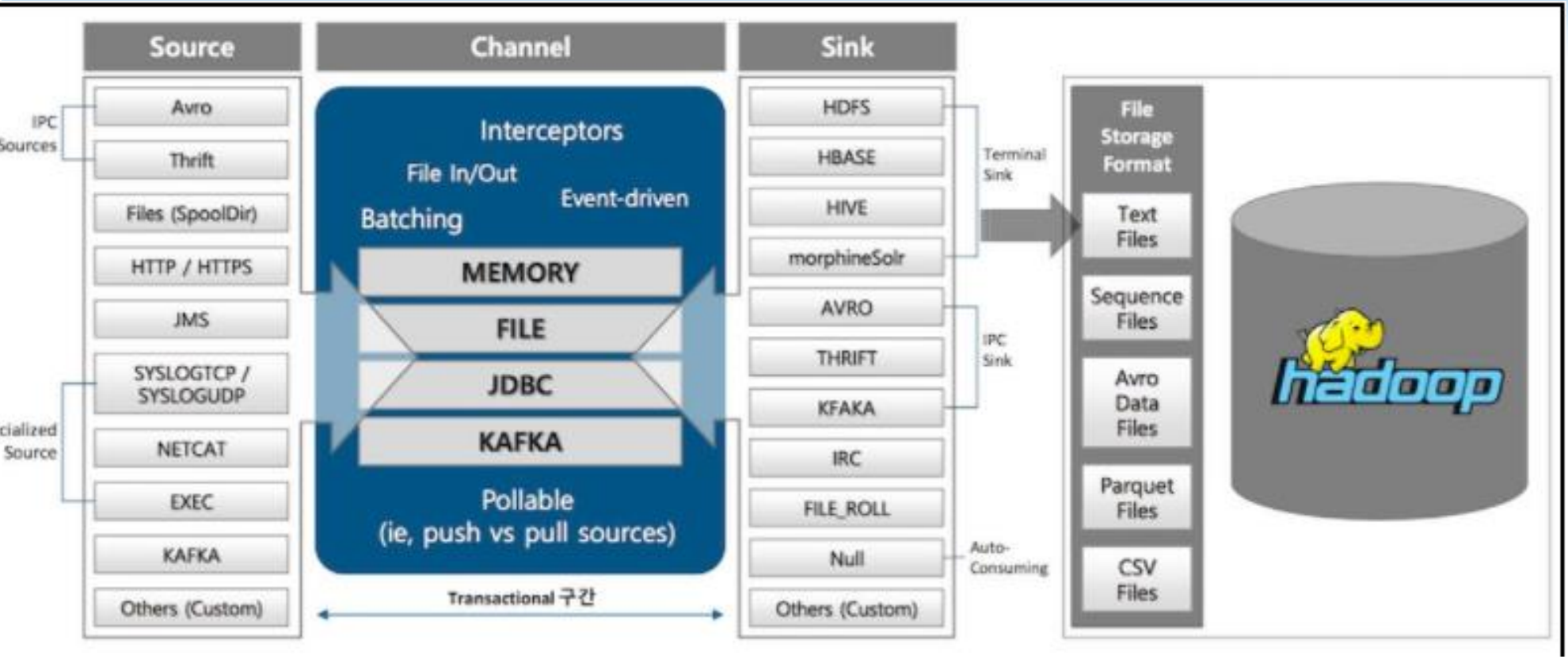
3강

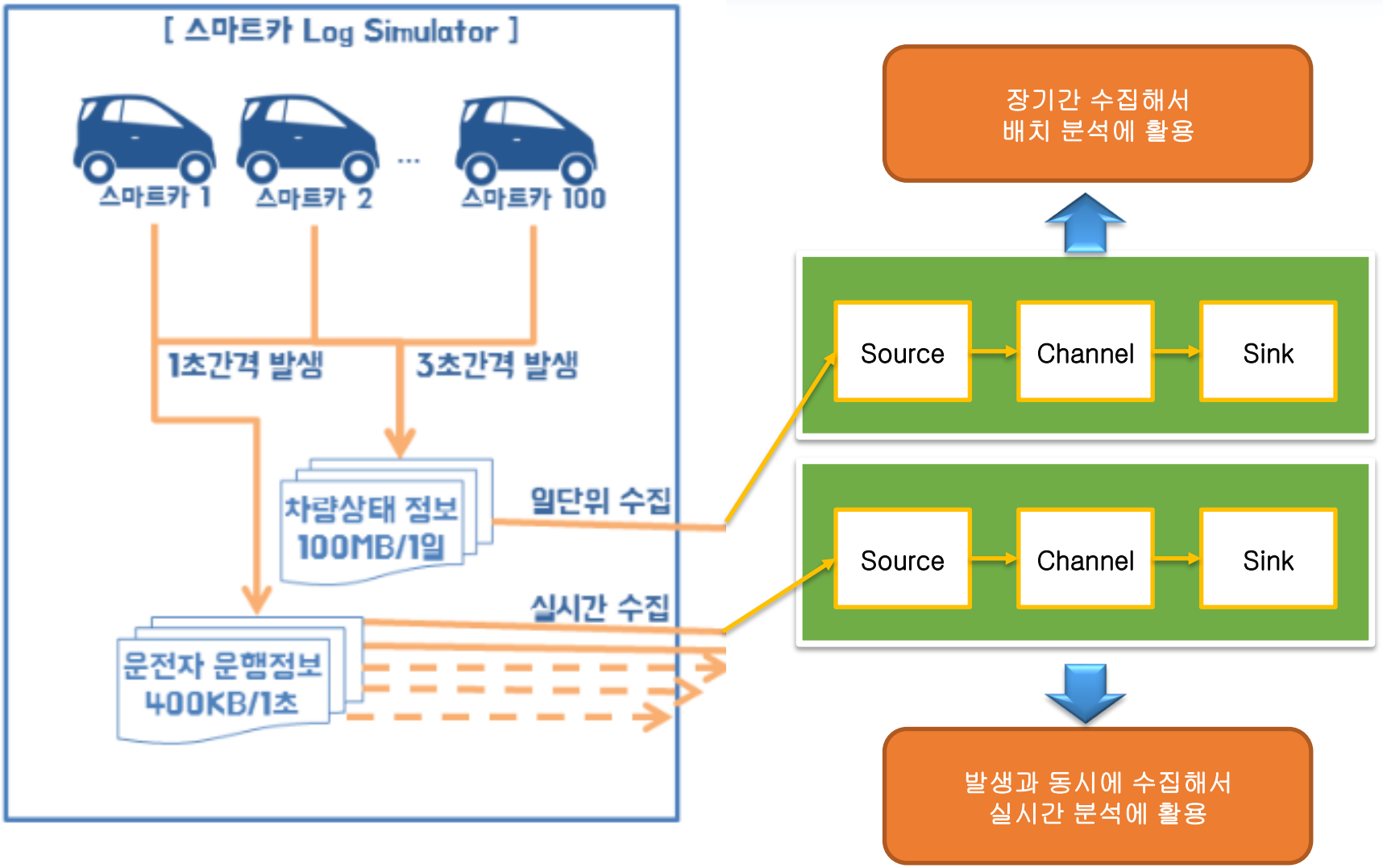
빅데이터 수집

Flume

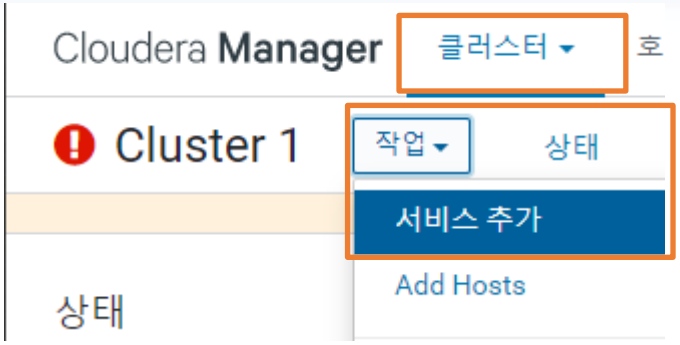
Flume이란

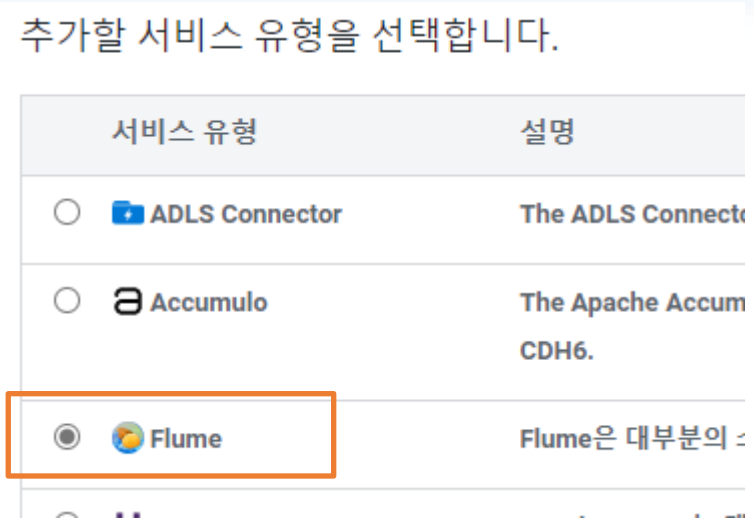

Source
Channel
Sink
IPC Sources
Avro
Thrift
Files (SpoolDir)
HTTP / HTTPS
JMS
SYSLOGTCP / SYSLOGUDP
cialized Source
NETCAT
EXEC
KAFKA
Others (Custom)
Interceptors
File In/Out
Event-driven
Batching
MEMORY
FILE
JDBC
KAFKA
Pollable
(ie, push vs pull sources)
Transactional 구간
