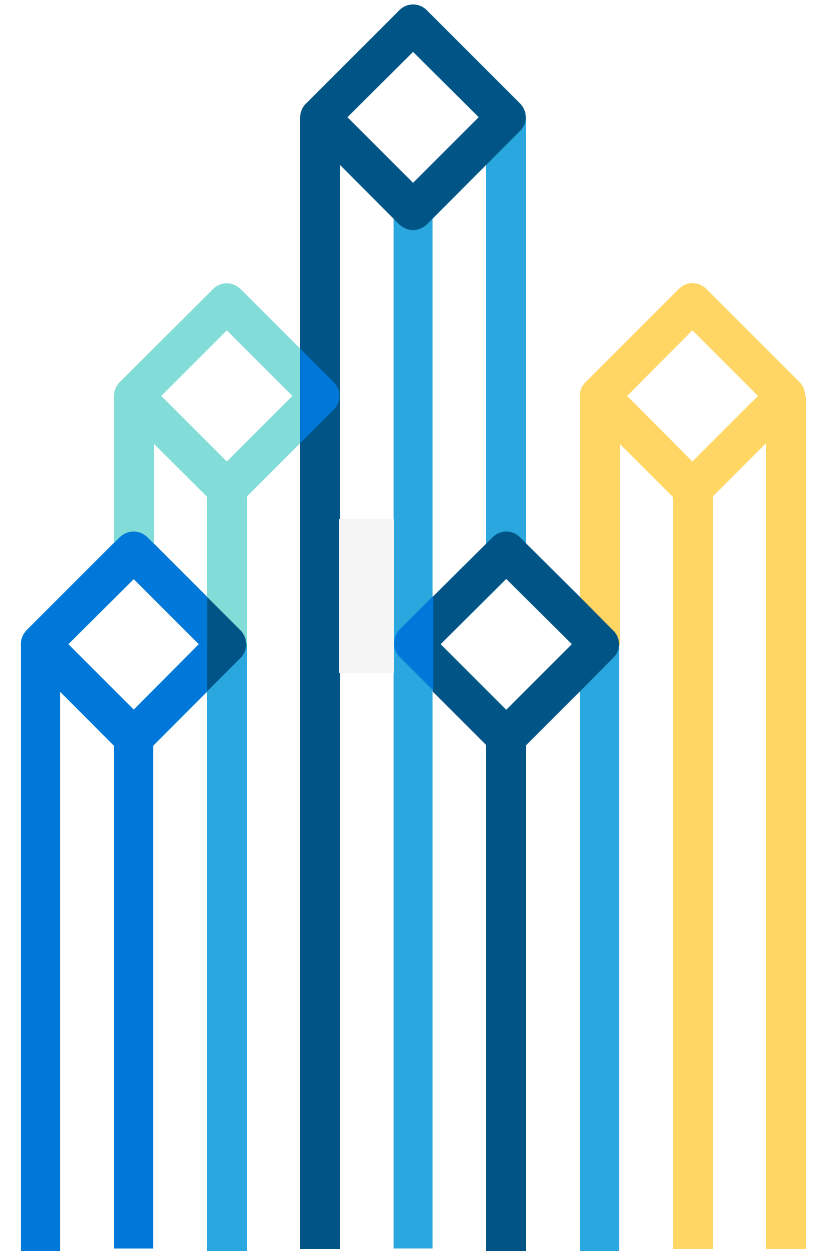




Hadoop 10th Birthday and Hadoop 3 Alpha

Dongjin Seo – Cloudera Korea, SE

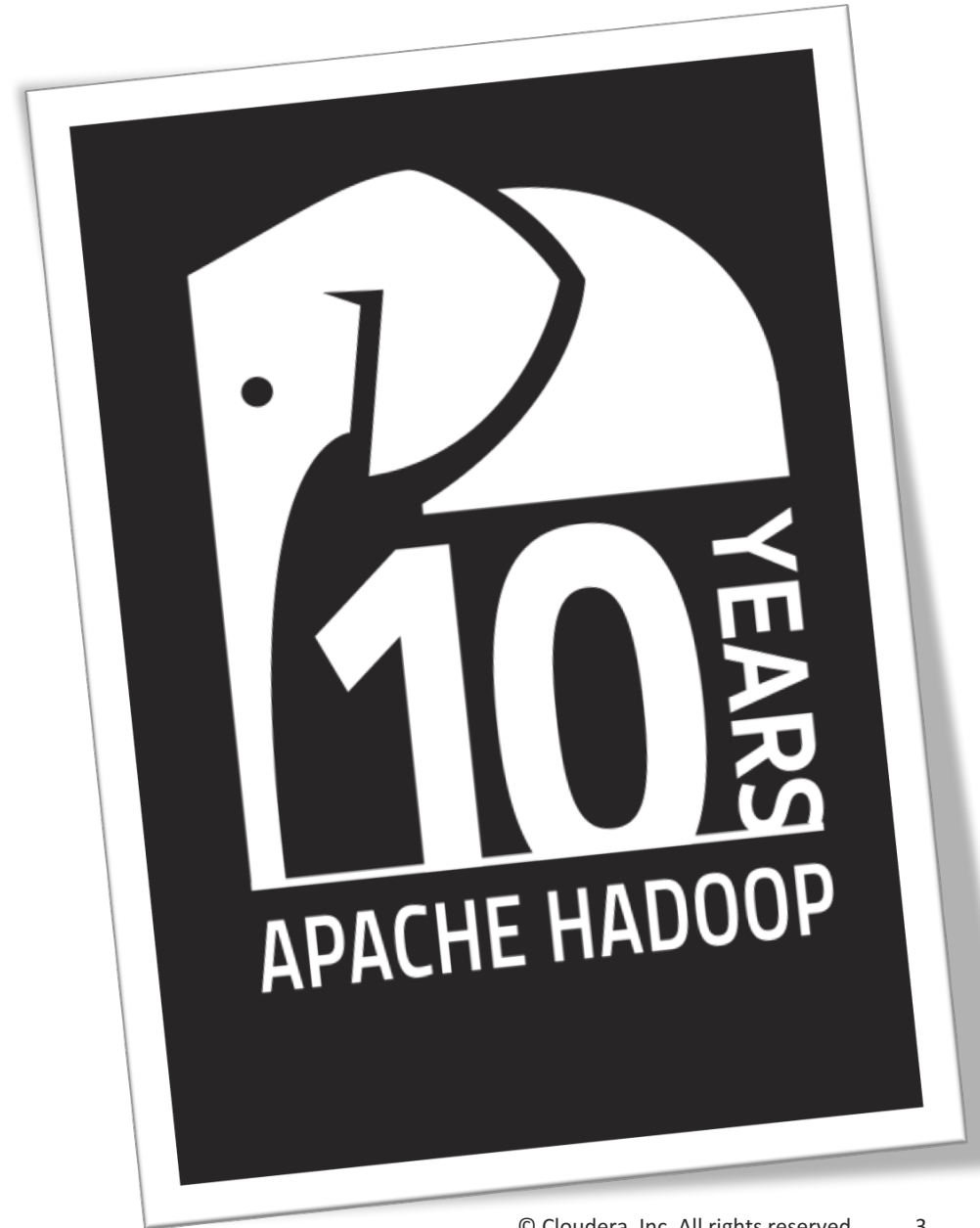


Agenda

I. Hadoop 10th Birthday

II. Hadoop 3 Alpha

Apache Hadoop at 10



Apache Hadoop's Timeline

The Invention Years 2002 ~ 2004

[2002]

- Doug Cutting and Mike Cafarella create Nutch, an open source web crawler

[2003]

- Google publishes its "Google File System" paper

[2004]

- Cutting & Cafarella implement Nutch features that will become HDFS
- Google publishes its "MapReduce" paper

The Incubation Years 2005 ~ 2007

[2005]

- Cafarella spearheads an implementation of MapReduce in Nutch

[2006]

- Cutting joins Yahoo!; starts Hadoop subproject by carving code from Nutch
- **First Apache release of Hadoop**

[2007]

- First Hadoop User Group meeting
- Community contributions begin to rise steeply

The Coming-Out Years 2008 ~ 2009

[2008]

- **Hadoop becomes a Top Level ASF project**
- Yahoo! launches world's largest Hadoop application
- **Hive**, Hadoop's first SQL framework, becomes a Hadoop sub-project
- Cloudera, first company to commercialize Hadoop, is founded

[2009]

- Cutting joins Cloudera as its chief architect

The Rapid Adaption Years 2010 ~ 2015

[2010-11]

