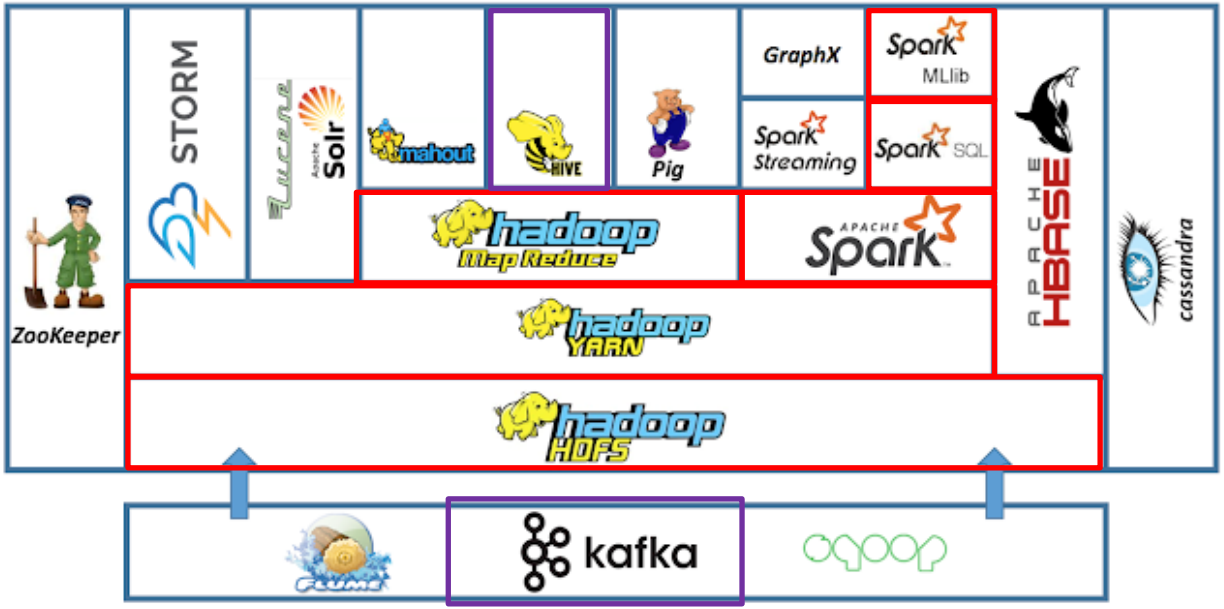
Windows 하에서의 빅데이터통계분석

숙명여대 여인권
inkwon@sookmyung.ac.kr

Hadoop Ecosystem

- 하둡 환경하에 빅데이터를 효율적으로 처리하기 위해 만들어진 서브 프로젝트의 집합



Hadoop Ecosystem

Data 수집 및 이관

- Sqoop: RDBMS ⇔ HDFS 사이에서 데이터를 이관(Retired)
- Flume: 다양한 로그 데이터 수집 및 모니터링
- Kafka: 실시간으로 스트림을 처리하는 분산 데이터 스트리밍 플랫폼

Hadoop Ecosystem

Data 저장

- **Hadoop HDFS**(Hadoop Distributed File System)
- Hbase: HDFS 상에 만들어진 비관계형 오픈소스 DB
- Cassandra: NoSQL 오픈소스 DB
- Hive: HDFS에 적재된 파일을 SQL 형식(HiveQL)으로 처리한 DW

Hadoop Ecosystem

Data 처리

- MapReduce: 병렬 프로그래밍으로 클러스터 내에서 분산 처리
- Yarn: 하둡 내 작업 스케쥴링, 클러스터 리소스 관리를 위한 프레임워크
- Spark: In-memory 기반 병렬 분산 처리하는 통합 컴퓨팅 엔진
- Pig: Hadoop에 기반하여 병렬로 데이터를 처리하는 엔진(Retired ?)
- Mahout: 하둡 내에서 분산처리가 가능하고 확장성을 가진 ML 라이브러리
- Storm: 클로저 프로그래밍 언어로 작성된 분산형 스트림 프로세싱 연산 프레임워크

Hadoop Ecosystem

Scheduling

- Zookeeper: 네이밍 레지스트리를 제공하는 중앙 집중식 서비스(분산 코디네이터)
- Oozie: 하둡의 Job을 관리하기 위한 서버 기반 워크플로우 스케쥴링 시스템

