

11. 하이브(Hive)와 임팔라(Impala)

하이브(Hive)

1. 특징

SQL On Hadoop 분석 도구인 Hive와 Impala는 어떤 차이가 있을까?

Hive : 특징

[1] Hive QL이라고 불리는 SQL 같은 언어를 제공하여 Hadoop 데이터(파일)를 Query를 이용해서 분석할 수 있게 해주며 Map-Reduce의 모든 기능을 지원합니다.

[2] Oracle DBMS가 Data Dictionary를 통해 Table Schema 정보를 관리하고 참조하듯이, Hive 메타스토어를 통해 Table, Column 정보를 관리하며, 실행데이터는 Hadoop HDFS에 저장합니다.

[3] **Hive Query는 Map-Reduce로 변환되어 실행됩니다.** Hive는 Query를 파싱, 실행계획 수립, 최적화 과정을 거쳐 Map-Reduce로 변환하여 처리하므로, 응답시간이 매우 길며, 대량 데이터의 Full-Scan에 최적화되어 있습니다. Map-Reduce 사용시 다수의 사용자 요청을 동시에 처리하는데 한계가 존재합니다.

[4] **대화형의 Online Query 사용에 부적합합니다.** Hive는 배치처리 기반의 Map-Reduce를 통해 데이터를 처리하므로, 사용자가 애드혹(Ad-Hoc) Query를 실행하거나, BI/시각화 도구, 어플리케이션에서 대화형 방식으로 분석할 경우 느립니다. 또한 전체 정렬을 단순 Hive QL로 실행하면 하나의 Reduce만 실행되어 급격한 성능 저하가 발생합니다.

[5] **데이터의 부분적인 수정/삭제(Record별 Update, Delete)가 불가능합니다.**

데이터의 부분적인 수정 불가는 Hadoop HDFS의 특징이며, Hive는 데이터를 HDFS에 저장하므로 HDFS의 특징을 그대로 계승합니다. 그로 인해 Hive Table에서는 Record별 수정하거나 삭제할 수 없고, 전체 데이터를 덮어쓰기 해야 하며, Hive로 처리할 데이터를 적재할 때는 특정 Partition 단위로 적재해 필요시 해당 Partition만 덮어쓰는 방식으로 데이터 관리구조를 설계해야 합니다. Hive 0.14부터는 INSERT, UPDATE, DELETE, MERGE 문을 ORC Stored 형식에서 부분적으로 지원합니다.

[6] **Transaction 관리 기능이 없어 롤백 처리 불가능합니다.** 나의 Hive Query는 여러 개의 잡(Job)과 Map-Reduce 프로그램으로 실행되며, 로컬 디스크에 중간 파일들을 만들어냅니다. 이때 특정 Map-Reduce 작업 하나가 실패하면 이미 성공한 잡들이 롤백 처리되지 않으며, 단지 실패한 잡을 재실행하기 위해 시도할 뿐입니다. 이러한 기본 원칙으로 인해 Hive에는 Transaction 관리 기능이 존재하지 않습니다. Hive 0.14부터는 ACID를 부분적으로 지원하고 있으나, 제약사항이 많아 자주 사용되지 않습니다.

[7] Hadoop HDFS, HBase(Column기반 데이터베이스), Accumulo(정렬된 분산 Column기반 저장소), Druid(BI/OLAP 전용 분석데이터 저장소), Kudu(빠른 Column기반 저장소) 등 다양한 저장소를 사용할 수 있습니다.

2. 데이터 모델

Table : 관계형 데이터베이스의 Table과 유사하고, Column을 포함하며, Table의 모든 데이터는 HDFS 디렉토리에 저장됩니다.

Partition : Table별 데이터가 저장되는 방식을 결정하며, 하나 이상의 Partition 키를 가질 수 있습니다.

예를 들어 Table T의 Column중 날짜 Column ds를 Partition키로 정의하고,

ds Column의 실제 데이터가 '2021-09-01'이라면, 화면에 보이는 것과 같이

"/user/hive/warehouse/production.db/T/ds=2021-09-01/" HDFS 디렉토리 하위에 파일로 저장됩니다.

Bucket : Table의 Column 해시를 기반으로 데이터를 지정된 개수의 파일로 분리해서 저장하며,

각 버킷은 Partition 디렉토리에 파일로 저장됩니다.