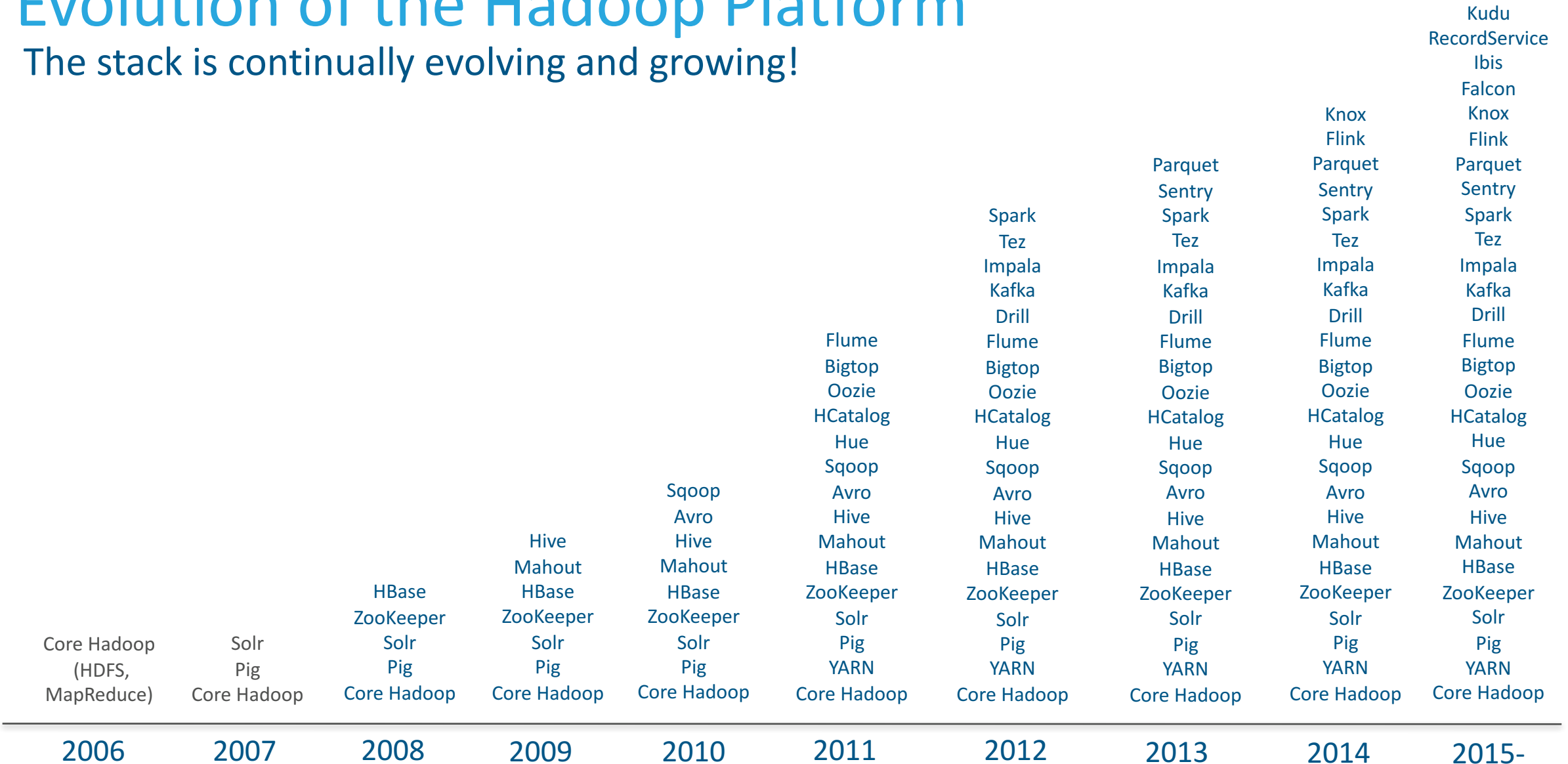
- The extended Hadoop community busily builds out a plethora of new components (Crunch, Sqoop, Flume, Oozie, etc)

[2012-15]

- HDFS NameNode HA, YARN, significant new features for enterprise adoption
- **Impala** joins the ecosystem
- **Spark** becomes a Top Level ASF project
- **Kudu**, the first native storage option for Hadoop since HBase, joins the ASF Incubator

Evolution of the Hadoop Platform

The stack is continually evolving and growing!



Why Did Hadoop Succeed?



Open source community and license

A large and diverse community of developers has historically made, and continues to make, the Hadoop ecosystem among the most active and engaged in history, while the Apache License lowers the barrier to entry for users.



Extensibility/adaptability

With the possible exception of Linux, no other complex platform has evolved on so many levels, and so quickly, to meet user requirements over time.



A strong focus on systems

The roots of Hadoop are in making distributed computing infrastructure more accessible by application developers. That continuing focus continues to bear fruit in areas like resource management and security.

Apache Hadoop 3 Alpha Release!!