HDFS
HBASE
HIVE
morphineSolr
AVRO
THRIFT
KFAKA
IRC
FILE_ROLL
Null
Others (Custom)
Terminal Sink
IPC Sink
Auto-Consuming
File Storage Format
Text Files
Sequence Files
Avro Data Files
Parquet Files
CSV Files
hadoop

Agent 생성

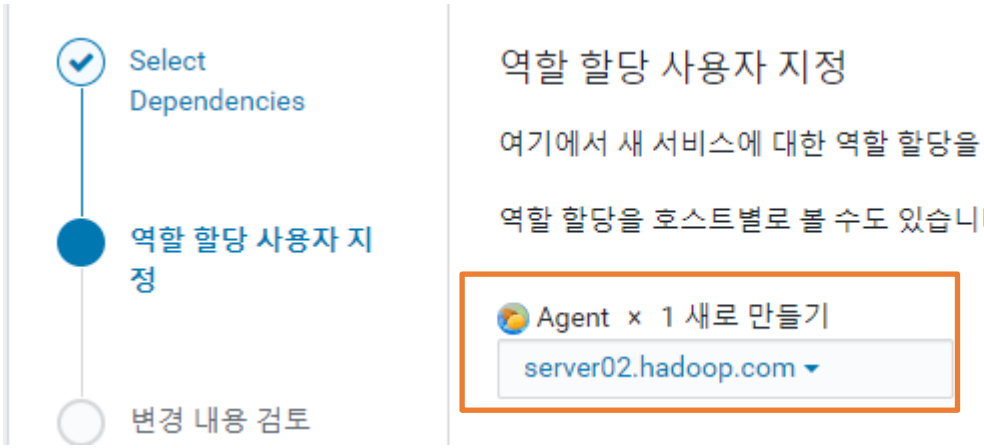

[스마트카 Log Simulator]
스마트카 1
스마트카 2
...
스마트카 100
1초간격 발생
3초간격 발생
차량상태 정보
100MB/1일
일단위 수집
실시간 수집
운전자 운행정보
400KB/1초
장기간 수집해서
배치 분석에 활용
Source
Channel
Sink
Source
Channel
Sink
발생과 동시에 수집해서
실시간 분석에 활용

