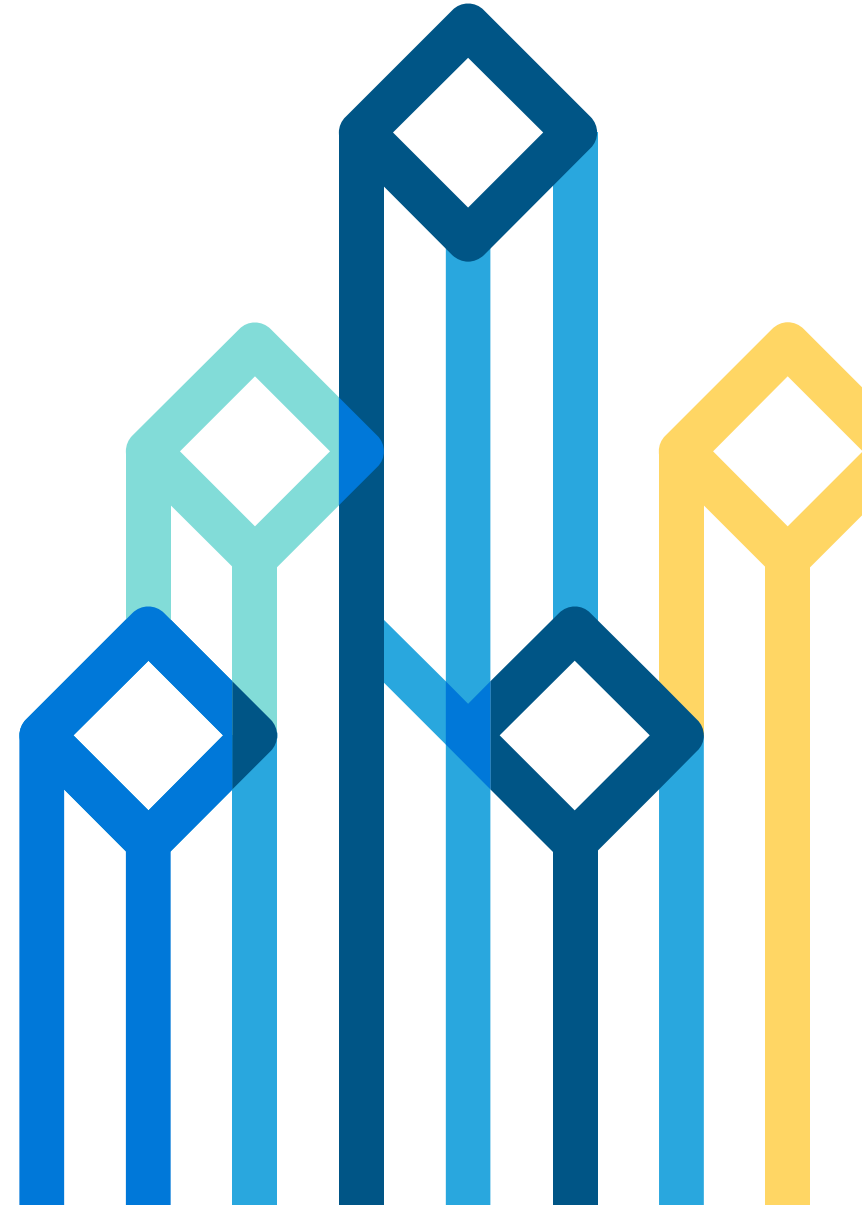


cloudera®

# hadoop 架构在云上 的应用实践



# 提纲

- Cloudera公司及业务简介
- Cloudera EDH 介绍
- Hadoop架构在云上的应用实践

# Cloudera

创立 成立于2008，**企业级Hadoop**产品提供商

ORACLE YAHOO! facebook Google

员工数量 超过1100名

全球支持 24x7 全球支持

创新的主动支持和预测支持项目

客户群 700+**全行业客户**(金融、电信、零售、能源、互联网、媒体等)

各行业的顶尖企业都有Cloudera Enterprise部署

强大的产业链 数百个生态链合作伙伴; Cloudera Connect Program (CCP)