프로그램 설치

빅데이터 통계분석의 최적 환경

- Unix-like OS(Linux, Mac OS) + scala
  - 대부분의 통계학 전공자에게는 창문 밖 세상
- 이번 시간에는 Windows 하에서 Python으로 …

프로그램 설치

프로그램 설치 시 주의할 점

- 빅데이터 분석 도구 간 의존성(dependency) 확인
- 버전 간 호환이 안되면 설치 및 작동이 안되거나 오류가 발생
  - Python: 최신버전 3.11.x, Anaconda는 3.10.x 기반
  - Cassandra 3.x : batch 파일 제공, python 2.7
    Cassandra 4.x : batch 파일 미제공, python 3.8, scala 1.12
    ⇒ docker로 설치
  - 설치된 spark 버전과 python의 pyspark 버전이 다르면 오류발생

프로그램 설치

프로그램	최신버전(6/27)	의존성
Python	3.11.4	Anaconda: python 3.10
Scala	3.3.0	
Hadoop	3.3.6	Winutil 3.2.2 & 3.3.1
Spark	3.4.1	Hadoop 3.2 & 3.3+scala 2.12 & 2.13
Kafka	3.5.0	Scala 2.12 & 2.13 3.3.+(Zookeeper vs Kraft)
Cassandra	4.1.2	Stable 4.0.10(docker) Python 3.7 & 3.8, scala 2.11 & 2.12 Spark 1.0~3.3(4.0:3.2, 4.x: 3.3)
HBase	2.5.5 3.0.0-α-4	Hadoop 3.2.3+ & 3.3.2+ 2.4.x: Hadoop 3.1.1+
Hive	3.1.3 4.0.0-alpha-2	Hadoop 3.x & Cygwin(Linux) 4.0.0-alpha-2: Hadoop 3.3.1

프로그램 설치

Java 설치

- Windows용 JDK 다운로드 후 설치
  - Java SE Development Kit 8 - Downloads – Oracle
  - C:₩java 폴더 생성 후 C:₩Program Files₩Java₩jdkX.X.X_X 복사
- 시스템 환경 변수 편집
  - Linux에서 .bashrc 파일 수정과 동일한 작업
  - 환경변수의 사용자변수 ⇒ 변수: JAVA_HOME, 값: C:₩java
  - 환경변수의 시스템변수 ⇒ Path : C:₩java₩bin, C:₩java₩jre₩bin
- 버전 확인
  - CMD> javac -version

프로그램 설치: Hadoop

Hadoop 다운로드, 압축해제, 환경설정

- Hadoop 3.3.1.tar.gz 압축 풀고 C:₩hadoop-3.3.1에 저장
- 시스템 환경 변수 편집
  - 사용자변수 ⇒ 변수: HADOOP_HOME, 값: C:₩hadoop-3.3.1
  - 사용자변수 ⇒ 변수: HADOOP_USER_NAME, 값: C:₩Users₩***
  - 시스템변수 ⇒ Path : C:₩hadoop-3.3.1₩bin

namenode와 datanode를 생성할 위치 지정

- d:₩hadoopDisk₩data 폴더 생성

Windows용 실행 및 설정 파일 복사

- Hadoopwin-3.3.1의 bin 폴더 내 파일을 c:₩Hadoop-3.3.1₩bin에 복사

프로그램 설치: Hadoop

설정파일 수정

- c:₩Hadoop-3.3.1₩etc₩hadoop으로 이동
- hadoop-env.cmd, core-site.xml, hdfs-site.xml, mapred-site.xml, yarn-site.xml 수정
  - core-site.xml: HDFS와 맵리듀스에서 공통적으로 사용되는 IO 설정 같은 하둡 코어 환경정보 설정
  - hdfs-site.xml: 파일복제 옵션, 저장위치 지정
  - mapred-site.xml: 맵리듀스 설정 관리
  - yarn-site.xml: yarn(맵리듀스기반 하둡의 문제(리소스 할당문제나 Spark 등의 새로운 플랫폼 출현)를 해결하기 위한 하둡의 서브 프로젝트) 설정 관리

프로그램 설치: Hadoop

Hadoop format

- CMD에서 실행

```
> hdfs namenode -format
```

  - d:₩hadoopDisk₩data에 namenode 생성
- %HADOOP_HOME%₩sbin 폴더로 이동
- namenode와 datanode 시작(관리자 권한 CMD)

```
> start-dfs.cmd
> jps
```

