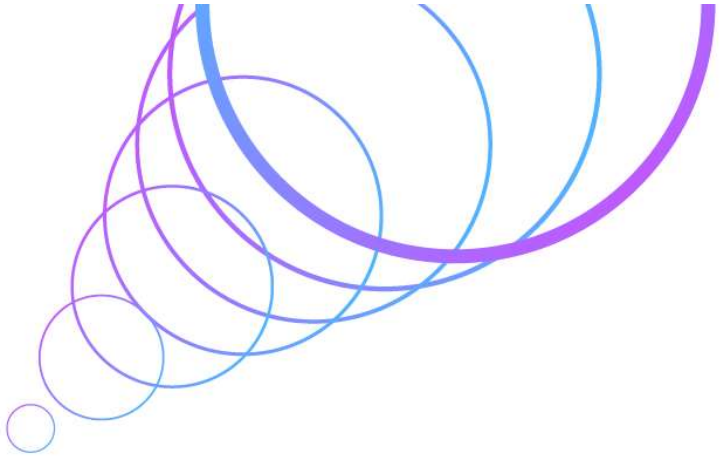
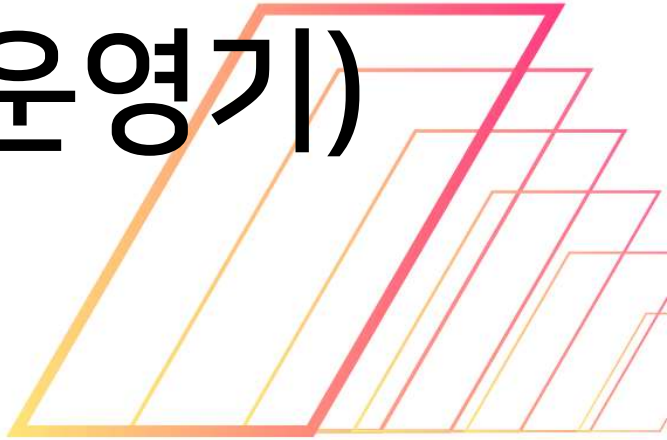




네이버 최대의 데이터 저장소 운영기 (HBase Locality기반 운영기)

