Hadoop Ecosystem on Docker

Docker是一种内核虚拟化技术，可以提供轻量级虚拟化，以便隔离进程和资源，而且不需要提供指令解释机制以及全虚拟化的其他复杂性，相当于C++中的NameSpace。容器可以有效的将单个操作系统管理的资源划分成孤立的组中，以便更好的孤立在组之间平衡有冲突的资源使用需求。将服务打包运行在Docker中有以下好处：

1. 环境一致性问题，开发环境、测试环境和生产环境不一致可能会导致有些bug并未在开发过程中被发现，而Docker通过镜像提供了除内核外完整的运行时环境，确保应用环境一致性
2. 持续交付和部署，开发人员可以通过定制应用镜像来实现持续集成、持续交付和部署，从而实现一次创建或配置，可以在任意地方正常运行
3. 更轻松的迁移，由于Docker确保了执行环境的一致性，使得应用迁移更加容易，轻易的将在一个平台上的应用，迁移到另一个平台，而不用担心运行环境的变化导致无法正常运行
4. 更轻松的维护和扩展，Docker使用分层存储以及镜像技术，使得应用重复部分的复用更为容器，也使得应用的维护更加简单。

随着大数据应用深入发展和Docker产品的成熟，业务开始将Docker和大数据平台（Hadoop）结合使用，主流的两种用法：

* 通过YARN来使用Docker容器进行部署应用，DockerContainerExecutor使得YARN NM服务可以将Container进程运行在Docker容器中，用户可以自定义运行YARN Container的Docker镜像。在Hadoop 3.0后，支持Yarn-native-service框架，简化Docker应用的创建和维护。但是目前Docker主流调度室是k8s和mesos，yarn支持Docker处于比较尴尬的地步
* 第二种方式是使用Docker来直接运行Hadoop，例如Hortonworks使用Cloudbreak技术将HDP运行在云平台上，MicroSoft的Windows Azure HDSinght等，即实现Hadoop as a Service。

目前更多的开发者将微服务及容器作为DevOps的技术基础，而且使用Docker Container来运行Hadoop的优势明显，同时大数据开发人员使用云存储存放数据，按需根据数据大小使用Hadoop及Spark的云集群对数据分析，其简化大数据平台（Hadoop）的应用开发部署，研发和测试团队集成交付更加敏捷高效、在线环境的运维更加有质量保障。BdasS的核心是Hadoop as a Service，将Hadoop运行在云服务中，包括YARN、HDFS、Spark、Hive及HBase等服务。.

