@joerg_schad

Myriad, Spark, Cassandra, and Friends

Big Data Powered by Mesos

DC/OS

Apache Big Data Europe

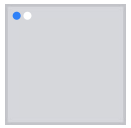
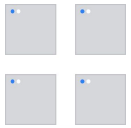
MESOSPHERE

Jörg Schad

Distributed Systems Engineer

@joerg_schad

Evolution of Applications

UNIT OF INTERACTION	PARTITION (LPAR)	SERVER	VIRTUAL MACHINE	DATACENTER
				
	MAINFRAME	**PHYSICAL (x86)**	**VIRTUAL**	**HYPERSCALE**

HYPERSCALE MEANS VOLUME *AND* VELOCITY

Days *Hours* *Minutes* *Seconds* *Microseconds*

Batch | **Micro-Batch** | **Event Processing**

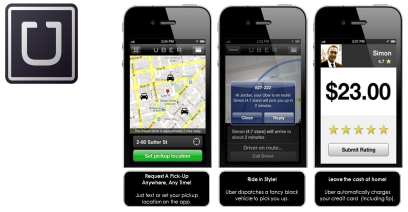
Reports what has happened using descriptive analytics

Solves problems using predictive and prescriptive analytics

Billing, Chargeback

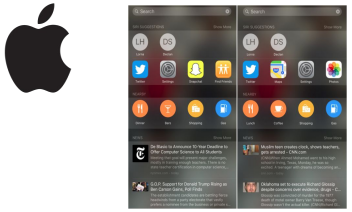
Product recommendations

Real-time Pricing and Routing

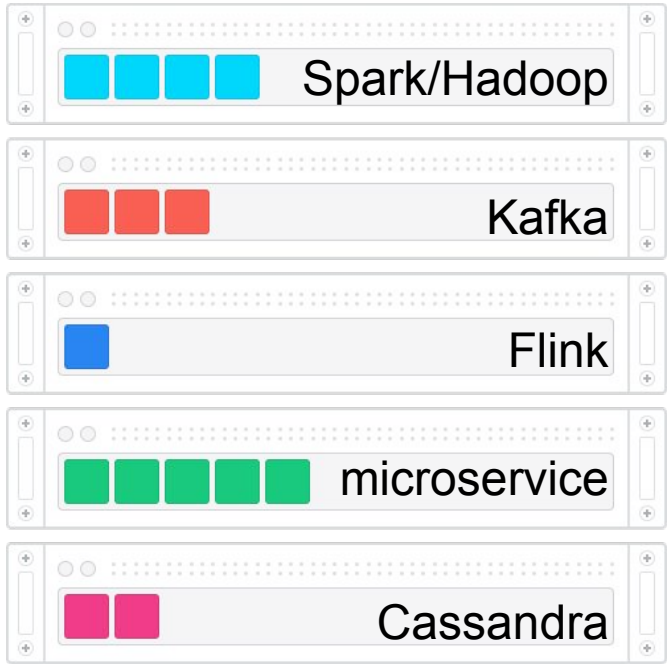


Real-time Advertising

Predictive User Interface



Naive Approach



Industry Average
12-15% utilization

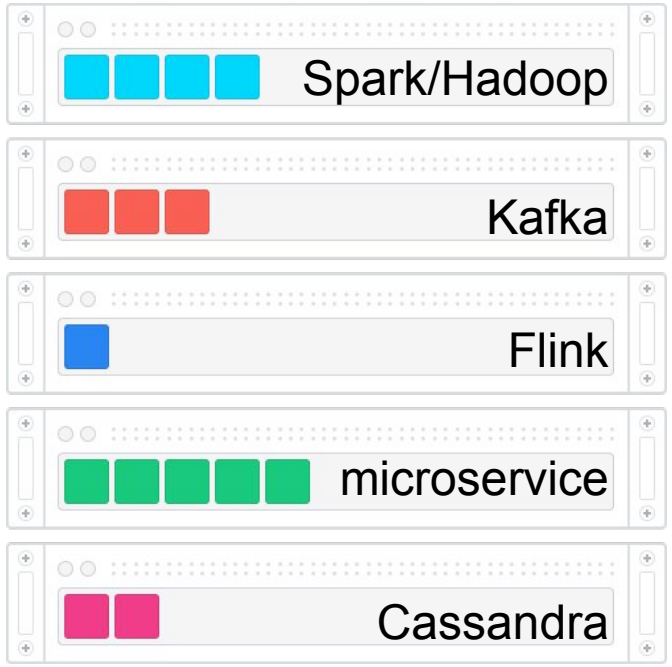
Typical Datacenter
siloed, over-provisioned servers,
low utilization

HYPERSCALE CHALLENGES

- Workload variability
- Efficiency
- Interoperability
- Flexibility
- Scalability
- High Availability
- Operability
- Portability
- Isolability
- Schedulability
- Shareability
- Extensibility
- Programmability
- Monitorability
- Debuggability
- Usability