培训和认证 超过80,000管理员、开发者等受训; 最有价值的大数据证书

开源领导者 Hadoop及其相关生态项目**最大贡献者**, 和Intel合作加速

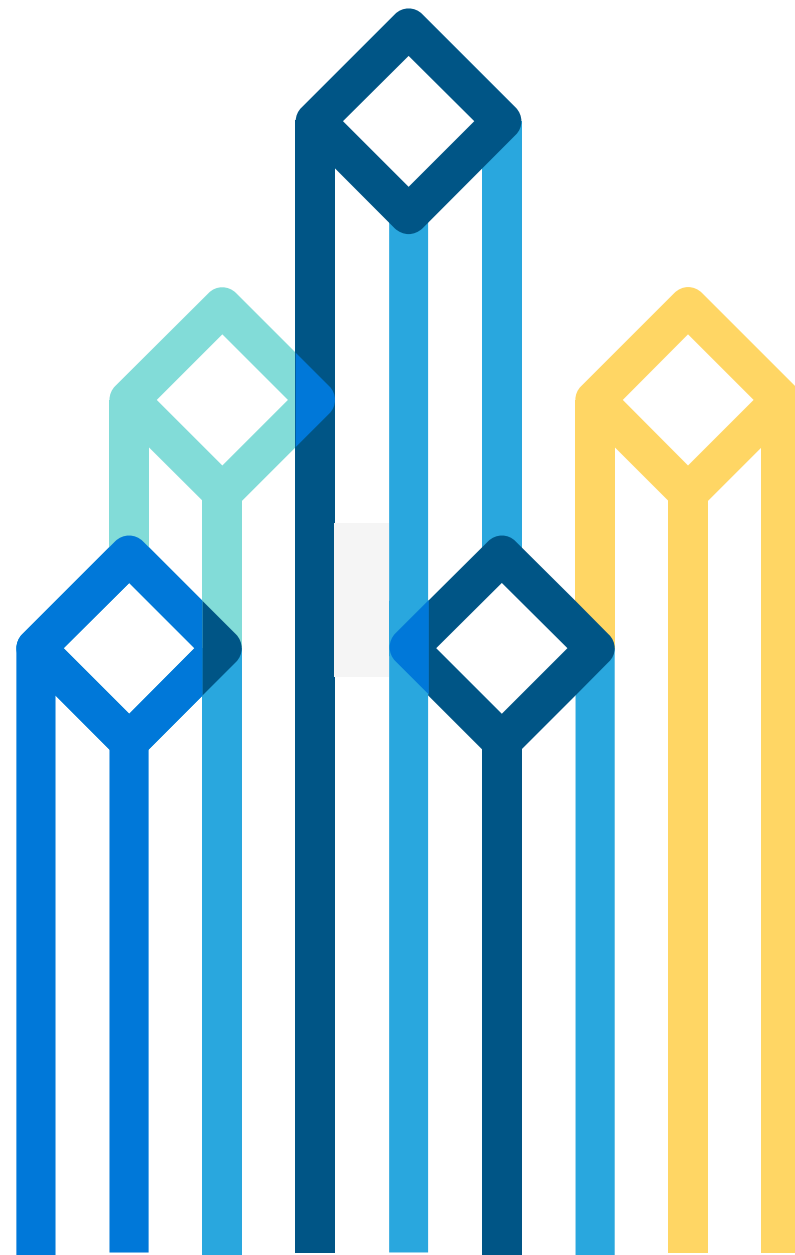
企业数据中心的革新

**Cloudera中国** 2014年9月成立, 上海是大中华区总部, 负责产品培训、

专业技术服务和产品支持, 在北京和广州有本地支持

cloudera®

# Cloudera EDH 介绍



# Cloudera和Apache开源社区

## Leaders Across the Hadoop Ecosystem



**Doug Cutting:**  
Hadoop创始人, Apache基金会主席

---

**22** 项目由Cloudera员工创建

---

**100** Committer 席位覆盖Hadoop生态圈

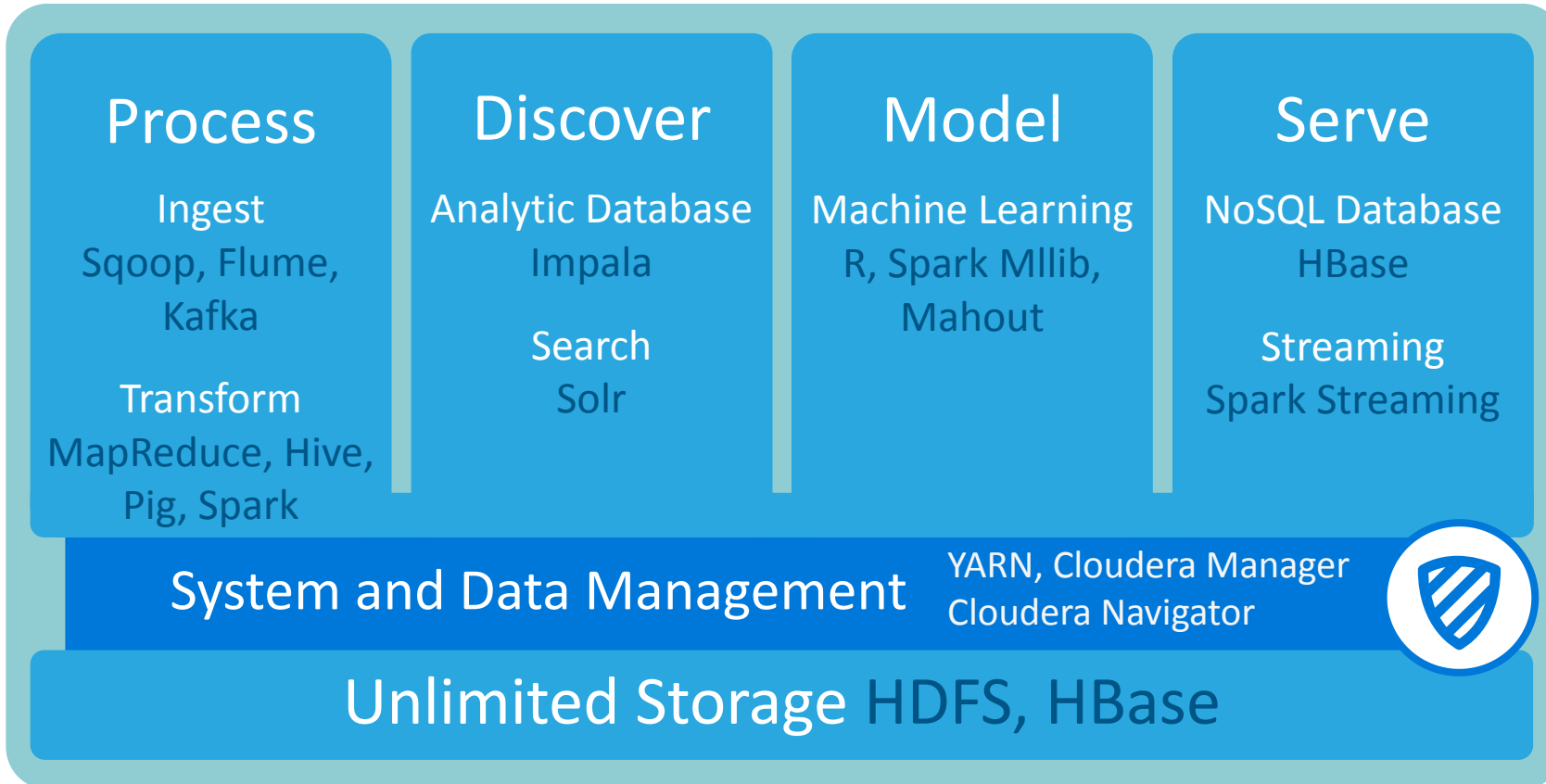
---

**数十万** 生产环境部署节点, 覆盖全行业

---

**15** Hadoop相关的原理及架构类书籍

# Cloudera Enterprise



## 完善的企业安全策略

- 身份认证, 授权, 审计, 数据安全
- 数据可管理性

## 开放标准

- 100%开源Hadoop及其相关组件
- 3rd标准的软件集成
- 开放API
- 标准云服务集成

## 统一平台

- 数据导入导出
- 可扩展存储
- 多样化的处理引擎
- 安全
- 资源管理
- 元数据管理

Deployment  
Flexibility



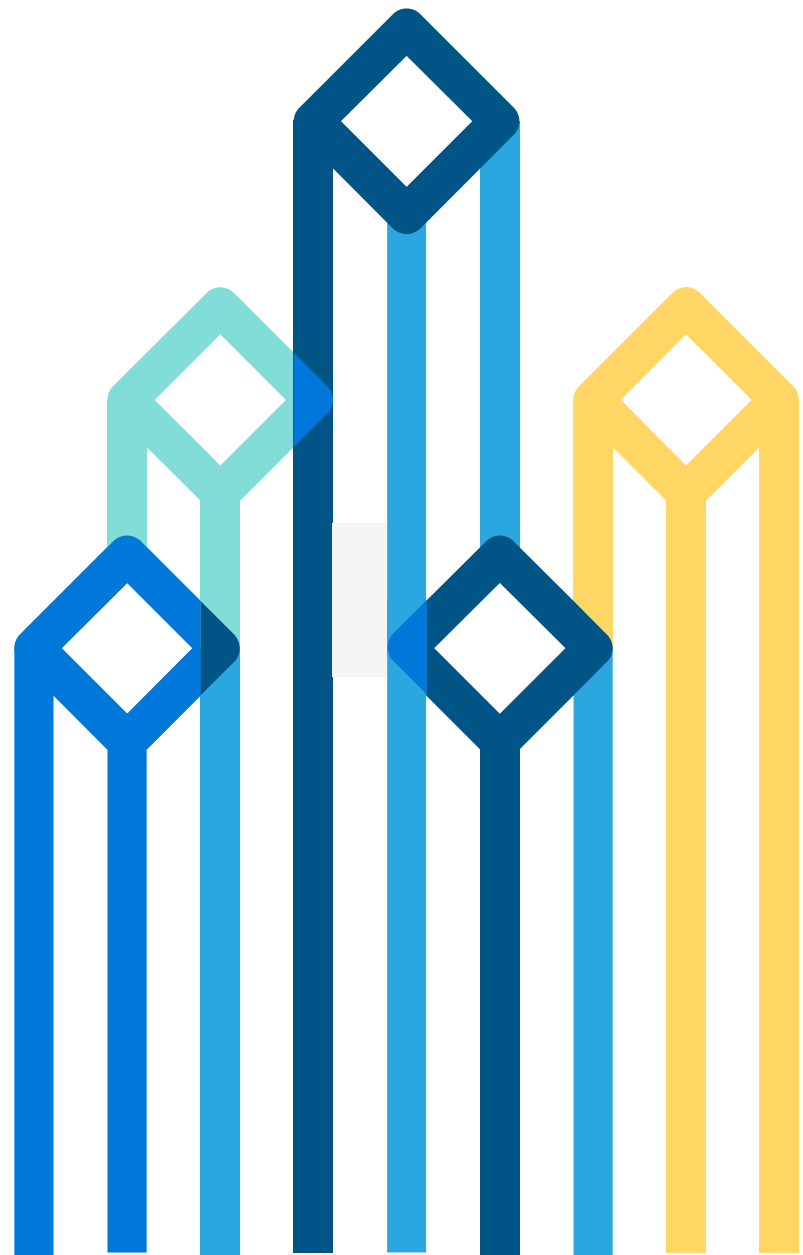
On-Premises  
Appliances  
Engineered Systems



Public Cloud  
Private Cloud  
Hybrid Cloud

cloudera

# Hadoop架构在云上的应用实践



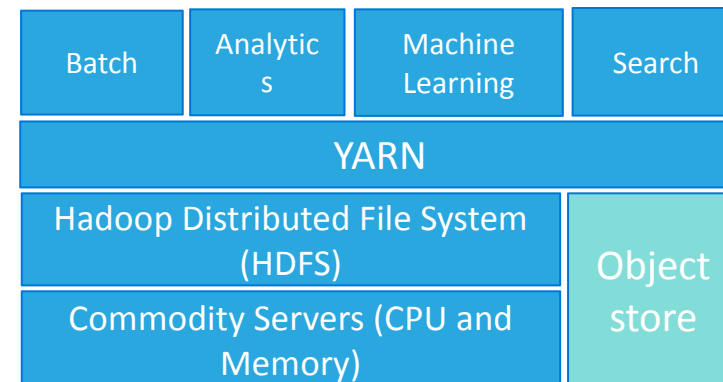
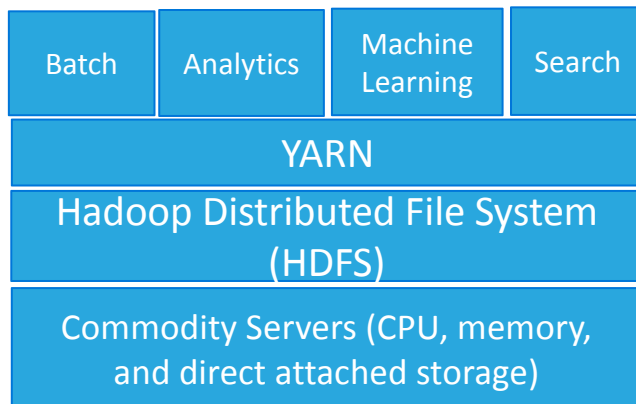
# Hadoop 栈: 本地 versus 云