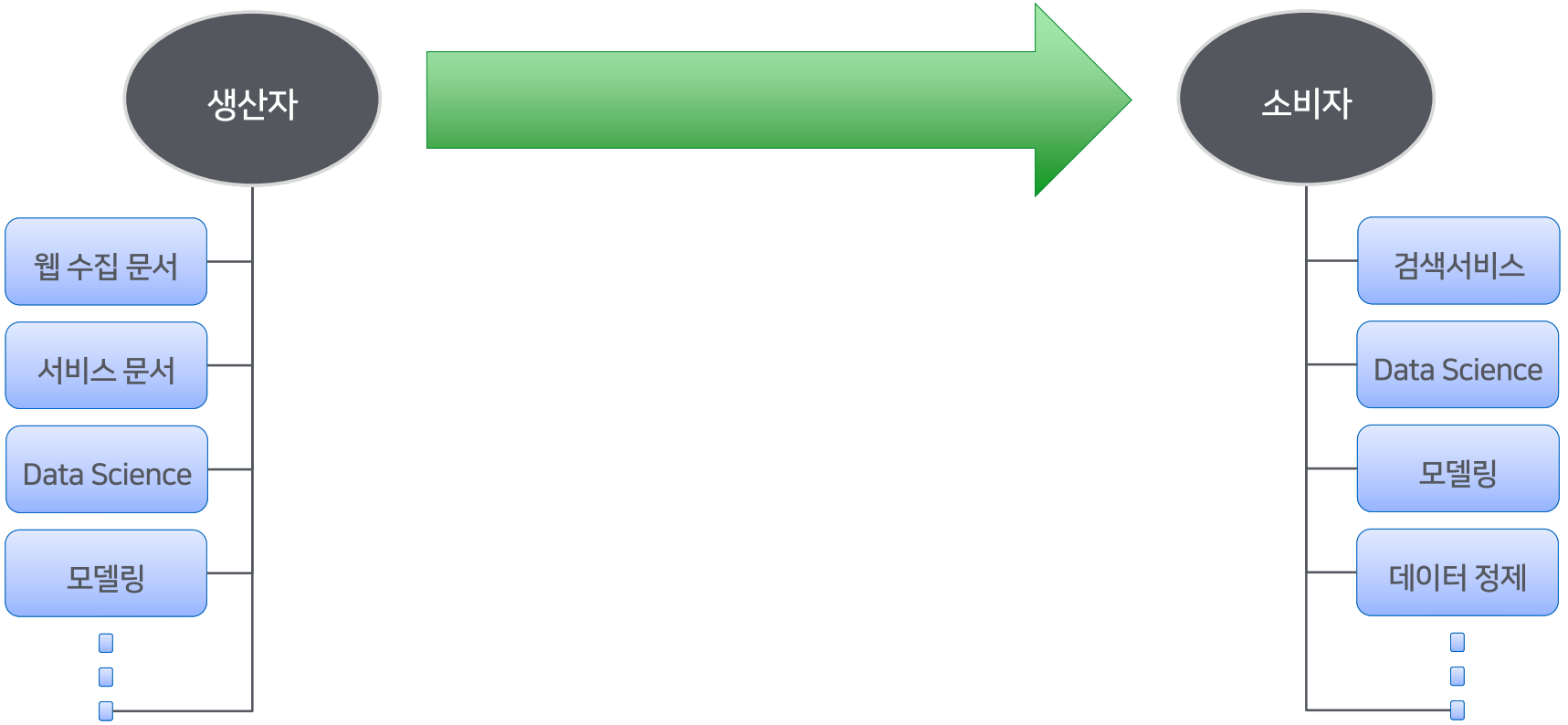
정한룡 NAVER

CONTENTS

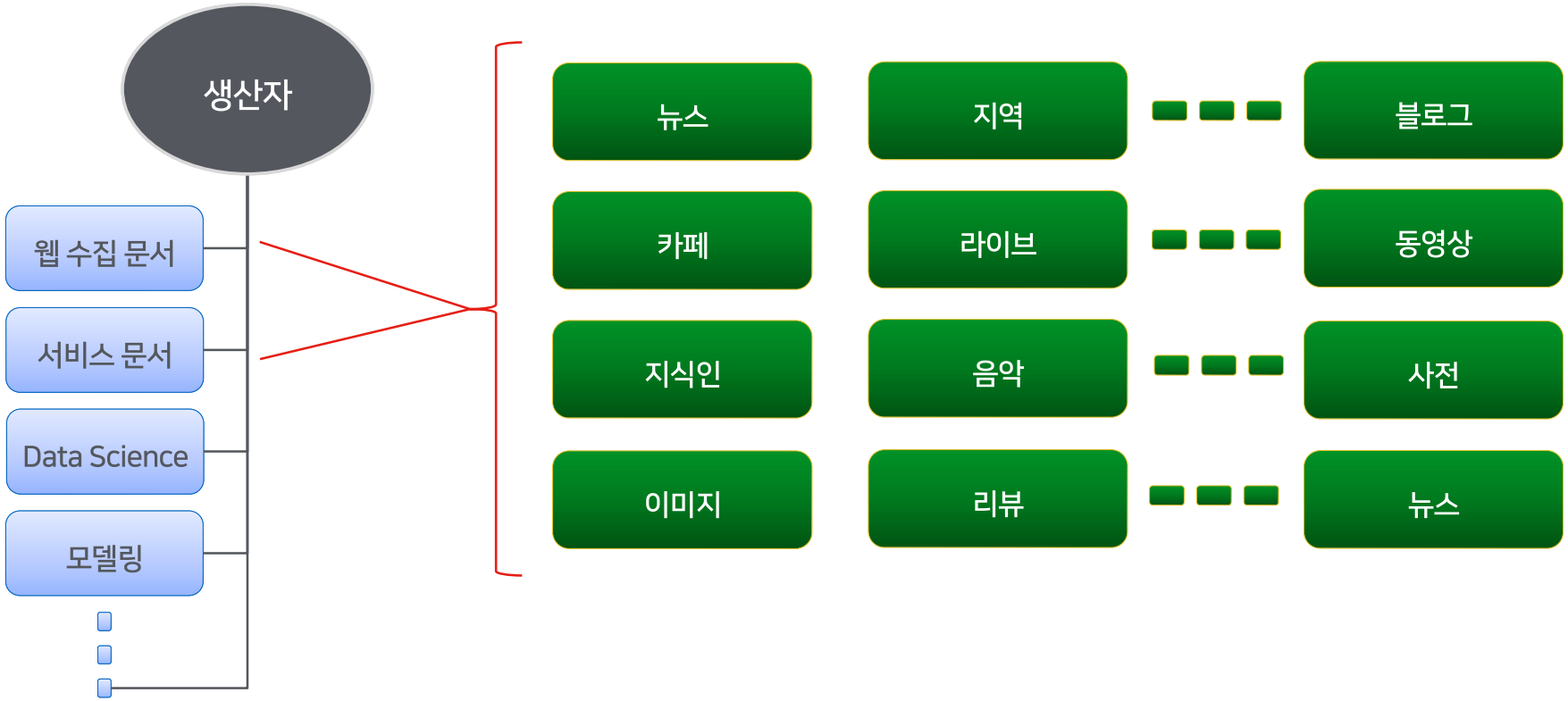
DEVIEW 2021

HBase기반 데이터 저장소

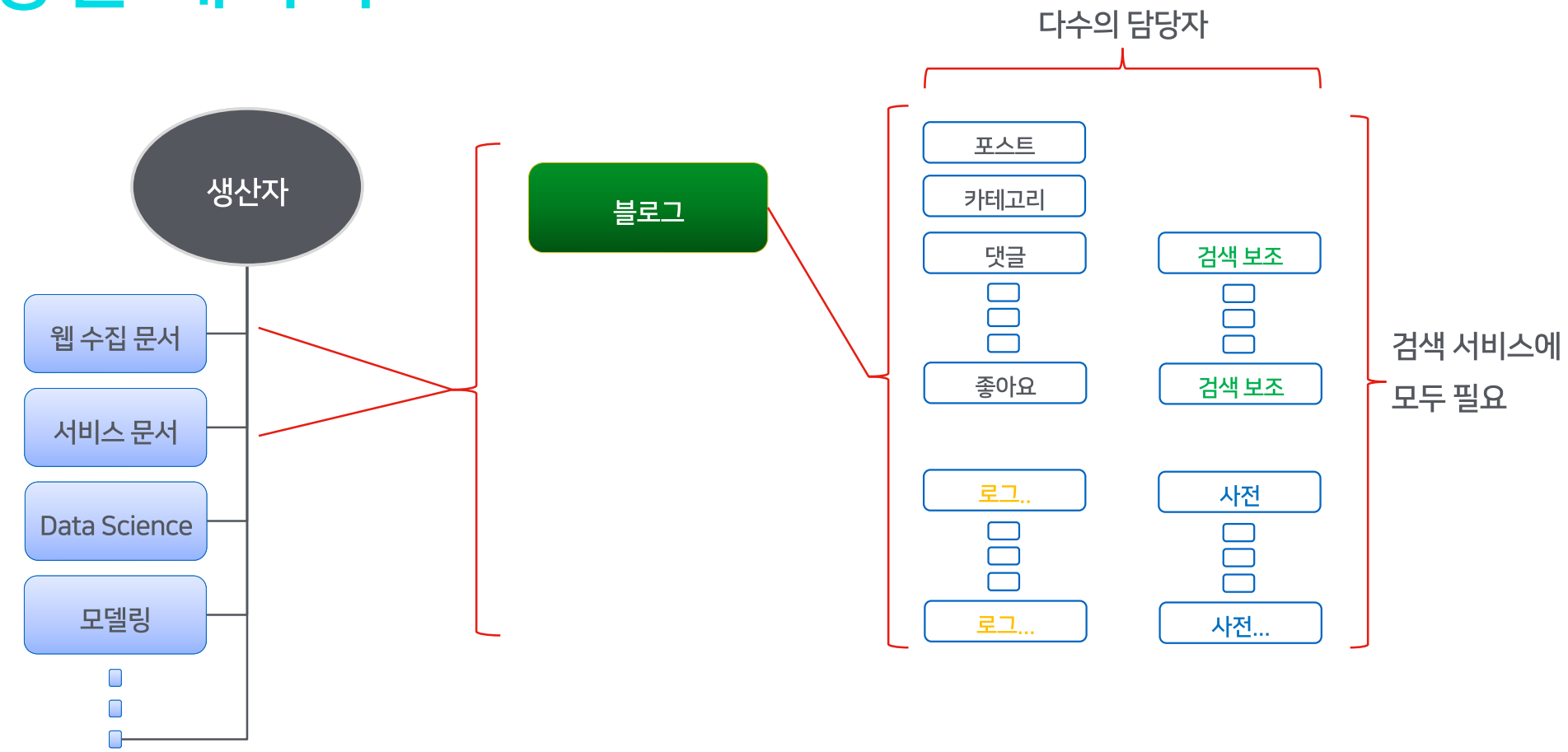
데이터 저장소가 있기 전


생산자
웹 수집 문서
서비스 문서
Data Science
모델링
소비자
검색서비스
Data Science
모델링
데이터 정제

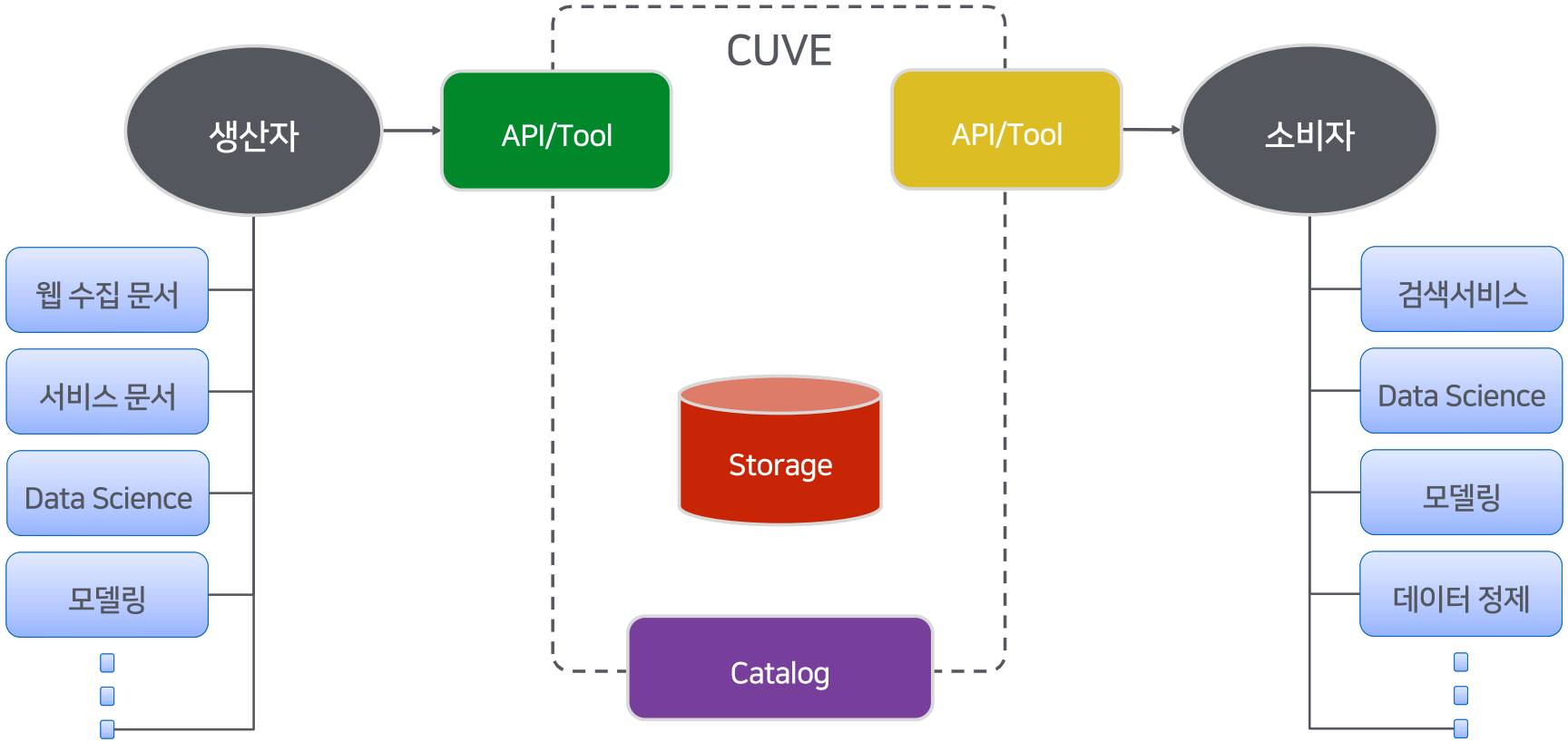
다수의 생산자


생산자
웹 수집 문서
서비스 문서
Data Science
모델링
뉴스
지역
블로그
카페
라이브
동영상
지식인
음악
사전
이미지
리뷰
뉴스

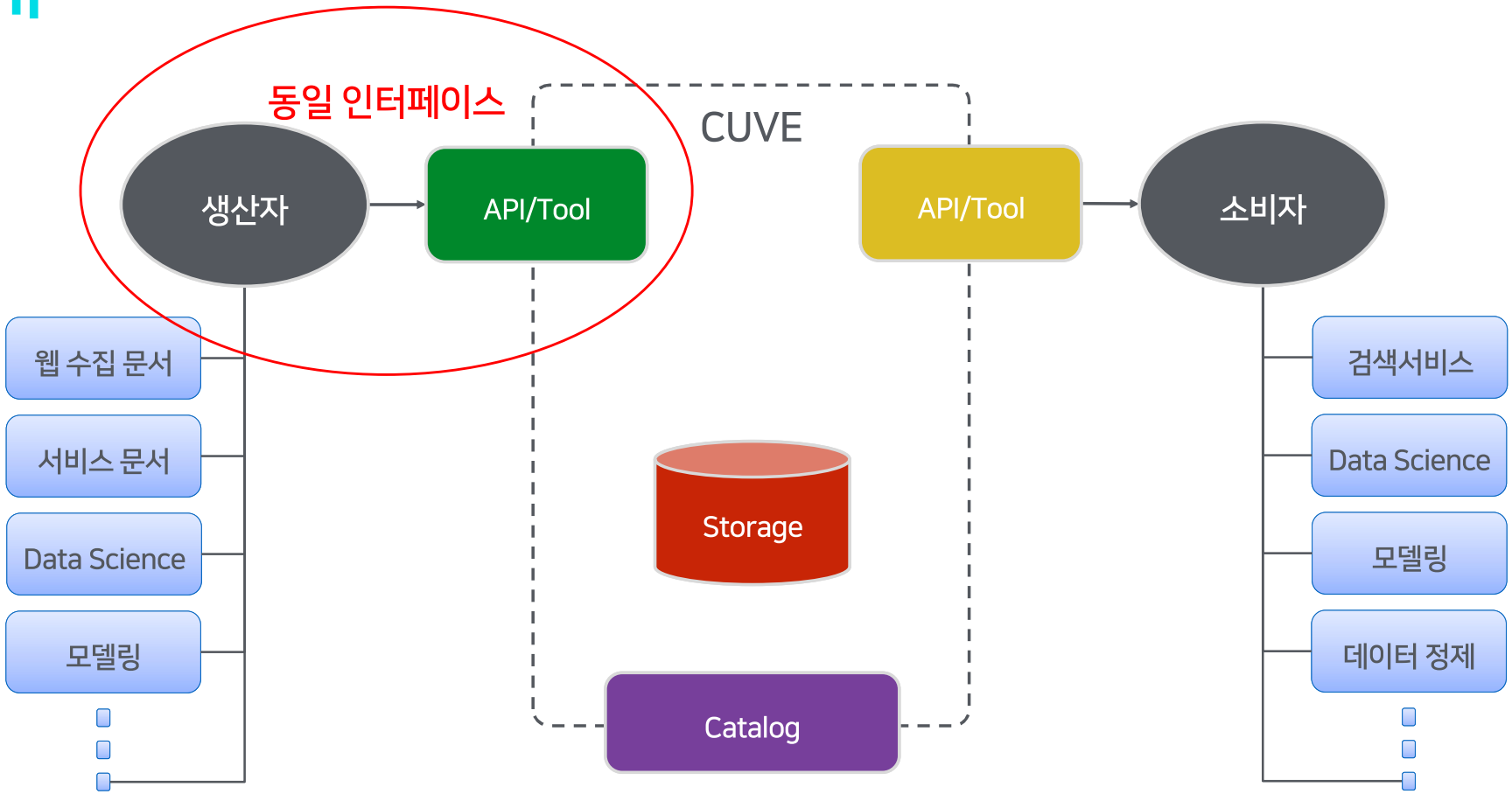
다양한 데이터


다수의 담당자
생산자
웹 수집 문서
서비스 문서
Data Science
모델링
블로그
포스트
카테고리
댓글
좋아요
로그..
로그...
검색 보조
검색 보조
사전
사전...
검색 서비스에
모두 필요

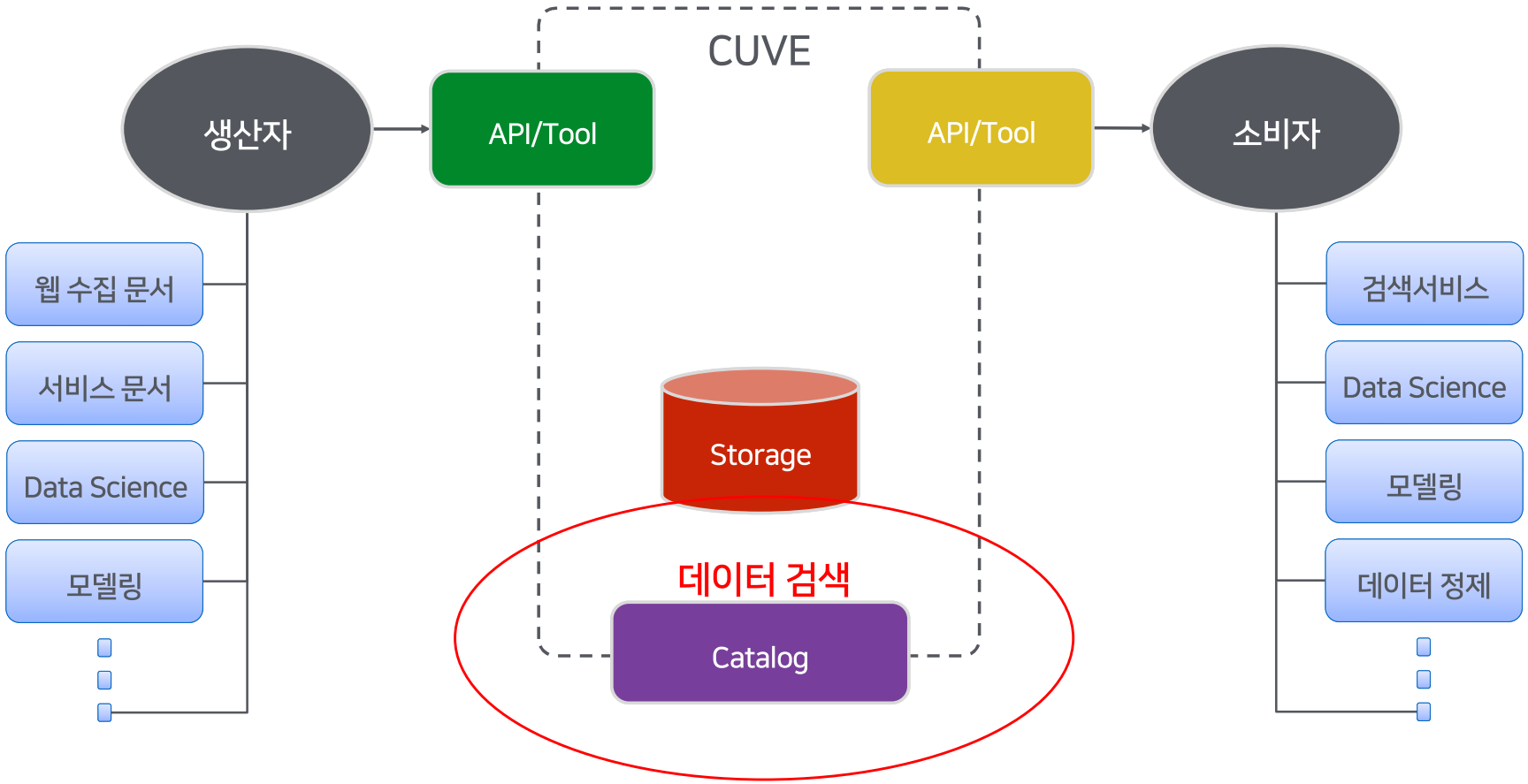
현재


CUVE
생산자
API/Tool
API/Tool
소비자
웹 수집 문서
서비스 문서
Data Science
모델링
Storage
Catalog
검색서비스
Data Science
모델링
데이터 정제

