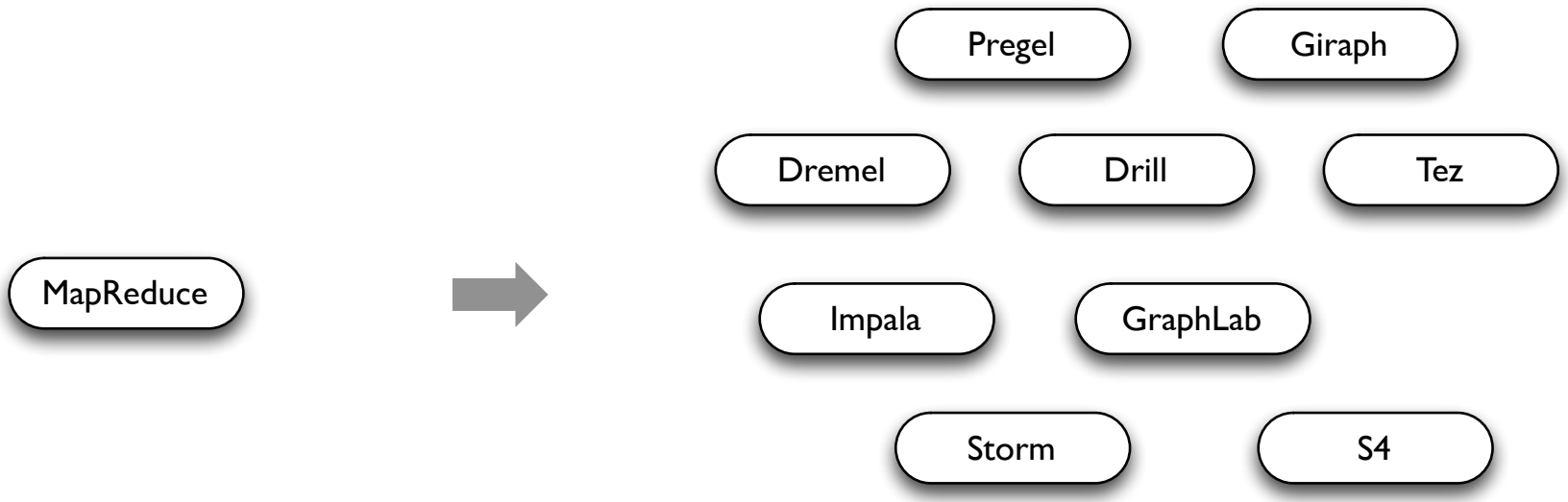
Those with kiddos understand: 10-y.o.'s play Minecraft… some subset of avid MC players now may be Google AI interns in less than a decade

qCraft: Quantum Physics In Minecraft
plus.google.com/u/1/+QuantumAILab/posts/grMbaaDGChH

Summary

Integrate The Pieces

Summary: *Specialized Topologies vs. General Engines*



General Batch Processing

Specialized Systems:
iterative, interactive, streaming, graph, etc.

Summary: *Apache Spark – an update on MapReduce*

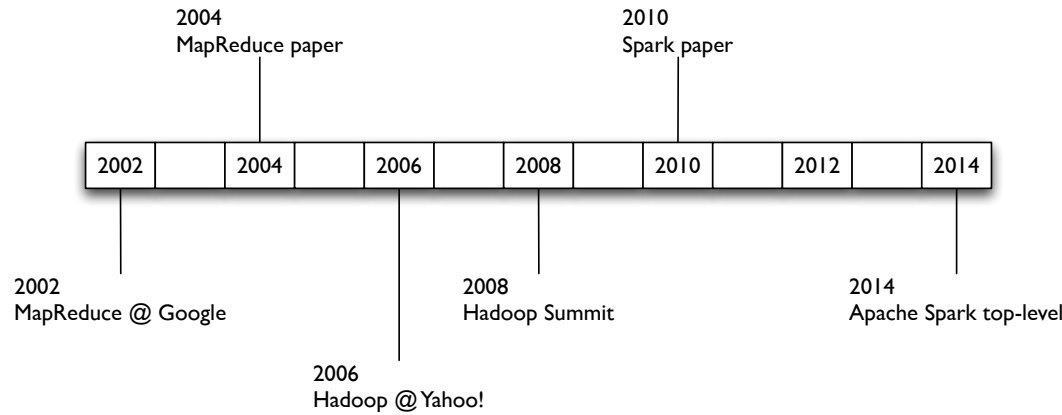
Unlike the various specialized systems, Spark's goal was to *generalize* MapReduce to support new apps within same engine