## 本地模式

- 直连存储
- 数据不在集群间共享
- 固定大小的集群
- 对所有用户共享一个集群

## 云模式

- 存储不一定是在本地
- 可在多个集群间共享数据
- 基于负载弹性伸缩
- 集群按照使用者的需要进行创建





# 新的Hadoop 部署生态

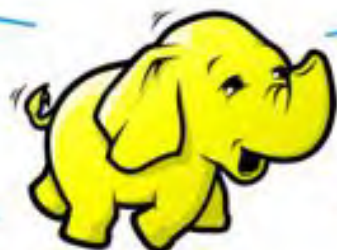
- 公有云
  - Amazon AWS
  - Microsoft Azure
  - Google Cloud
  - Aliyun
  - Amazon EMR
- 传统的中央存储
  - EMC DSSD, EMC Isilon
  - HGST Active Archive System



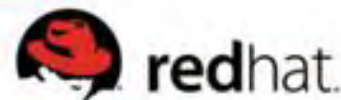
cloudera®  
IMPALA



APACHE  
HBASE



Google Cloud Platform



# 新型部署的特点

- HDFS将Block管理外包给第三方服务
  - 理论上可以支持无限的文件
  - 小文件/大量读写的问题都得以解决
- HDFS 提供大数据应用统一的访问层
  - HDFS FileSystemInterface
- 后端存储服务通常有一致的访问时间
  - 云存储(Amazon S3, Aliyun OSS)
  - 失去了局部性
  - 没有访问速度区别

# HDFS on Cloud

- 趋势: 数据已经在云上
- 为公有云优化
  - 在云上搭建HDFS可以为HDFS带来
    - 弹性伸缩性
  - HDFS 可以为Cloud带来
    - 丰富的大数据生态系统
    - 强一致性
    - 成熟的数据监管

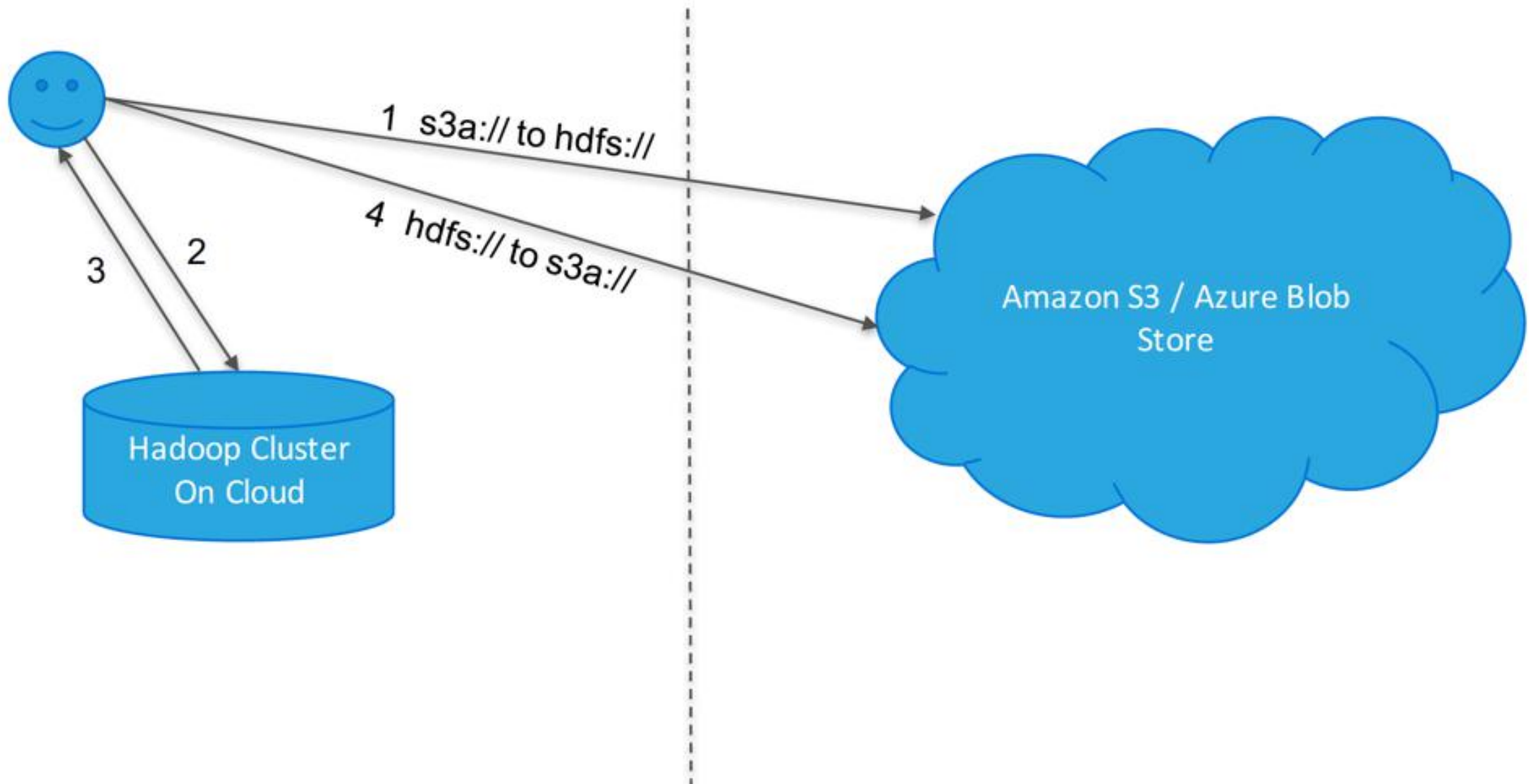
# 三种主要公有云部署形式, 之一

- 直接在虚拟机/云上部署Hadoop集群
  - 与直接在物理机器上部署Hadoop模式几乎完全一致
  - 每个DataNode都建议挂载永久存储设备(i.e. AWS EBS)
  - 需要注意的是:
    - 通常HA NameNode, QJM, ZooKeeper等需要保证部署在不同物理机器上或者Availability Zone里
    - 保证真正意义的HA
  - 优点: 传统的安装经验可以直接使用
  - 缺点: 人为干预高, 难于做性能诊断

# 云部署之二

- Hadoop 作为运算集群
- 数据存储在第三方对象存储服务商
  - 通过HDFS cloud connector 来访问云存储中数据
    - AmazonS3 / MicrosoftAzure / Google CloudStorage connectors
  - HDFS 作为缓存区和工作区
- 优点: 较少的维护HDFS 集群的烦恼
- 缺点:
  - 每次做数据分析都需要导入导出数据
  - 违背了Hadoop的” Move Computation to Data“ 的原则

# Hadoop Cloud Connectors



# 云部署之三

- Hadoop直接操作后端Shared Storage
  - 多个解决方案正在积极开发中
    - Elastic HDFS on Amazon S3
    - Microsoft Azure Data Lake
    - EMC DSSD
  - 利用后端shared storage可以被所有DataNode访问的特性
    - 由DataNode充当代理来访问后端的存储
  - 对上层应用(Hbase,Hive, Impala等)透明



# Elastic HDFS on Amazon S3

- 传统HDFS的运营难处在于:
  - 难于简单的扩大缩小集群
    - 由于固定的Block到DN的关系，变更集群大小需要大量的数据迁移工作  
(Decommision/Rebalance)
  - 通常为了存储而非计算来设计集群的大小
    - 易造成低利用率

# Elastic HDFS on Amazon S3

- 设计目标:
  - 允许Hadoop上层应用直接访问S3
    - 提供数据的强一致性(Strong Consistency) 访问
  - 一次性的运算集群(DisposableHadoopCluster)
  - 集群规模可以弹性伸缩(Elastic Scale Out / In)