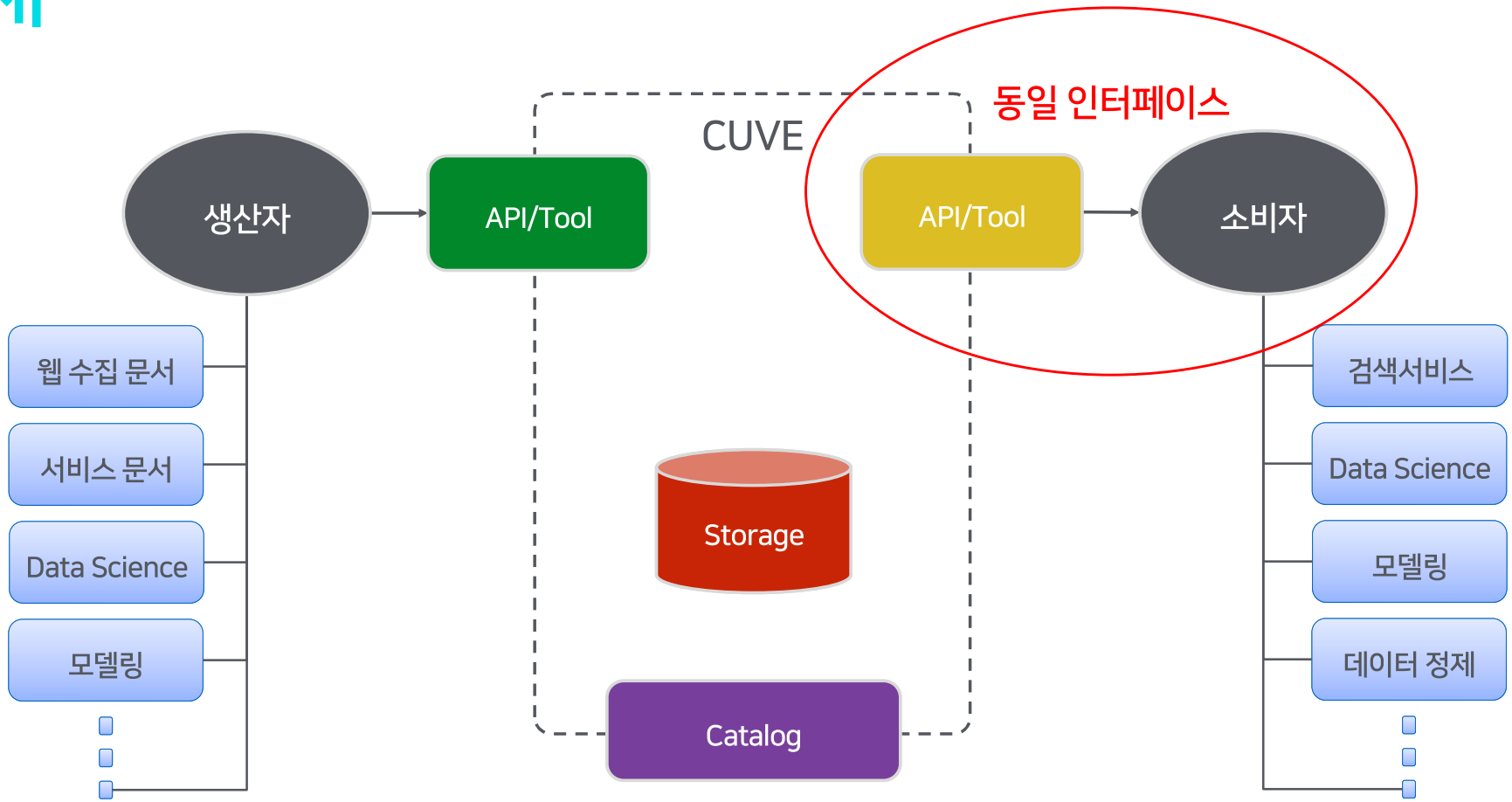
현재


동일 인터페이스
CUVE
생산자
API/Tool
API/Tool
소비자
웹 수집 문서
서비스 문서
Data Science
모델링
Storage
Catalog
검색서비스
Data Science
모델링
데이터 정제

현재


CUVE
생산자
API/Tool
API/Tool
소비자
웹 수집 문서
서비스 문서
Data Science
모델링
Storage
데이터 검색
Catalog
검색서비스
Data Science
모델링
데이터 정제

현재


동일 인터페이스
CUVE
생산자
API/Tool
API/Tool
소비자
웹 수집 문서
서비스 문서
Data Science
모델링
Storage
Catalog
검색서비스
Data Science
모델링
데이터 정제

