빅데이터 실시간 적재

- 실시간 적재 활용 기술

빅데이터 적재 소개

1. 빅데이터 실시간 적재 개요

빅데이터 실시간 적재에 대한 기본 정의와 일반 적재와의 차이를 설명한다.

2. 빅데이터 실시간 적재에 활용하는 기술

빅데이터 실시간 적재에서 사용할 4가지 기술(HBase, 레디스, 스톰, 에스퍼)을 소개하고 각 기술별 주요 기능과 아키텍처, 활용 방안을 알아본다.

3. 실시간 적재 파일럿 실행 1단계 – 실시간 적재 아키텍처

스마트카 데이터의 실시간 적재와 관련된 요구사항을 구체화하고, 실시간 적재 요건을 해결하기 위한 파일럿 아키텍처를 제시한다.

4. 실시간 적재 파일럿 실행 2단계 – 실시간 적재 환경 구성

스마트카의 실시간 적재 아키텍처에 대한 설치 및 환경을 구성한다. HBase, 레디스, 스톰 순으로 설치하게 된다.

5. 실시간 적재 파일럿 실행 3단계 – 실시간 적재 기능 구현

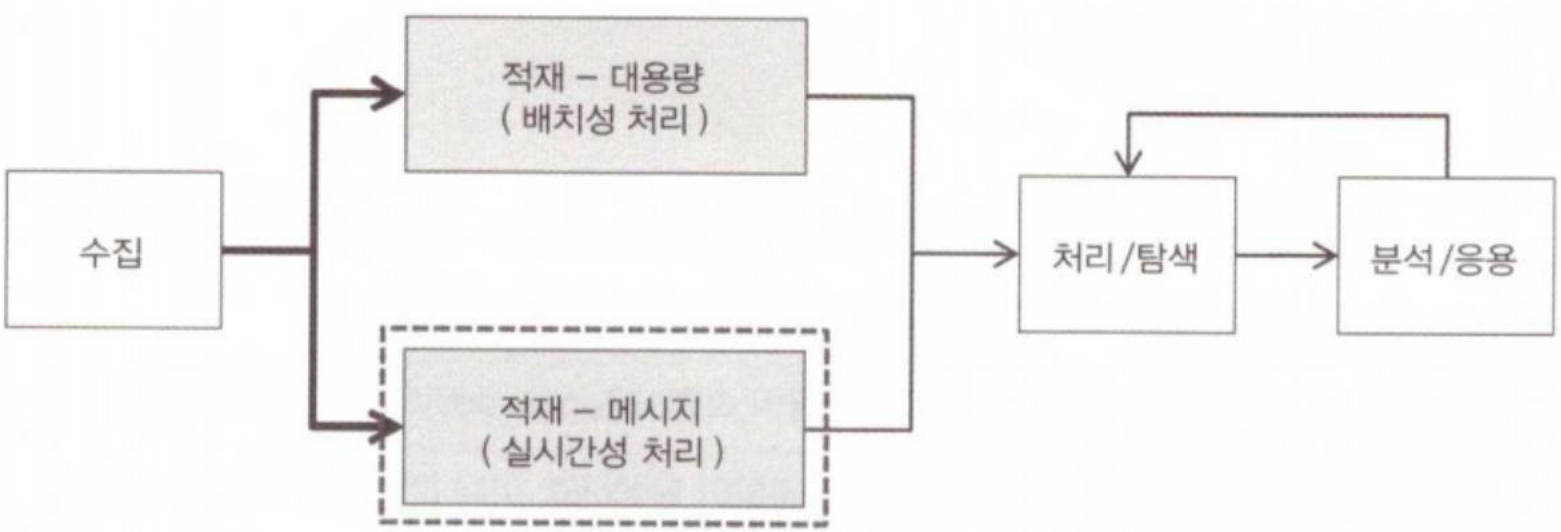
카프카와 스톰을 이용해 실시간 데이터 처리 기능을 구현하고, HBase와 레디스에 적재하는 기능과 방식을 이해한다.

6. 실시간 적재 파일럿 실행 4단계 – 실시간 적재 기능 테스트

로그 시뮬레이터를 이용해 실시간 데이터를 발생시키고 카프카, 스톰의 기능을 점검한 후HBase, 레디스에 적재된 실시간 데이터를 확인한다. 추가로 실시간 적재 개발 환경을 구성한다.

빅데이터 실시간 적재 개요

➢ 빅데이터 구축 단계 – 실시간 적재



- 스마트카 운전자의 실시간 운행 정보를 분석한 후 적재하는 영역을 다룸.
- 적재하기 직전에 실시간 분석 작업(집계, 분류, 관계 등)을 수행해 그 결과를 인메모리 시스템에 전달해 주변 응용 시스템과 빠르게 공유.
- 실시간으로 발생하는 대규모 메시지성 데이터를 영구적으로 저장하기에 하둡은 효율성이 떨어지므로 HBase 같은 NoSQL 데이터베이스를 사용.

빅데이터 실시간 적재에 활용하는 기술 – HBase

➢ HBase 소개

- 하둡의 HDFS를 저장소로 사용하는 컬럼 기반 지향(Column-Oriented)의 NoSQL 데이터베이스.
- 논리적 관점에서 로우키(row key)와 컬럼패밀리(column family), 컬럼 퀄러파이어(column qualifier)의 중첩 맵 구조로 저장.
- 물리적 관점에서는 컬럼 패밀리 단위로 생성되는 HFile이라는 파일에 저장.
- NoSQL DB는 데이터를 키/값(key/value) 형식으로 단순하게 구조화하는 대신 고성능의 쓰기/읽기가 가능하도록 만든 데이터베이스.
- HBase의 경우 특히 쓰기 성능에 좀 더 최적화되어 있으며, 대용량 처리가 필요한 대규모 NoSQL 아키텍처 구성이 필요할 때 자주 사용.
- 구글의 BigTable 논문을 모델로 2006년 말에 개발 시작, 2008년 하둡의 서브 프로젝트가 됐음.