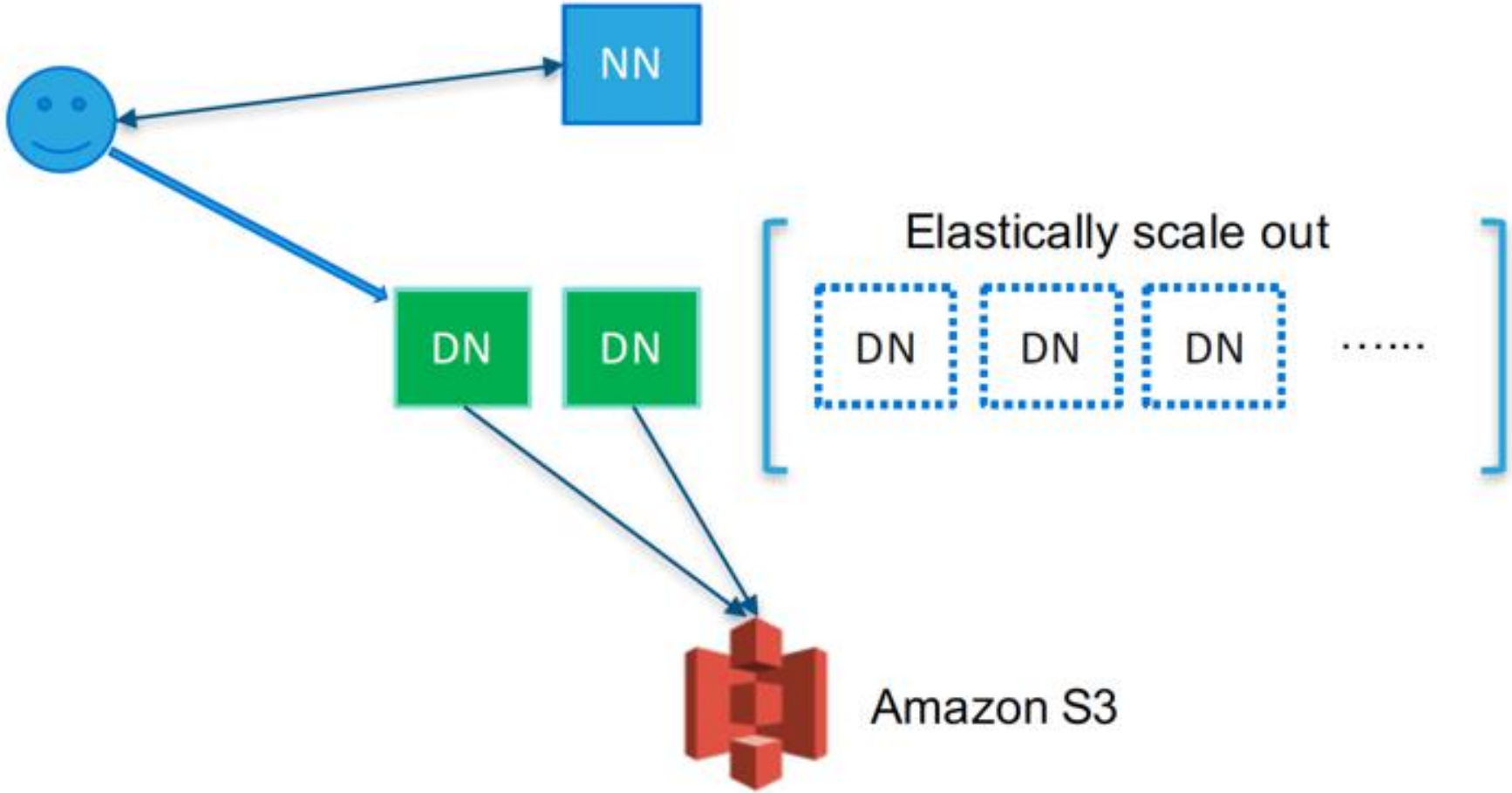
# Elastic HDFS on Amazon S3

- 数据
  - 永久数据保存在S3上，保证
    - 99.9999999999 % 持久性(Durability)
    - 99.99 % 可用性(Availability)
  - HDFS 无需用3 份副本
  - HDFS DataNode 作为Proxy来访问数据, 提供HDFS语义

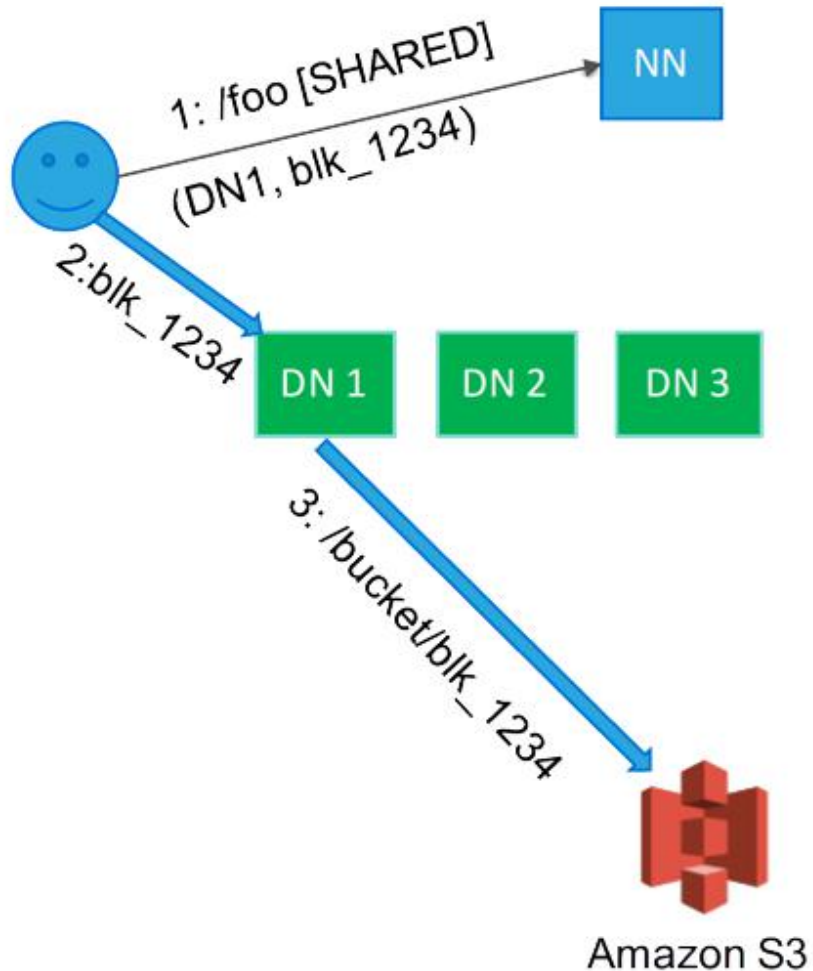
# Elastic HDFS on Amazon S3

- 元数据(Metadata)
  - NameNode无需保证Block 到DataNode的关联性
    - 无需BlockManager 和full block report
    - 有效的减少NameNodememory footprint, Garbage Collection, RPC throughput
  - NameNode可将Client导向任意DataNode来访问S3 数据
    - 在改变HadoopCluster大小后, 不需要再平衡数据(Rebalance)