Two reasonably small additions are enough to express the previous models:

- *fast data sharing*
- *general DAGs*

This allows for an approach which is more efficient for the engine, and much simpler for the end users

Summary: *Apache Spark – an update on MapReduce*



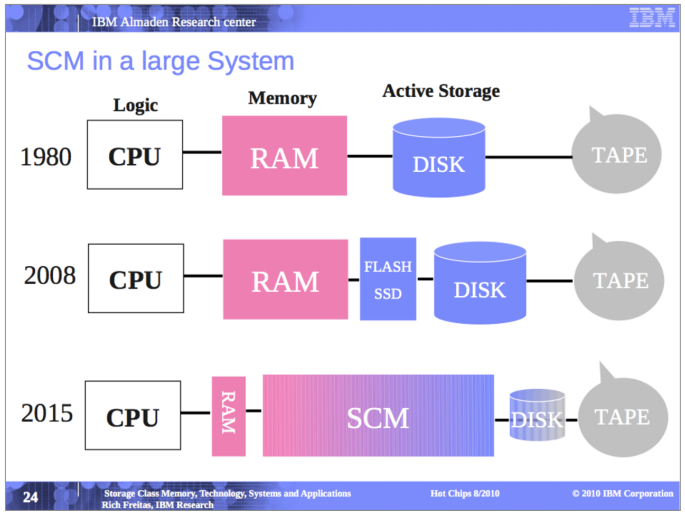
Spark: Cluster Computing with Working Sets
Matei Zaharia, Mosharaf Chowdhury,
Michael J. Franklin, Scott Shenker, Ion Stoica
USENIX HotCloud (2010)
people.csail.mit.edu/matei/papers/2010/hotcloud_spark.pdf

Resilient Distributed Datasets: A Fault-Tolerant Abstraction for In-Memory Cluster Computing
Matei Zaharia, Mosharaf Chowdhury, Tathagata Das,
Ankur Dave, Justin Ma, Murphy McCauley,
Michael J. Franklin, Scott Shenker, Ion Stoica
NSDI (2012)
usenix.org/system/files/conference/nsdi12/nsdi12-final138.pdf

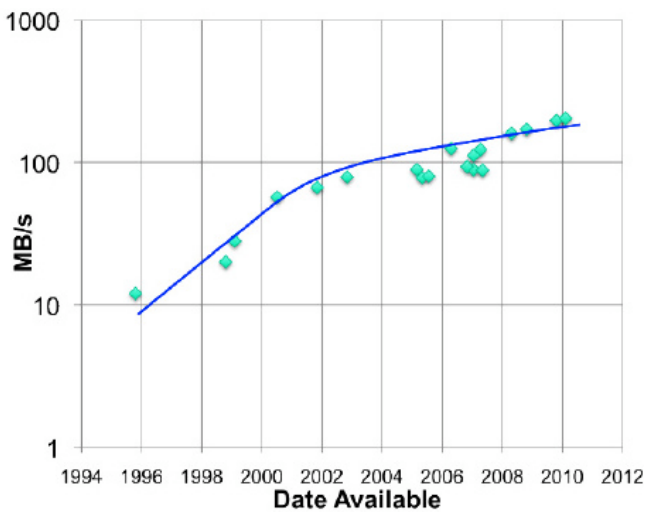
Summary: *Apache Spark, estimating Pi*

```
import scala.math.random
import org.apache.spark._

/** computes an approximation to pi */
object SparkPi {
  def main(args: Array[String]) {
   val sc = new SparkContext(args(0), "SparkPi",
      System.getenv("SPARK_HOME"), SparkContext.jarOfClass(this.getClass))
    val slices = if (args.length > 1) args(1).toInt else 2
    val n = 100000 * slices
    val count = sc.parallelize(1 to n, slices).map { i =>
      val x = random * 2 - 1
      val y = random * 2 - 1
      if (x*x + y*y < 1) 1 else 0
    }.reduce(_ + _)

    println("Pi is roughly " + 4.0 * count / n)
    sc.stop()
  }
```