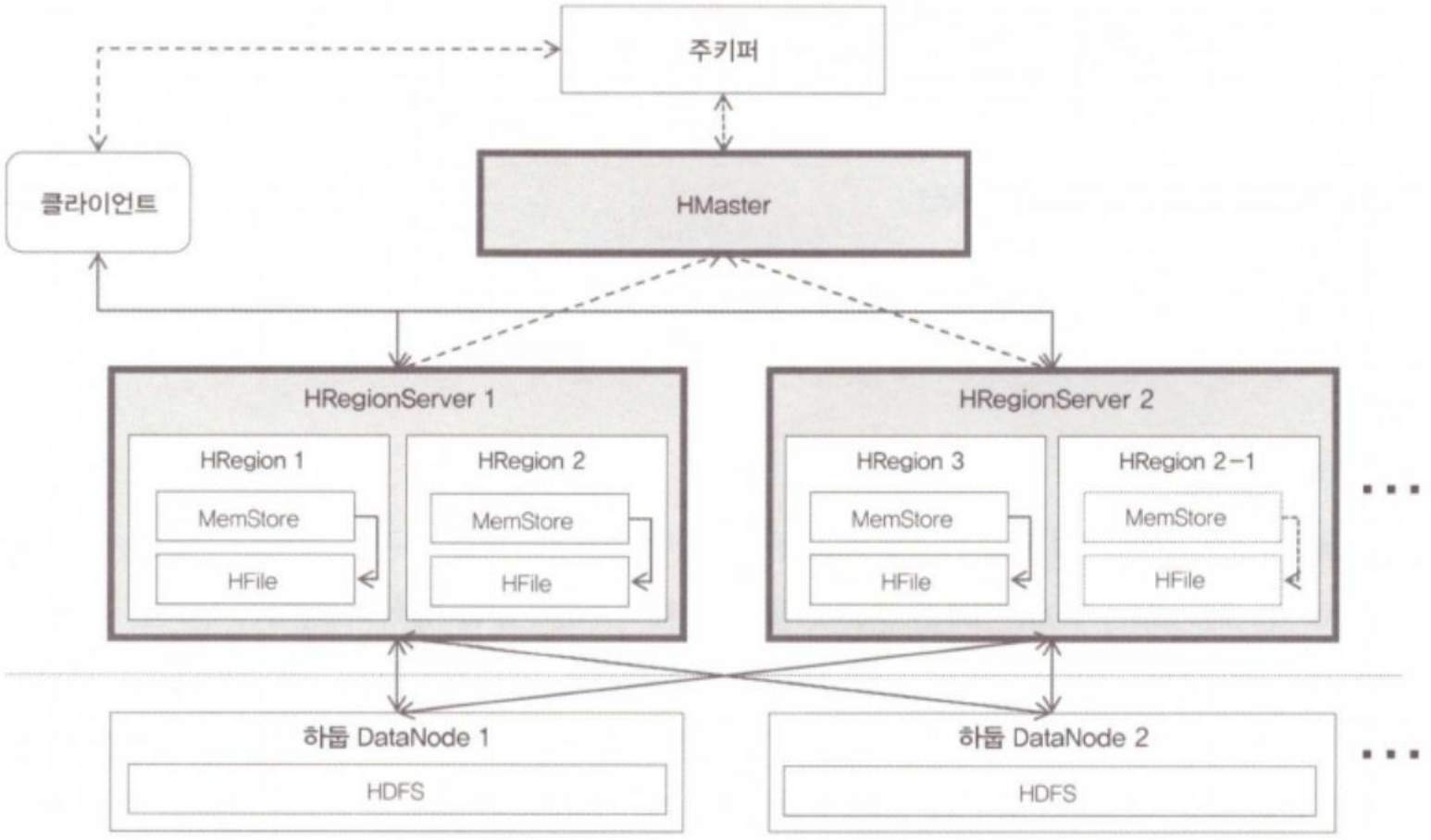
빅데이터 실시간 적재에 활용하는 기술 – HBase

- HBase 기본 요소

공식 홈페이지		http://hbase.apache.org
주요 구성 요소	HTable	칼럼 기반 데이터 구조를 정의한 테이블로서, 공통점이 있는 칼럼들의 그룹을 묶은 칼럼 패밀리와 테이블의 로우를 식별해서 접근하기 위한 로우키로 구성
	HMaster	HRegion 서버를 관리하며, HRegion들이 속한 HRegion 서버의 메타 정보를 관리
	HRegion	HTable의 크기에 따라 자동으로 수평 분할이 발생하고, 이때 분할된 블록을 HRegion 단위로 지정
	HRegionServer	분산 노드별 HRegionServer가 구성되며, 하나의 HRegionServer에는 다수의 HRegion이 생성되어 HRegion을 관리
	Store	하나의 Store에는 칼럼 패밀리가 저장 및 관리되며, MemStore와 HFile로 구성됨
	MemStore	Store 내의 데이터를 인메모리에 저장 및 관리하는 데이터 캐시 영역
	HFile	Store 내의 데이터를 스토리지에 저장 및 관리하는 영구 저장 영역
라이선스	Apache	
유사 프로젝트	BigTable, Cassandra, MongoDB	