데이터 조회 및 Join시 필요한 버킷의 파일만 사용하여 Query 성능을 향상시키기 위함입니다.

임팔라(Impala)

: Hadoop에 저장된 데이터에 대한 빠른 조회용 SQL Query Engine으로, Hive와 유사한 역할

1. 특징

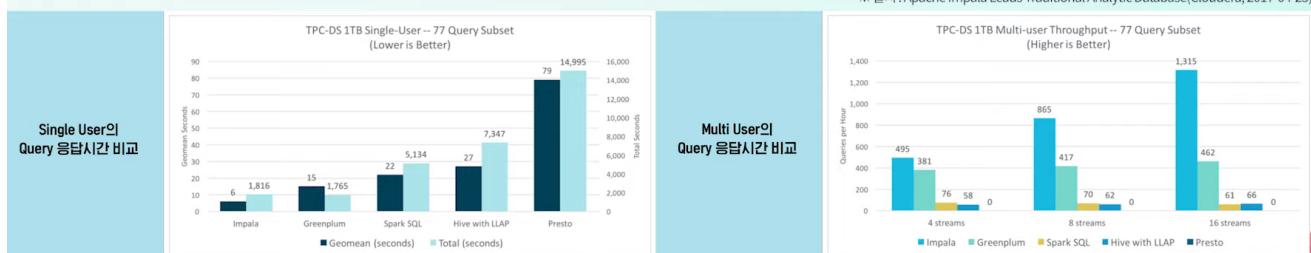
- Hive가 갖고 있는 실시간성 Query 성능 문제와 멀티 사용자 지원 해결을 위해, 2010년 Google의 드레멜(Dremel) 논문에서 영감을 받아 Cloudera에서 개발하여 오픈소스화 했음

[1] Hadoop용 인터랙티브 SQL로, 빠른 응답을 위한 분석용 혹은 Ad-Hoc Query에 최적화되어 있습니다.

[2] Hive가 Map-Reduce를 통해 처리하는 반면, Impala는 자체 분산 Query Engine을 사용하여 응답속도가 빠릅니다.

SQL on Hadoop 도구로 Hive외 Facebook에서 개발한 Presto, Spark SQL 등이 존재하며, TPC-DS 기반 성능 벤치마크에서 Single, Multi User의 Query 응답시간 모두 Impala가 가장 우수합니다. 성능 벤치마크 시 Impala 2.8, Greenplum Database 4.3.9.1, Spark SQL 2.1, Presto 0.160, Hive 2.1 with LLAP from HDP 2.5를 사용합니다.

※ 출처: Apache Impala Leads Traditional Analytic Database(Cloudera, 2017-04-25)



[3] 표준 ANSI-92 SQL을 준수하고, Hive QL과 호환성을 제공합니다.

[4] 단일 명령문에 대한 Transaction을 지원하나, 다중 명령문에 대한 Transaction은 지원하지 않습니다.

[5] Hadoop HDFS, HBase, Amazon S3, Azure ADLS, Azure ABFS 등 다양한 저장소를 사용할 수 있으며, Hive, Hue, Kudu 등과 연계하여 사용 가능합니다.