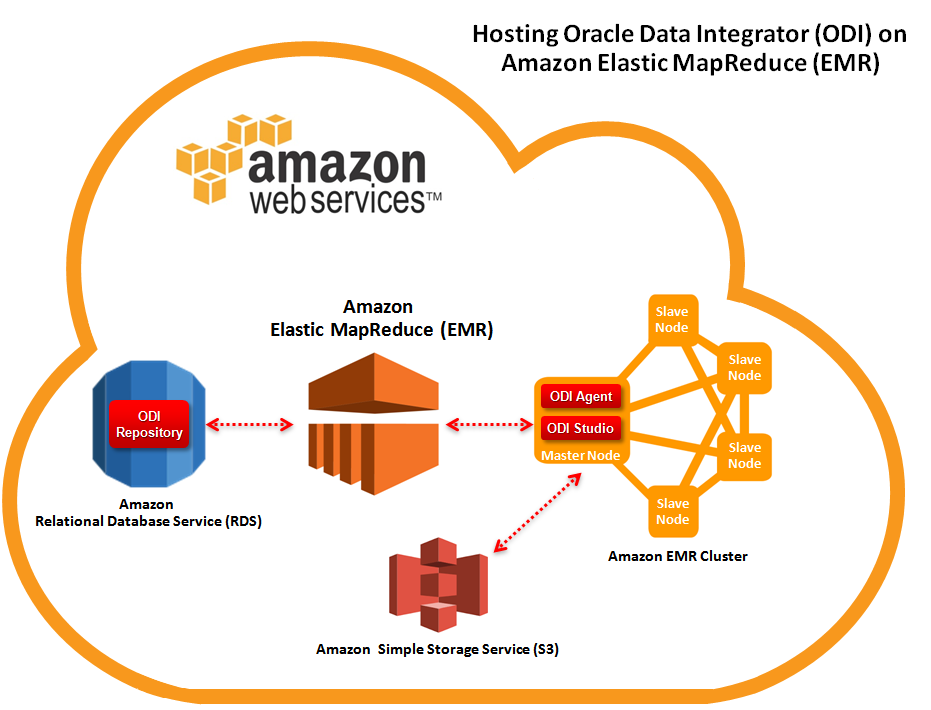
# 1.Hadoop As a Service

开源大数据框架Apache Hadoop已经成为大数据处理的事实标准，但是Hadoop集群的搭建和使用需要大量的工作，因此就有了HaaS(Hadoop as a Service)，开发者或者用户可以做到按需使用，包括存储空间和计算能力。Amazon、Cloudera、HortonWorks、微软等公司都发布了自己的Hadoop-as-a-Service产品，其出发点都在于减少Hadoop和基础设施的管理，但是HaaS没有一定的标准，但是要满足以下要求：

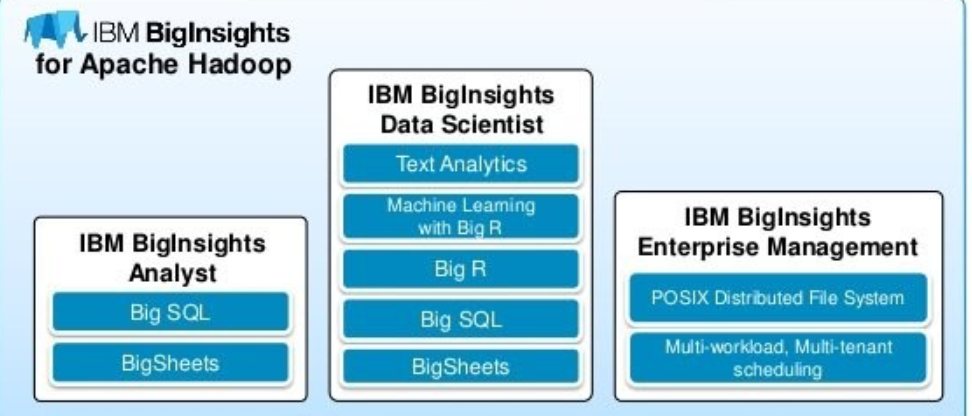
1. 功能丰富，不仅要具备基本功能：YARN、HDFS、Hive等，还需要具备其他数据分析的工具，如Pig、R等，用户登录到服务后要保证相关的计算操作立即可用
2. 使用HDFS存储数据，以其他格式存储需要一定的成本转换成HDFS格式
3. 弹性，按需扩充存储和处理能力
4. 不停机操作
5. 自动配置，使用HaaS的优点是最大限度的降低Hadoop专家的需要，因此有必要自行配置最佳数量和类型的节点