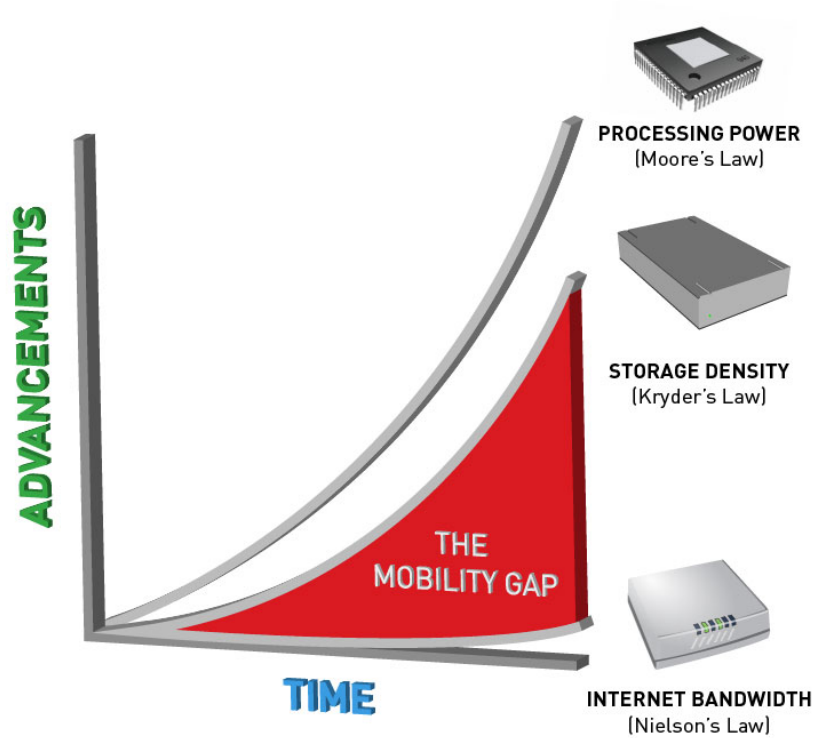
Summary: *Datacenter Economics, since 2002*



Rich Freitas, **IBM Research**



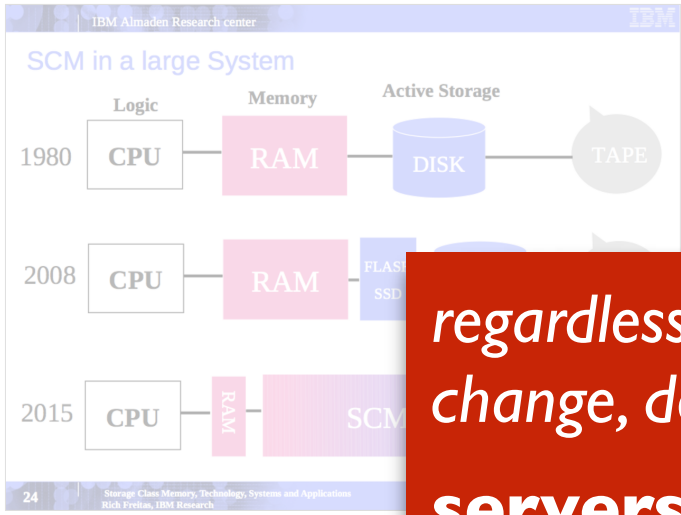
storagenewsletter.com/rubriques/hard-disk-drives/hdd-technology-trends-ibm/



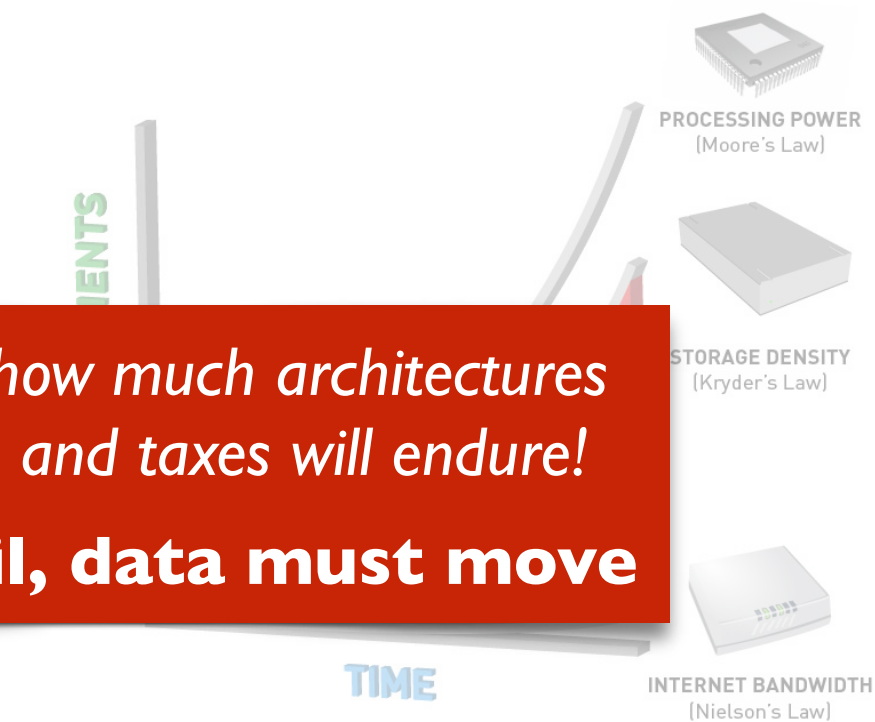
pistoncloud.com/2013/04/storage-and-the-mobility-gap/