Flume 설치

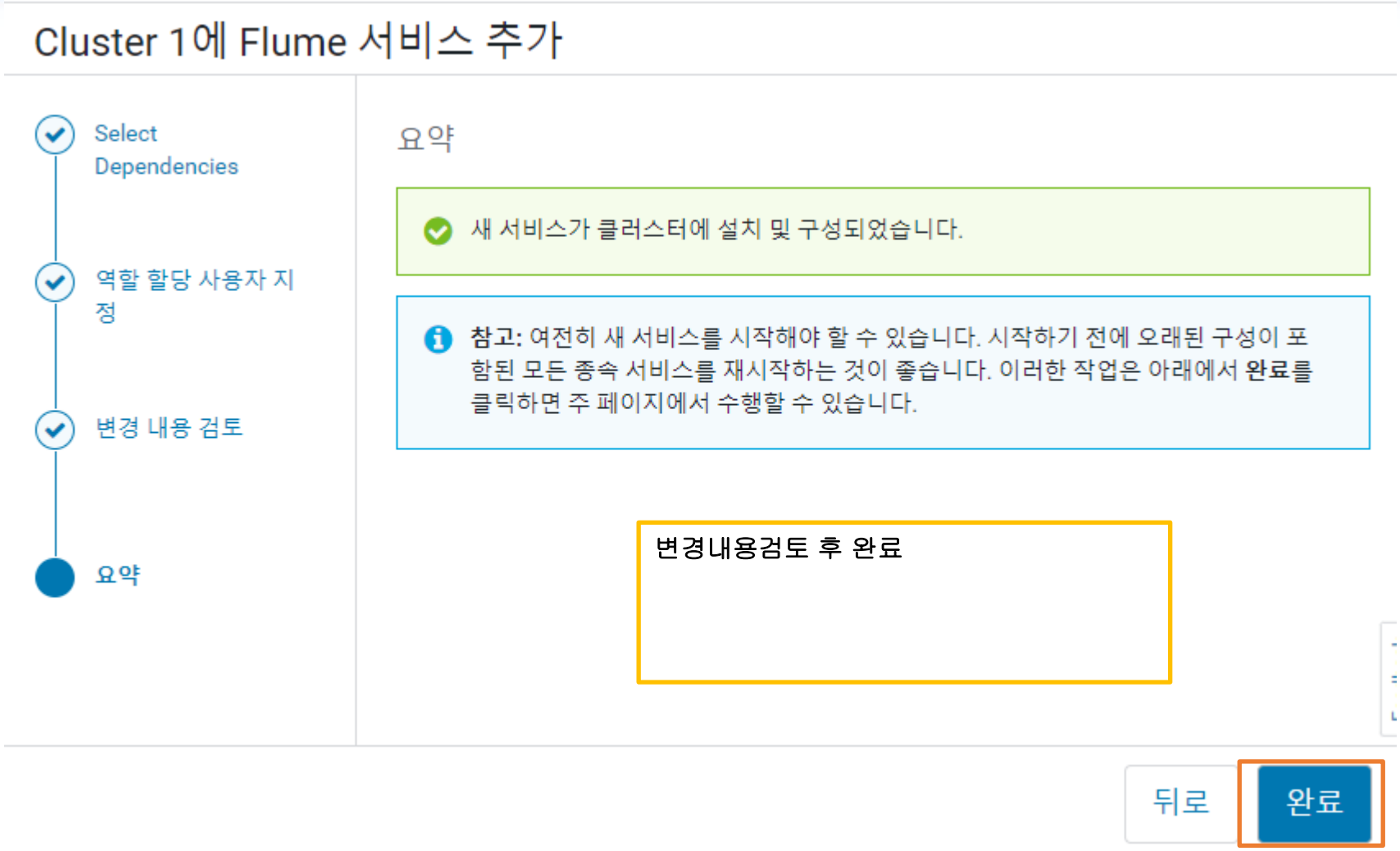




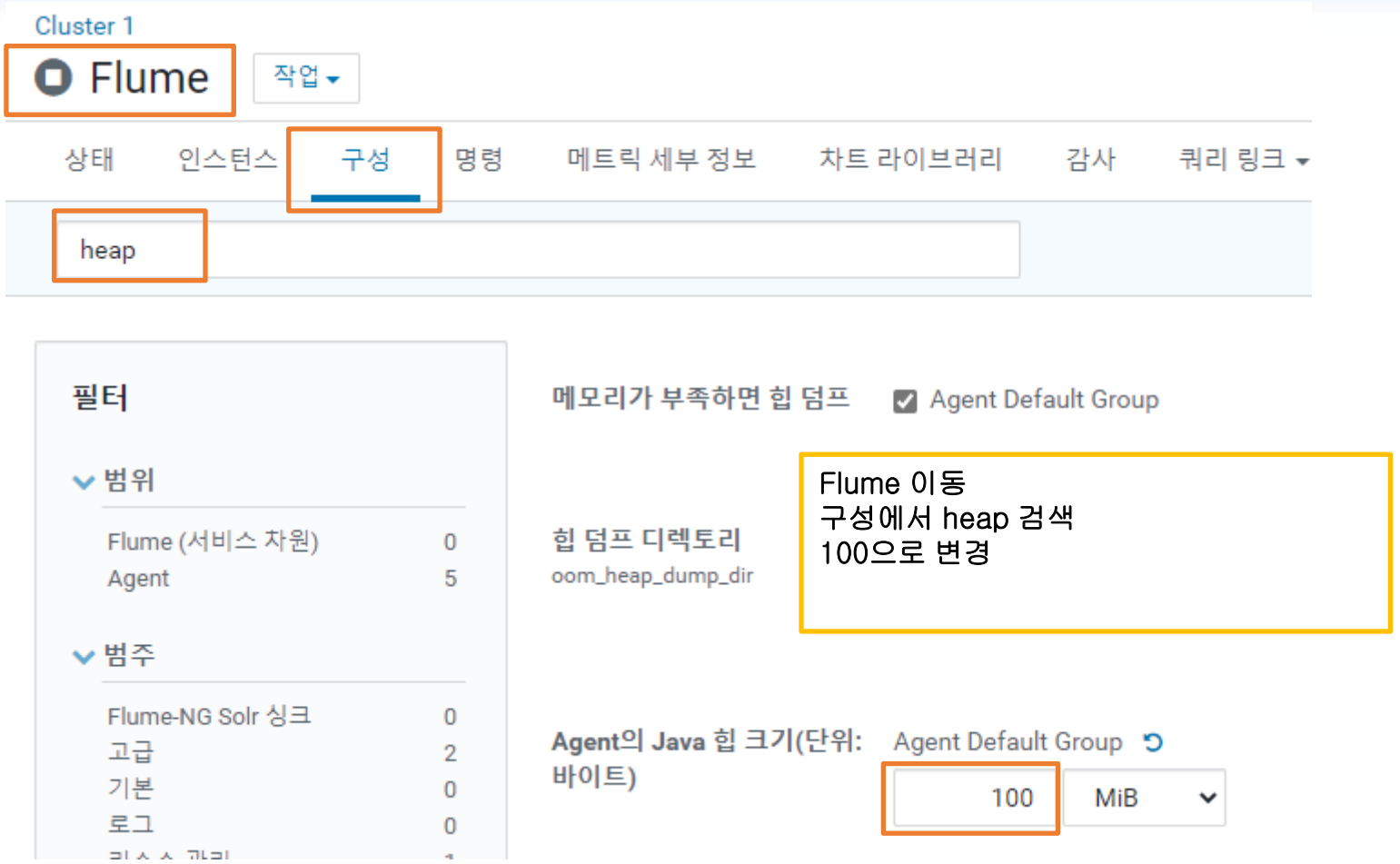
클러스터 클릭하여 홈으로 이동
작업 -> 서비스 추가 클릭
Flume 선택
Agent에서 server2 선택



설치 완료


Cluster 1에 Flume 서비스 추가
Select Dependencies
역할 할당 사용자 지정
변경 내용 검토
요약
요약
새 서비스가 클러스터에 설치 및 구성되었습니다.
참고: 여전히 새 서비스를 시작해야 할 수 있습니다. 시작하기 전에 오래된 구성이 포함된 모든 종속 서비스를 재시작하는 것이 좋습니다. 이러한 작업은 아래에서 완료를 클릭하면 주 페이지에서 수행할 수 있습니다.
변경내용검토 후 완료
뒤로
완료

메모리 설정


Cluster 1
Flume
작업
상태
인스턴스
구성
명령
메트릭 세부 정보
차트 라이브러리
감사
쿼리 링크
heap
필터
범위
Flume (서비스 차원) 0
Agent 5
범주
Flume-NG Solr 싱크 0
고급 2
기본 0
로그 0
메모리가 부족하면 힙 덤프
Agent Default Group
힙 덤프 디렉토리
oom_heap_dump_dir
Flume 이동
구성에서 heap 검색
100으로 변경
Agent의 Java 힙 크기(단위: 바이트)
Agent Default Group
100
MiB