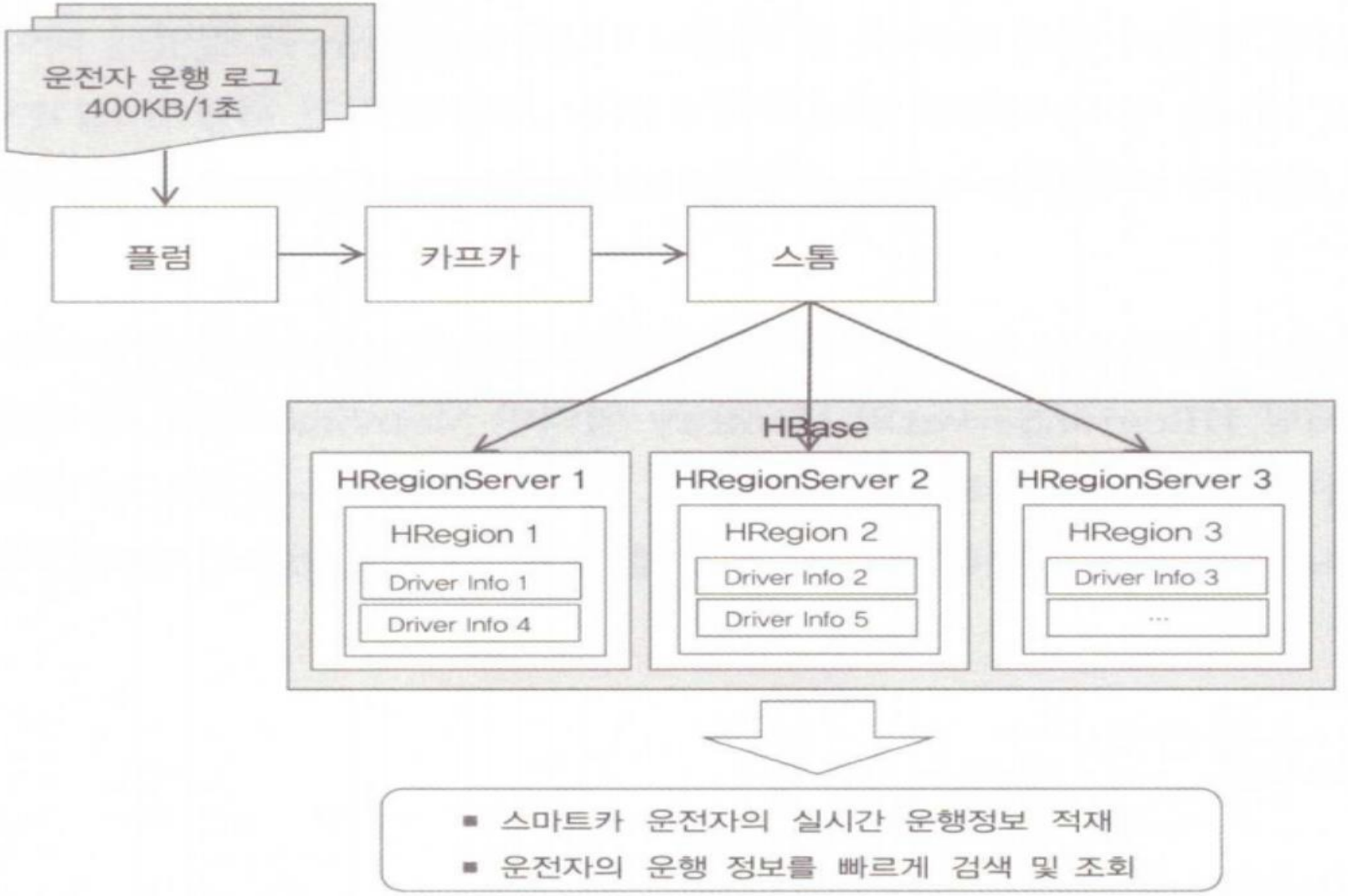
빅데이터 실시간 적재에 활용하는 기술 – HBase

- HBase 아키텍처



빅데이터 실시간 적재에 활용하는 기술 – HBase

- HBase 활용 방안



빅데이터 실시간 적재에 활용하는 기술 – Redis

➢ 레디스 소개

- IMDG(In-Memory Data Grid) 소프트웨어 – 분산 캐시 시스템이면서 NoSQL 데이터베이스 처럼 대규모 데이터 관리 능력도 갖추고 있음.
- key/value 형식의 데이터 구조를 분산 서버상의 메모리에 저장하면서 고성능의 응답 속도를 보장.
- 문자열 외의 다양한 바이너리 파일도 저장할 수 있고 집합 연산 기능도 제공.
- hash, set, list 등 다양한 자료구조를 지원하기 때문에 다양한 유형의 자료 구조를 저장할 수 있고, 또한 데이터를 구조화해서 저장할 수 있어 단순 key/value 이상의 데이터 복잡성도 처리 가능.
- 인메모리 데이터를 영구적으로 저장할 수 있는 스냇샵 기능 제공.
- 데이터 유실에 대비한 AOF(Append Only File) 기능으로 정합성 보장.