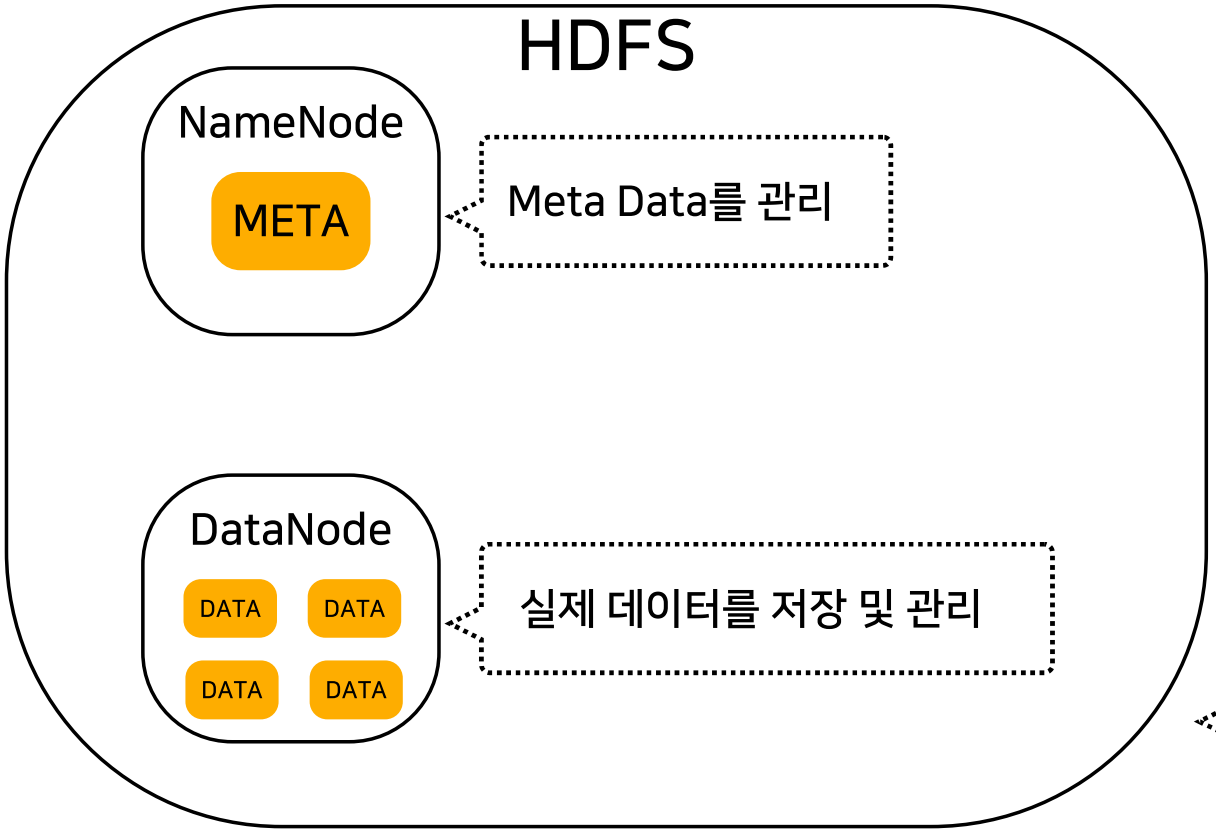
HBase Key구조 및 자세한 건…

DEVIEW 2021

2017 DEVIEW

HBase 기반 검색 데이터 저장소

HDFS에 대해서

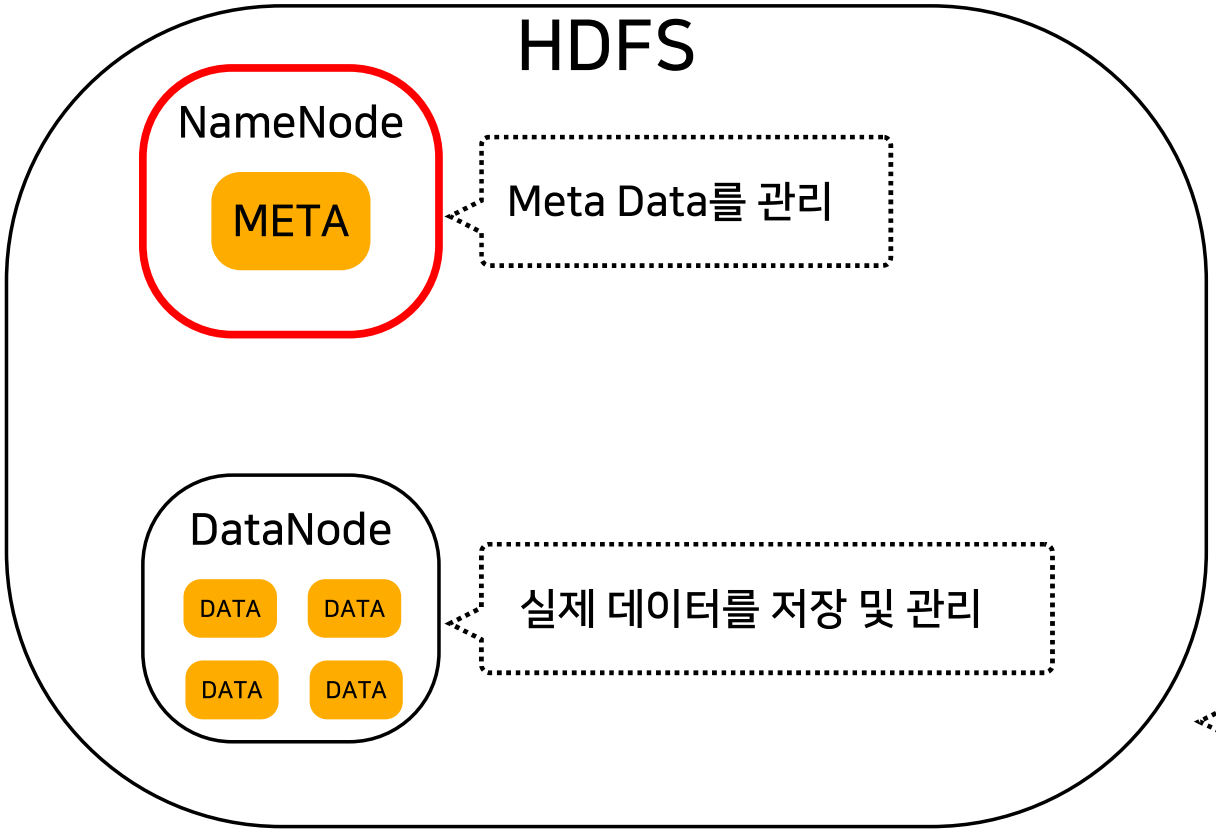


Hadoop Distributed File System

대용량 데이터를 분산된 서버에 저장하기 위하여 개발된 시스템

- 장애복구 (Replication)
- 스트리밍의 방식 데이터 접근
- 대용량 데이터 저장
- 데이터 무결성

HDFS에 대해서

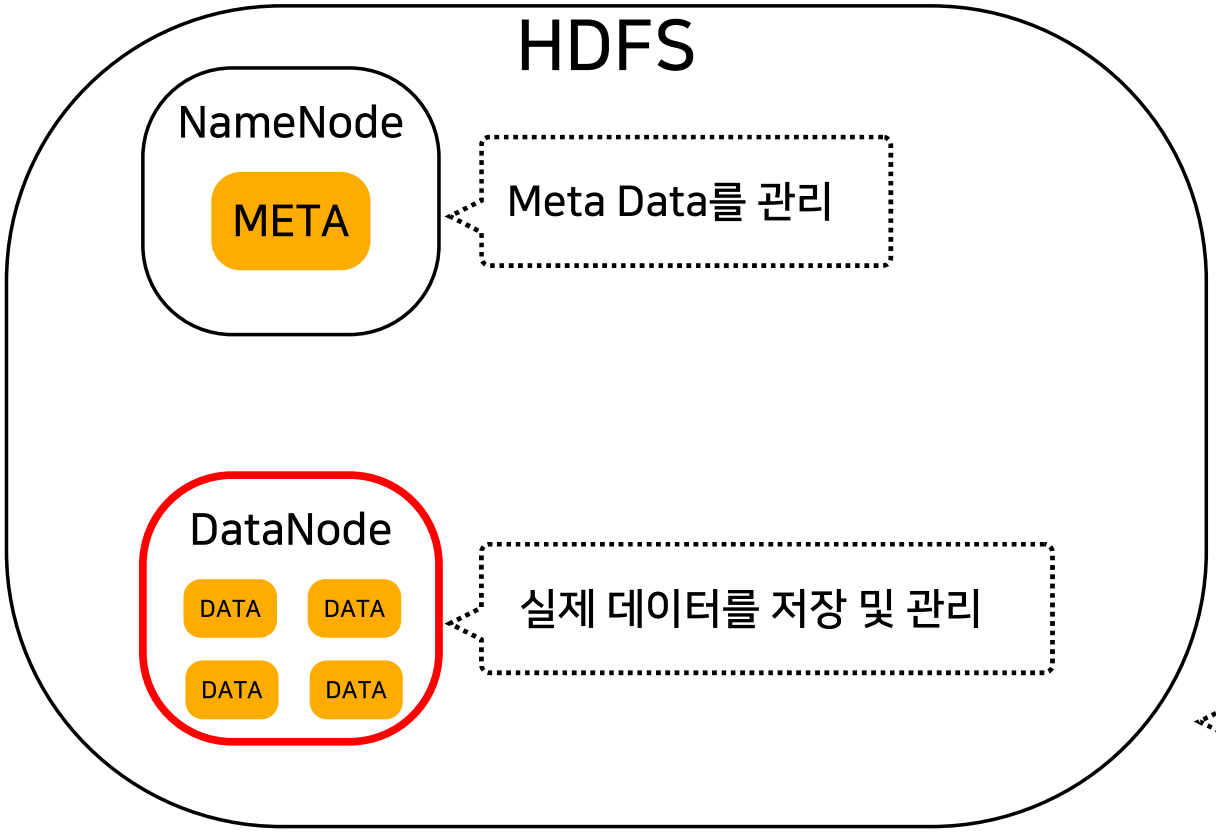


Hadoop Distributed File System

대용량 데이터를 분산된 서버에 저장하기 위하여 개발된 시스템

- 장애복구 (Replication)
- 스트리밍의 방식 데이터 접근
- 대용량 데이터 저장
- 데이터 무결성

HDFS에 대해서



Hadoop Distributed File System

대용량 데이터를 분산된 서버에 저장하기 위하여 개발된 시스템

- 장애복구 (Replication)
- 스트리밍의 방식 데이터 접근
- 대용량 데이터 저장
- 데이터 무결성

HBase에 대해서

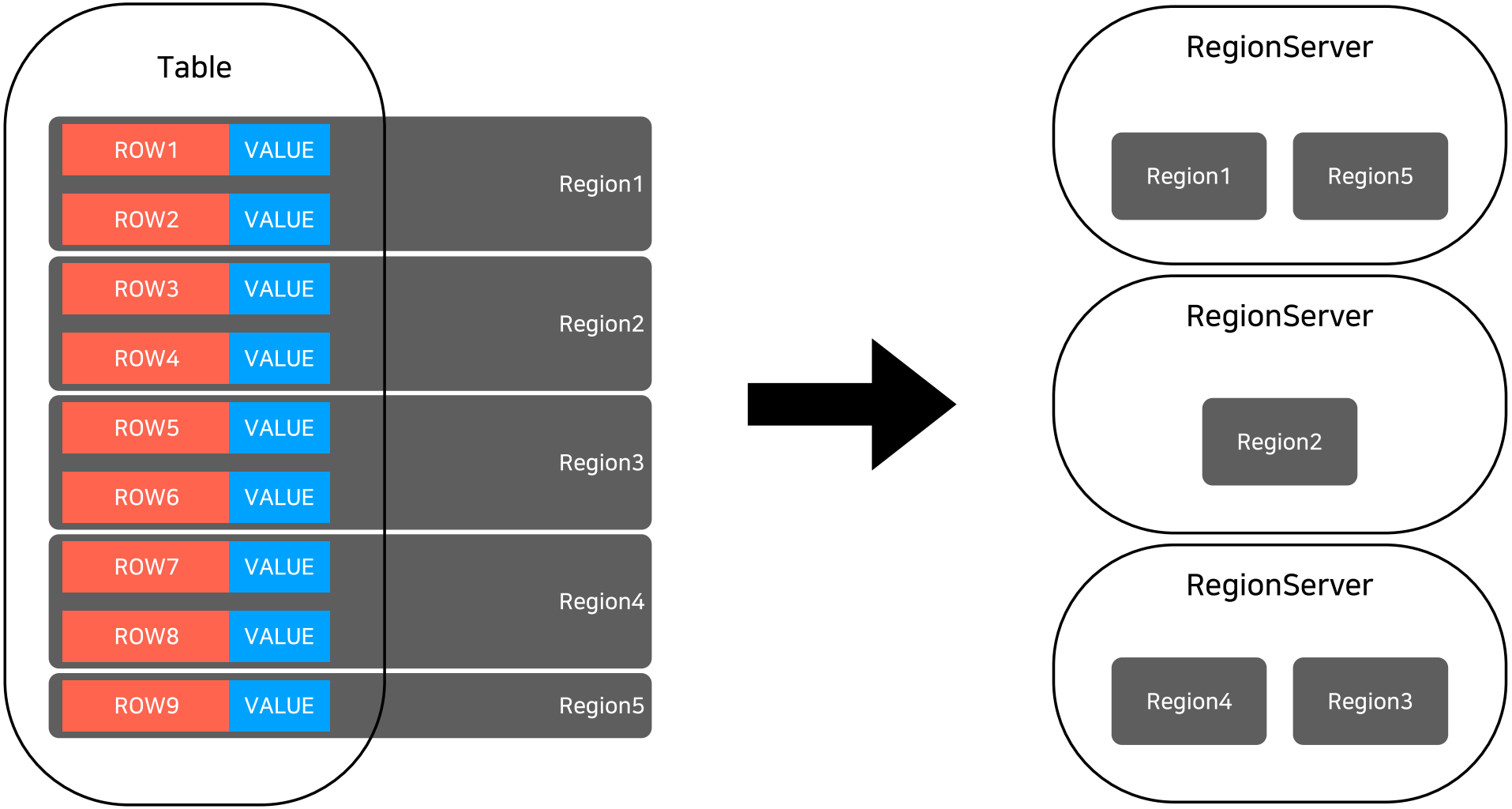
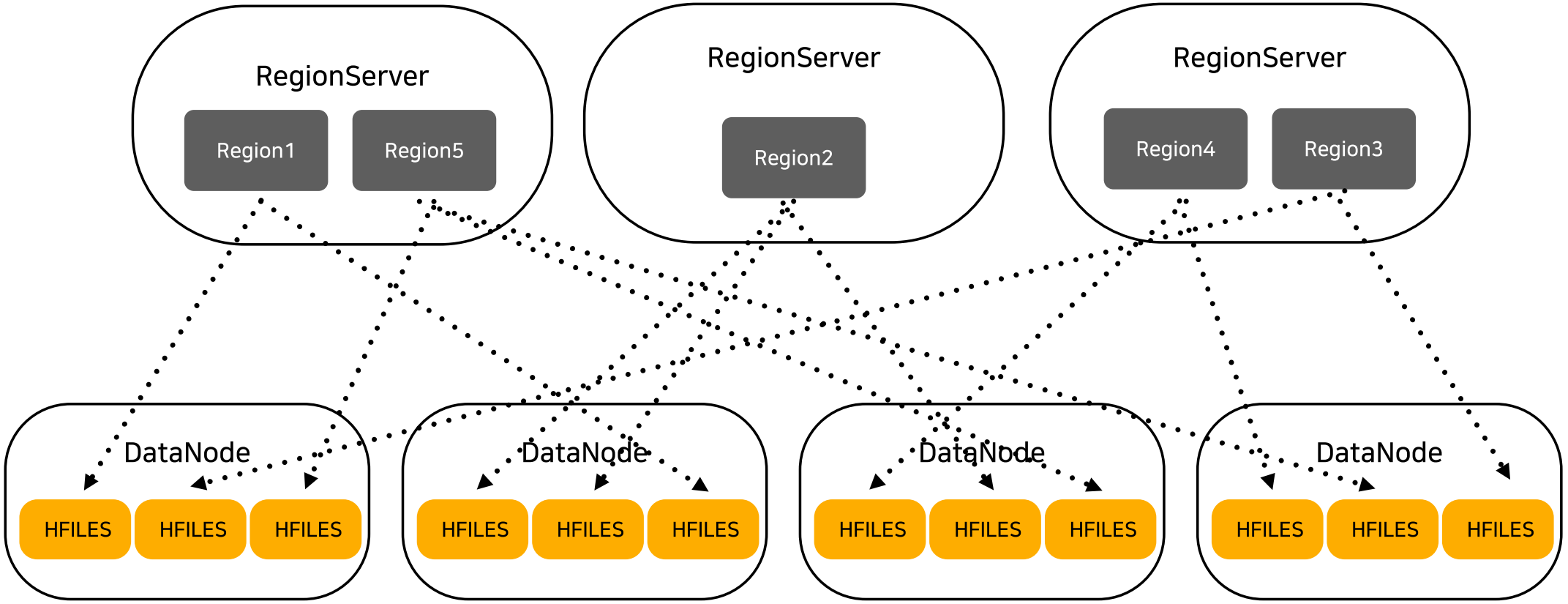
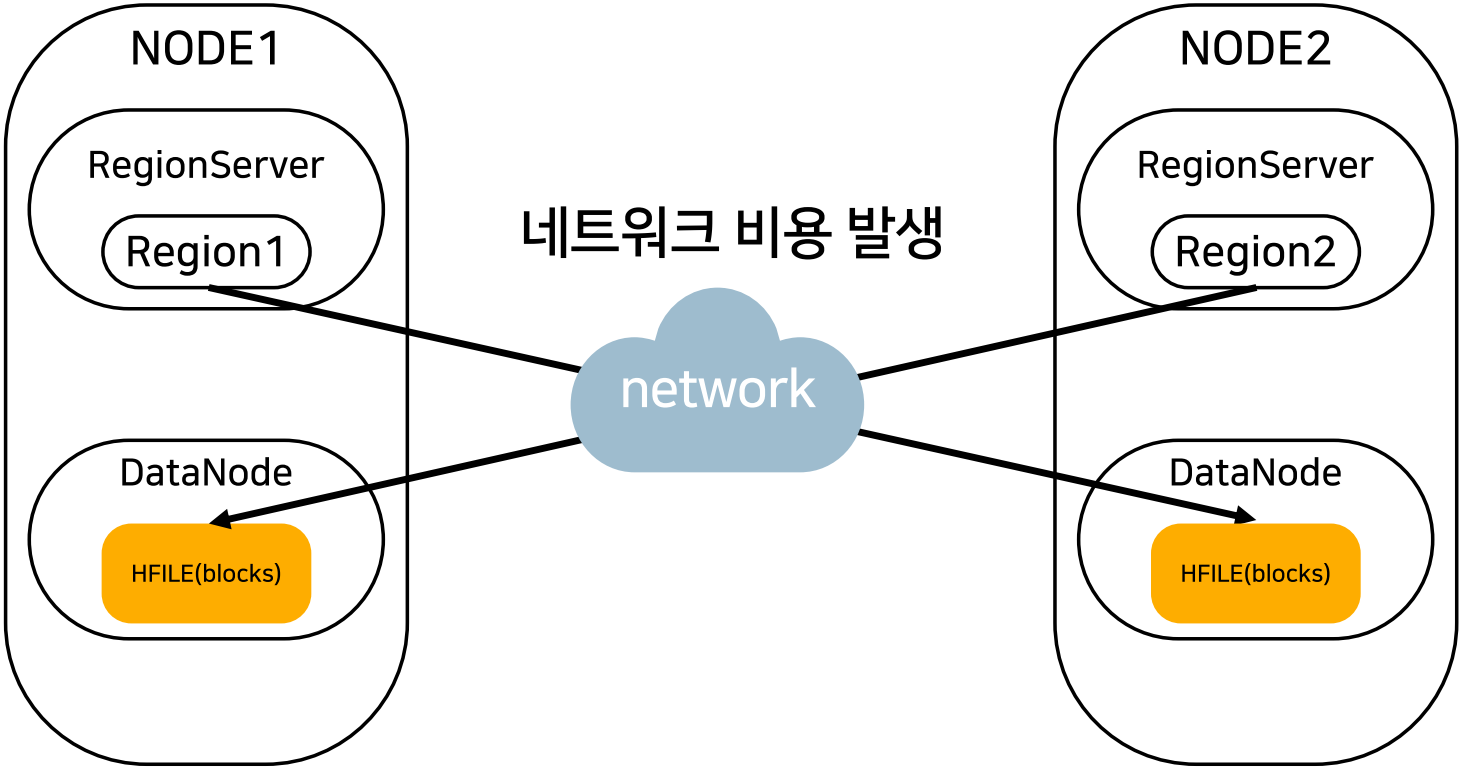