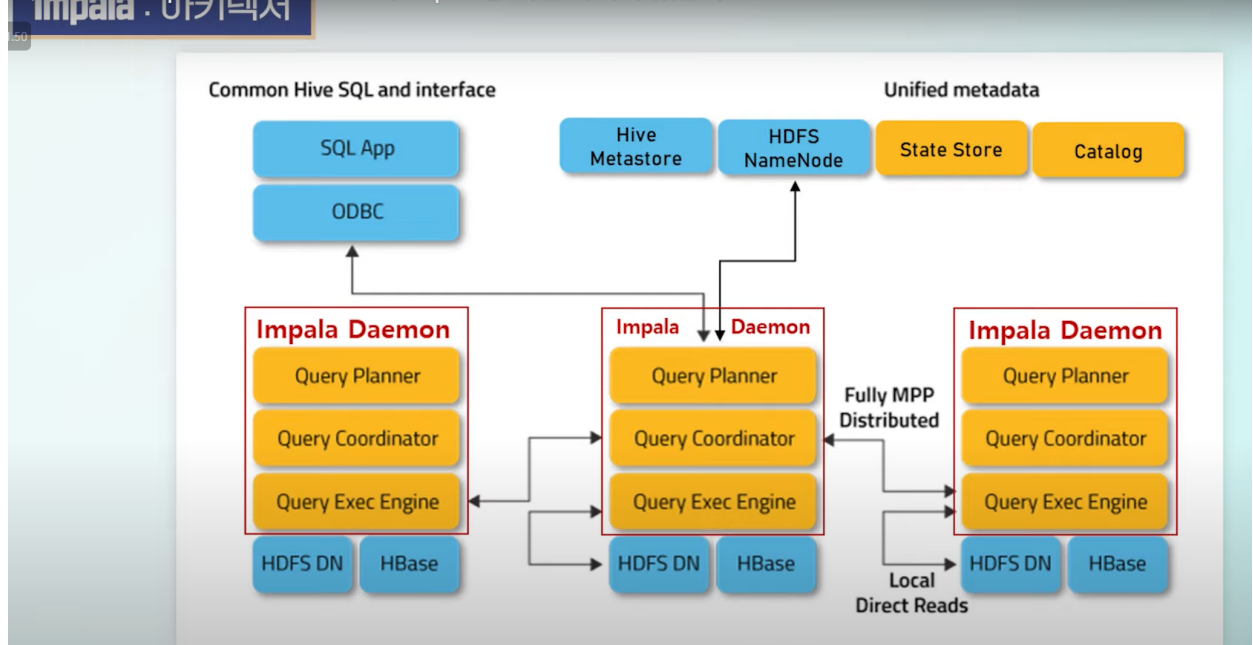
[6] 전용 메타스토어를 사용할 필요가 없으며, 기존 Hive 메타스토어를 사용합니다.

[7] Impala도 데이터를 HDFS에 저장하므로, 데이터의 부분적인 수정/삭제(Record별 Update, Delete)가 불가능합니다.

[8] Hive는 Hadoop 설치 및 관리 도구인 Open-Source Apache Ambari에 포함되어 있지만, Impala는 상용 Cloudera CDP 배포본에만 포함되어, 별도 설치가 필요합니다.

2. 아키텍처

SQL On Hadoop 분석 도구인 Hive와 Impala는 어떤 차이가 있을까?



- 임팔라 요소(주황), 하둡 요소(파랑)
- 임팔라 : Impala Daemon, StateStore, Catalog Service로 구성
 - Impala Daemon : Hadoop Data Node에서 실행되는 Impala 프로세스로, 사용자 쿼리를 요청받아 Query Planner, Coordinator, Executor 역할을 수행하며, 데이터 파일을 읽고 씀 (impalad라는 Daemon 프로세스로 표시)
 - Impala Statestore : Cluster에 있는 모든 Impala Daemon의 상태를 확인하고, 그 결과를 각 Daemon에 지속적으로 전달하며, Catalog Service에서 요청받은 메타데이터 동기화 작업에 대한 브로드캐스팅 역할 수행 (statestore라는 Daemon 프로세스로 표시)
 - Impala Catalog Service : Impala SQL문에서 Cluster의 모든 Impala Daemon으로 메타데이터 변경 사항을 전달 (catalogd라는 Daemon 프로세스로 표시)

- Impala Daemon에서 직접 메타 데이터를 변경한 경우, 자동 동기화 vs. Hive나 HDFS에서 직접 변경한 경우, Refresh문으로 동기화 작업을 수행해야 함

3. 데이터 객체

Database



Table



View



Function

- RDBMS와 동일

4. Hive와 Impala 비교

구분	RDBMS	Hive	Impala
개발된 언어	Java 혹은 C++	Java	C++
데이터 저장	O (Tablespace에 저장)	X (Hadoop HDFS 이용)	X (Hadoop HDFS 이용)
대량 데이터 관리	X	O	O
DDL 지원	O	O	O
DML 지원	O	△ (Record별 Update/Delete 불가)	△ (Record별 Update/Delete 불가)
Trigger/Index 지원	O	X	X
표준 SQL 지원	O	△ (SQL과 유사한 Hive QL)	O
응용프로그램 접속	JDBC, ODBC	JDBC, ODBC, Thrift	JDBC, ODBC
Transaction 지원	O	X	O
용도	OLTP, OLAP용	OLAP용(배치성)	OLTP용(실시간성)

- Hive - 배치성 대량 데이터 조회 및 분석에 적합
- Impala - OLTP성 데이터 조회 및 분석에 적절
=> 용도에 따라 선별적 사용이 필요