- 대규모 빅데이터 아키텍처에서는 실시간으로 분석한 결과를 레디스에 저장해서 주변 업무 시스템과 결과를 빠르게 공유하는 데 주로 활용.

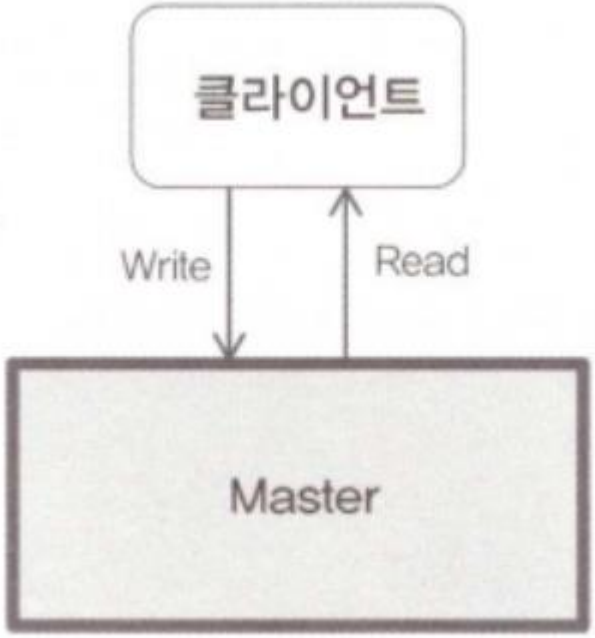
빅데이터 실시간 적재에 활용하는 기술 – Redis

- 레디스 기본 요소

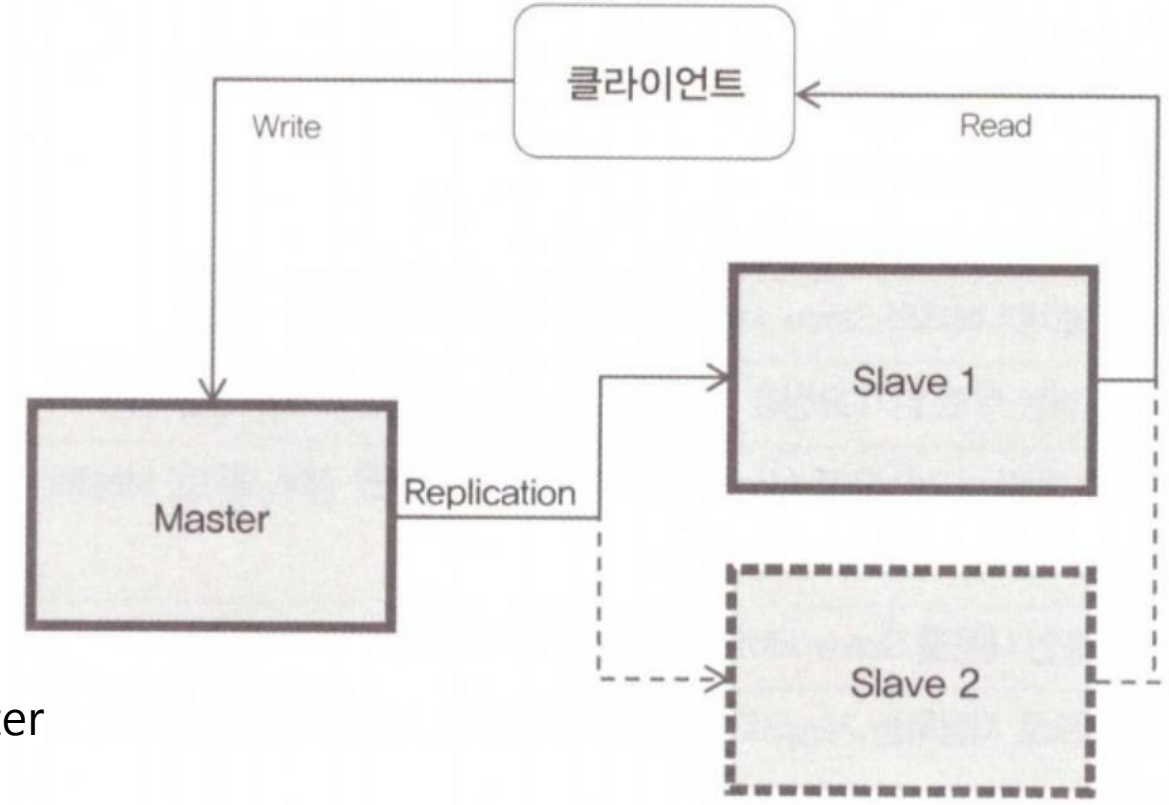
공식 홈페이지	 redis http://www.redis.io	
주요 구성 요소	Master	분산 노드 간의 데이터 복제와 Slave 서버의 관리를 위한 마스터 서버
	Slave	다수의 Slave 서버는 주로 읽기 요청을 처리하고, Master 서버는 쓰기 요청을 처리
	Sentinel	레디스 3.x에서 지원하는 기능으로, Master 서버에 문제가 발생할 경우 새로운 Master를 선출하는 기능
	Replication	Master 서버에 쓰인 내용을 Slave 서버로 복제해서 동기화 처리
	AOF/Snapshot	데이터를 영구적으로 저장하는 기능으로, 명령어를 기록하는 AOF와 스냅샷 이미지 파일 방식을 지원
라이선스	BSD	
유사 프로젝트	jBoss Infinispan, MemCached, Mambase	