# Elastic HDFS架构

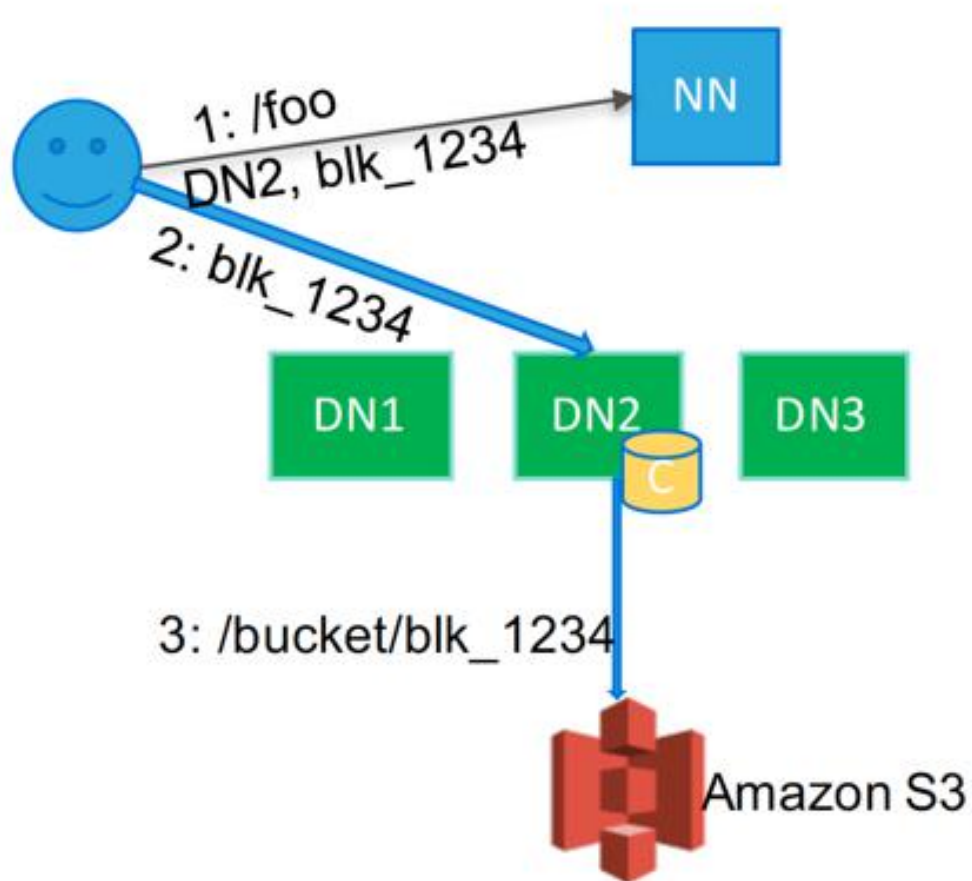


# Write Pipeline



1. HDFS Client 向NN发出写请求,NN 随机返回一个DN
2. Client 将数据写入DN1
3. DN1将数据首先写入本地的临时目录中
4. 在一个Block完成以后, DataNode 将数据推送到S3

# Read Pipeline



1. 用户向NameNode 询问文件(/foo) 的位置
2. NameNode返回任意DataNode (DN2)
3. Client 访问DataNode (DN2)
4. DN2从S3中读取数据, 缓存在本地,并返回给Client

# 其他应用透明访问S3的数据

- 写入数据时
  - 应用直接通过Client-DataNode 协议
  - 无需知道后端存储的具体细节
- 读取数据时
  - DataNode 缓存这份数据
    - HDFS数据是不可更改(Immutable)
  - 提供Client 数据访问局部性
    - 方便任务调度和I/O优化(e.g., Impala)

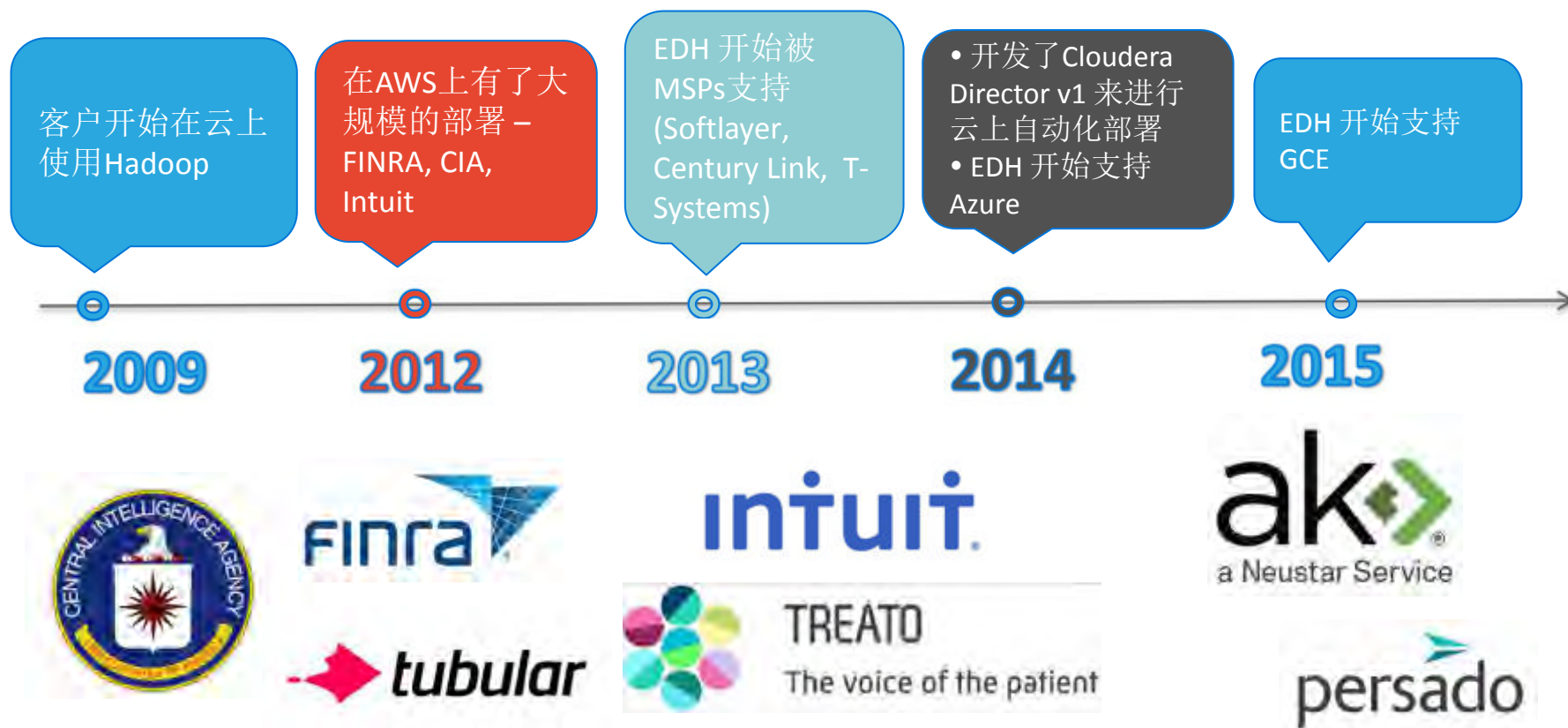


# Elastic Scale Out / In

- HDFS 集群的规模可以快速的扩张或者缩小
  - NameNode具有详尽准确的DataNode信息
  - 所有DataNode上的数据都是作为Staging和Cache data而存在的
    - 新增的DataNode可以立即响应任意的读写请求
    - 关闭DataNode等效于清空Cache
- HDFS集群的大小取决于
  - 运算能力的需求
  - 而不是存储能力的需求

# Cloudera云解决方案的时间线

CDH 是在目前主要的云供应商中最流行的Hadoop发行版



# 云上的架构模式

ETL / Modeling – MR, Spark



短期运行集群  
可伸缩的负载  
不需要本地存储

Source Data

Object Storage (S3, WASB, Swift, Google Blob Store)