Table
ROW1
VALUE
ROW2
VALUE
Region1
ROW3
VALUE
ROW4
VALUE
Region2
ROW5
VALUE
ROW6
VALUE
Region3
ROW7
VALUE
ROW8
VALUE
Region4
ROW9
VALUE
Region5
RegionServer
Region1
Region5
RegionServer
Region2
RegionServer
Region4
Region3

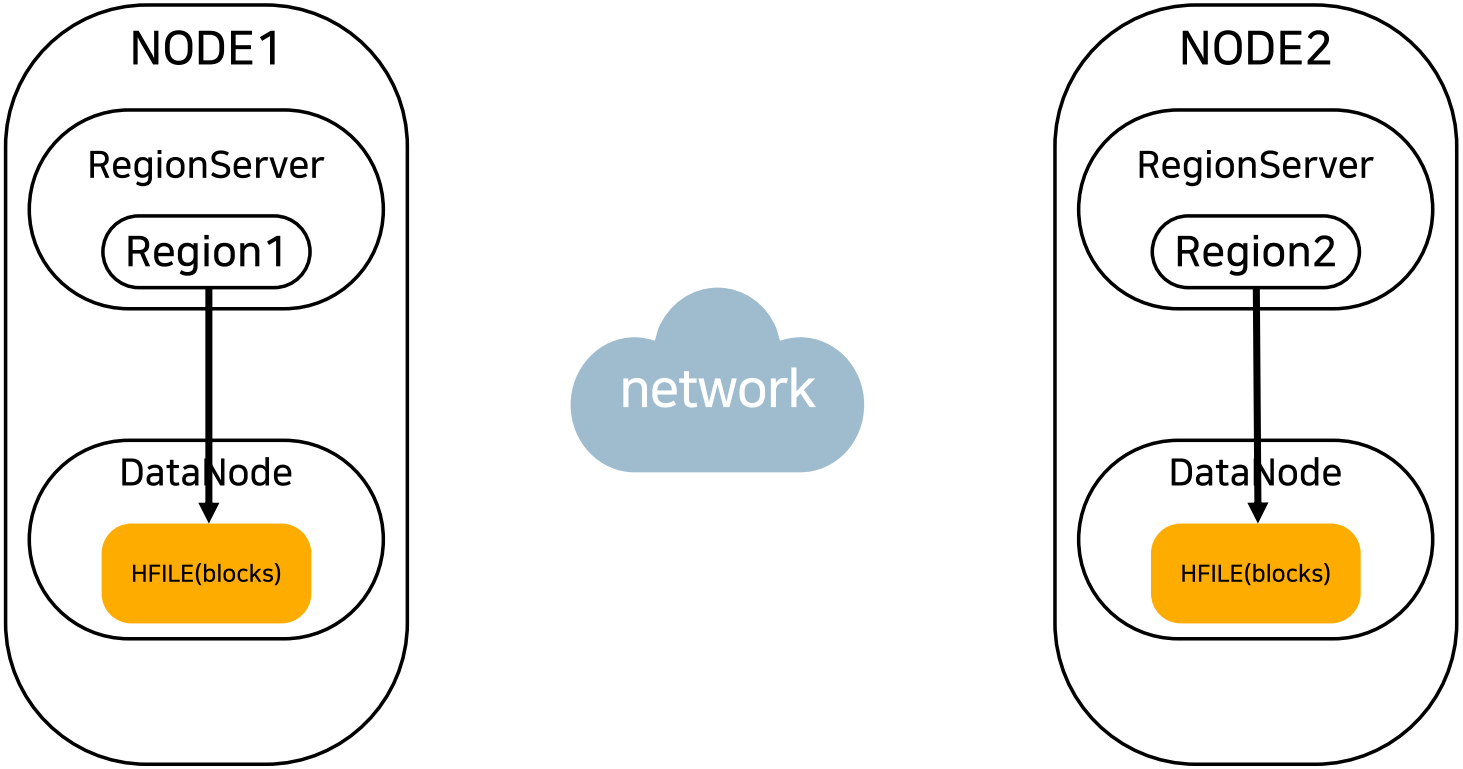
HBase에 대해서


RegionServer
Region1
Region5
RegionServer
Region2
RegionServer
Region4
Region3
DataNode
HFILES
HFILES
HFILES
DataNode
HFILES
HFILES
HFILES
DataNode
HFILES
HFILES
HFILES
DataNode
HFILES
HFILES
HFILES

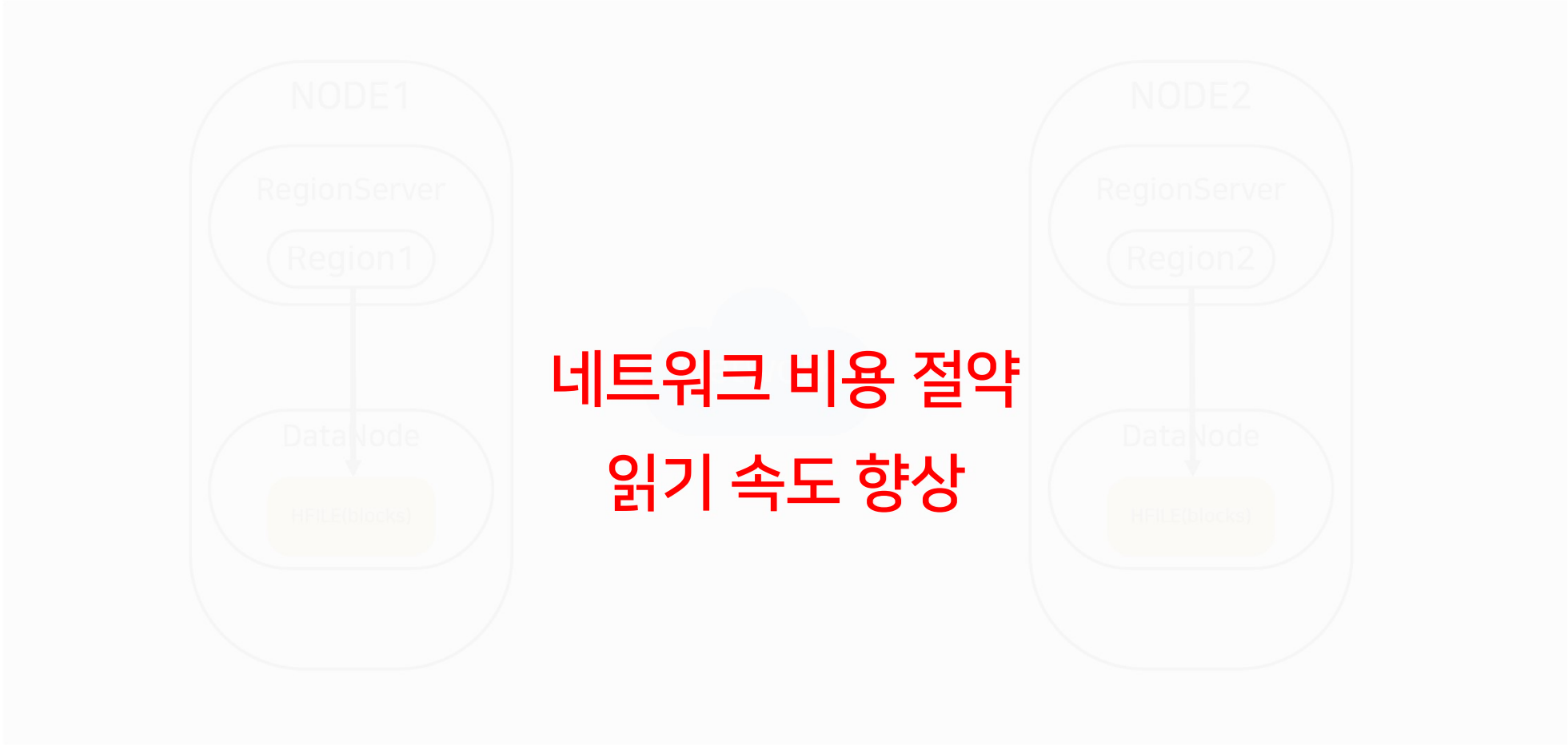
Region – HFILES가 같은 장비에 있다면?



Region – HFILES가 같은 장비에 있다면?



Region – HFILES가 같은 장비에 있다면?