目前业界已有的HaaS产品较多，典型的产品如下：

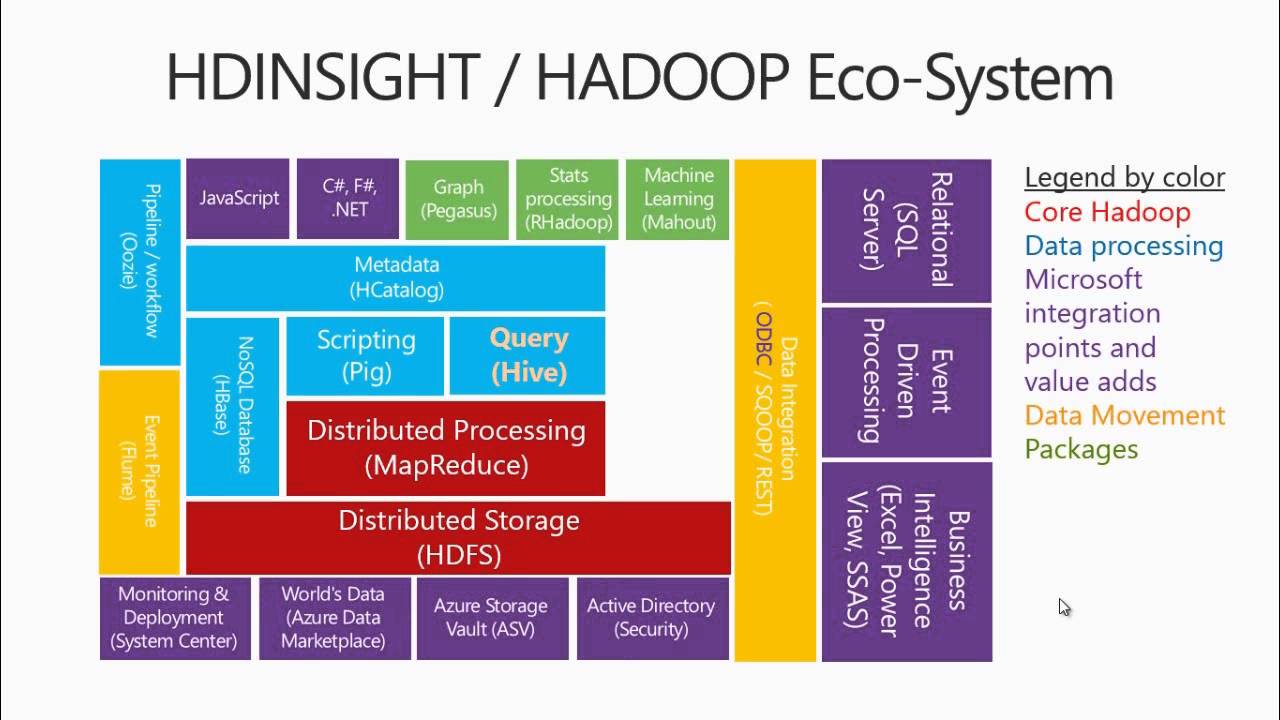
* Amazon Web Service，Amazon EMR，托管数据分析服务框架，企业、研究人员、数据分析和开发人员可以轻松经济的掌握海量数据的能力，托管的服务包括Hadoop、Zeppelin、Tez、Hbase、Spark等，无需要担心集群设置、管理或调整，



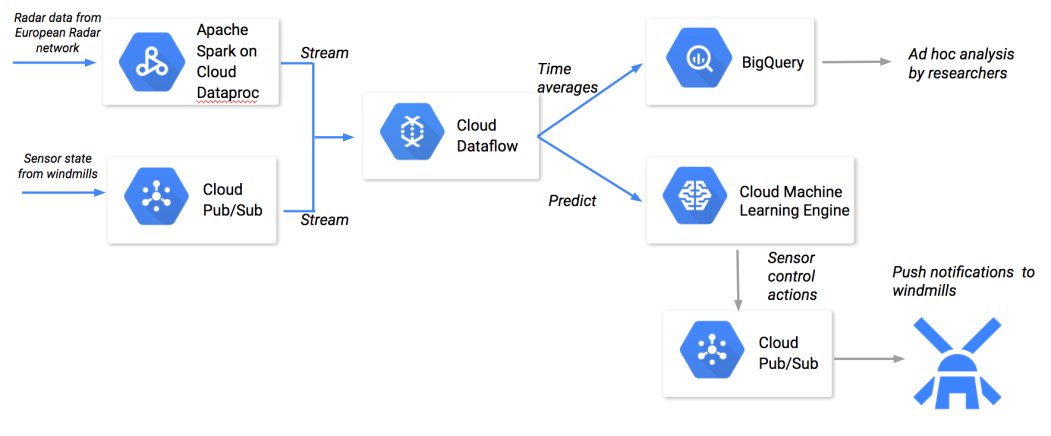
* IBM BigInsights，IBM的HaaS平台，旨在帮助公司对大数据进行挖掘分析，提供数据分析的开源工具Hadoop及一些IBM开发的技术



* Microsoft Azure HDInsight，是微软提供的基于Hortonworks HDP的Hadoop组件的云发行版，可以通过HDInsight轻松、快速且经济有效的处理大数据数据，可以使用Hadoop、Spark、Hive、LLAP、Kafka等最常用的开源框架。

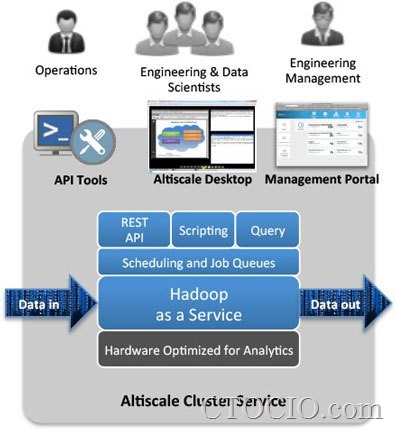


* Google GCE中的Cloud Dataproc，在云端使用Spark和Hadoop，提供一个强大且完整的数据处理、分析和机器学习平台，



* Altiscal

Altisacle将Hadoop以HaaS的模式运行在云端，用户可以通过ssh登录到Alstica桌面环境（托管在Amazon上的服务）来使用Hadoop，系统架构如下图所示：