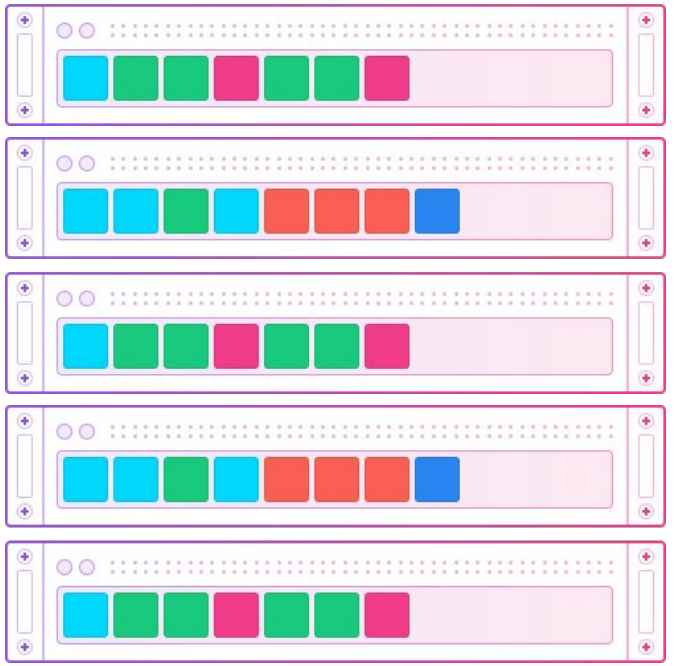
Mesos

SILOS OF DATA, SERVICES, USERS, ENVIRONMENTS



Industry Average
12-15% utilization

Typical Datacenter
siloed, over-provisioned servers, low utilization

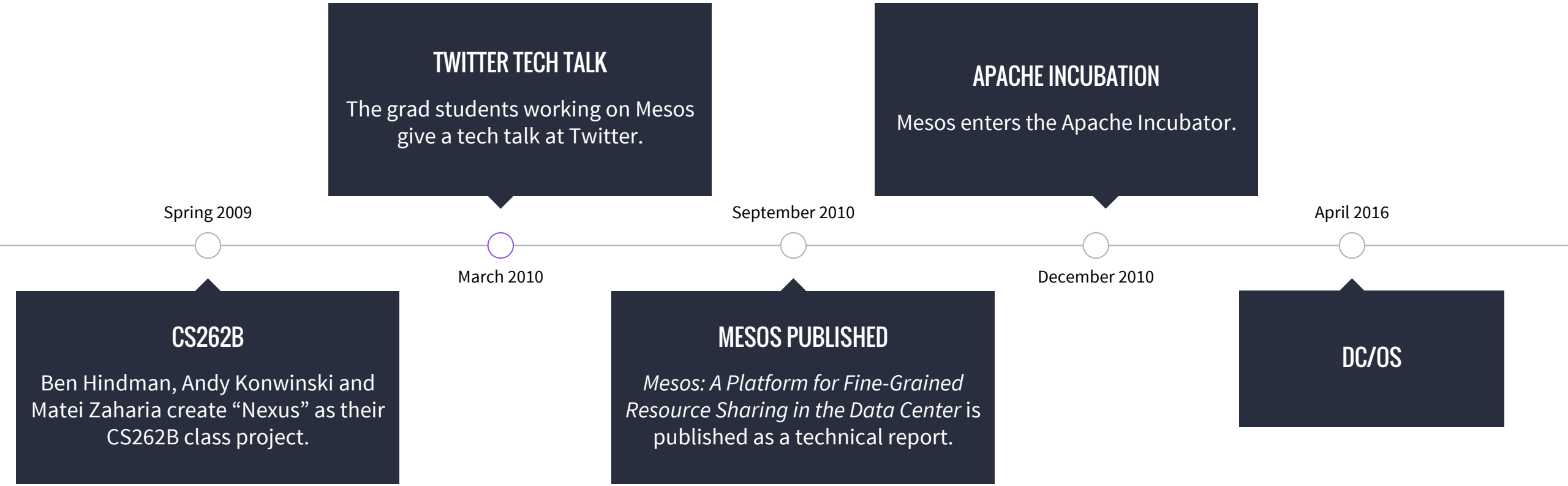


Mesos Multiplexing
30-40% utilization, up to 96% at some customers

4X

Mesos Datacenter
automated schedulers, workload multiplexing onto the same machines

THE BIRTH OF MESOS



TECHNOLOGY

Mesos: A Platform for Fine-Grained Resource Sharing in the Data Center

Benjamin Hindman, Andy Konwinski, Matei Zaharia,
Ali Ghodsi, Anthony D. Joseph, Randy Katz, Scott Shenker, Ion Stoica
University of California, Berkeley

Sharing resources between batch processing frameworks

- Hadoop
- MPI
- Spark

VISION

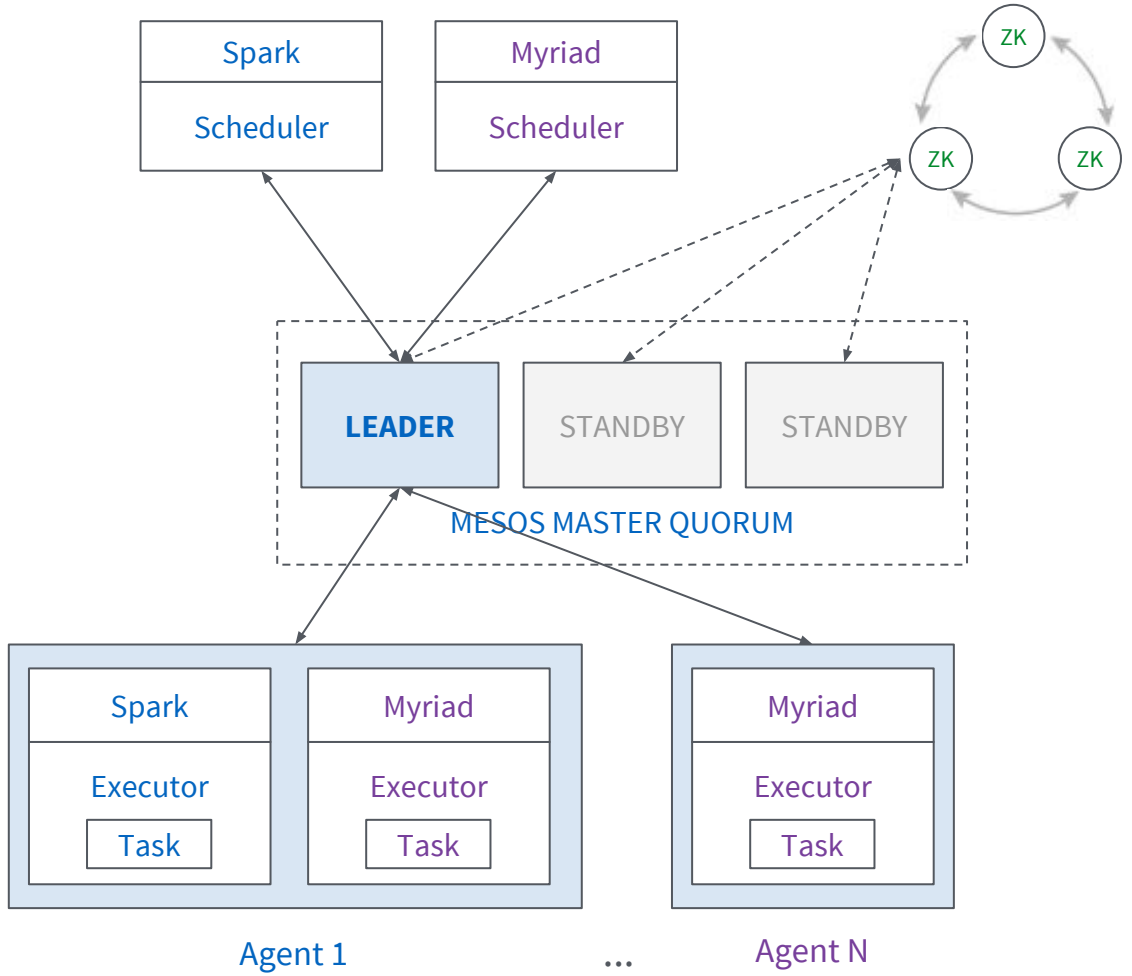
The Datacenter Needs an Operating System