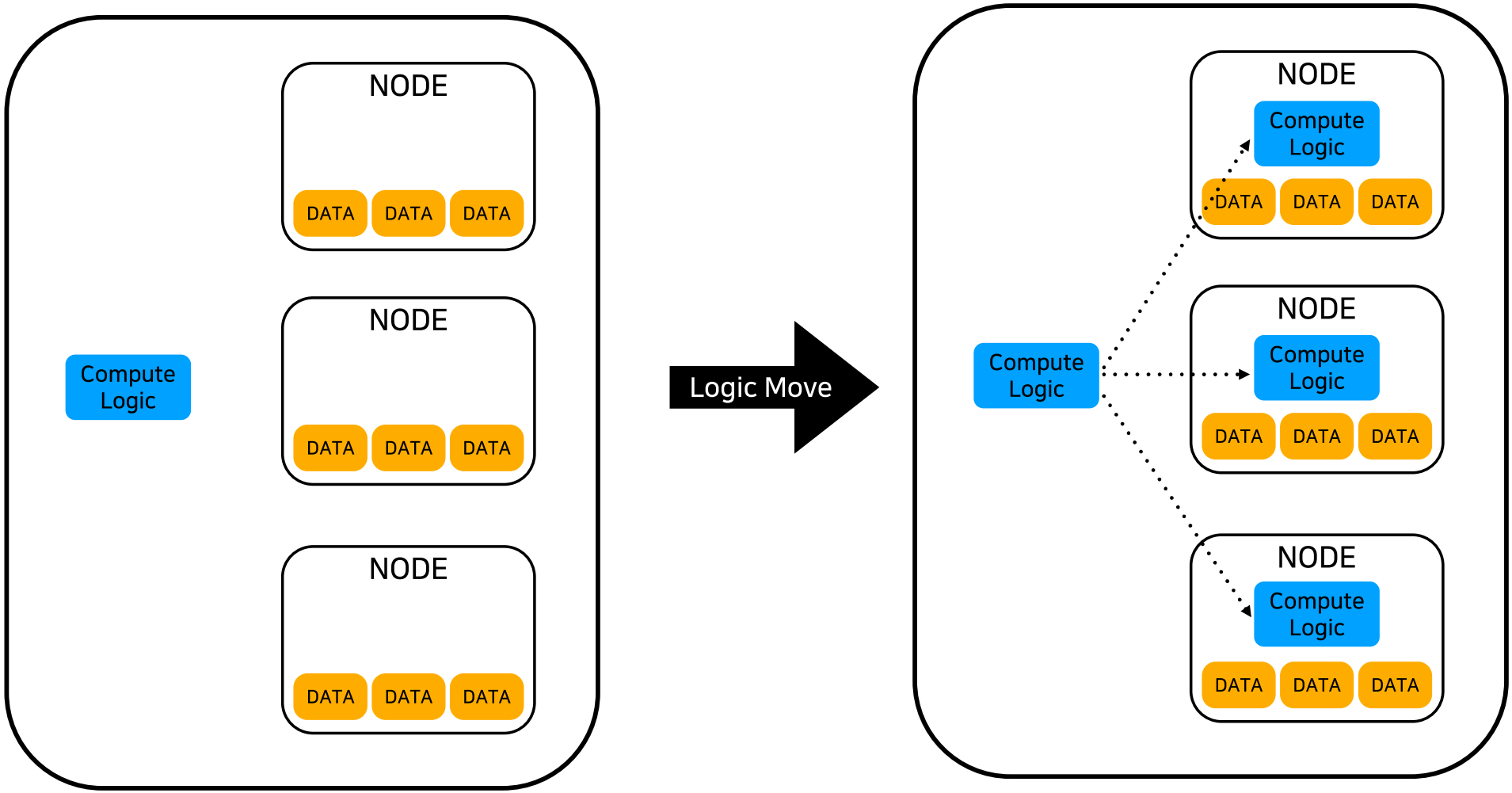


네트워크 비용 절약

읽기 속도 향상

HBase Locality

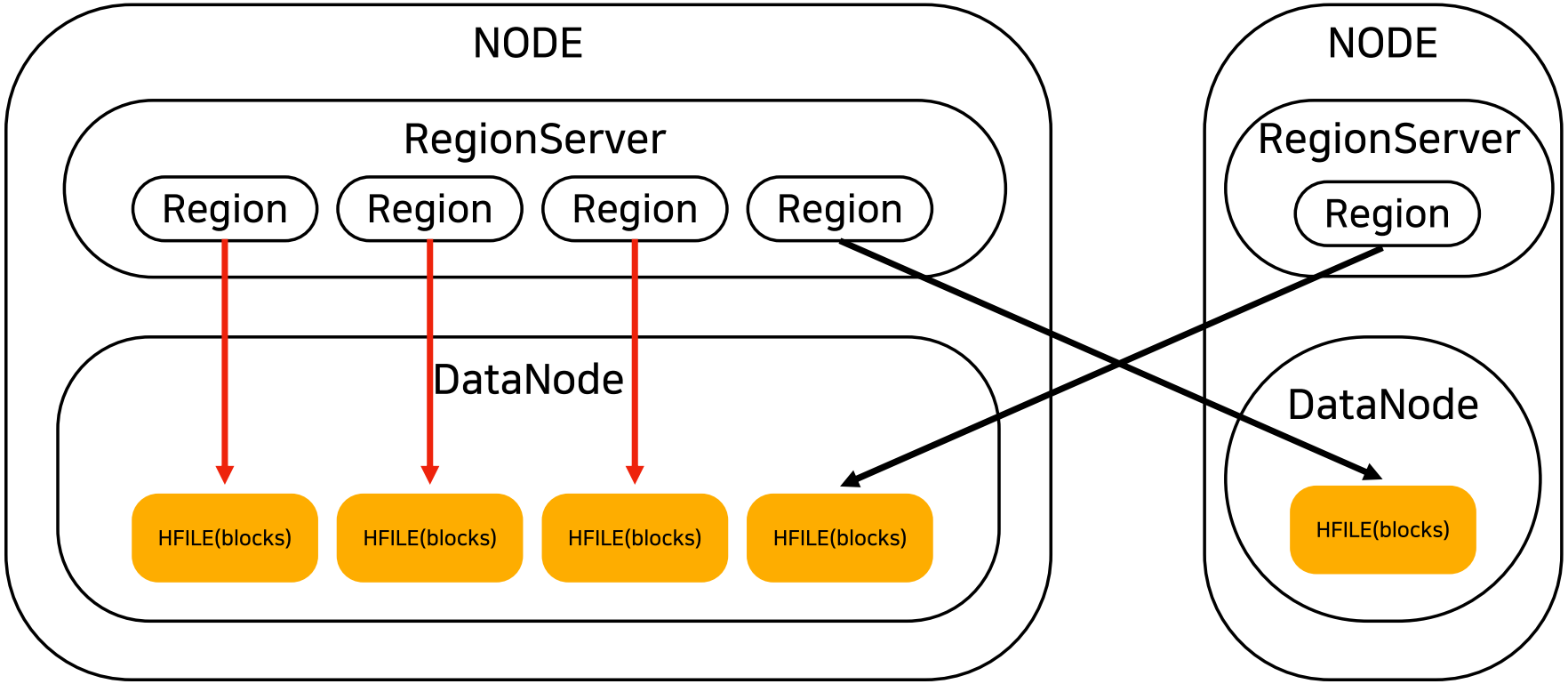
Data Locality



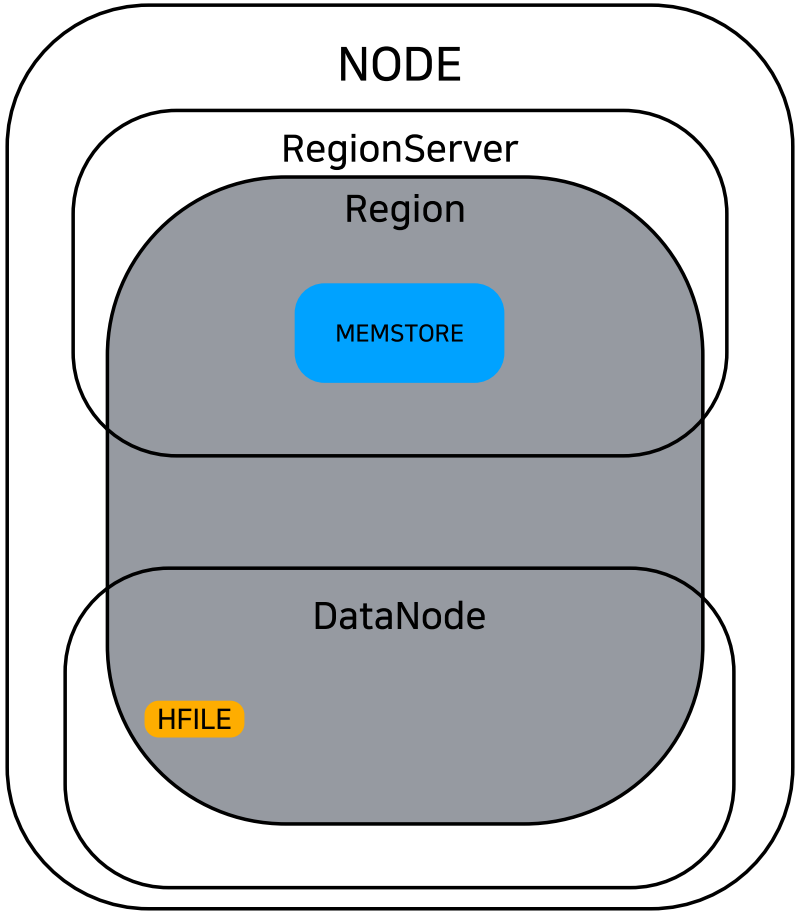
HBase Locality

DEVIEW 2021

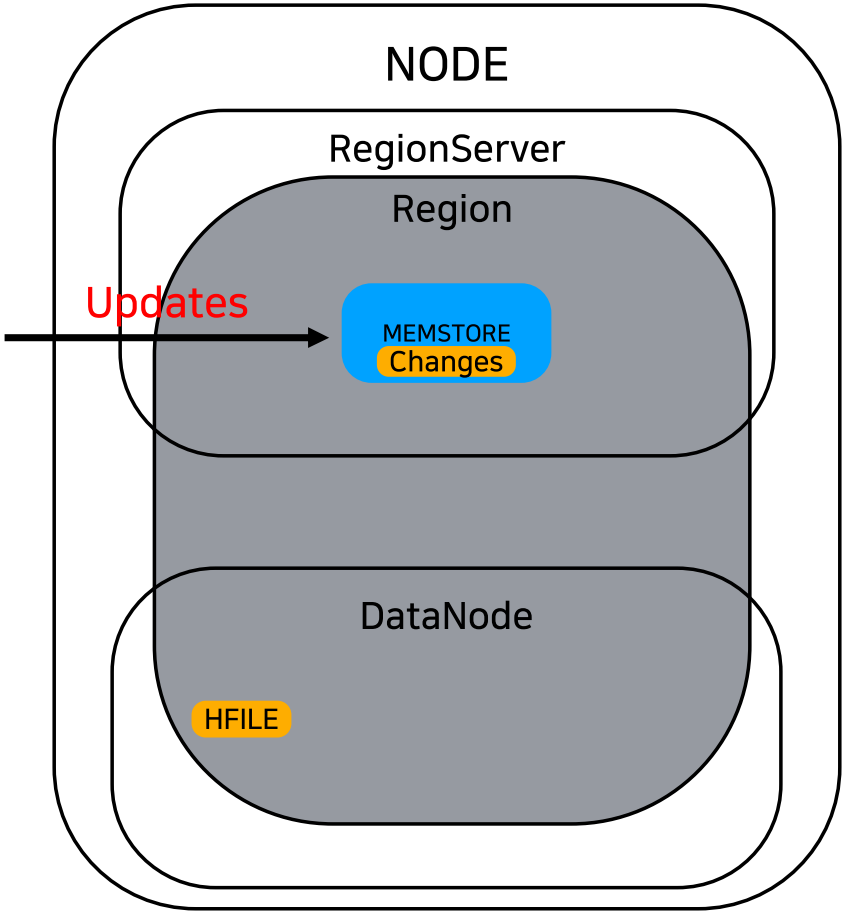
Region의 물리 데이터가 Local DataNode에 있는 것