  - d:₩hadoopDisk₩data에 datanode 생성
  - localhost:9870 접속

프로그램 설치: Hadoop

MapReduce 실행

- ResourceManager와 NodeManager 시작(관리자 권한 CMD)

```
> start-yarn.cmd
> jps
```

  - localhost:8088, localhost:8042 접속

일괄 실행 및 종료

```
> start-all.cmd
> stop-all.cmd
```

- start-all.cmd 권장하지 않음

하둡 주요 명령어

주요 명령어

- (관리자 권한) CMD 실행
  - **파일 삭제는 반드시 관리자 권한**
- Usage : hadoop fs [GENERIC_OPTIONS] [COMMAND_OPTIONS]
  - hadoop fs –mkdir –p {paths}
  - hadoop fs –put {localsrc} ... {dst}
  - hadoop fs –ls -R {args}
  - hadoop fs –get -crc {src} {localdst} .
  - hadoop fs –rm –f -R URI ..
  - hadoop fs –cp URI ..

MapReduce 예제

Word Count

- "worddata.txt"를 /Input에 put하고 계산 결과를 /Output에 출력
- (관리자 권한) CMD 실행

```
> hadoop fs –mkdir /Input
> hadoop fs –put worddata.txt /Input
> cd %HADOOP_HOME%₩share₩hadoop₩mapreduce
> hadoop jar hadoop-mapreduce-examples-3.3.1.jar wordcount
/Input/worddata.txt /Output
> hadoop fs –cat /Output/*
```

- 폴더에 여러 파일이 있는 경우 폴더만 쓰면 모든 파일을 분석함
- localhost:9870 ☞ Utilities ☞ Browse the filesystem

프로그램 설치: Spark

Spark 다운로드, 압축해제, 환경설정

- Hadoop에 맞는 Spark 선택(3.4.0) 후 압축해제 C:₩spark-3.4.0에 저장
- 시스템 환경 변수 편집
  - 사용자변수 ⇒ 변수: SPARK_HOME, 값: C:₩spark-3.4.0
  - 시스템변수 ⇒ Path : C:₩spark-3.4.0₩bin;
    C:₩Program Files₩R₩R-X.X.X₩bin

Spark 실행 : CMD에서

- spark-shell/pyspark/spark 실행
- localhost:4040 확인

Python 패키지 : pip install pyarrow pyspark==3.4.0

Pyspark vs Pandas 자료 읽기 쓰기

- 읽기: pandas.read_xxx, spark.read.xxx, spark.read.format('xxx').load
- 쓰기: pandas: df.to_xxx, spark: df.write.xxx, df.write.format('xxx').save

xxx	pandas		pyspark		xxx	pandas		pyspark	
	읽기	쓰기	읽기	쓰기		읽기	쓰기	읽기	쓰기
avro			O	O	csv	O	O	O	O
excel	O	O			feather	O	O		
fwf	O	X			hdf	O	O		
html	O				json	O	O	O	O
ocr	O	O	O	O	**parquet**	O	O	O	O
pickle	O	O			sql	O	O	O	O
table	O	X	O		xml	O	O	O	O

프로그램 설치: Kafka

Kafka 다운로드, 압축해제, 환경설정

- Kafka 3.4.1(scala 2.12) 압축해제 C:₩kafka-3.4.1에 저장
- 시스템 환경 변수 편집
  - 사용자변수 ⇒ 변수: KAFKA_HOME, 값: C:₩kafka-3.4.1
  - 시스템변수 ⇒ Path : C:₩kafka-3.4.1₩bin₩windows

Kafka 환경 실행

- 3.3버전부터 Kraft 사용 가능
- ZooKeeper와 Kraft 중 하나만 사용

Python 패키지 : pip install kafka-python

- Kafka: ~2.4, python: ~3.8, coverage: 86%

프로그램 설치: Kafka

ZooKeeper를 이용한 Kafka 환경

- 관리자 권한 CMD 실행

```
> cd %KAFKA_HOME%
> bin₩windows₩zookeeper-server-start.bat config₩zookeeper.properties
```

```
> cd %KAFKA_HOME%
> bin₩windows₩kafka-server-start.bat config₩server.properties
```

Topic 생성

```
> cd %KAFKA_HOME%
> bin₩windows₩kafka-topics.bat --create --topic testevent --bootstrap-server localhost:9092
```

프로그램 설치: Kafka

Topic에 event 쓰기

```
> cd %KAFKA_HOME%
> bin₩windows₩kafka-console-producer.bat --topic testevent --bootstrap-server localhost:9092
> This is ....
```