Matei Zaharia, Benjamin Hindman, Andy Konwinski, Ali Ghodsi,
Anthony D. Joseph, Randy Katz, Scott Shenker, Ion Stoica
University of California, Berkeley

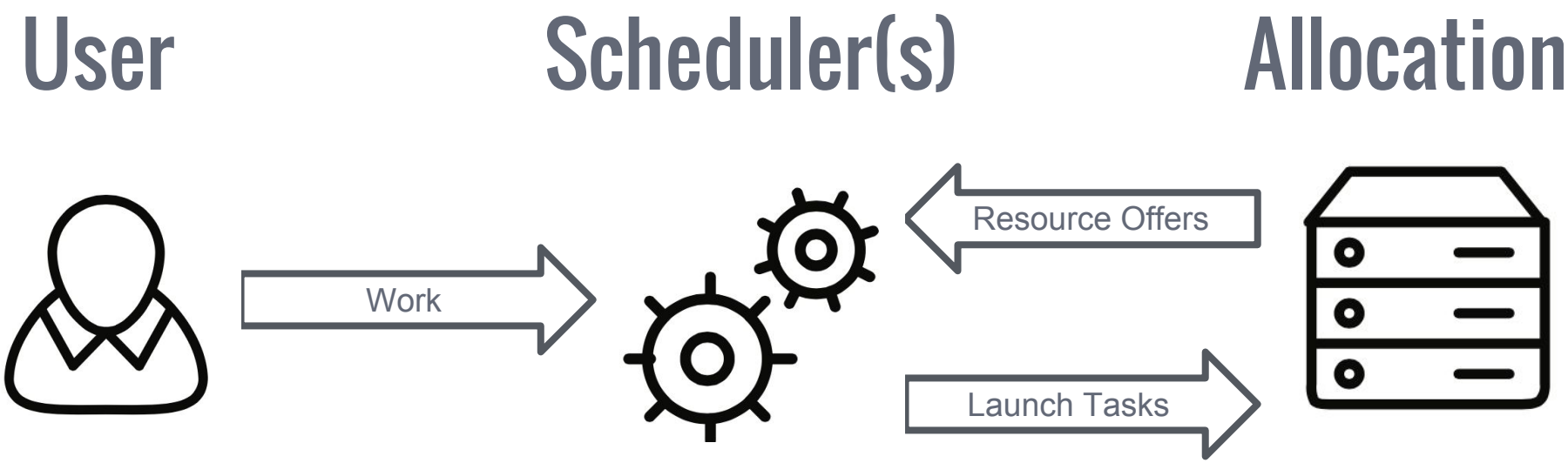
What does an operating system provide?

- Resource management
- Programming abstractions
- Security
- Monitoring, debugging, logging