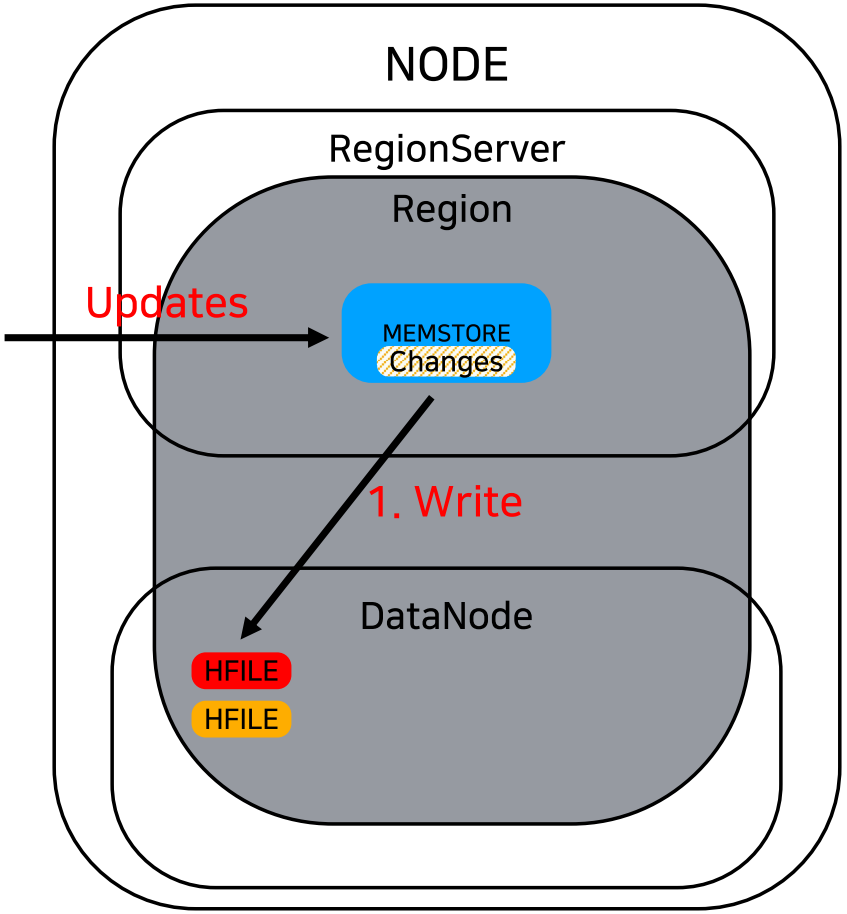
HBase에서 Locality를 올리는 방법



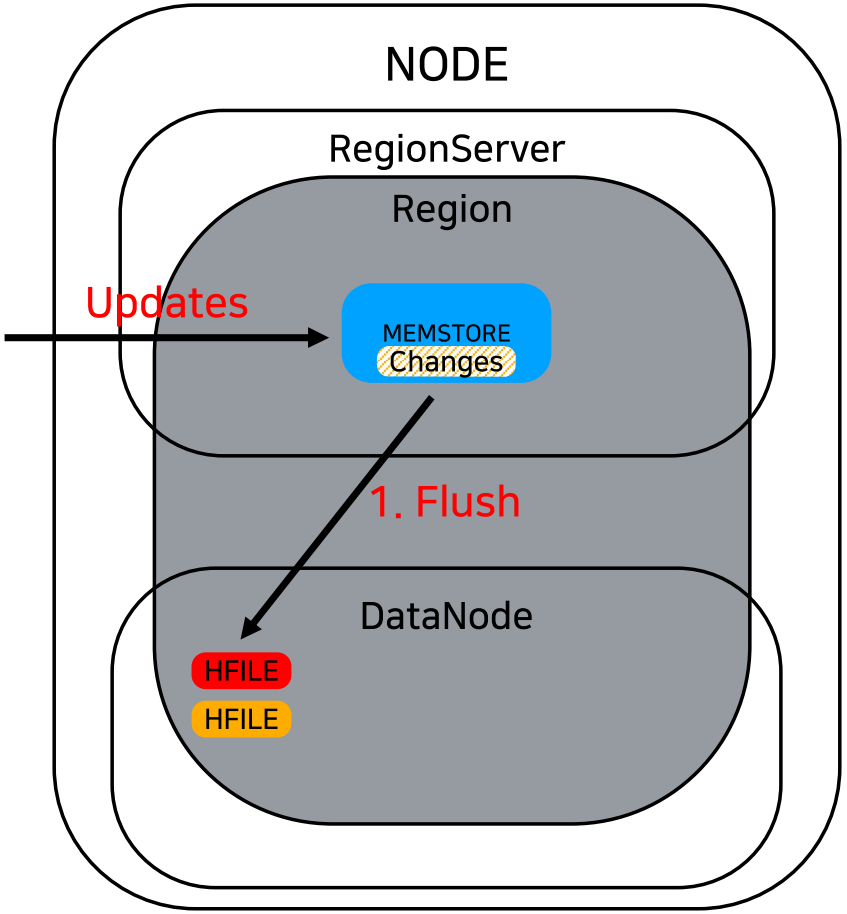
HBase에서 Locality를 올리는 방법



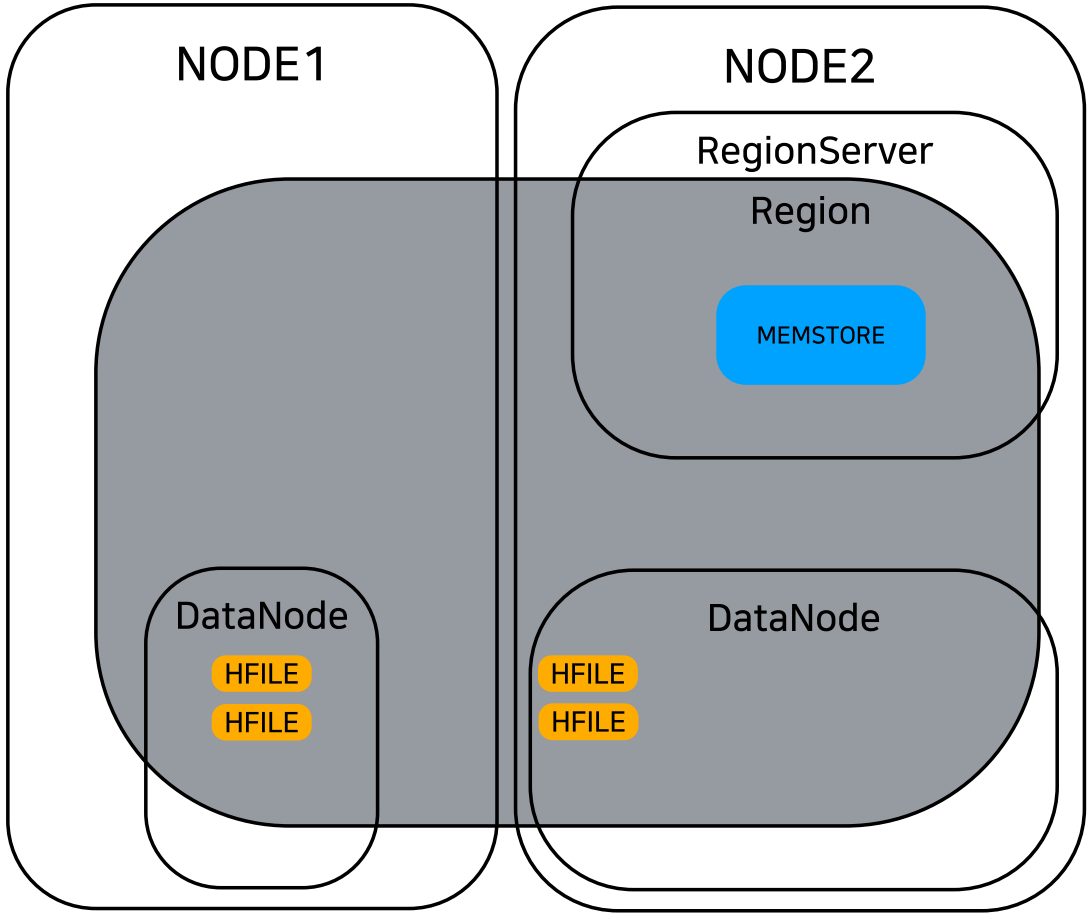
HBase에서 Locality를 올리는 방법



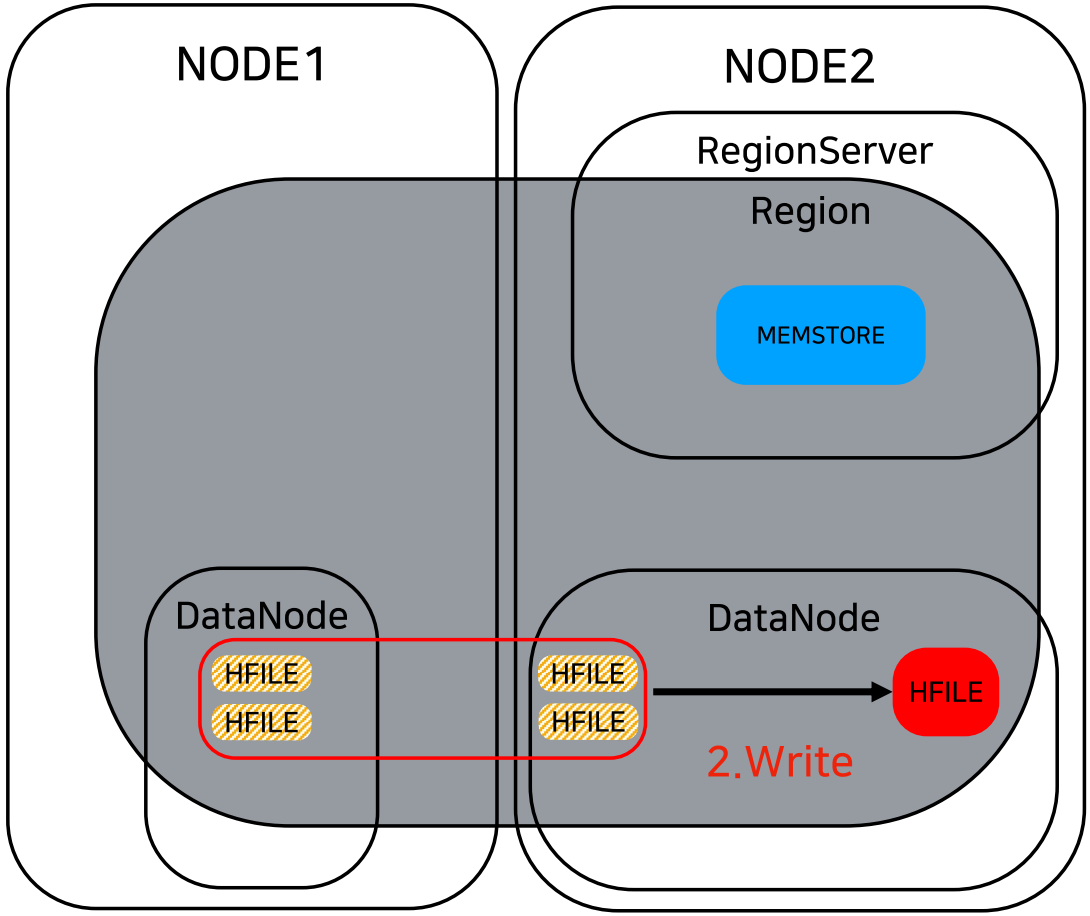
HBase에서 Locality를 올리는 방법



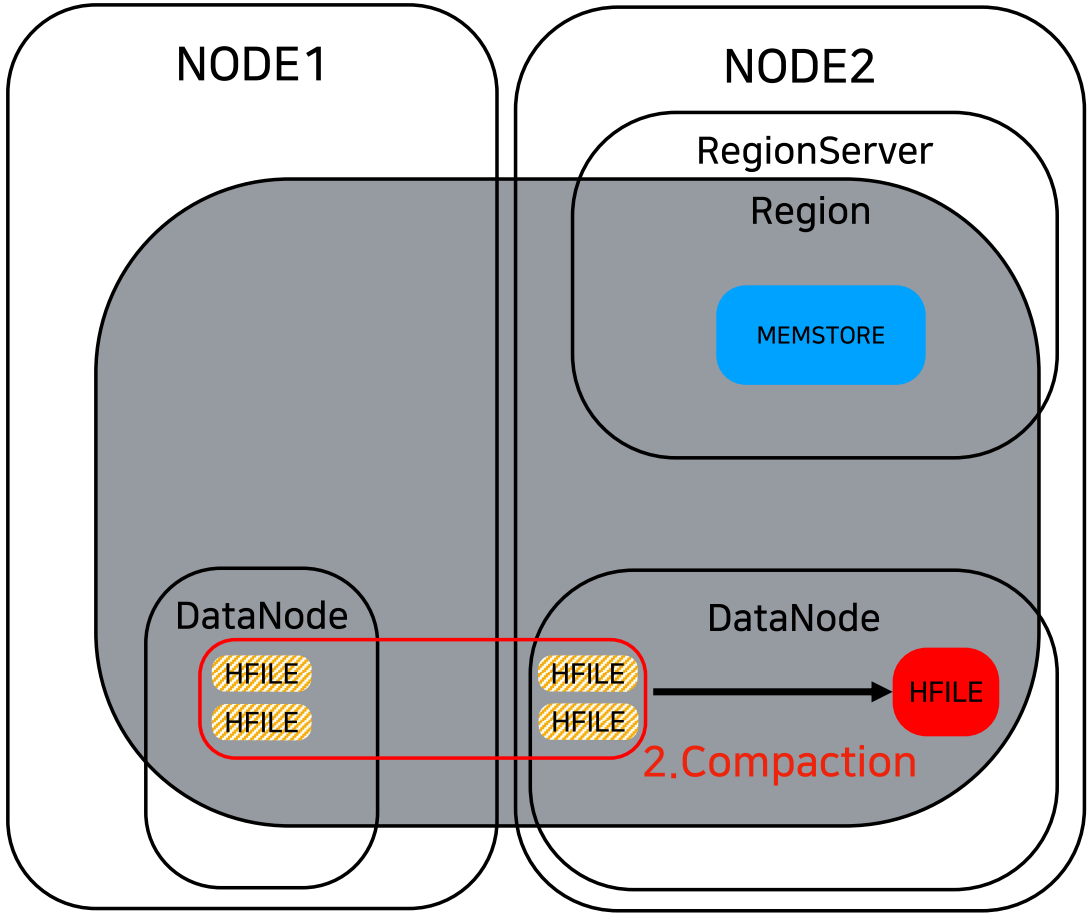