빅데이터 실시간 적재에 활용하는 기술 – Redis

➢ 레디스 아키텍처



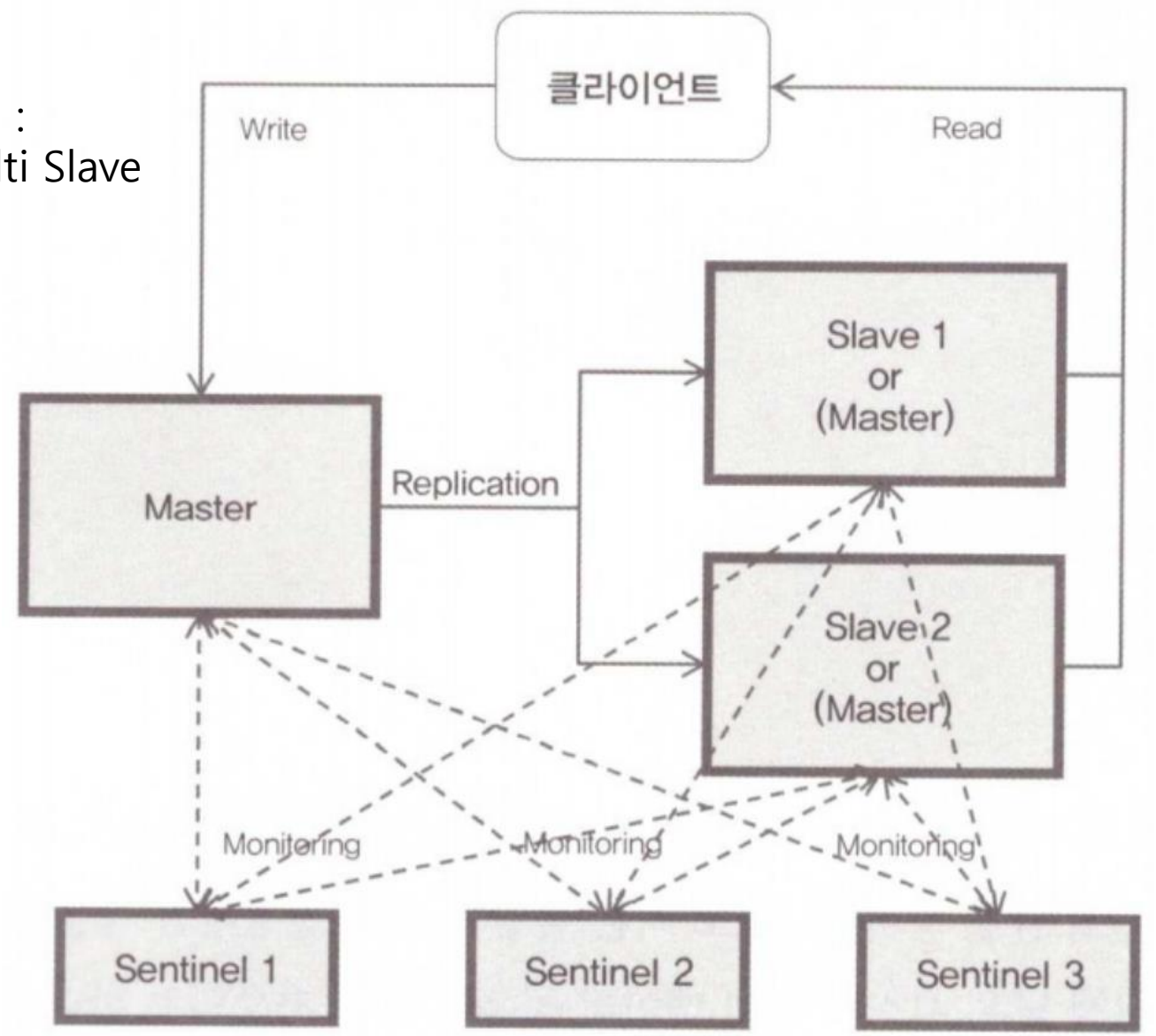
- 레디스 아키텍처1: Single Master



- 레디스 아키텍처2: Single Master / Multi Slave

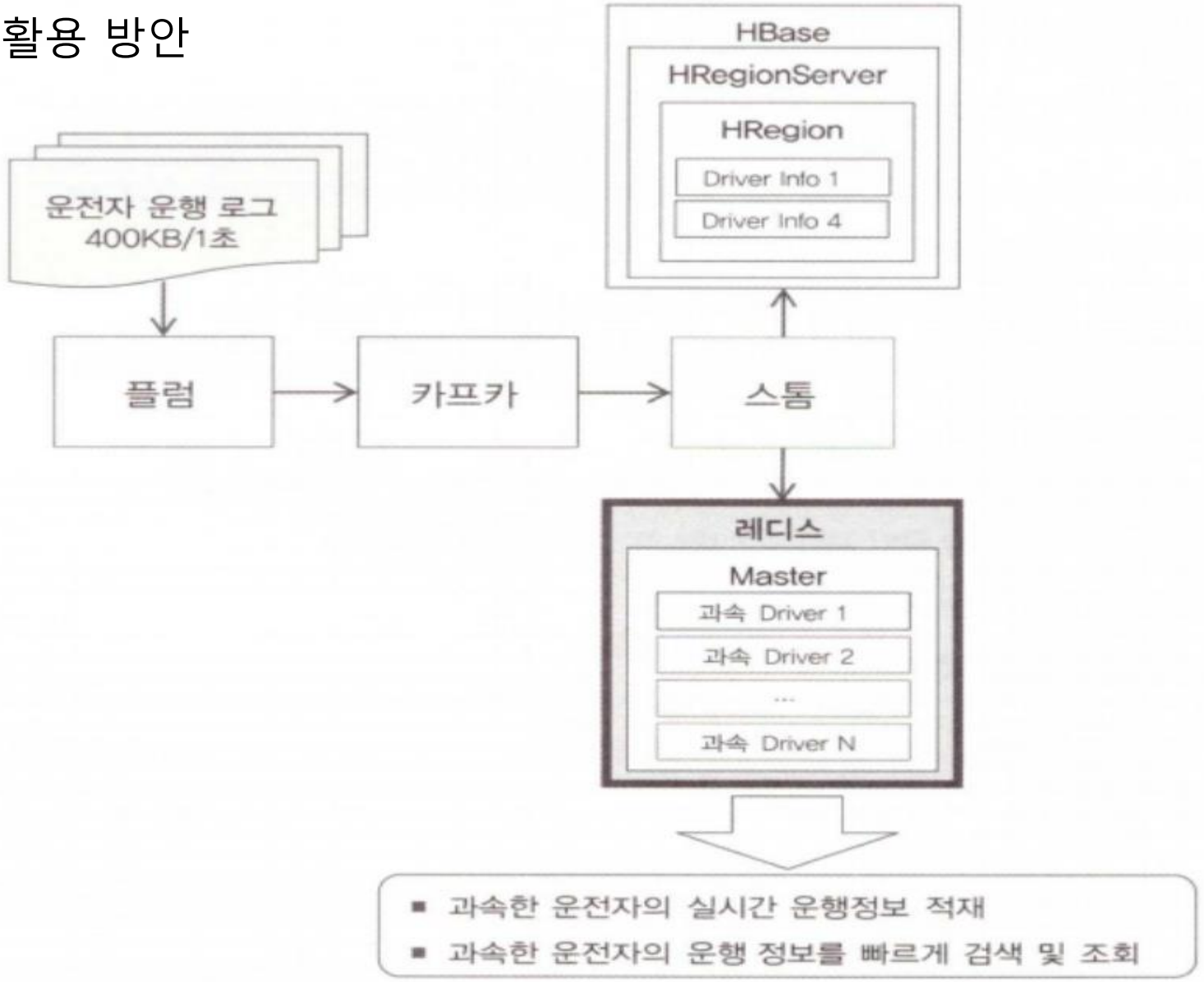
빅데이터 실시간 적재에 활용하는 기술 – Redis

- 레디스 아키텍처
  - 레디스 아키텍처3 :
    HA Master / Multi Slave



빅데이터 실시간 적재에 활용하는 기술 – Redis

- Redis 활용 방안



빅데이터 실시간 적재에 활용하는 기술 – STORM

➢ 스톰 소개

- 실시간 분산 처리기 - 빅데이터 프로젝트에서 실시간 데이터를 병렬 프로세스로 처리하기 위한 소프트웨어.
- 데이터를 적재하기 전에 발생과 동시에 이벤트를 감지해서 처리하는 방식.
- 2011년 트위터가 백타이프라는 회사로부터 인수했고, 바로 스톰을 오픈소스 프로젝트로 공개.
- 2014년 9월 아파치 최상위 프로젝트로 승격.