Major new changes



Release Date: 03 September, 2016
Changes: about 3,000



HDFS Erasure Coding (HDFS-EC)



YARN Timeline Service v.2



Shell Script Rewrite



MapReduce task-level native optimization



Support for Multiple Standby NameNodes



Java 8 Minimum Runtime Version



New Default Ports for Several Services



Intra-DataNode Balancer



Reworked daemon and task heap management

Apache Hadoop 3 Alpha Major New Changes

Major new changes



Release Date: 03 September, 2016

Changes: about 3,000



HDFS Erasure Coding (HDFS-EC)

Erasure Coding ?

- Fault-tolerance를 위한 데이터 보존 기법 중 하나로 흔히 RAID-5에서 사용되는 기법입니다. 데이터 저장 시 EC Codec으로 데이터를 균일한 사이즈의 Data cell/Parity cell로 인코딩하며 이와 반대로 데이터 로드 시 Data cell과 Parity cell로 구성된 EC Group에서 유실된 cell에 대해서 해당 그룹에 남아있는 cell들로부터 재구성하여 원본 데이터 복구하는 디코딩 작업을 실행

HDFS-EC

- ✓ EC의 Reed-Solomon algorithm을 수행하는 Intel ISA-L (Intelligence Storage Acceleration Library) 사용하여 스토리지 성능, 처리량, 보안, 안정성 개선
- ✓ EC는 Exclusive-OR 공식 기반이지만 Multiple failure 을 보장하지 못하는 불안정한 부분이 존재하여, Reed-Solomon algorithm을 적용하여 Multiple failures를 보장
- ✓ Cloudera + Intel + Hadoop Community 합작하여 빌드
- ✓ 각 개별 디렉토리에 'hdfs erasurecode -setPolicy' 커맨드로 policy 적용

Erasure Coding의 필요성

- ✓ 복제개수 3개는 장애대응에 용이하나 기회비용이 비쌈 (200% overhead in storage space and other resources (e.g. network bandwidth when writing the data))
- ✓ 일반적인 Operations 에서는 추가적으로 복제된 블록에 대해 접근을 잘 하지 않음

Erasure Coding의 효과

- ✓ 더 적은 용량으로도 Fault-tolerance 보장
- ✓ 3x replication 에 비해 storage cost ~50% 감소 효과

Apache Hadoop 3 Alpha Major New Changes

Major new changes



Release Date: 03 September, 2016
Changes: about 3,000



YARN Timeline Service v.2

YARN Timeline Service v.2

- Event, Metrics와 같은 컨테이너 관련 정보 및 Map, Reduce Task 관련 Application 정보를 WebUI를 통해 확인 할 수 있도록 제공되는 서비스

1) Improving scalability and reliability of Timeline Service

✓ 확장성이 높고, 분산 저장 가능한 아키텍처(e.g. HBase)를 채택하여 신뢰성 향상

2) enhancing usability by introducing flows and aggregation

✓ YARN Application 의 단계별 논리적 Flow 제공
✓ Flow 레벨에서의 metrics에 대한 Aggregation 지원



Shell Script Rewrite

✓ 오랫동안 가지고 있던 버그 수정 및 새로운 기능 추가를 위해 Hadoop 셸스크립트가 수정 됨
✓ 기존 셸스크립트 버전에 대해 완벽하게 호환되지 않아 Hadoop 환경변수를 사용하거나 shell command를 이용하는 사용자에게 영향주는 부분에 대한 검토가 필요

more info:

<https://issues.apache.org/jira/browse/HADOOP-9902>

<https://issues.apache.org/jira/secure/attachment/12599817/more-info.txt>

Apache Hadoop 3 Alpha Major New Changes

Major new changes



Release Date: 03 September, 2016
Changes: about 3,000



MapReduce task-level native optimization

NativeTask ?

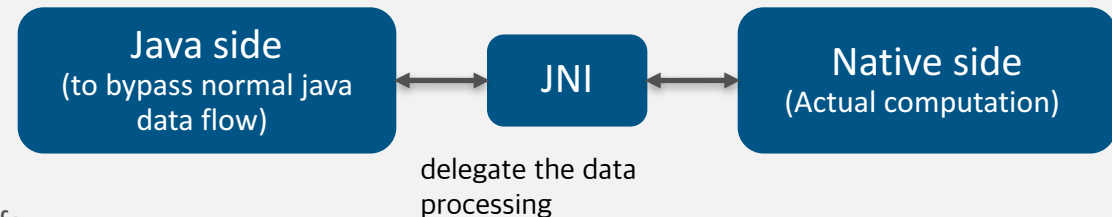
- 데이터 프로세싱에 초점을 맞춘 native computing unit으로 Hadoop MapReduce를 위한 고성능 C++ API & runtime