MESOS ARCHITECTURE



2-Level Scheduling

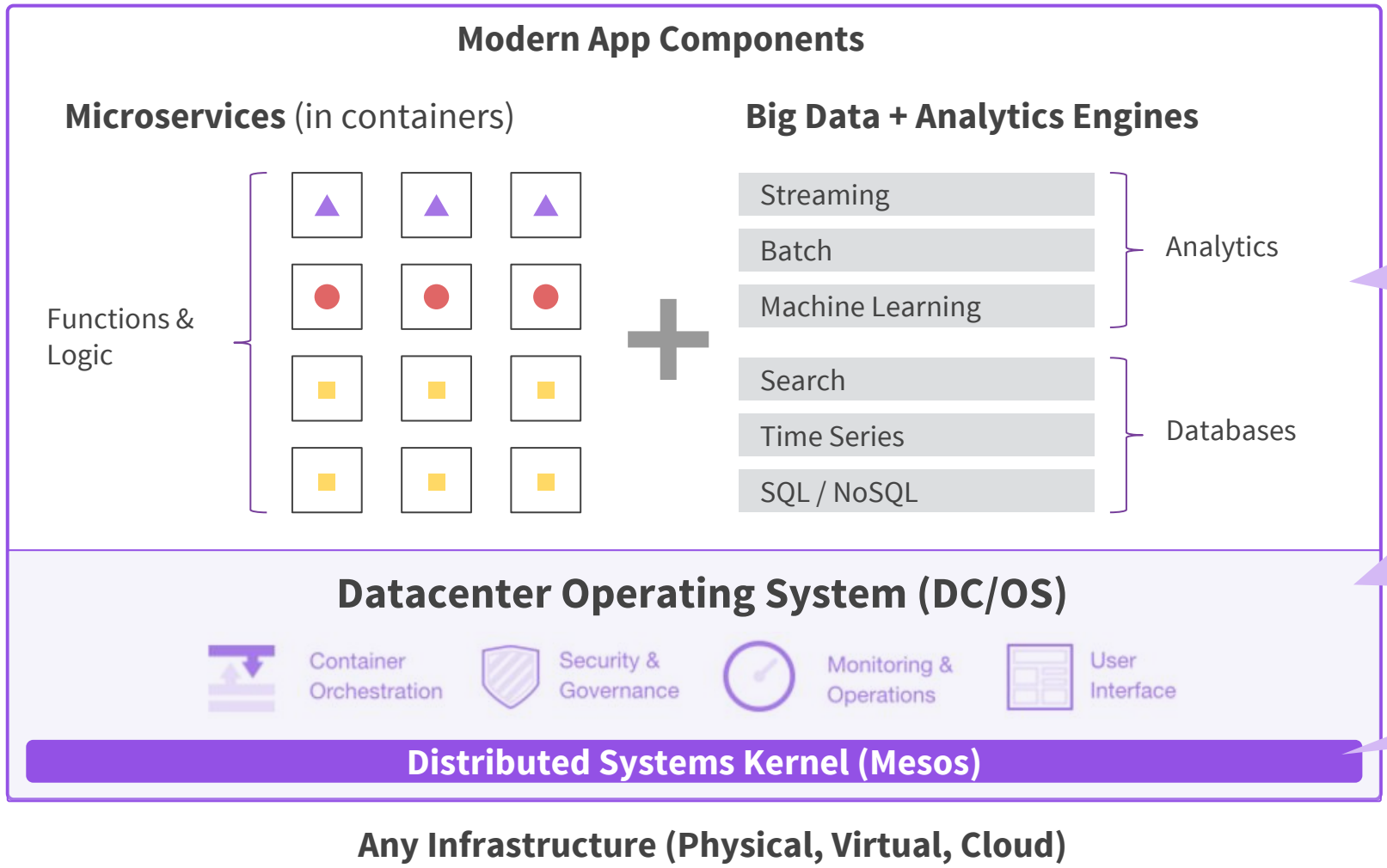


Apache Mesos

- A top-level Apache project
- A cluster resource negotiator
- Scalable to 10,000s of nodes
- Fault-tolerant, battle-tested
- An SDK for distributed apps
- Native Docker support

DC/OS

DC/OS ENABLES MODERN DISTRIBUTED APPS



DC/OS (~30 OSS components)

- UI and CLI, Cluster Installer/Bootstrapper
- **Resource Management**
- **Container Orchestration: Services** & Jobs
- **Services Catalog**, Package Management
- Virtual Networking, Load Balancing, DNS
- Logging, Monitoring, Debugging