서비스 시작

작업 시작 클릭

Kafka

Kafka

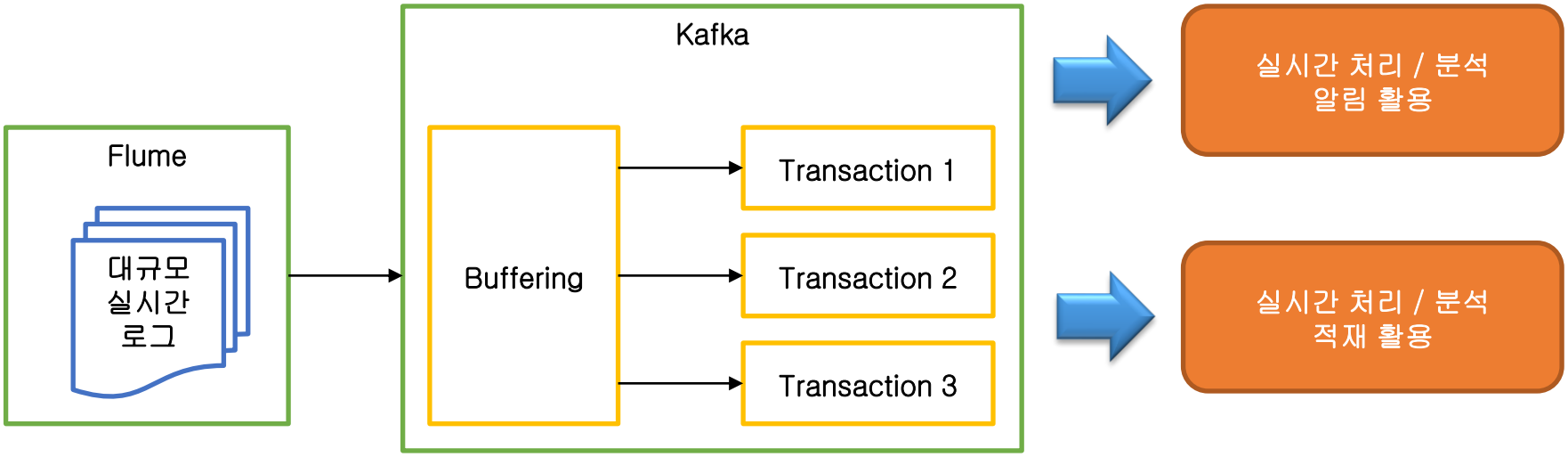
❖ 정의

- Message Oriented Middleware
- 대규모 메시지 데이터를 비동기 방식으로 중계
- 버퍼링을 통한 시스템의 안정적 전송

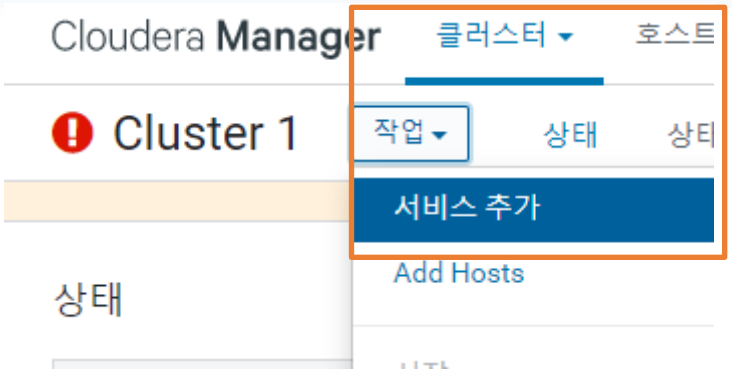
❖ 주요 기능

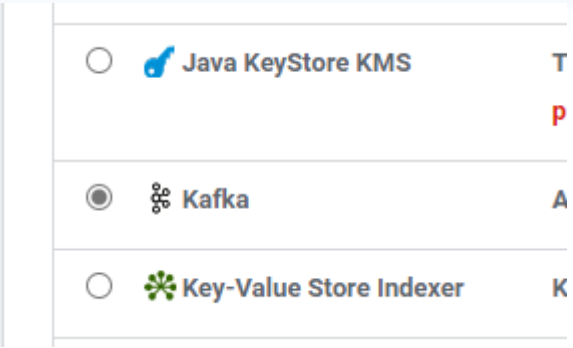
- Broker : 서비스 인스턴스
- Topic : Broker에서 데이터의 발행/소비 처리를 위한 저장소
- Provider : Broker의 특정 Topic에 데이터 전송
- Consumer : Broker의 특정 Topic에서 데이터 수신

카프카 활용

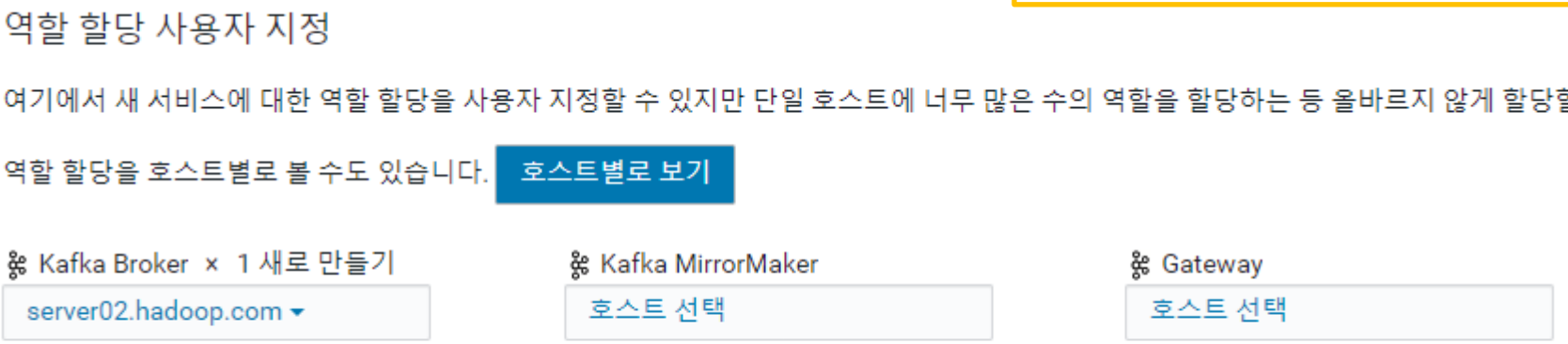

Flume
대규모
실시간
로그
Kafka
Buffering
Transaction 1
Transaction 2
Transaction 3
실시간 처리 / 분석
알림 활용
실시간 처리 / 분석
적재 활용