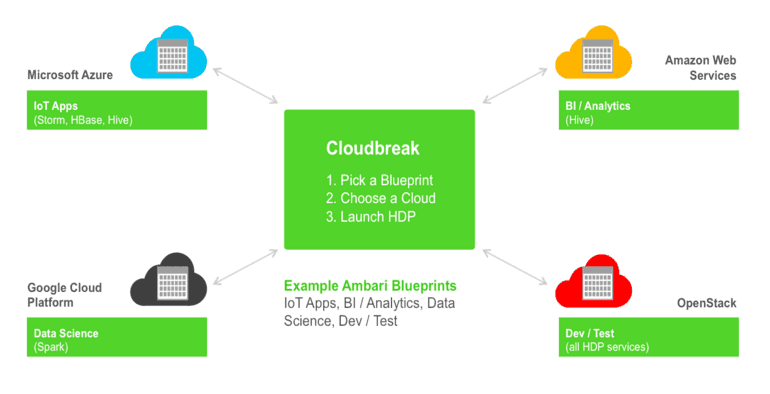
https://www.technavio.com/blog/top-16-companies-in-the-hadoop-as-a-service-hdaas-market

# 2.Hadoop On Docker已有方案

使用Docker启动Hadoop集群可以简化大数据平台的应用开发部署，使研发和测试团队集成交付更加敏捷高效。因此Hadoop On Docker就是将硬件（云服务器）通过Docker搭建成一个超级服务器，每个Docker容器作为一个Hadoop实例，根据需求动态增加实例，以满足计算性能要求。目前Hadoop On Docker有多种方案包括BlueData EPIC Lite，Cloudbreak及Pachyderm等

## 2.1 Hortonworks Cloudbreak

基于Hadoop的docker service API，建立于Apache Ambari 、Docker Containers、Serf和dnsmasq这类云提供者的API之上，使用Docker镜像在任何主流云平台启动HDP，包括微软Azure、AWS、Google Cloud Platform等，如下图：



Cloudbreak的Docker镜像为sequenceiq/ambari，根据$AMBARI-ROLE的变量启动的Docker Container为Ambari Server和Ambari Agent。但是在云环境中IP地址是不可知的，因此使用Serf，将首个启动的Container的IP传入Serf，通过使用Gossip协议，Serf自动发现其他容器，并设置DNS名称及配置节点间的路由，Serf会随着节点的加入和离开自动更新配置运行于容器中的DNS服务器dnsmasq。Cloudbreak启动Hadoop Cluster的步骤如下：

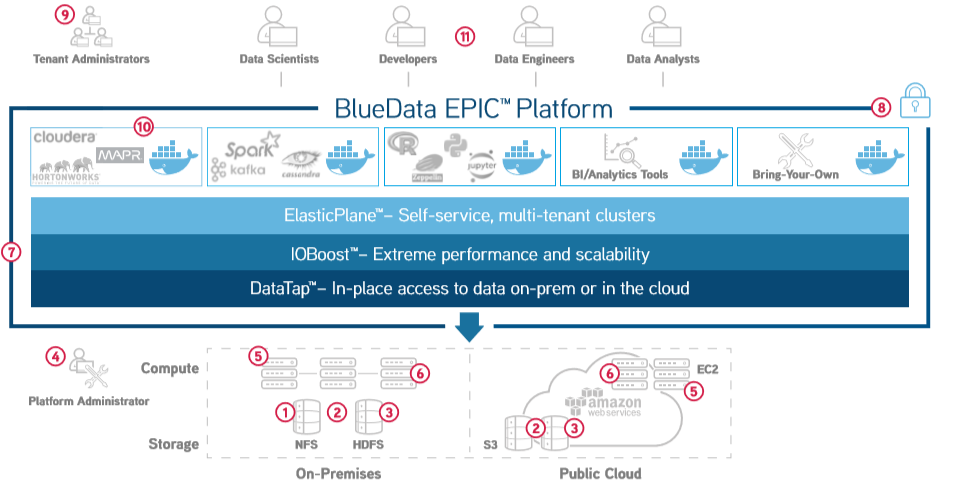
1. Pick a Blueprint，Cloudbreak使用Ambari Blueprints来声明Hadoop集群的定义
2. Choose a Cloud，选择云基础服务来创建集群，包括Server、网络和安全配置
3. Launch Cluster，Cloudbreak获取Docker Container资源后，启动Apache Ambari并且应用Blueprint启动集群。