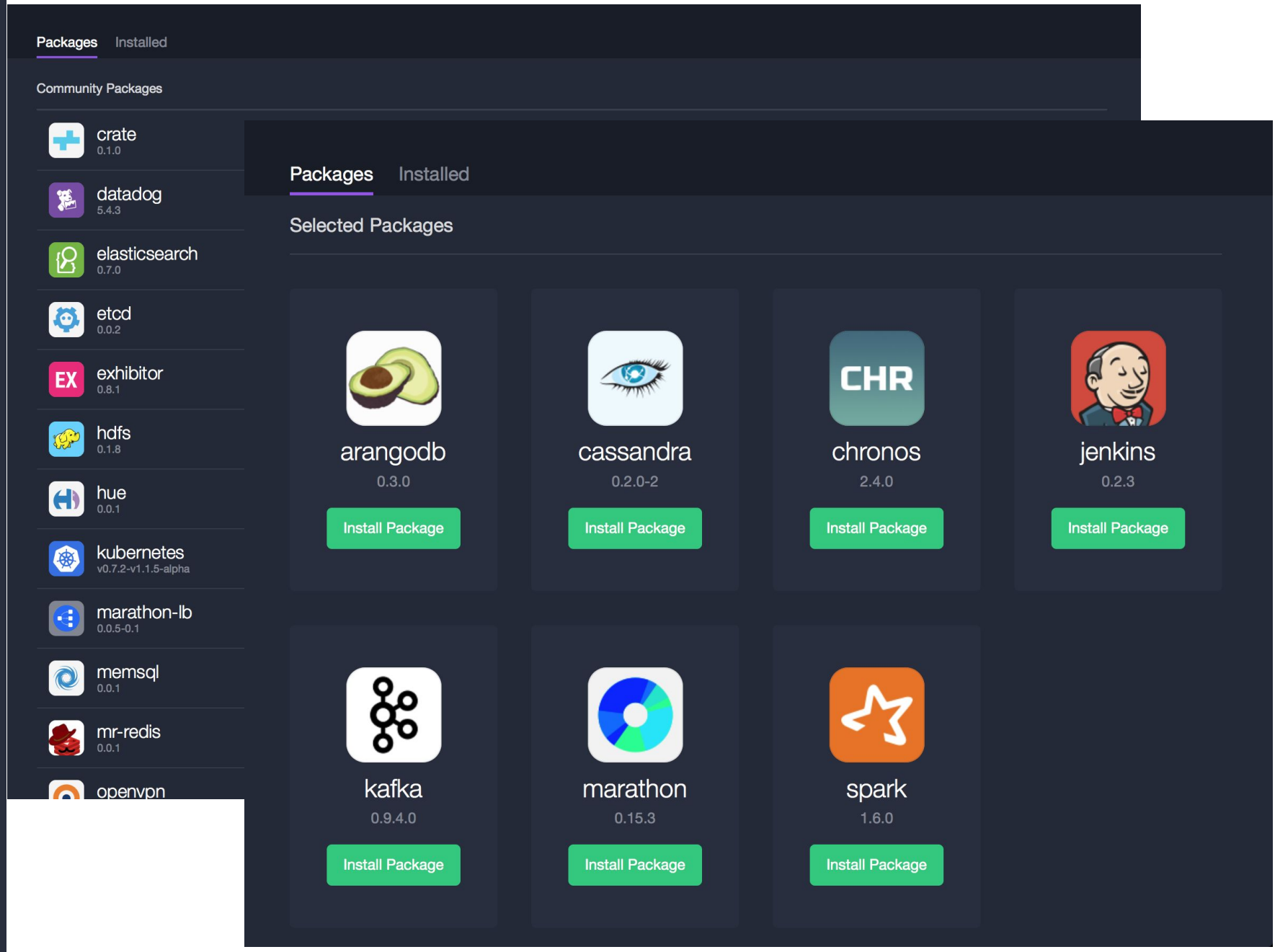
ENTERPRISE DC/OS

- TLS Encryption
- Identity & Access Management
- Secrets Management
- Enterprise-grade Support

MARATHON: The DC/OS init system

- Marathon is a DC/OS service for long-running **services** such as:
  - web services
  - application servers
  - databases
  - API servers
- Services can be Docker images or JARs/tarballs plus a command
- Marathon is not a Platform as a Service (PaaS), but a powerful RESTful API that can be used for building your own PaaS
  https://mesosphere.github.io/marathon/docs/generated/api.html

THE UNIVERSE



DC/OS Big Data Stack

THE SMACK Stack

Apache Spark: distributed, large-scale data processing

Apache Mesos: cluster resource manager

Akka: toolkit for message driven applications

Apache Cassandra: distributed, highly-available database

Apache Kafka: distributed, highly-available messaging system

DATA PROCESSING AT HYPERSCALE

EVENTS

Ubiquitous data streams from connected devices

INGEST

Ingest millions of events per second

STORE

Distributed & highly scalable database

ANALYZE

Real-time and batch process data

ACT

Visualize data and build data driven applications

Sensors

Devices

Clients

Apache Kafka

Apache Cassandra

Apache Spark

Akka

DC/OS

USE CASES

IOT APPLICATIONS: Harness the power of connected devices and sensors to create groundbreaking new products, disrupt existing business models, or optimize your supply chain.

ANOMALY DETECTION: Detect in real-time problems such as financial fraud, structural defects, potential medical conditions, and other anomalies.

PREDICTIVE ANALYTICS: Manage risk and capture new business opportunities with real-time analytics and probabilistic forecasting of customers, products and partners.

PERSONALIZATION: Deliver a unique experience in real-time that is relevant and engaging based on a deep understanding of the customer and current context.

