**[과제] 대용량 데이터 처리**

**1. Oracle**

* 프로세스, 디스크 영역, 메모리 영역으로 구성

**A. 프로세스**

* 서버 프로세스와 백그라운드 프로세스로 구분
* 서버 프로세스는 클라이언트와 통신하면서 사용자의 각종 명령어를 처리
* 서버 프로세스는 대표적으로 SQL 파싱, 최적화, 요청 결과 전송 등을 처리하며,  
  백그라운드 프로세스에게 작업을 위임하기 위한 시스템 콜 요청을 수행
* 백그라운드 프로세스는 데이터 파일을 읽어서 DB 버퍼 캐시에 적재하는 등  
  내부적으로 DB 시스템이 잘 돌아가게 해주는 역할을 수행
* 백그라운드 프로세스에는 장애 복구를 위한 SMON, 리소스 복구를 위한 PMON,  
  버퍼 캐시 처리를 위한 DBWR, 로그 파일 기록을 위한 LGWR 등이 존재

**B. 디스크 영역**

* 데이터 파일, 임시 데이터 파일, 로그 파일로 구분
* 데이터 파일은 가장 작은 단위인 블록부터 익스텐트, 세그먼트, 테이블 스페이스로 나눠지며,  
  하나의 블록에 여러 데이터를 포함
* 임시 데이터 파일은 대량의 작업 수행 중 메모리 공간이 부족해지면  
  중간에 결과 집합을 잠시 저장하는 용도로 사용되는 파일
* 로그 파일은 변경사항이 아직 기록되지 않은 상태에서 장애 발생 시 복구하기 위해 사용
* 데이터베이스를 시작할 때 항상 참조되는 컨트롤 파일과  
  환경설정 등의 정보를 포함하는 파라미터 파일도 별도로 존재

**C. 메모리 영역**

* 공유 메모리 영역, 프로세스 메모리 영역으로 구분
* 공유 메모리 영역은 여러 프로세스가 동시에 액세스할 수 있는 메모리 영역
* 공유 메모리 영역의 프로세스로는 DB 버퍼 캐시, 메타데이터와 사용자가 수행한 SQL문을 저장하는 공유 풀, 모든 변경사항을 기록하는 로그 버퍼가 존재
* 프로세스 메모리 영역은 세션과 커서에 대한 상태 정보를 저장하는 용도로 사용

2. Hive

* 하둡 에코시스템에서 데이터 프로세싱을 위해 사용하는 데이터 웨어하우징용 솔루션
* RDB와 같은 형태로 HDFS에 저장된 데이터의 구조를 정의하는 방법을 제공하며,  
  SQL과 유사한 HiveQL 쿼리를 이용해 데이터를 조회하는 방법 제공
* 기존의 RDB와 같이 미리 스키마를 정의하고 그 틀에 맞게 데이터를 입력하지 않고,  
  데이터를 먼저 저장하고 메타스토어를 통해 스키마를 입히는 과정을 거침

**A. Hive 구성 요소**

* UI: 사용자가 쿼리 및 기타 작업을 시스템에 제출하는 사용자 인터페이스 (CLI, Beeline 등)
* Driver: 쿼리를 입력 받고 작업을 처리, JDBC/ODBC 인터페이스 API 제공
* Compiler: 메타스토어를 참고하여 쿼리 구문을 분석하고 실행 계획을 생성
* Metastore: DB, 테이블, 파티션의 정보를 저장
* Execution Engine: 컴파일러에 의해 생성된 실행 계획을 수행

**B. Hive와 Oracle의 구조적 차이**

* Hive는 Java VM 환경에서 실행되는 데이터 웨어하우스 소프트웨어지만,  
  Oracle은 Mac OS, Windows, Linux와 같은 대중적인 OS 상에서 실행되며,  
  둘을 구성하는 언어에 있어서도 Hive는 Java, Oracle은 C와 C++이라는 차이가 있음
* 서로 다른 노드에 각각의 데이터를 저장하기 위해  
  Hive는 샤딩 파티셔닝 방식만을 사용하지만, Oracle은 수평적 파티셔닝 방식도 사용
* Hive는 데이터를 가공하기 위한 목적의 사용자 정의 MapReduce API를 지원하지만,  
  Oracle은 그러한 API를 지원하지 않음
* Hive는 내부 메소드에 접근하기 위한 목적의 JDBC, ODBC 엔진으로 구성되며,  
  Oracle은 JDBC, ODBC, ODP.NET, OCI 엔진으로 구성
* 기본적인 ANSI-SQL 구문을 사용하는 Oracle과 달리,  
  Hive는 SQL과 유사한 HiveQL (Hive Query Language)를 사용
* HDFS가 한 번 저장한 파일을 수정할 수 없기 때문에  
  HiveQL에서 UPDATE와 DELETE를 사용할 수 없고, 읽기 전용 뷰만 지원
* Oracle에서는 데이터를 입력하기 위한 스키마가 미리 정의되지만,  
  Hive는 메타스토어를 통해 저장된 데이터에 스키마를 부여함
* Oracle은 정형 데이터만을 저장하지만, Hive는 비정형 데이터도 동시에 저장 가능하며,  
  저장하는 데이터의 크기 또한 Hive는 빅데이터를 처리할 수 있는 구조를 가지지만,  
  Oracle은 상대적으로 작은 규모의 데이터셋을 처리하는 RDBMS를 사용
* 위와 같은 목적의 차이 때문에 Hive는 INT, FLOAT, STRING과 같은 기본 타입 외에  
  STRUCT, MAP, ARRAY와 같은 복합 타입을 지원

**3. 참고자료**

* <https://jeong-pro.tistory.com/147>
* <https://spidyweb.tistory.com/238>
* <https://www.geeksforgeeks.org/difference-between-hive-and-oracle/>
* <https://db-engines.com/en/system/Hive%3BOracle>