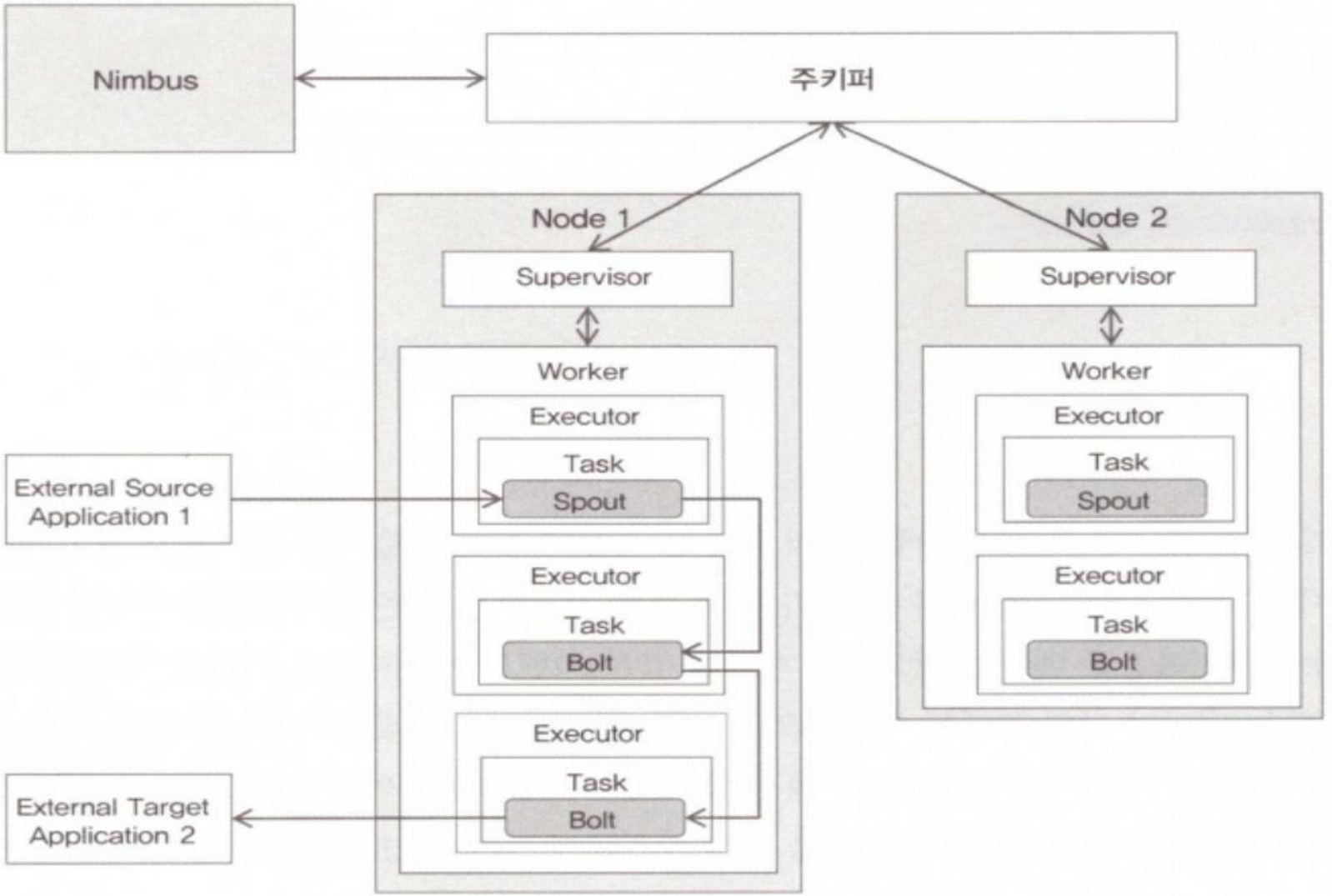
빅데이터 실시간 적재에 활용하는 기술 – STORM

- 스톰 기본 요소

공식 홈페이지	 APACHE STORM™ http://storm.apache.org/	
주요 구성 요소	Spout	외부로부터 데이터를 유입받아 가공 처리해서 튜플을 생성, 이후 해당 튜플을 Bolt에 전송
	Bolt	튜플을 받아 실제 분산 작업을 수행하며, 필터링(Filtering), 집계(Aggregation), 조인(Join) 등의 연산을 병렬로 실행
	Topology	Spout-Bolt의 데이터 처리 흐름을 정의, 하나의 Spout와 다수의 Bolt로 구성
	Nimbus	Topology를 Supervisor에 배포하고 작업을 할당, Supervisor를 모니터링하다 필요 시 페일오버(Fail-Over) 처리
	Supervisor	Topology를 실행할 Worker를 구동시키며 Topology를 Worker에 할당 및 관리
	Worker	Supervisor 상에서 실행 중인 자바 프로세스로 Spout와 Bolt를 실행
	Executor	Worker 내에서 실행되는 자바 스레드
	Tasker	Spout 및 Bolt 객체가 할당
라이선스	Apache	
유사 프로젝트	Samza, S4, Akka, Spark Stream	

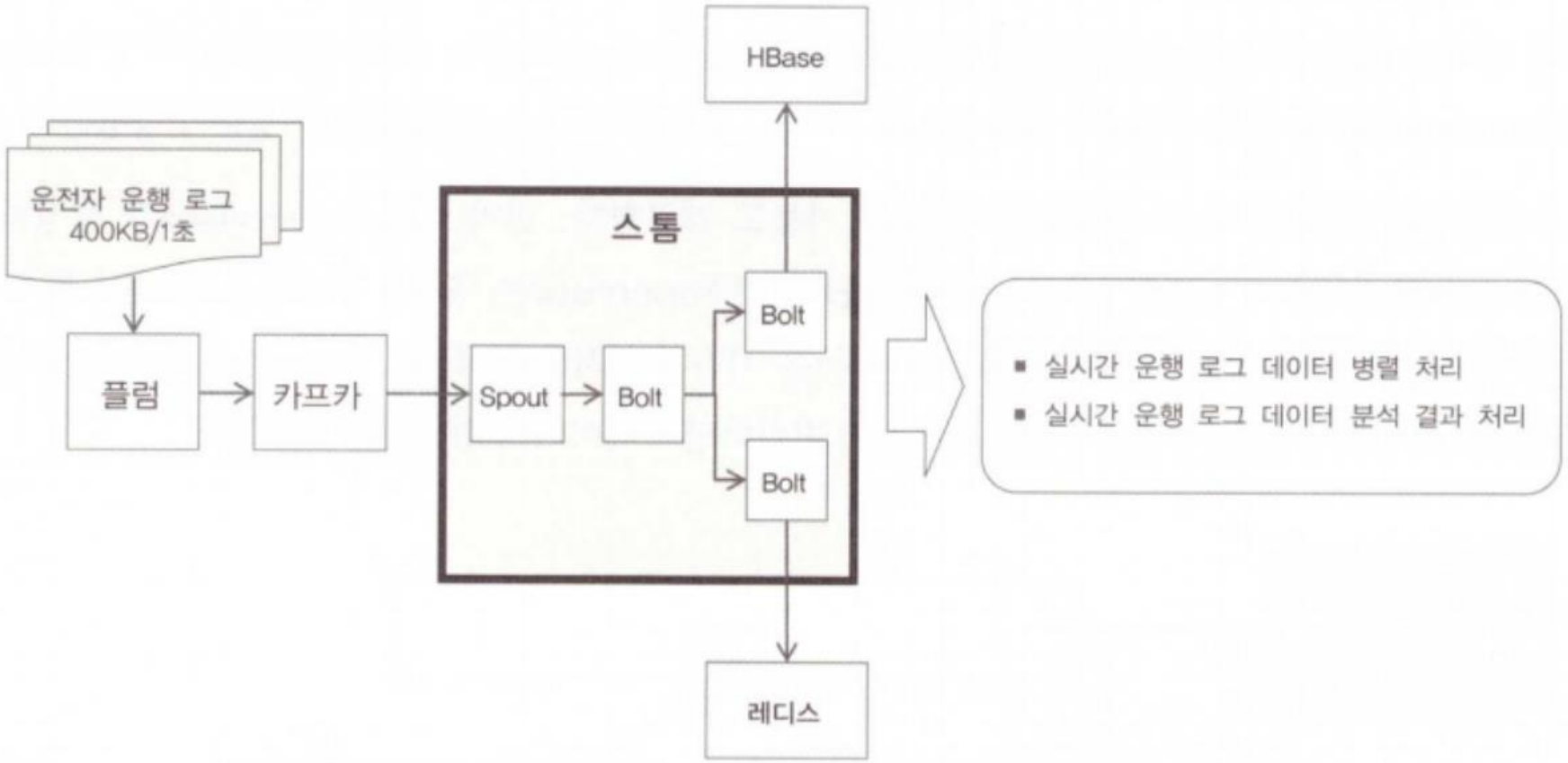
빅데이터 실시간 적재에 활용하는 기술 – STORM

- STORM 아키텍처



빅데이터 실시간 적재에 활용하는 기술 – STORM

- STORM 활용 방안



빅데이터 실시간 적재에 활용하는 기술 – Esper

➢ 에스퍼 소개

- 실시간 스트리밍 데이터의 복잡한 이벤트 처리가 필요할 때 사용하는 룰 엔진.
- 실시간으로 발생하는 데이터로부터 복잡한 패턴을 찾고, 그 패턴에 따른 이벤트를 처리하는 기능에 적합.
- CEP 처리를 위한 다양한 조건과 복합적인 이벤트를 하나의 룰로 쉽게 정의할 수 있어 CEP 처리 및 관리가 수월.
  - ✓ CEP(Complex Event Processing) 엔진 – 실시간으로 발생하는 데이터 간의 관계를 복합적으로 판단 및 처리하는 것.
- EsperTech사로 부터 지난 2006년 에스퍼 0.7.0 알파 버전이 최초 공개 후 새로운 버전들이 계속 릴리스 되는 중.