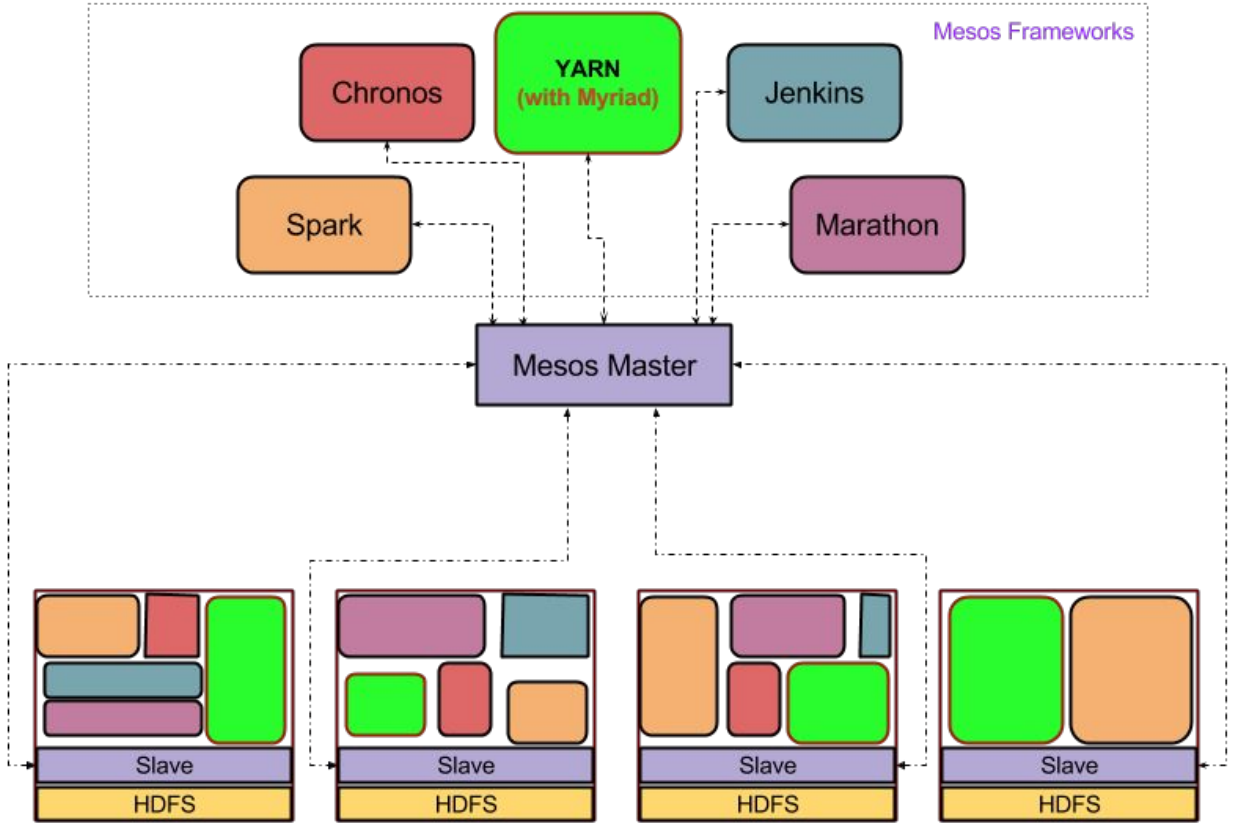
DATA PROCESSING AT HYPERSCALE

Myriad

Yarn on Mesos

- Mesos Framework
- Flexible YARN Cluster

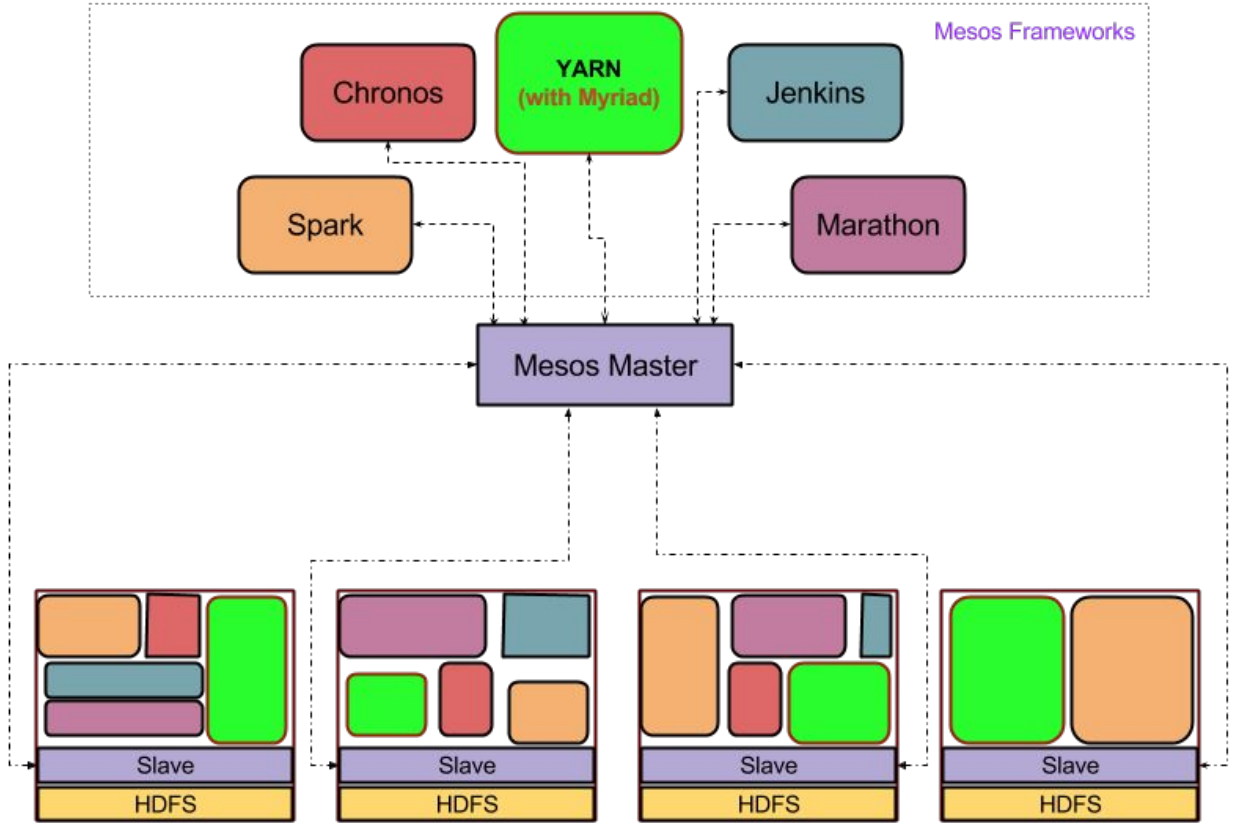
- Mesos manges DC
- YARN manges Hadoop



Myriad

Why?

- Avoid isolated cluster
  - Co-locate with Tier 1 services
  - Make Ops happy!
- Elasticity
- Fault-tolerenance
  - Automatic RM restart
- Multitenancy
- Resource isolation



2nd Day SERVICE OPERATIONS

- Configuration **Updates** (ex: Scaling, re-configuration)
- Binary **Upgrades**
- Cluster **Maintenance** (ex: Backup, Restore, Restart)
- **Monitor** progress of operations
- **Debug** any runtime blockages