https://hortonworks.com/open-source/cloudbreak/#blog

https://www.oschina.net/translate/docker-ships-hadoop-to-the-cloud?cmp

## 2.2 BlueData EPIC Lite

EPIC Lite使用Docker Containe在单主机或者AWS上运行虚拟的Hadoop或者Spark集群，在集群内部整合了Apache Ambari和Cloudera Manager，支持通用大数据分析应用和“自建应用”能力，其目标就是在企业内部快速部署Hadoop环境。EPIC Lite系统架构图如下所示：



在EPIC中，文件系统部署在虚拟集群中，节点存储空间使用EPIC Host或者Amazon S3 Bucket，EPC Host可以是运行Linux的物理机或者AWS EC2 主机，在每个主机上都安装EPIC Services。EPIC Lite从App Store中获取运行服务的Docker镜像，目前支持的有：

• Cloudera CDH 5.7 with Cloudera Manager (CDH 5.8 and 5.9 are

available upon request)

• Hortonworks HDP 2.4 with Ambari 2.2

• MapR 5.1 with MapR Control System

• Apache Spark 2.0.1 with Zeppelin Notebook

• Apache Spark 1.6.1 with Job Server and Zeppelin Notebook

• DataStax Distribution of Apache Cassandra 3.9

• DataStax Distribution of Apache Cassandra 2.1.10

• Apache Kafa 0.9.0.1

• Ubuntu 16.04 (Xenial Xerus)

• CentOS 7 / CentOS 6

• RHEL 7 / RHEL 6

用户也可以自定义Docker镜像。

在前期，BlueData平台是基于OpenStack，使用KVM虚拟机，2016年后引入Docker容器技术，目前使用Kubernetes。

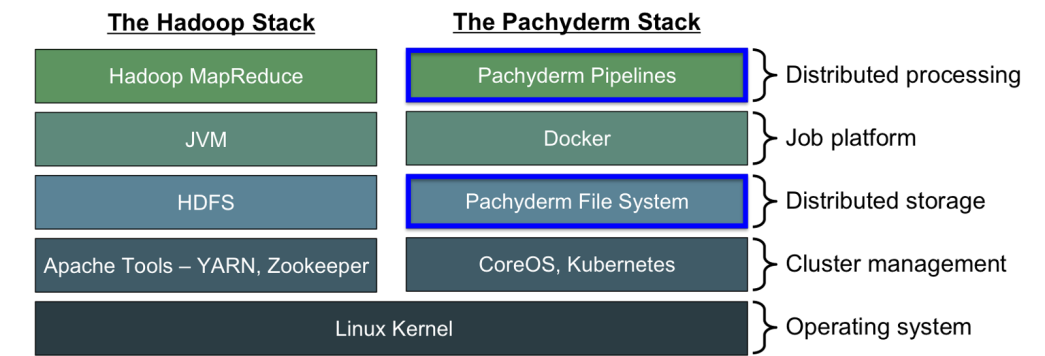
https://www.bluedata.com/product/architecture/

http://info.bluedata.com/rs/693-TGY-247/images/BD-EPIC-AWS-Spec-R7\_Update.pdf

http://www.jifang360.com/news/201579/n124769831.html

## 2.3 Pachyderm

Pachyderm将Docker和Hadoop结合起来，使得大数据的分析过程更加敏捷，其使用Kubernetes和CoreOS的etcd来管理集群，Pachyderm是一个架构在Hadoop和Docker上全新的存储和计算引擎，其功能框架图如下所示：