meanwhile, spinny disks haven't changed all that much…

Summary: *Datacenter Economics, since 2002*



Rich Freitas



regardless of how much architectures change, death and taxes will endure!

servers fail, data must move

pistoncloud.com/2013/04/storage-and-the-mobility-gap/

storagenewsletter.com/rubriques/hard-disk-drives/hdd-technology-trends-ibm/

meanwhile, spinny disks haven't changed all that much…

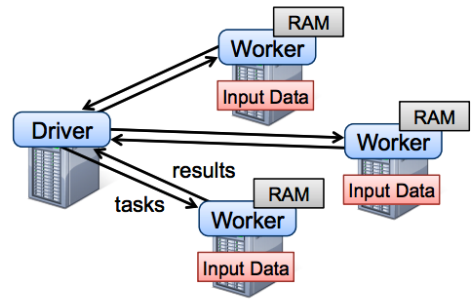
Summary: *Integrate The Pieces…*

unified platform for big data analytics: batch, streaming, interactive, graph, ML, SQL, etc.

The State of Spark, and Where We're Going Next
Matei Zaharia
Spark Summit (2013)
youtu.be/nU6vO2EJAb4

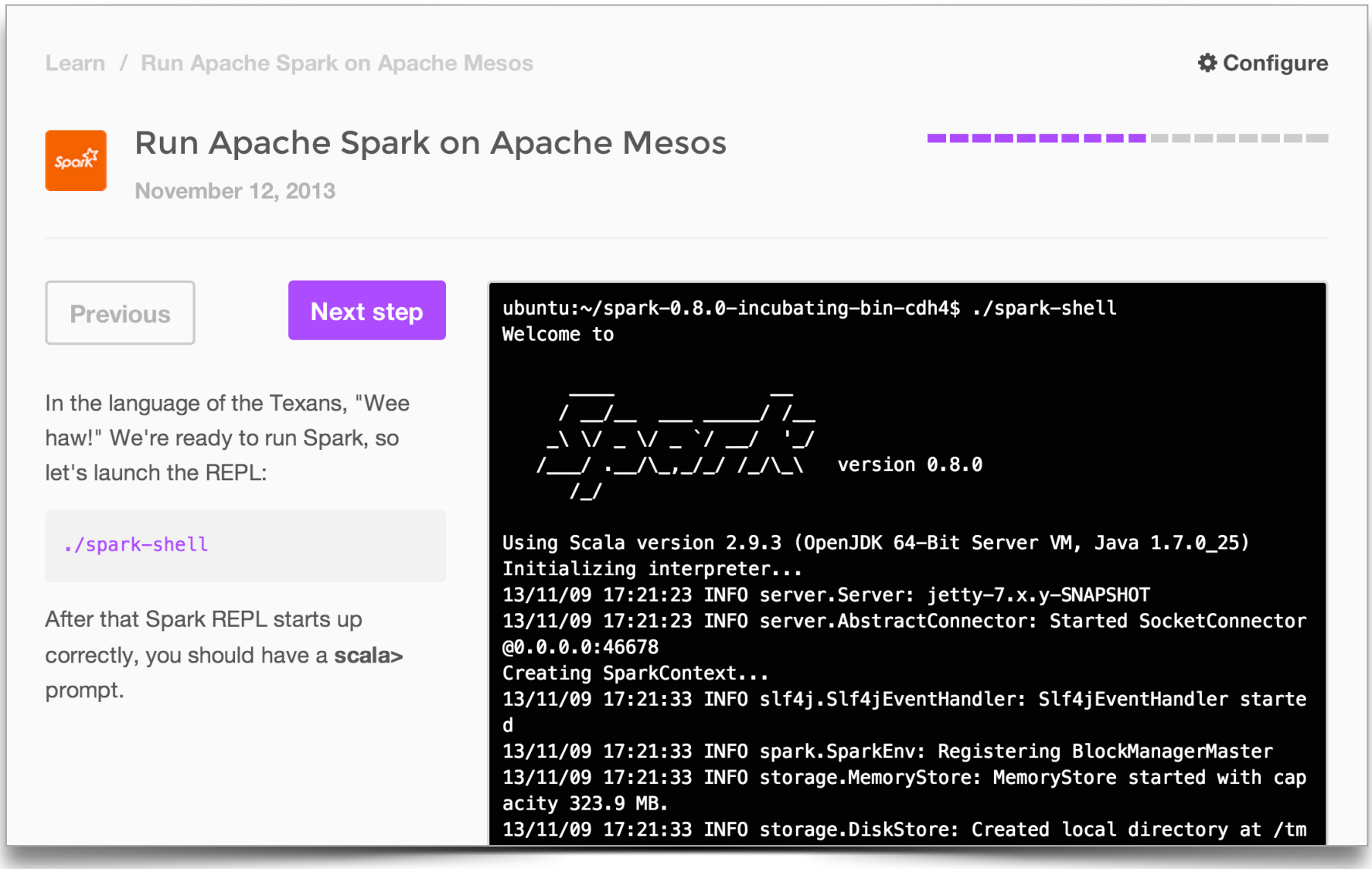
Spark SQL: Manipulating Structured Data Using Spark
Michael Armbrust, Reynold Xin
databricks.com/blog/2014/03/26/Spark-SQL-manipulating-structured-data-using-Spark.html