설정

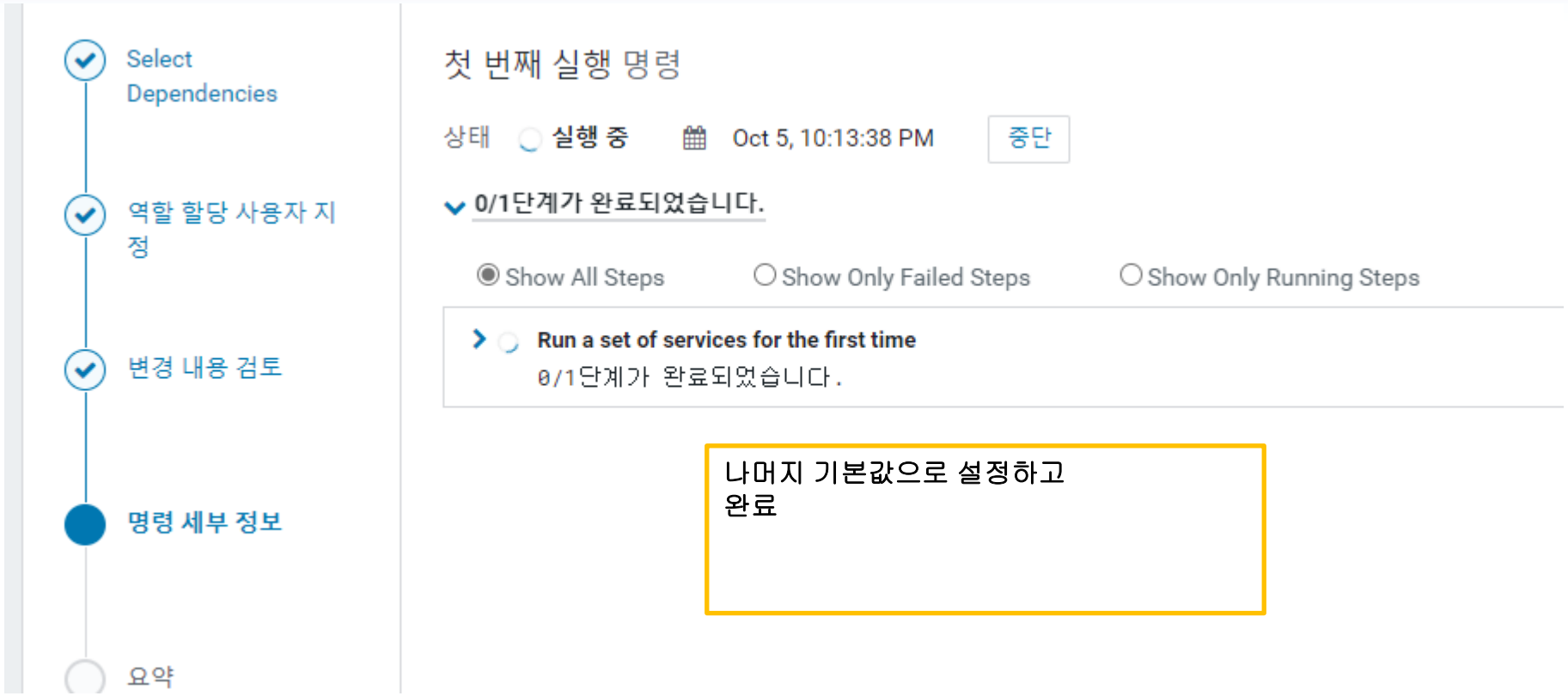




클러스터 클릭
작업 -> 서비스 추가
Kafka 선택
Broker를 server2로 변경



설치 완료



나머지 기본값으로 설정하고
완료

Cluster 1

Kafka 작업

상태 인스턴스 구성 명령 차트 라이브러리 감사 쿼리 링크

data reten

필터

범위

Data Retention Time
log.retention.ms

Kafka Broker Default Group

10 분

Flume 설정

구성 파일 분석

- Sources, channels, sinks 이름 지정
  - SmartCar_Agent.sources = SmartCarInfo_SpoolSource
  - SmartCar_Agent.channels = SmartCarInfo_Channel
  - SmartCar_Agent.sinks   = SmartCarInfo_LoggerSink

- 소스 타입 설정
  - SmartCar_Agent.sources.SmartCarInfo_SpoolSource.type = spooldir
- 스풀 위치 지정
  - SmartCar_Agent.sources.SmartCarInfo_SpoolSource.spoolDir = /home/pilot-pjt/working/car-batch-log
- 삭제 정책 설정
  - SmartCar_Agent.sources.SmartCarInfo_SpoolSource.deletePolicy = immediate
- Channel에 전송할 batch 크기
  - SmartCar_Agent.sources.SmartCarInfo_SpoolSource.batchSize = 1000

구성 파일 분석

- Channel 설정
  - SmartCar_Agent.channels.SmartCarInfo_Channel.type = memory
- Channel에 저장할최대 event 수
  - SmartCar_Agent.channels.SmartCarInfo_Channel.capacity = 100000
- Source에서 가져오거나 sink에 전달할 최대 event 수
  - SmartCar_Agent.channels.SmartCarInfo_Channel.transactionCapacity = 10000

- 정상 동작 확인 위해 flume으로 수집된 데이터를 logge로 전송
  - SmartCar_Agent.sinks.SmartCarInfo_LoggerSink.type = logger
- SpoolSource와 Memory chnnel 연결
  - SmartCar_Agent.sources.SmartCarInfo_SpoolSource.channels = SmartCarInfo_Channel
- Loggger와 Memory chnnel 연결
  - SmartCar_Agent.sinks.SmartCarInfo_LoggerSink.channel = SmartCarInfo_Channel