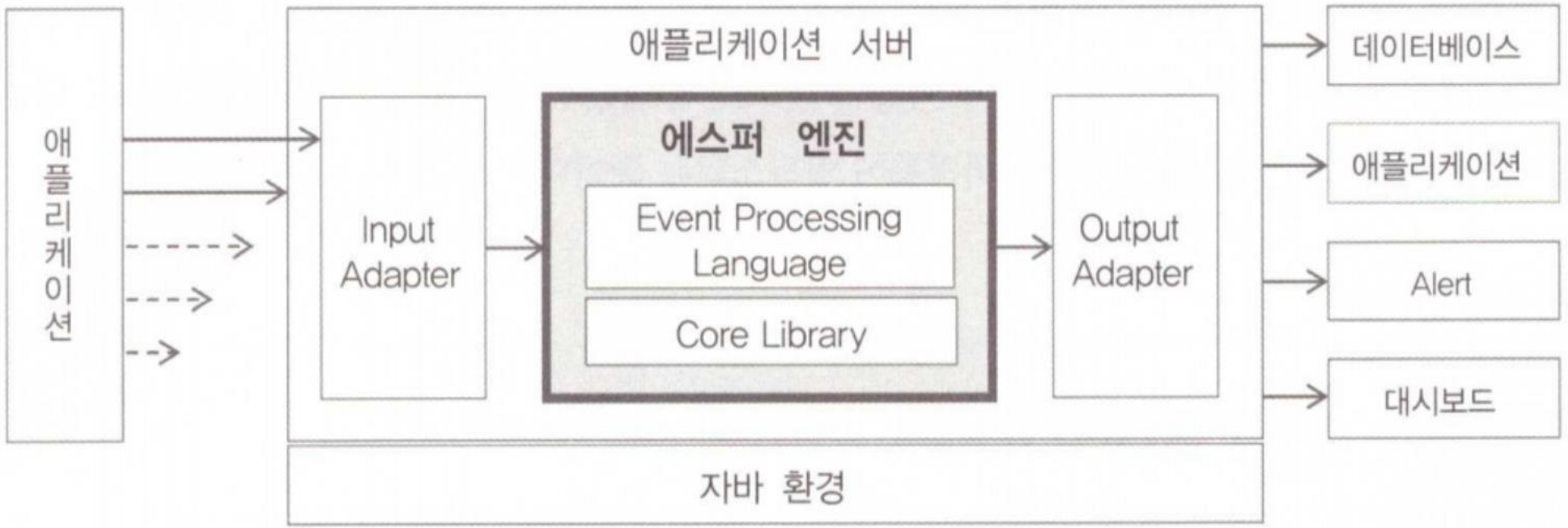
빅데이터 실시간 적재에 활용하는 기술 – 에스퍼

- 에스퍼 기본 요소

공식 홈페이지	EsperTech http://www.espertech.com	
주요 구성 요소	Event	실시간 스트림으로 발생하는 데이터들의 특정 흐름 또는 패턴을 정의
	EPL	유사 SQL을 기반으로 하는 이벤트 데이터 처리 스크립트 언어
	Input Adapter	소스로부터 전송되는 데이터를 처리하기 위한 어댑터 제공 (CSV, Socket, JDBC, Http 등)
	Output Adapter	타깃으로 전송하는 데이터를 처리하기 위한 어댑터 제공 (HDFS, CSV, Socket, Email, Http 등)
	Window	실시간 스트림 데이터로부터 특정 시간 또는 개수를 설정한 이벤트들을 메모리 상에 등록한 후 EPL을 통해 결과를 추출
라이선스	GPL	
유사 프로젝트	Drools	