NativeTask의 필요성

- ✓ **I/O bottleneck.** Most Hadoop workloads are data intensive, so if no compression is used for input, mid-output, and output, I/O(disk, network) could be a bottleneck.
- ✓ **Inefficient implementation.** (Map side sort, Serialization/Deserialization, Shuffle, Data locality, Scheduling & starting overhead)
- ✓ **Inflexible programming paradigm.** - limits its performance

NativeTask의 효과

- ✓ NativeTask 를 map output collector에 적용하여 shuffle-intensive job의 성능을 30% 이상 향상



more info:

<https://issues.apache.org/jira/browse/MAPREDUCE-2841>

Apache Hadoop 3 Alpha Major New Changes

Major new changes



Release Date: 03 September, 2016
Changes: about 3,000



Support for Multiple Standby NameNodes

- ✓ 기존 single Active Namenode, single Standby Namenode로만 구성이 가능했던 아키텍처에서 다중 Standby Namenode로 구성이 가능하도록 변경
 - ✓ 더 높은 수준의 High-Availability를 제공
 - ✓ Namenode 는 5개 초과 X, 3개를 추천함 → communication overheads 고려
- more info:
<http://hadoop.apache.org/docs/r3.0.0-alpha1/hadoop-project-dist/hadoop-hdfs/HDFSHighAvailabilityWithQJM.html>



Java 8 Minimum Runtime Version

- ✓ Java 7에 대한 오라클의 공식적인 지원이 종료(April 2015)되어 Java 8으로 변경할 수 밖에 없는 상황이 됨
- ✓ 이에 따라 Hadoop 3의 최소 자바 버전은 Java 8 로 변경 됨

Apache Hadoop 3 Alpha Major New Changes

Major new changes



Release Date: 03 September, 2016
Changes: about 3,000



New Default Ports for Several Services

- ✓ Hadoop 시작 시 bind error를 피하기 위해 Namenode, Secondary NN, Datanode, KMS의 port를 변경
 - ✓ 대규모 클러스터에서의 rolling restart의 신뢰성 향상에 기여할 것으로 예상
- Namenode ports: 50470 --> 9871, 50070 --> 9870, 8020 --> 9820
Secondary NN ports: 50091 --> 9869, 50090 --> 9868
Datanode ports: 50020 --> 9867, 50010 --> 9866, 50475 --> 9865, 50075 --> 9864
KMS port: 16000 --> 9600

more info:

<https://issues.apache.org/jira/browse/HDFS-9427>

<https://issues.apache.org/jira/browse/HADOOP-12811>



Intra-DataNode Balancer

- ✓ 디스크 추가/변경으로 발생하는 Datanode 내의 디스크들에 대한 데이터 적재 불균형을 HDFS Command(diskbalancer)로 해결

more info:

<http://hadoop.apache.org/docs/r3.0.0-alpha1/hadoop-project-dist/hadoop-hdfs/HDFSCommands.html>

Apache Hadoop 3 Alpha Major New Changes

Major new changes



Reworked daemon and task heap management

- ✓ 호스트의 메모리 사이즈를 기반으로 자동으로 튜닝해주는 기능 추가
- ✓ HADOOP_HEAPSIZE는 deprecated 됨
- ✓ 기존에 비해 간단하게 map/reduce heap 사이즈 설정이 가능하게 되어 요구되는 heap사이즈를 task설정이나 Java 옵션으로 명시해야되는 수고가 없어졌음

more info:

<https://issues.apache.org/jira/browse/HADOOP-10950>

<https://issues.apache.org/jira/browse/MAPREDUCE-5785>



3.0.0-alpha1

Release Date: 03 September, 2016

Changes: about 3,000

Conclusion

- ❖ Hadoop의 핵심인 Component(HDFS, MARECUDE)의 큰 변화는 또 다른 큰 발전을 위한 발돋움
 - ✓ HDFS 저장공간 활용도 증가 → ROI 증가, TOC 감소 → 초기 도입에 대한 장벽이 낮아짐
 - ✓ Multiple Standby Namenode, heap 튜닝 기능 → 운영환경에 대한 용이성 증가 → 운영에 투자되는 비용을 클러스터 확장, 개선과 같은 곳에 투자할 수 있는 기회 증가
- ❖ 변화한 Hadoop과 연관 Component들의 변화가 기대 됨
 - ✓ HIVE 의 Query 속도 개선?
 - ✓ HBase의 Thruthput 증가?
 - ✓ 데이터 압축 효율성 증가?
 - ✓ etc
- ❖ 이러한 큰 변화 뒤에는 Community와 구성원들의 관심과 사랑이 있기에 가능



cloudera

감사합니다

djseo@cloudera.com