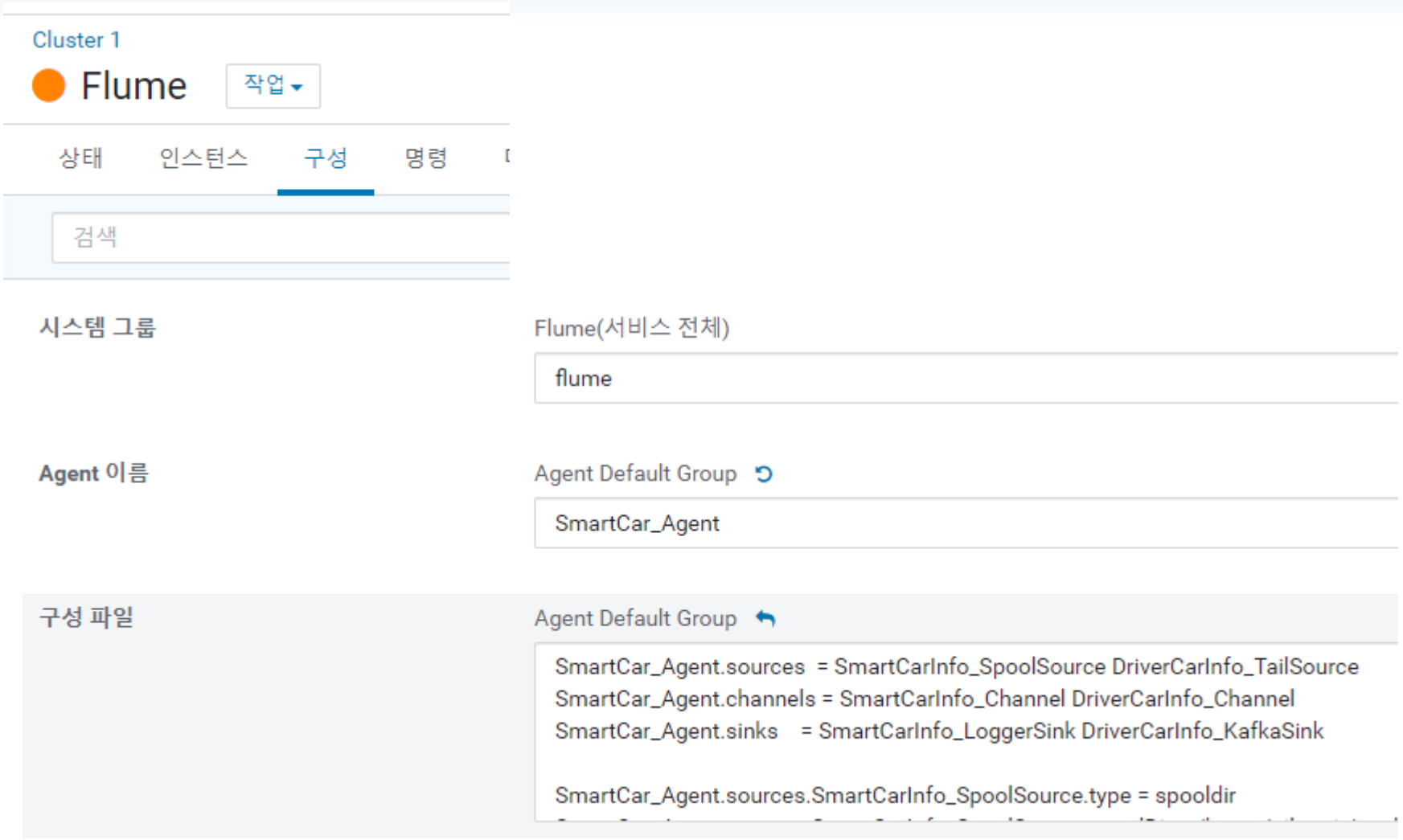
interceptors

- ❖ 이해하기
  - Source와 Channel 중간에서 데이터 가공하는 역할
  - 플럼 Source에서 유입되는 데이터 중 일부를 수정/추가/가공/정제 등
  - 플럼 데이터 전송 단위 : Event = Header + Body
  - interceptors는 Header 특정값 추가, Body 데이터 가공
- ❖ 변수선언
  - SmartCar_Agent.sources.SmartCarInfo_SpoolSource.interceptors = filterInterceptor
- ❖ 정규 표현식을 이용한 필터링
  - SmartCar_Agent.sources.SmartCarInfo_SpoolSource.interceptors.filterInterceptor .type = regex_filter
- ❖ 14자리 날짜 형식으로 시작하는 데이터
  - SmartCar_Agent.sources.SmartCarInfo_SpoolSource.interceptors.filterInterceptor .regex = ^₩₩d{14}
- ❖ 제외된 이벤트 처리
  - SmartCar_Agent.sources.SmartCarInfo_SpoolSource.interceptors.filterInterceptor .excludeEvents = false

카프카 연동

- 외부 수행 명령 결과를 flume으로 가져와 수집
  - SmartCar_Agent.sources.DriverCarInfo_TailSource.type = exec
  - SmartCar_Agent.sources.DriverCarInfo_TailSource.command = tail -F /home/pilot-pjt/working/driver-realtime-log/SmartCarDriverInfo.log
  - SmartCar_Agent.sources.DriverCarInfo_TailSource.restart = true
  - SmartCar_Agent.sources.DriverCarInfo_TailSource.batchSize = 1000
- 카프카 연동
  - SmartCar_Agent.sinks.DriverCarInfo_KafkaSink.type = org.apache.flume.sink.kafka.KafkaSink
  - SmartCar_Agent.sinks.DriverCarInfo_KafkaSink.topic = SmartCar-Topic
  - SmartCar_Agent.sinks.DriverCarInfo_KafkaSink.brokerList = server02.hadoop.com:9092
  - SmartCar_Agent.sinks.DriverCarInfo_KafkaSink.requiredAcks = 1
  - SmartCar_Agent.sinks.DriverCarInfo_KafkaSink.batchSize = 1000

구성 파일 등록


Cluster 1
Flume
작업
상태
인스턴스
구성
명령
검색
시스템 그룹
Flume(서비스 전체)
flume
Agent 이름
Agent Default Group
SmartCar_Agent
구성 파일
Agent Default Group
SmartCar_Agent.sources = SmartCarInfo_SpoolSource DriverCarInfo_TailSource
SmartCar_Agent.channels = SmartCarInfo_Channel DriverCarInfo_Channel
SmartCar_Agent.sinks = SmartCarInfo_LoggerSink DriverCarInfo_KafkaSink
SmartCar_Agent.sources.SmartCarInfo_SpoolSource.type = spooldir

Kafka 동작 확인

- [root@server02 /]# kafka-console-producer --broker-list server02.hadoop.com:9092 -topic SmartCar-Topic
- >20/10/11 13:01:43 INFO clients.Metadata: Cluster ID: djul1ODDShmRdbl_Ufhxhg
- test message
- >