BI / Analysis – Impala, Search



长期运行集群  
预先分配集群大小  
少量的本地存储

Seed data

App delivery – HBase



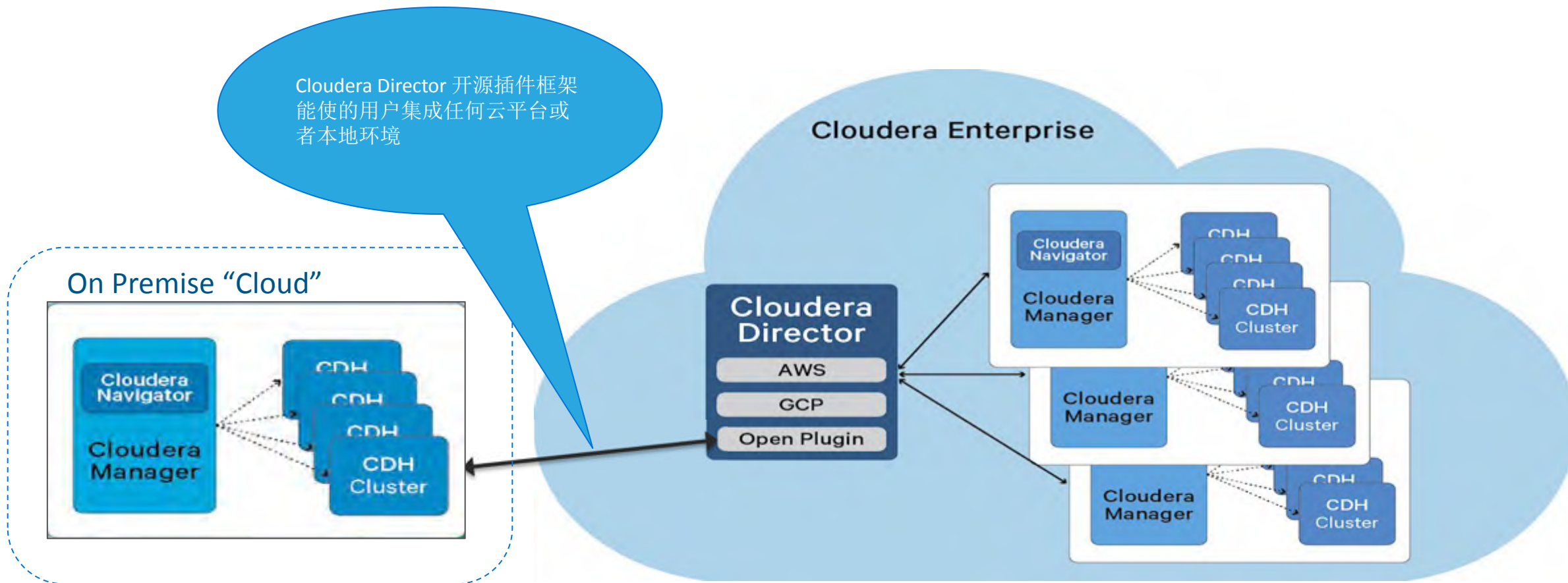
固定的集群大小  
周期同步  
默认需要本地存储

DR/Backup

# 云部署方式

Cloudera Director 建立 Cloudera Manager 来监控一个或多个集群，无论是在本地还是在云端

Cloudera Director 开源插件框架能使用的用户集成任何云平台或者本地环境



# Cloudera 特色

## 部署

部署简单;无后续配置需求

支持快速启动: AWS Quickstart and GCE

支持弹性扩展计算和存储能力

## 可管理性

可以管理多个平台环境 - 本地和云

一个仪表板来监控所有部署好的集群

克隆集群

为已保存的工作负载生成模板

## 可集成性

提供了开放的API以满足二次开发包括简单的脚本和自动化调度的需求

为云提供商提供了开放的 API 插件

## 企业级

深度安全集成: Kerberos & Sentry

Cloudera Navigator: Audit / Lineage / Workflow

高可用和弹性服务

Flexible Integration for controlled deployment in the cloud



唯一支持多平台的  
Hadoop 发行版



Microsoft Azure



cloudera



# Cloudera: 把 EDH 带到云上

易迁移性: 支持多个平台

Private Cloud

Physical

vmware



Public Cloud



Windows Azure

灵活性: 计费灵活并且支持灵活

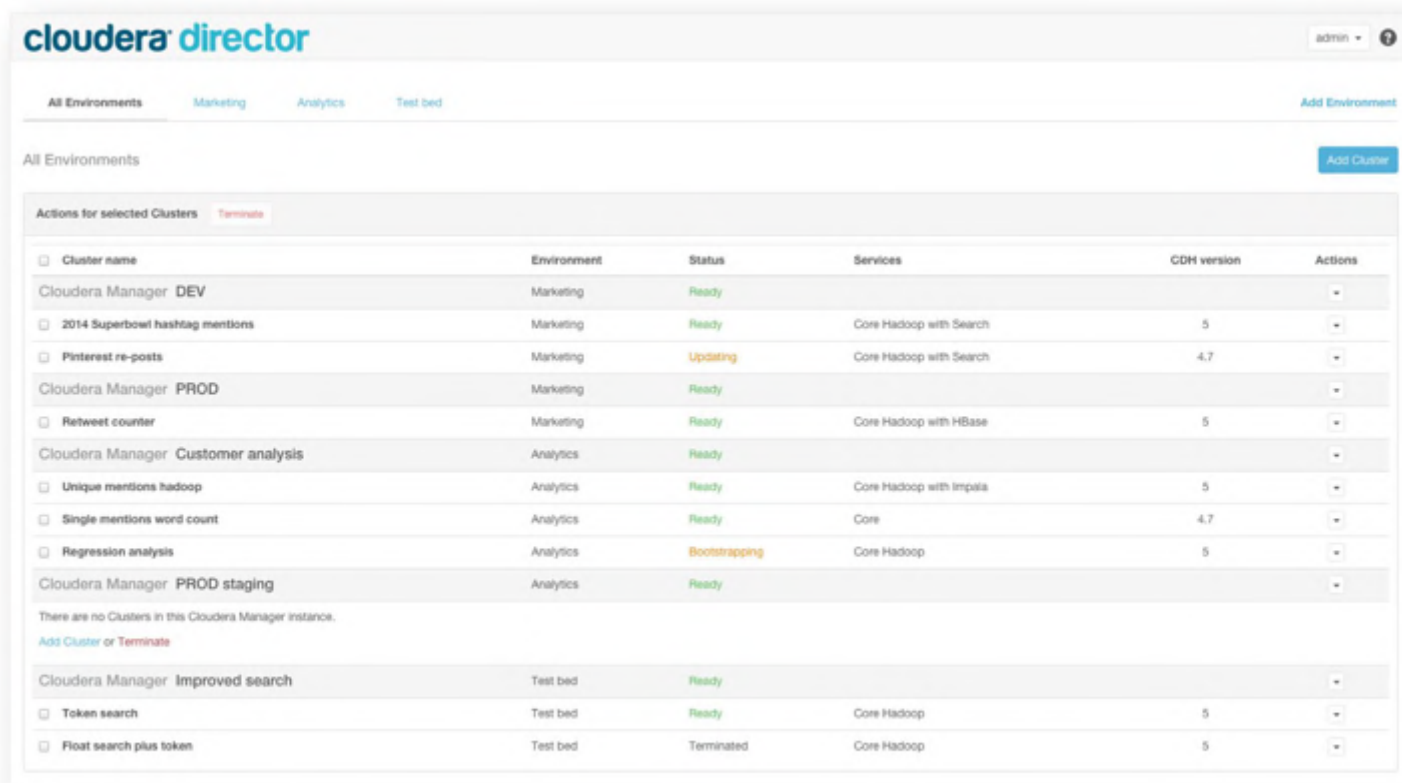
- 在云上支持传统的计费和支持方式
- 支持按照使用量计费

新选择: 正在增长的生态环境

快速扩张的云服务商和生态系统来提供基于云的服务

\* Scheduled for Roadmap

# 云上的Hadoop 不折中



The screenshot shows the Cloudera Director web interface. At the top, there's a navigation bar with 'All Environments', 'Marketing', 'Analytics', and 'Test bed' tabs. Below this, there's a table listing various clusters. The table has columns for 'Cluster name', 'Environment', 'Status', 'Services', 'CDH version', and 'Actions'. The clusters listed include 'Cloudera Manager DEV', '2014 Superbowl hashtag mentions', 'Pinterest re-posts', 'Cloudera Manager PROD', 'Retweet counter', 'Cloudera Manager Customer analysis', 'Unique mentions hadoop', 'Single mentions word count', 'Regression analysis', 'Cloudera Manager PROD staging', 'Cloudera Manager Improved search', 'Token search', and 'Float search plus token'. The status of clusters varies, with some being 'Ready', some 'Updating', and one 'Terminated'.

Cluster name	Environment	Status	Services	CDH version	Actions
Cloudera Manager DEV	Marketing	Ready			
2014 Superbowl hashtag mentions	Marketing	Ready	Core Hadoop with Search	5	
Pinterest re-posts	Marketing	Updating	Core Hadoop with Search	4.7	
Cloudera Manager PROD	Marketing	Ready			
Retweet counter	Marketing	Ready	Core Hadoop with HBase	5	
Cloudera Manager Customer analysis	Analytics	Ready			
Unique mentions hadoop	Analytics	Ready	Core Hadoop with Impala	5	
Single mentions word count	Analytics	Ready	Core	4.7	
Regression analysis	Analytics	Bootstrapping	Core Hadoop	5	
Cloudera Manager PROD staging	Analytics	Ready			
There are no Clusters in this Cloudera Manager instance.					
Cloudera Manager Improved search	Test bed	Ready			
Token search	Test bed	Ready	Core Hadoop	5	
Float search plus token	Test bed	Terminated	Core Hadoop	5	

## Cloudera Director

业界第一款跨平台的，自服务的解决方案，用来部署和管理云上的企业级的hadoop.