Developing own Services

DC/OS Commons

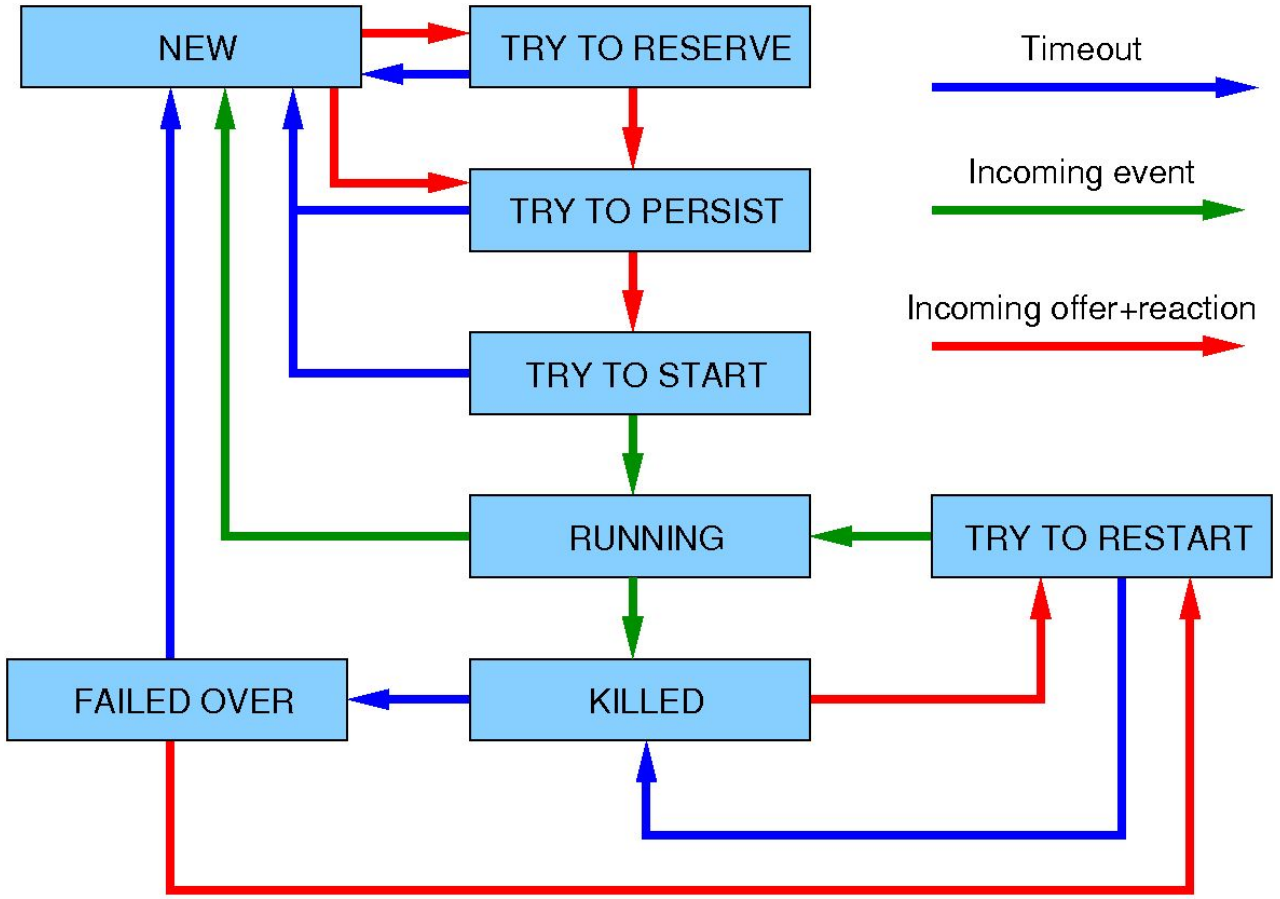
Challenge Fault-Tolerance

Every Big Data Scheduler needs to implement:

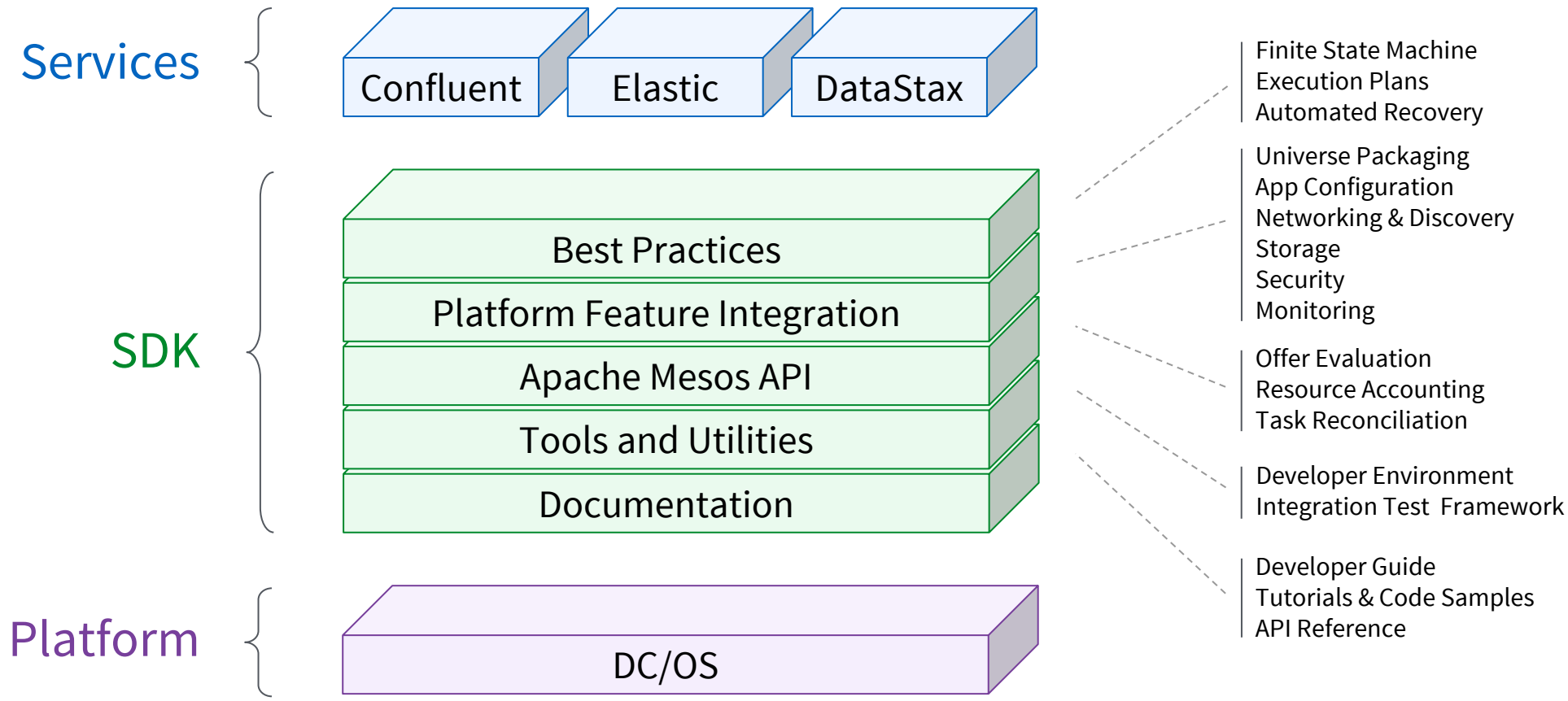
- Reliable data recovery
  - Reserved resources
  - Persistent volumes
- Minimize re-replication
  - Transient failures (like network partitions) shouldn’t lead to re-replication of data