Beyond Word Count:
Productionalizing Spark Streaming
Ryan Weald
spark-summit.org/talk/weald-beyond-word-count-productionalizing-spark-streaming/



Summary: *Integrate The Pieces…*

mesosphere.io/learn/run-spark-on-mesos/



Big Data, really?

Probably a few orders of magnitude more data than the entirety of what Facebook has today – *each day, per use case*

We'll need all that functional programming, machine learning, datacenter computing, etc., just to handle the data rates

Along with the caveat that the skills and tech that served well for ad-tech may not apply so well for the business contexts ahead

It's time to get busy with the math

Calendar:

Spark Summit
SF, Jun 30
spark-summit.org/2014

OSCON
PDX, Jul 20
oscon.com/oscon2014/

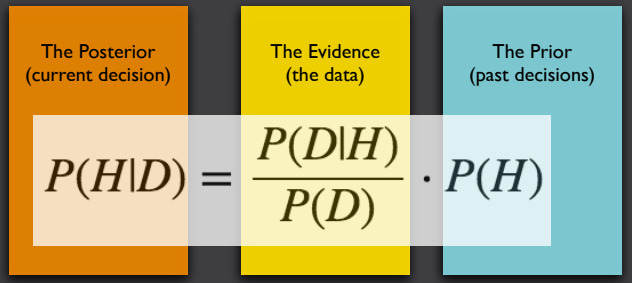
#MesosCon
Chicago, Aug 21
events.linuxfoundation.org/events/mesoscon

Strata NYC + Hadoop World
NYC, Oct 15
strataconf.com/stratany2014

New Book:

Just Enough Math

with Allen Day @MapR Asia



advanced math for business people, to leverage open source for Big Data

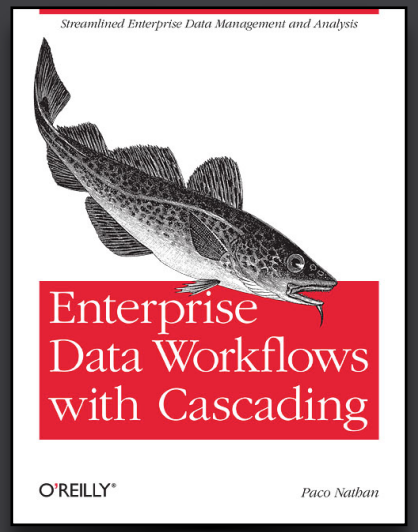
galleys: July 2014 @ OSCON

oscon.com/oscon2014/public/schedule/detail/34873

Enterprise Data Workflows with Cascading

O'Reilly (2013)

shop.oreilly.com/product/0636920028536.do



monthly newsletter for updates, events, conference summaries, etc.:

liber118.com/pxn/