Event 읽기

```
> cd %KAFKA_HOME%
> bin₩windows₩kafka-console-consumer.bat --topic testevent --from-beginning --bootstrap-server localhost:9092
```

종료: Ctrl+C

프로그램 설치: Kafka

기존 시스템의 자료 연결

- Plugin.path에 연결시키는 파일 추가

```
> cd %KAFKA_HOME%
> echo "plugin.path=libs₩connect-file-3.4.1.jar"
```

- config/connect-standalone.properties에 plugin.path=%KAFKA_HOME%₩libs₩connect-file-3.4.1.jar 추가할 수 있음

- 예제 데이터파일 만들기

```
> echo windows-kafka > test.txt
> echo test exmples >> test.txt
```

프로그램 설치: Kafka

Standalone 모드에서 connectors 실행

```
> cd %KAFKA_HOME%
> bin₩windows₩connect-standalone.bat config₩connect-standalone.properties config₩connect-file-source.properties config₩connect-file-sink.properties
```

```
> cd %KAFKA_HOME%
> bin₩windows₩kafka-console-consumer.bat --bootstrap-server localhost:9092 --topic connect-test --from-beginning
```

- 데이터 파일 확인 및 추가

```
> more test.sink.txt
> echo add data > test.txt
```

프로그램 설치: HIVE

Hive 다운로드, 압축해제, 환경설정

- Hive 3.1.2 압축해제 C:₩hive-3.1.2에 저장
- Apache Derby 압축해제 C:₩derby-10.14.2에 저장: Apache DB project
- 시스템 환경 변수 편집: 사용자변수
  - 변수: HIVE_HOME, 값: C:₩hive-3.1.2
  - 변수: DERBY_HOME, 값: C:₩derby-10.14.2
  - 변수: HIVE_LIB, 값: %HIVE_HOME%₩lib
  - 변수: HIVE_BIN_PATH, 값: %HIVE_HOME%₩bin
  - 변수: HADOOP_USER_CLASSPATH_FIRST, 값: true ⇒ 시스템변수
- 시스템 환경 변수 편집: 시스템변수
  - Path: C:₩hive-3.1.2₩bin; C:₩derby-10.14.2₩bin

프로그램 설치: HIVE

파일 복사

- Derby lib 폴더의 파일을 Hive lib에 복사
- Hive lib 폴더의 guava-19.0.jar를 guava-27.0-jre.jar(제공)로 대체
- Hive conf 폴더에 hive-site.xml(제공) 저장
- Hive bin 폴더에 hive-cmd파일들(제공) 저장

Cygwin 설치

- 설치과정에 vim 패키지 선택

프로그램 설치: HIVE

관리자 권한으로 Hadoop 실행: %HADOOP_HOME%₩sbin

```
> start-dfs
> start-yarn
```

Derby 실행

```
> startNetworkServer -h 0.0.0.0
```

Hive 실행

- C:₩와 D:₩(데이터드라이브, NTFS)에 cygdrive 폴더 생성
- 관리자 권한 CDM(symbolic links 생성)

```
> mklink /J D:₩cygdrive₩d₩ D:₩
> mklink /J C:₩cygdrive₩c₩ C:₩
```

프로그램 설치: HIVE

Cygwin 실행 후 vi .bashrc

```
export HADOOP_HOME='/cygdrive/c/hadoop-env/hadoop-3.3.1'
export PATH=$PATH:$HADOOP_HOME/bin
export HIVE_HOME='/cygdrive/c/hadoop-env/apache-hive-3.1.2'
export PATH=$PATH:$HIVE_HOME/bin
export HADOOP_CLASSPATH=$HADOOP_CLASSPATH:$HIVE_HOME/lib/*.jar
```

- 저장(Esc ⇒ : ⇒ wq)후 source .bashrc 실행

Hive scripts₩metastore₩upgrade₩derby₩hive-schema-3.1.0.derby.sql 수정

- 'NUCLEUS_ASCII' & 'NUCLEUS_MATCHES' function 주석처리

프로그램 설치: HIVE

Cygwin에서

```
$ schematool -dbType derby –initSchema
$ hive --service hiveserver2 start
```

- Hive-schema-3.1.0.-derby.sql 수정하지 않으면 Metastore 초기화에서 오류 발생
- 관리자 권한으로 새로운 Cygwin 실행

```
$ hive
```