Pachyderm Pipelines处理数据时只需要创建一个容器化的应用并从本地文件系统上读写数据即可。

http://blog.csdn.net/u012331758/article/details/78106751

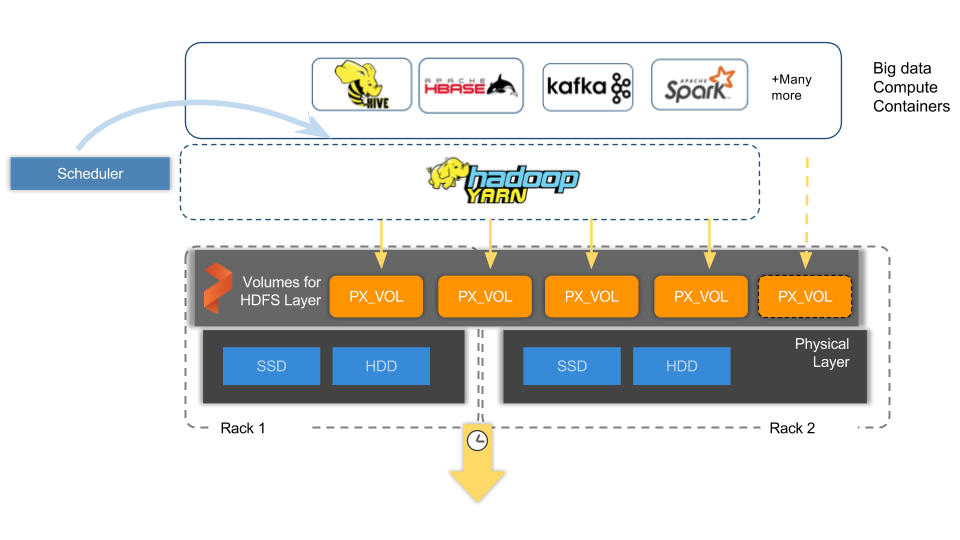
https://www.infoq.com/news/2015/02/pachyderm-build-modern-hadoop

https://github.com/pachyderm/pachyderm

http://pachyderm.readthedocs.io/en/latest/deployment/deploy\_intro.html

**2.4 Hadoop on Docker With Portworx**

使用Kubernetes、Mesos及Docker Swarm等编排工具构建Hadoop（On Docker）集群，都会影响Hadoop的数据本地性，从而降低Job的性能，Portworx可以解决这个问题，使用Portworx的Hadoop on Docker系统架构图如下所示：



使用Portworx来辅助Hadoop on Docker的优点如下：

* Hadoop节点（DN,NN及JN）的快速恢复
* 提高运行在同一个节点上的Hadoop集群的资源利用率，基于数据卷将集群的节点均衡分布到不同主机上
* 使Hadoop具有数据本地性，提高作业性能，Portworx可以使MR Task运行在有数据副本上，不管数据使用哪种文件系统（Ceph、Gluster），也支持Amazon EBS（网络延迟）
* 可以在线动态的扩展HDFS Volumes
* 使用Portworx框架，简化Hadoop的安装和配置
* 使用Hadoop透明的使用云存储，管理方式与本地存储一致

https://docs.portworx.com/applications/hadoop-docker.html

# 3.Hadoop on Kubernetes

使用Docker启动Hadoop集群可以简化大数据平台的应用开发部署，使研发和测试团队集成交付更加敏捷高效，大数据应用开发人员倾向于利用Docker来运行应用程序，目前在云上提供Hadoop集群的产品比较多，但是开销比较大，因此直接在私有集群的裸机上使用Docker来提供Hadoop集群服务也是一种选择。构建Hadoop On Docker集群需要考虑以下因素：

* 使用Docker启动Hadoop集群，需要跨主机多Container组成一个服务，需要涉及到容器网络、调度、监控与容错等，因此需要选择容器编排工具，目前主流的编排工具包括：Swarm,Fleet,Kubernetes及Marathon等，其中Kubernetes带有服务发现和复制等功能，可以提供一个可容错和可扩展的系统，已经成为编排工具的首先。
* Hadoop集群本身的管理问题：多Docker构成了一个服务的载体，但是用户无法感知其中运行的Hadoop集群状况；Hadoop生态系统包括多个应用，例如HBase,Spark,Hive及其他大数据分析工具。如果要启动一个长时间可用的Hadoop套件集群，必须通过Hadoop集群管理工具来实现，目前主流的工具是Apache Ambari和Cloudera Manager。
* 其他辅助工具：为了保证Hadoop数据本地性可以使用Portworx