State Machine

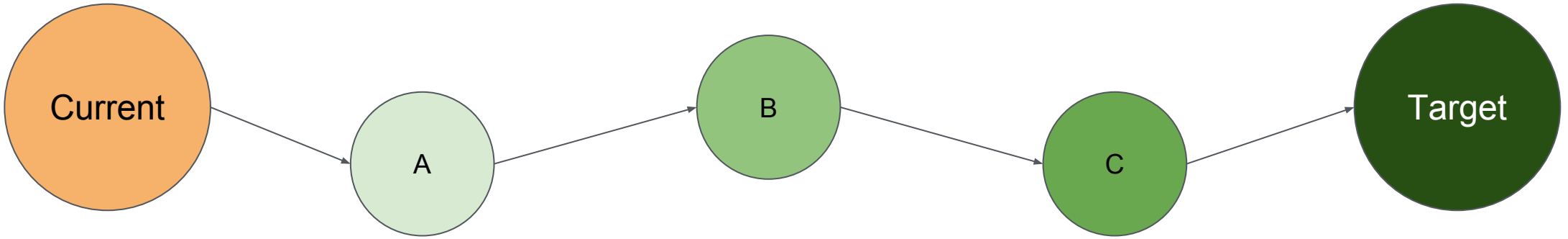
- a **State Machine** for each task:
- state kept in zookeeper
- framework runs event loop
- and handles state changes



DC/OS Commons SDK

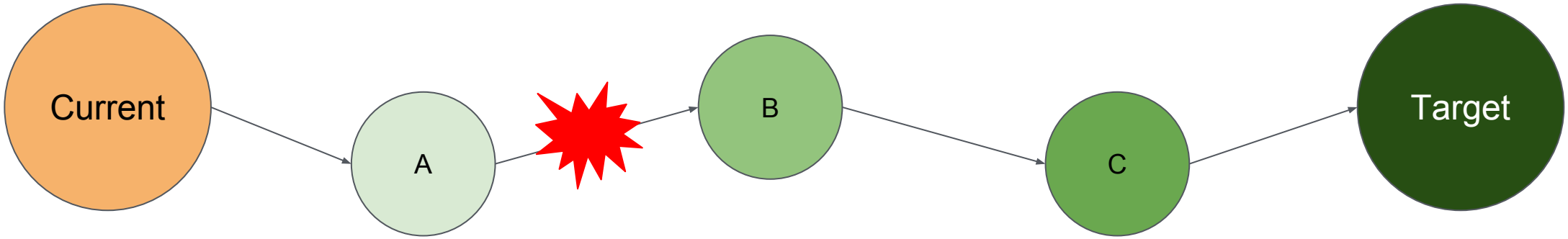


Declarative Design



- Human friendly way of thinking
- Debuggable by design
- Monitor progress
- Fault-tolerant

FAULT-TOLERANCE



TAKEAWAYS

- **Elastic**: Scale your cluster and apps, with minimal operational overhead or cluster reaction time
- **Multi-workload**: Hadoop, Spark, Cassandra, Kafka, and arbitrary microservices/containers/scripts
- **Resilient**: Every DC/OS component is replicated and fault-tolerant;
  SDK makes it easy to build a resilient app scheduler to handle task failures
- **Scalable**: Proven in production on clusters of 10,000s nodes
- **Efficient**: Improve cluster utilization, reduce costs, and increase productivity by letting developers focus on apps, not infrastructure
- **Isolated**: cgroups and namespaces to isolate cpu/gpu, mem, network/ports, disk/filesystem (with/without docker runtime)

Questions?

DC/OS

www.dcos.io

Join the DC/OS Community

Connect with our community of users and browse the latest DC/OS news.

GitHub

Are you interested in helping us make DC/OS even better? Let's work together! Check out our source code on GitHub.

View repositories →

Slack

Have any questions? Our Slack channel is the best place to get help. Just send us a request to automatically receive your invitation.

Join chat →

Mailing List

Want to stay in the loop and connect with other community members? Our public mailing list has all the latest updates. Join the discussion.

Join users@dcos.io →

RESOURCES

- https://dcos.io
- https://mesos.apache.org/
- https://github.com/mesosphere/dcos-cassandra-service
- https://github.com/mesosphere/dcos-kafka-service
- https://myriad.incubator.apache.org
- https://github.com/mesosphere/dcos-commons

DATACENTER RESOURCE MANAGEMENT

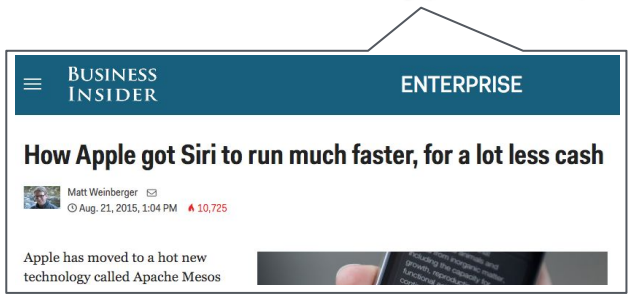
Production-proven Web-Scale Cluster Resource Managers

Borg/Omega	Tupperware/Bistro	Apache Mesos
~2001	*~2007*	*2010+*
Proprietary	Proprietary	Open Source (Apache License)


CADE METZ BUSINESS 03.05.13 6:30 AM

WIRED

RETURN OF THE BORG: HOW TWITTER REBUILT GOOGLE'S SECRET WEAPON


BUSINESS INSIDER ENTERPRISE

How Apple got Siri to run much faster, for a lot less cash

Matt Weinberger
Aug. 21, 2015, 1:04 PM 10,725

Apple has moved to a hot new technology called Apache Mesos

- Built at UC Berkeley AMPLab by **Ben Hindman** (Mesosphere Co-founder)
- Built in collaboration with Google to overcome some Borg Challenges
- Production proven at scale on 10Ks hosts @ Twitter