- 免费下载，免费使用
- 支持企业级别的用户
- 自服务，统一管理
- 支持混合云管理并首先支持了AWS



# Cloudera Director: 自服务的hadoop



管理员



用户



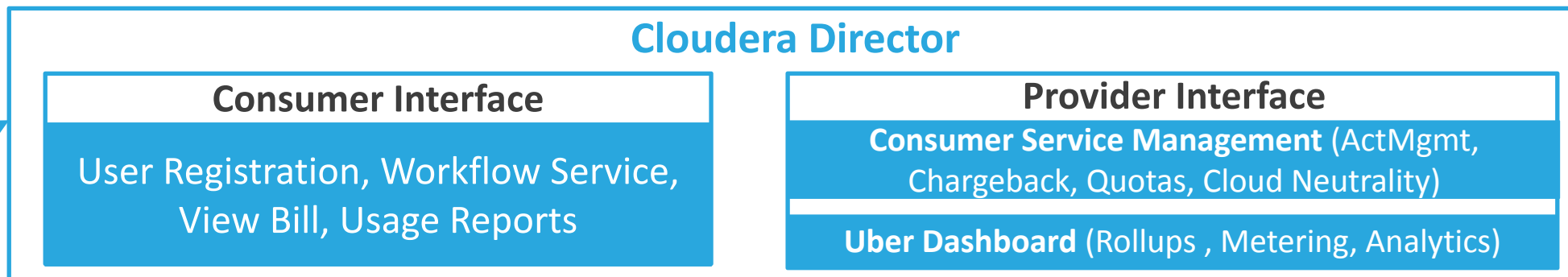
开发者



应用



Customer Portal



Cloudera Director

Consumer Interface

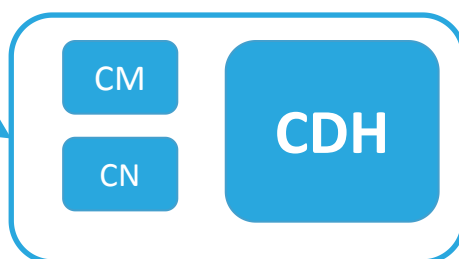
User Registration, Workflow Service,  
View Bill, Usage Reports

Provider Interface

Consumer Service Management (ActMgmt,  
Chargeback, Quotas, Cloud Neutrality)

Uber Dashboard (Rollups, Metering, Analytics)

cloudera®

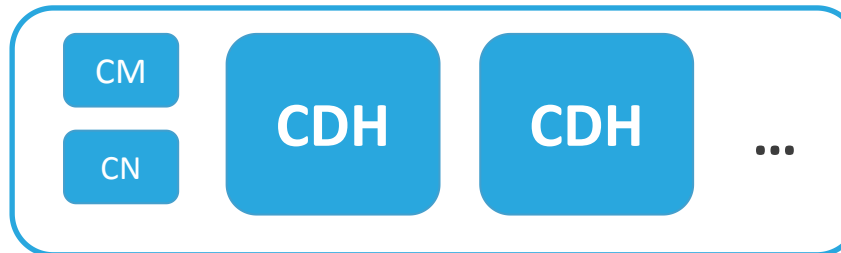


CM

CN

CDH

...



CM

CN

CDH

CDH

...

...



AWS



OpenStack



VMW

...

# Cloudera Director的优势

提供给客户的优点	独特的能力	特性
简化集群生命周期管理	简单的 UI 来进行 建立, 扩展和拆除集群工作	<ul style="list-style-type: none"><li>• 自服务的建立和拆除集群</li><li>• 针对高峰负载动态伸缩</li><li>• 简化集群的clone</li><li>• 为重复部署提供了云时代的蓝图</li></ul>
消除锁定	灵活开放的平台	<ul style="list-style-type: none"><li>• 100% 开源Hadoop 发行版</li><li>• 为混合部署提供原生支持</li><li>• 支持在同个工作流中进行第三方软件的部署</li><li>• 支持客户特定的负载部署</li></ul>
加速实现业务价值	企业就绪的安全和管理能力	<ul style="list-style-type: none"><li>• 支持复杂的集群架构</li><li>• 在容量确定的情况下, 最小化集群规模</li><li>• 提供方便的管理工具</li><li>• 安全和数据治理合规</li><li>• 通过优化的云存储连接器来进行备份和灾难恢复</li></ul>
减少支持费用	监控和指标收集工具	<ul style="list-style-type: none"><li>• 支持多集群的 dashboard</li><li>• 为计费提供跟踪信息</li></ul>

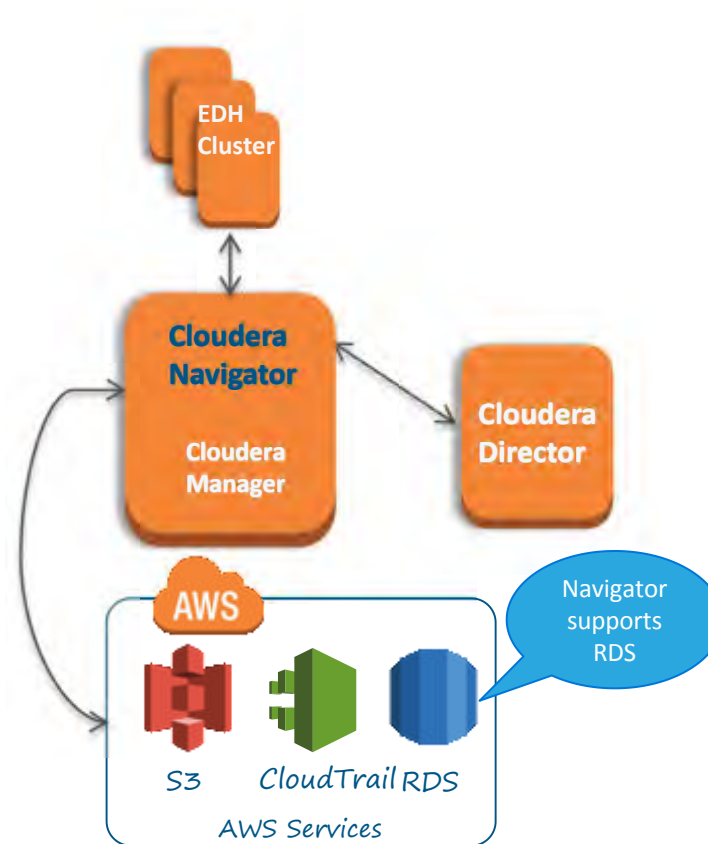
# 云上Navigator 的路线图

Navigator的核心能力:

- 审计
- 发现血缘关系
- 数据发现
- 元数据管理和生命周期管理

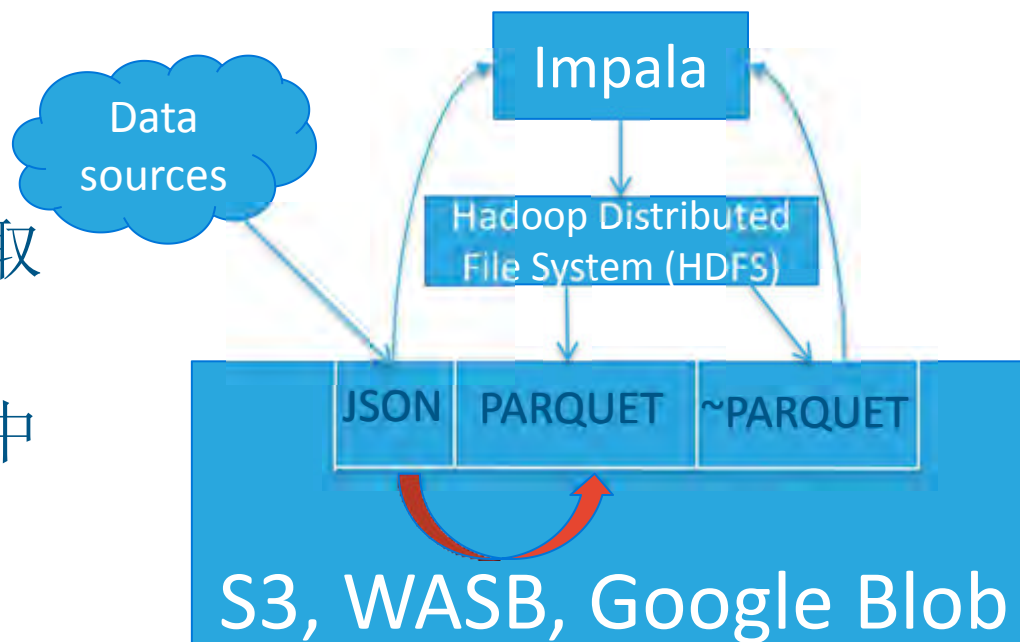
路线图项目

- 数据发现支持S3文件
- 与S3 notifications & Lambda相集成
- 与 CloudTrail 集成
- 审计 Director 的动作
- 存储Navigator 数据在 S3 上