빅데이터 실시간 적재에 활용하는 기술 – 에스퍼

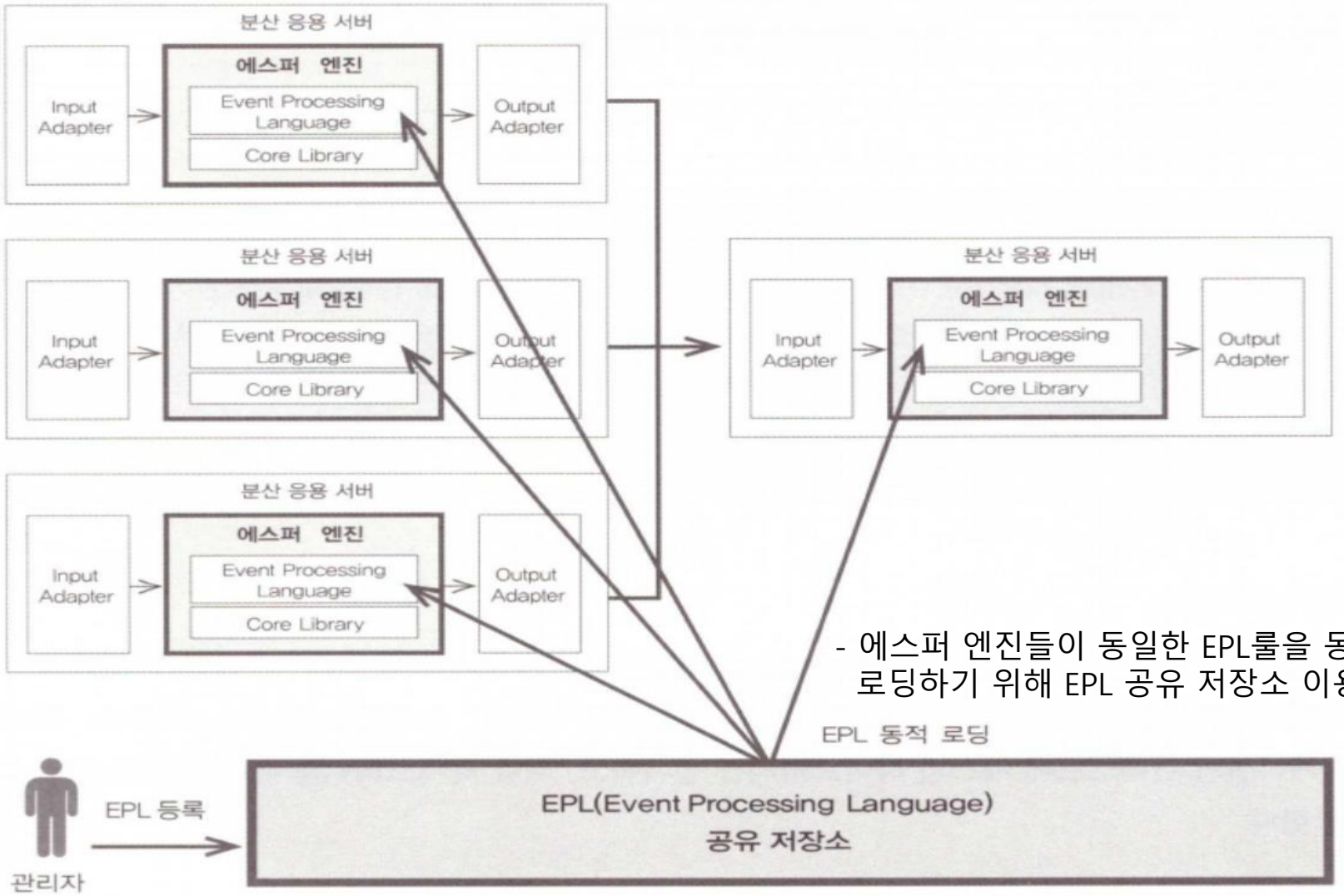
➢ 에스퍼 아키텍처 1



- 에스퍼 CEP 엔진은 단순 자바 라이브러리 프로그램에 불과.
- 애플리케이션 서버(톰캣, 제이보스, OSGI, 스톰등) 또는 애플리케이션의 컨텍스트에 에스퍼 라이브러리를 설치하고, 해당 라이브러리를 이용해 CEP 프로그래밍을 하면 됨.

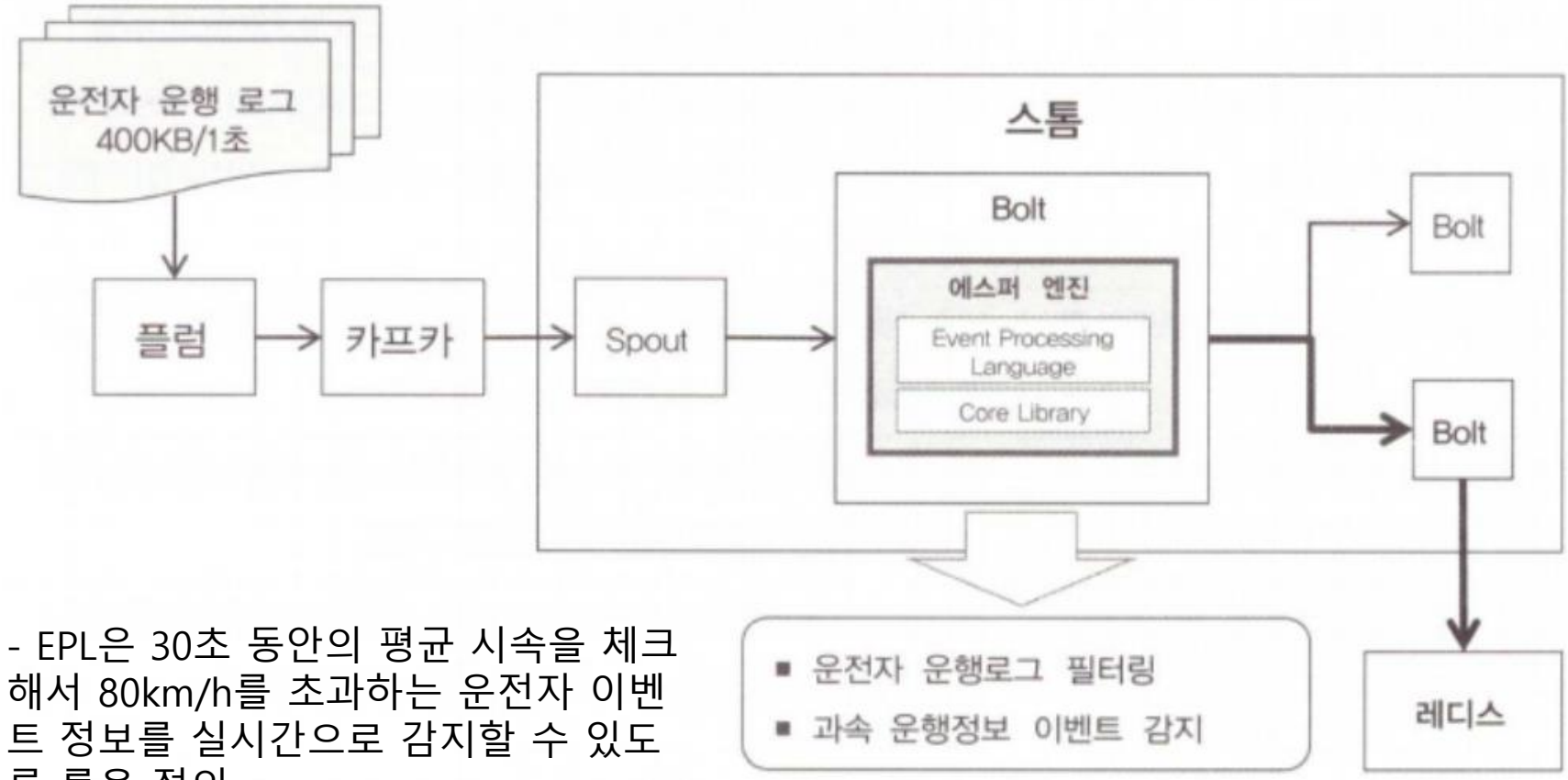
빅데이터 실시간 적재에 활용하는 기술 – 에스퍼

➢ 에스퍼 아키텍처 2 – 대규모 분산 아키텍처



빅데이터 실시간 적재에 활용하는 기술 – 에스퍼

➢ 에스퍼 활용 방안



- EPL은 30초 동안의 평균 시속을 체크해서 80km/h를 초과하는 운전자 이벤트 정보를 실시간으로 감지할 수 있도록 룰을 정의.