- [root@server02 /]# kafka-console-consumer --bootstrap-server server02.hadoop.com:9092 --topic SmartCar-Topic --partition 0 --from-beginning
- >20/10/11 13:01:43 INFO clients.Metadata: Cluster ID: djul1ODDShmRdbl_Ufhxhg
- test message

수집 기능 테스트

자동차 시뮬레이터 동작

- [root@server02 working]java -cp bigdata.smartcar.loggen-1.0.jar com.wikibook.bigdata.smartcar.loggen.CarLogMain 20200101 3&
- [root@server02 working]# java -cp bigdata.smartcar.loggen-1.0.jar com.wikibook.bigdata.smartcar.loggen.DriverLogMain 20200101 3 &

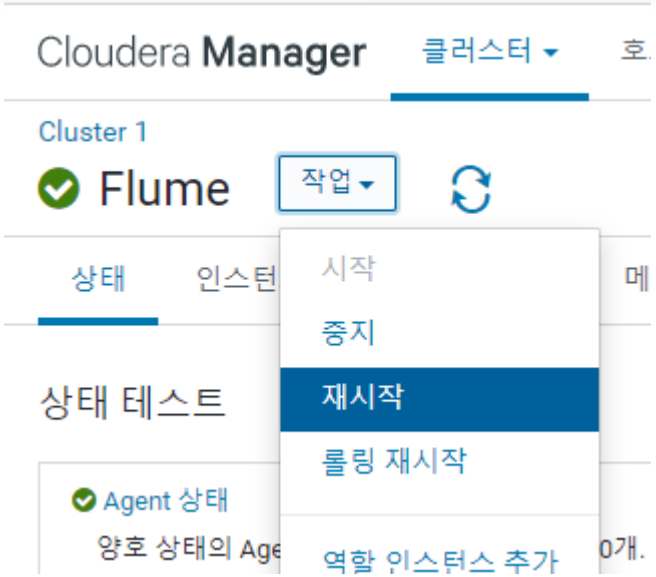
- [root@server02 working]# cd SmartCar/
- [root@server02 SmartCar]# cat SmartCarStatusInfo_20200101.txt

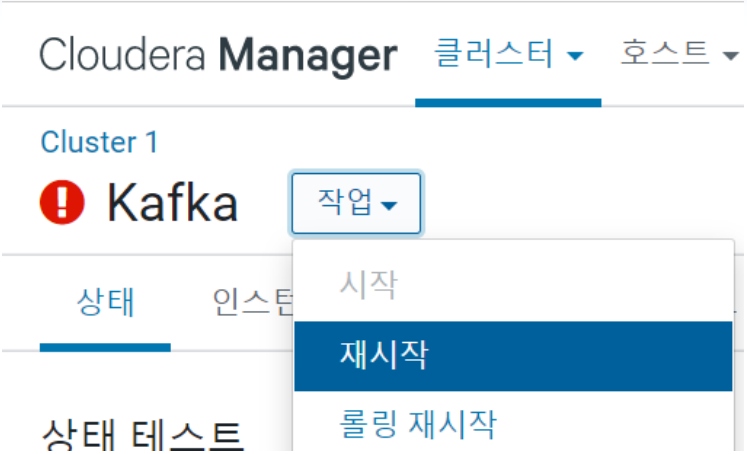
- [root@server02 SmartCar]# cd ..
- [root@server02 working]# cd driver-realtime-log/
- [root@server02 driver-realtime-log]# tail -f SmartCarDriverInfo.log

분석 정보 이동

- [root@server02 driver-realtime-log]# cd ..
- [root@server02 working]# cd SmartCar/
- [root@server02 SmartCar]# ls
- SmartCarStatusInfo_20200101.txt
- [root@server02 SmartCar]# mv SmartCarStatusInfo_20200101.txt /home/pilot-pjt/working/car-batch-log/

서비스 재시작


Cloudera Manager
클러스터
Cluster 1
Flume
작업
시작
중지
재시작
롤링 재시작
상태
상태 테스트
Agent 상태


Cloudera Manager
클러스터
호스트
Cluster 1
Kafka
작업
시작
재시작
롤링 재시작
상태
상태 테스트

수집 기능 점검

- [root@server02 ~]# tail -f /var/log/flume-ng/flume-cmf-flume-AGENT-server02.hadoop.com.log

```
2020-10-11 16:52:40,749 INFO org.apache.flume.sink.L
 31 30 31 32 33 35 39 32 34 2C 45 20190101235924,E }
2020-10-11 16:52:40,749 INFO org.apache.flume.sink.L
 31 30 31 32 33 35 39 32 38 2C 45 20190101235928,E }
```

- [root@server02 ~]# kafka-console-consumer --bootstrap-server server02.hadoop.com:9092 --topic SmartCar-Topic --partition 0

```
20190101003428,B0003,4,0,F,N,37,D04
20190101003426,C0001,2,0,R3,R,10,D06
20190101003438,H0002,1,0,F,N,5,D04
20190101003430,B0003,3,0,F,N,52,D05
```

자동차 시뮬레이터 종료

```
[root@server02 ~]# ps -ef | grep smartcar.log
root      4557     1  1 16:51 ?        00:00:04 java -cp bigdata.smartcar.loggen-1.0.jar com.wikibook.bigdata.smartcar.loggen.DriverLogMain 20190101 3
root      4584     1  3 16:51 ?        00:00:07 java -cp bigdata.smartcar.loggen-1.0.jar com.wikibook.bigdata.smartcar.loggen.CarLogMain 20190101 3
root      5467  5033  0 16:54 pts/1    00:00:00 grep smartcar.log
[root@server02 ~]# kill -9 4557
[root@server02 ~]# kill -9 4584
[root@server02 ~]# ps -ef | grep smartcar.log
root      5504  5033  0 16:55 pts/1    00:00:00 grep smartcar.log
```