# Impala: 未来云中的数据库

- JSON 数据将直接被加载到 S3 中
- 自动转换格式为 Parquet 以提高性能
- HDFS 作为一个 修改和读取 S3 中 parquet 数据的缓存
- Impala 将会从 HDFS 和 S3 中读取数据



# Cloudera的云合作伙伴

## Amazon Web Services (AWS)

Cloudera Director Integration	Yes
Object Storage Integration	Yes
Reference Architecture	Yes
Full Platform Support	Yes
Other Storage	Local Storage

## Windows Azure

Cloudera Director Integration	Planned
Object Storage Integration	In Progress
Reference Architecture	Yes
Full Platform Support	Yes
Other Storage	Premium VHD

## Google Cloud Platform (GCP)

Cloudera Director Integration	Yes
Object Storage Integration	TBD
Reference Architecture	Yes
Full Platform Support	Yes
Other Storage	Local Storage

## VMWare/OpenStack

Cloudera Director Integration	Planned
Object Storage Integration	NA
Reference Architecture	Planned
Full Platform Support	Planned
Other Storage	Local Storage

# Why Cloudera?

## 云上的hadoop，不能有折中

### 最多特性的hadoop平台

- 为更低的TCO和更快实现业务价值准备的多个框架
- 持续的创新
- 最多的合作伙伴应用集成

### 成功的企业级Hadoop平台

- 安全和数据治理合规的平台
- 端到端管理
- 世界级的支持和hadoop经验

### 为云而建的hadoop平台

- 独特的混合云管理软件
- 大量的优化的云合作伙伴生态系统
- 多个云提供商之间的跨平台性

# 补充材料

Download Now – [www.cloudera.com/downloads](http://www.cloudera.com/downloads)

Try It Out – [AWS Quickstart](#)

Learn More – [www.cloudera.com/director](http://www.cloudera.com/director)

# 集群管理界面

**cloudera director** admin ?

All Environments Marketing Analytics Test bed Add Environment

All Environments Add Cluster

Actions for selected Clusters Terminate

<input type="checkbox"/> Cluster name	Environment	Status	Services	CDH version	Actions
Cloudera Manager DEV					
<input type="checkbox"/> 2014 Superbowl hashtag mentions	Marketing	Ready	Core Hadoop with Search	5	⌵
<input type="checkbox"/> Pinterest re-posts	Marketing	Updating	Core Hadoop with Search	4.7	⌵
Cloudera Manager PROD					
<input type="checkbox"/> Retweet counter	Marketing	Ready	Core Hadoop with HBase	5	⌵
Cloudera Manager Customer analysis					
<input type="checkbox"/> Unique mentions hadoop	Analytics	Ready	Core Hadoop with Impala	5	⌵
<input type="checkbox"/> Single mentions word count	Analytics	Ready	Core	4.7	⌵
<input type="checkbox"/> Regression analysis	Analytics	Bootstrapping	Core Hadoop	5	⌵
Cloudera Manager PROD staging					
There are no Clusters in this Cloudera Manager instance.					
<a href="#">Add Cluster</a> or <a href="#">Terminate</a>					
Cloudera Manager Improved search					
<input type="checkbox"/> Token search	Test bed	Ready	Core Hadoop	5	⌵
<input type="checkbox"/> Float search plus token	Test bed	Terminated	Core Hadoop	5	⌵



# 申请云资源

cloudera director

## Add Environment

Environment name	<input type="text" value="Nightly_US_West_1"/>	?
Cloud provider	<input type="text" value="Amazon Web Services"/>	?
Region	<input type="text" value="us-west-1"/>	?
Access key ID	<input type="text" value="GASJFASKLJFKASFASAKJ"/>	?
Secret access key	<input type="password" value="....."/>	
	<input checked="" type="checkbox"/> Make access keys available to Hadoop	?
EC2 Public key name	<input type="text" value="Pubkey"/>	?
SSH username	<input type="text" value="Testuser"/>	?
	<input checked="" type="radio"/> Use password <input type="radio"/> Use private key	
SSH password	<input type="password" value="....."/>	
	<input checked="" type="checkbox"/> Associate public IP addresses with instances	?

# 部署 Cloudera Manager

cloudera director

## Add Cloudera Manager

Environment Nightly\_US\_West\_1

Cloudera Manager name

TusharTest2



Enable Cloudera Enterprise trial



Instance Template

ManagerTemplate



View



Repository

Override default Cloudera Manager repository



# 增加集群

## Add Cluster

Environment:  ? Cloudera Manager:  ?

Clone from existing  ?

Create from scratch

Cluster name:  ?

Products

Name	Version	?
CDH	<input type="text" value="5"/>	

Services

- Core Hadoop  
HDFS, YARN (MapReduce 2 included), ZooKeeper, Oozie, Hive, Hue, Sqoop
- Core Hadoop with HBase  
HDFS, YARN (MapReduce 2 included), ZooKeeper, Oozie, Hive, Hue, Sqoop, HBase
- Core Hadoop with Impala  
HDFS, YARN (MapReduce 2 included), ZooKeeper, Oozie, Hive, Hue, Sqoop, Impala
- Core Hadoop with Search  
HDFS, YARN (MapReduce 2 included), ZooKeeper, Oozie, Hive, Hue, Sqoop, Solr
- Core Hadoop with Spark  
HDFS, YARN (MapReduce 2 included), ZooKeeper, Oozie, Hive, Hue, Sqoop, Spark
- All Services  
HDFS, YARN (MapReduce 2 included), ZooKeeper, Oozie, Hive, Hue, Sqoop, HBase, Impala, Solr, Spark

Instance groups

Name	Count	Instance Template
masters	<input type="text" value="1"/>	<input type="text" value="Select a Template"/>
workers	<input type="text" value="10"/>	<input type="text" value="Select a Template"/>
gateway	<input type="text" value="1"/>	<input type="text" value="Select a Template"/>

# 扩展集群

## cloudera director

### Grow Cluster

**Cluster name** Nightly\_Cluster

<b>Products</b>	Name	Version
	CDH	4.7.0

**Services** ZOOKEEPER, HDFS, MAPREDUCE, YARN, HBASE

<b>Instance groups</b>	Name	Count	Instance Template
	<a href="#">master1</a>	<input type="text" value="1"/>	<input type="text" value="master1"/> <input type="button" value="View"/>
	<a href="#">master3</a>	<input type="text" value="1"/>	<input type="text" value="master3"/> <input type="button" value="View"/>
	<a href="#">master2</a>	<input type="text" value="1"/>	<input type="text" value="master2"/> <input type="button" value="View"/>
	<a href="#">workers</a>	<input type="text" value="3"/>	<input type="text" value="workers"/> <input type="button" value="View"/>

# 命令行

```
vagrant@node-0:~$ ssh -i .ssh/id_rsa -o StrictHostKeyChecking=no 192.168.33.10
tethys:cli asavu$ vim sandbox.sample.conf
tethys:cli asavu$ ./bin/launchpad bootstrap sandbox.sample.conf
Installing Cloudera Manager ...
* Starting ... done
* Requesting an instance for Cloudera Manager ..... done
* Running custom bootstrap script on 192.168.33.10 ..... done
* Inspecting capabilities of 192.168.33.10 ..... done
* Normalizing 192.168.33.10 ..... done
* Installing ntp (1/2) .... done
* Installing curl (2/2) ..... done
* Rebooting 192.168.33.10 .... done
* Waiting for 192.168.33.10 to boot ..... done
* Installing repositories for Cloudera Manager ..... done
* Installing oracle-j2sdk1.6 (1/5) .... done
* Installing cloudera-manager-daemons (2/5) ..... done
* Installing cloudera-manager-server (3/5) .... done
* Installing cloudera-manager-server-db-2 (4/5) .... done
* Installing cloudera-manager-agent (5/5) .... done
* Starting embedded PostgreSQL database .... done
* Starting Cloudera Manager server .... done
* Waiting for Cloudera Manager server to start ..... done
* Configuring Cloudera Manager .... done
* Starting Cloudera Management Services .... done
* Inspecting capabilities of 192.168.33.10 ..... done
* Done ...
Cloudera Manager ready.
Creating cluster CDH4.Sandbox.Byon ...
* Starting ... done
* Requesting 3 instance(s) ..... done
* Running custom bootstrap script on new instance(s) ..... done
* Inspecting capabilities of new instance(s) ..... done
* Running basic normalization scripts ..... done
* Registering instance(s) with Cloudera Manager ... done
* Waiting for Cloudera Manager to deploy agents on instances ....
```



**cloudera**

Thank You