根据前两章的内容分析和总结，使用Kubernetes启动Hadoop on Docker集群的方案图如下：

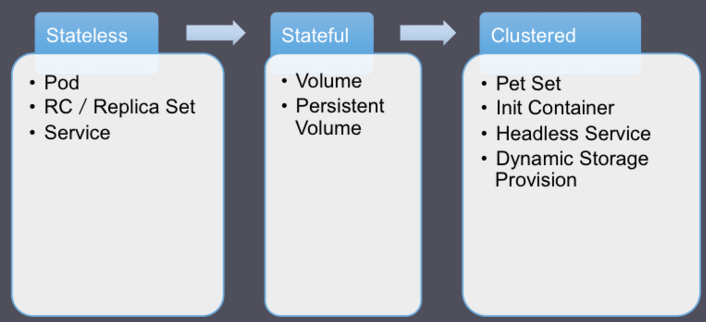


使用Kubernetes搭建长运行的Hadoop集群还要解决下面两个问题：

1. **Hadoop集群的运行状态处理**

在Kubernetes中运行的服务从简单到复杂分为三类：无状态服务，例如nginx或者Web Server，这些服务本身不需要保持数据，可以随意的扩容或缩容；普通有状态服务，例如MySQL，需要保持运行过程中产生的数据; 有状态的集群服务，这类服务需要启动多个Container，每个成员需要标识来形成群组关系。

Kubernetes针对这三种服务进行启动方式不同，如下图所示：

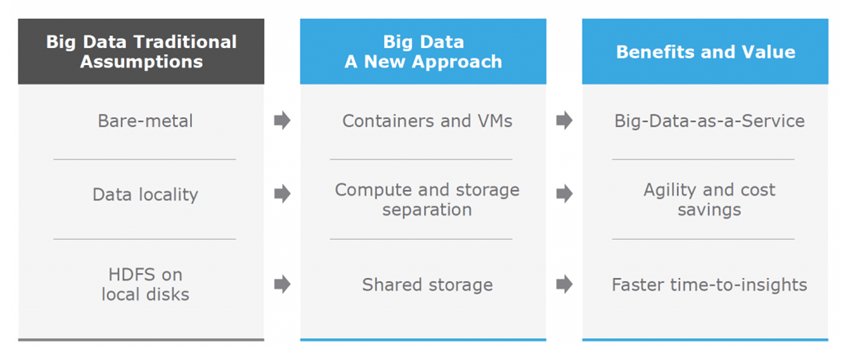


Hadoop生态系统包括Hadoop、Spark、Kafka、BI、机器学习、数据科学等多种服务。这些服务中一些服务如Spark符合Kubernetes支持的无状态、无服务架构，但是大多数系统是有状态的，例如Kafka 节点需要BrokerId，ZK需要myid来组成集群式服务，将这些有状态的服务运行在Kubernetes中需要一定的工作量。

1. **Hadoop计算与存储解耦**

在传统Hadoop集群中，服务部署在物理裸机上，数据存储到DAS挂载的磁盘上。启动的Job具有数据本地性的优势，Task启动时将其分配到数据所在的节点上，减少了网络造成的延迟。但是当前网络设备的发展比磁盘的速度快，10GB网络已经比较常见，数据本地性带来的优势越来越小，可以考虑将计算和存储解耦。

计算和存储解耦后，Hadoop的服务化构建成本会更低，其带来的优势也会更明显，



将计算和存储解耦后，使用Docker部署Hadoop节点，不再需要考虑数据的加载或者同步，节点通过共享文件系统使用数据，节点的宕机及迁移不用担心数据的丢失及节点之间数据的均衡造成的开销。Hadoop分布式文件系统协议本身只是接口，实现该接口的存储系统都可以作为Hadoop底层的存储，如下图：



这些文件系统都扩展自抽象基础类FileSystem，该类提供了用于文件系统的操作方法，包括创建目录、删除文件、重命名等，FileSystem通过反射机制来获取相应文件系统的对象。Hadoop Trunk中已经整合了众多的文件系统，下图是目前Hadoop的FileSystem的类层次关系：



将Hadoop计算和存储解耦后，Hadoop on Docker的部署图如下所示：



将Hadoop的计算和存储解耦后，支持多集群会变得容易，任何现有或将来的集群都可以共享文件系统来使用数据，集群迁移也可以将数据无丢失的复位到目标集群。集群在动态的扩展或者缩少时不用担心丢失数据。

解耦后带来的缺点失去计算的本地性的，在Hadoop on Kubernetes社区中有相关的项目来解决这个问题：

<https://github.com/apache-spark-on-k8s/kubernete>-HDFS

其核心实现是重构了机架感知策略，并且将Block副本放置配置为：

org.apache.hadoop.hdfs.server.blockmanagement.BlockPlacementPolicyWithNodeGroup