HBase에서 Locality를 올리는 방법



HBase에서 Locality를 올리는 방법



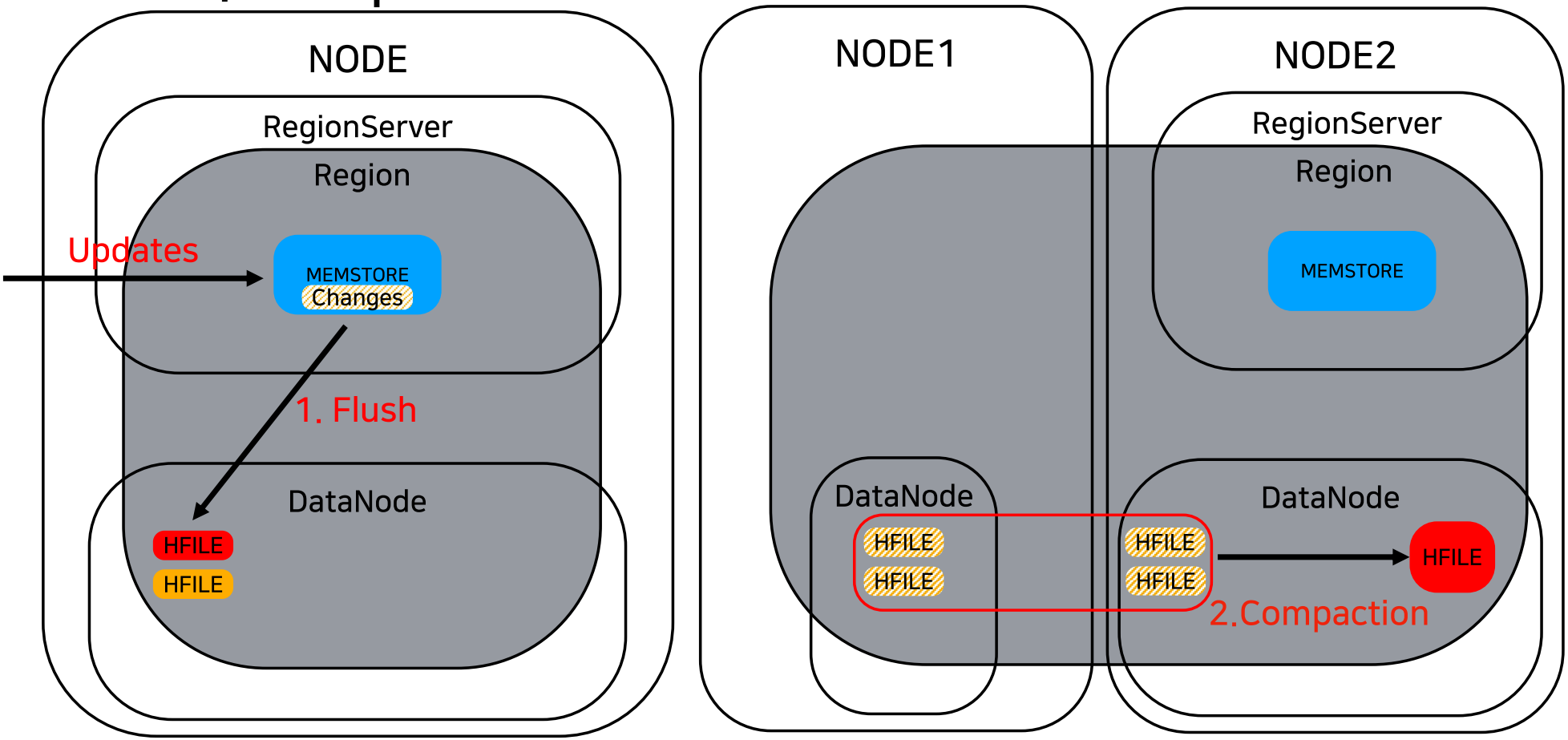
HBase에서 Locality를 올리는 방법



HBase에서 Locality를 올리는 방법

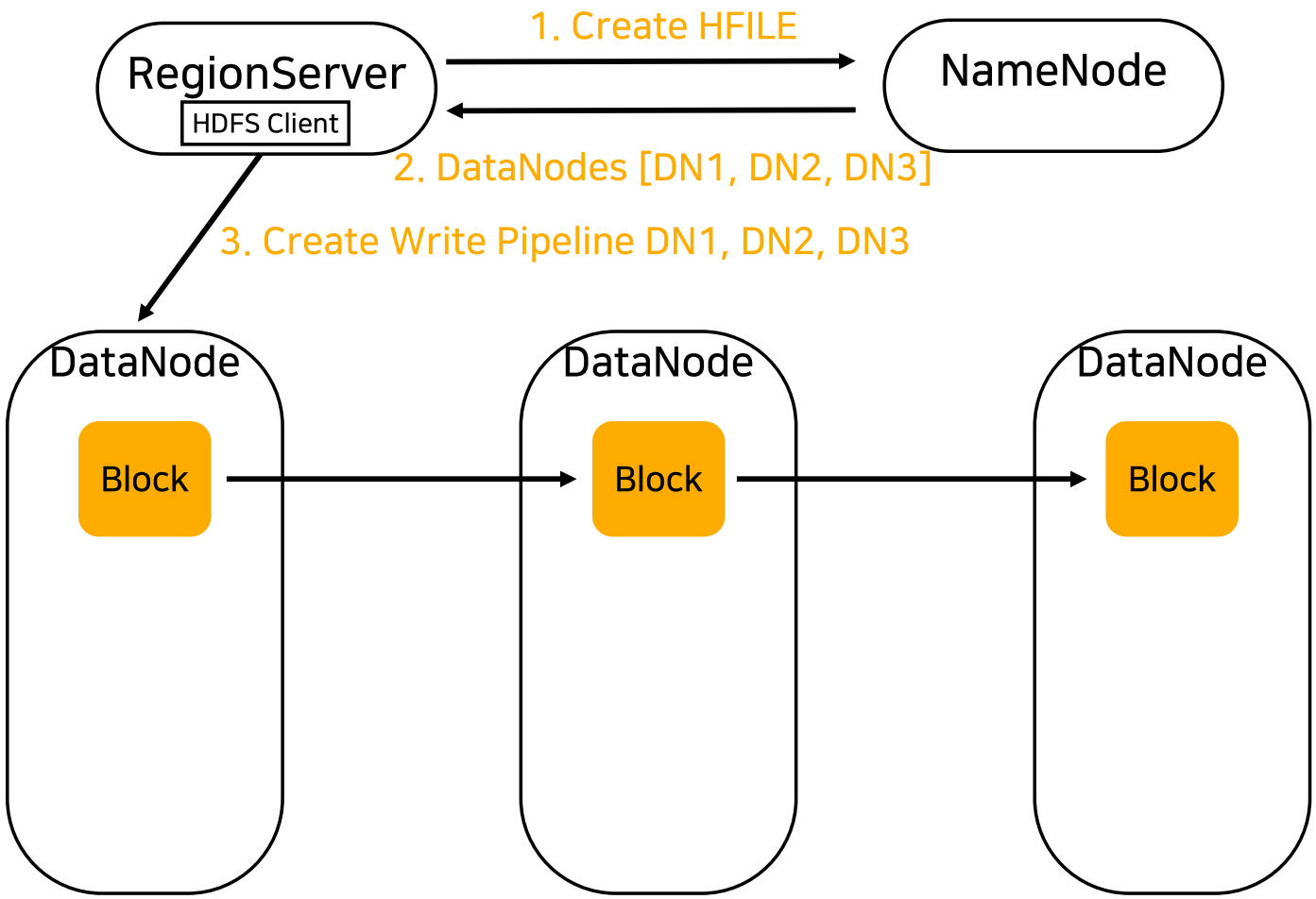
DEVIEW 2021

Flush와 Compaction

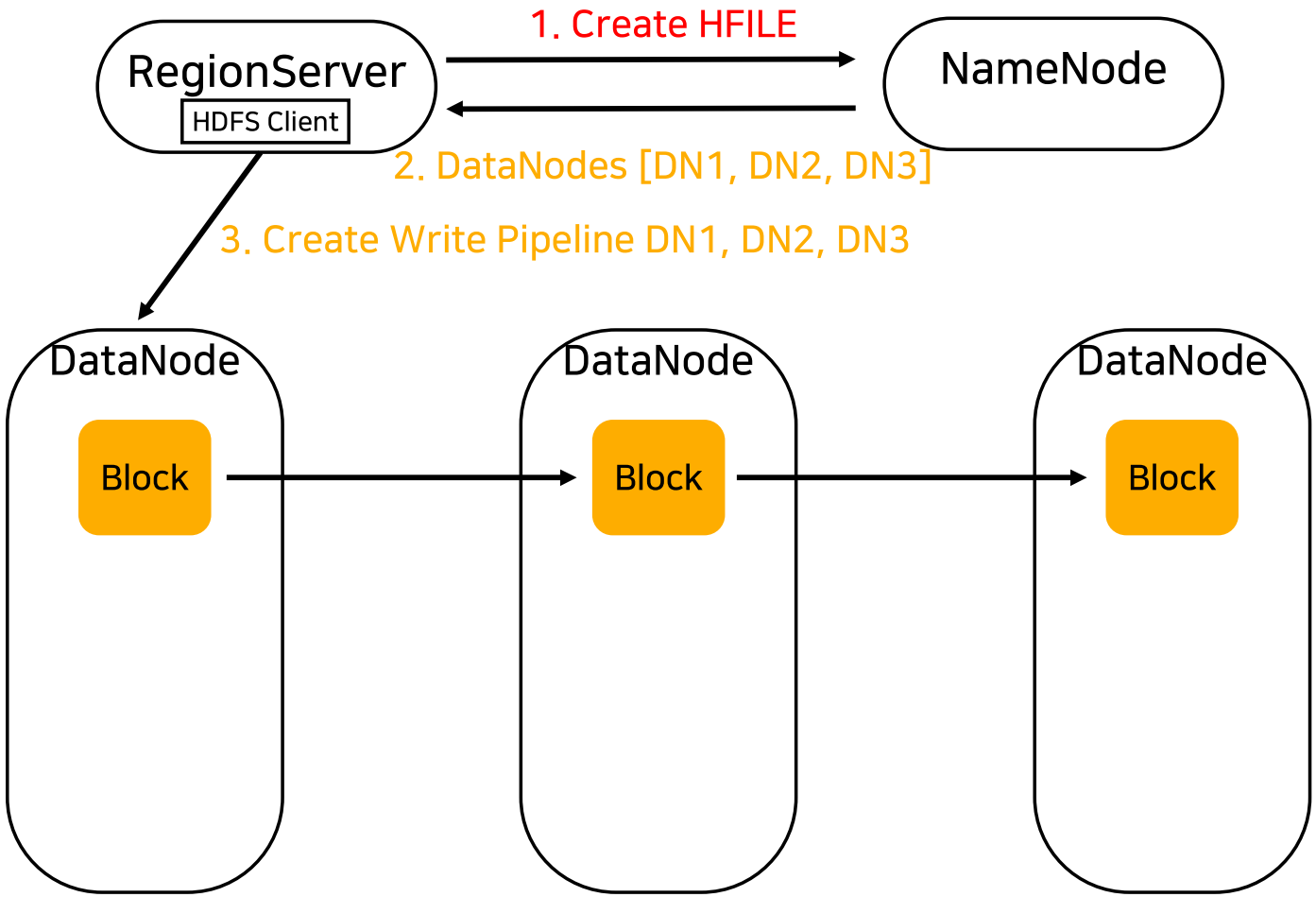


HBase에서 Locality를 올리는 방법

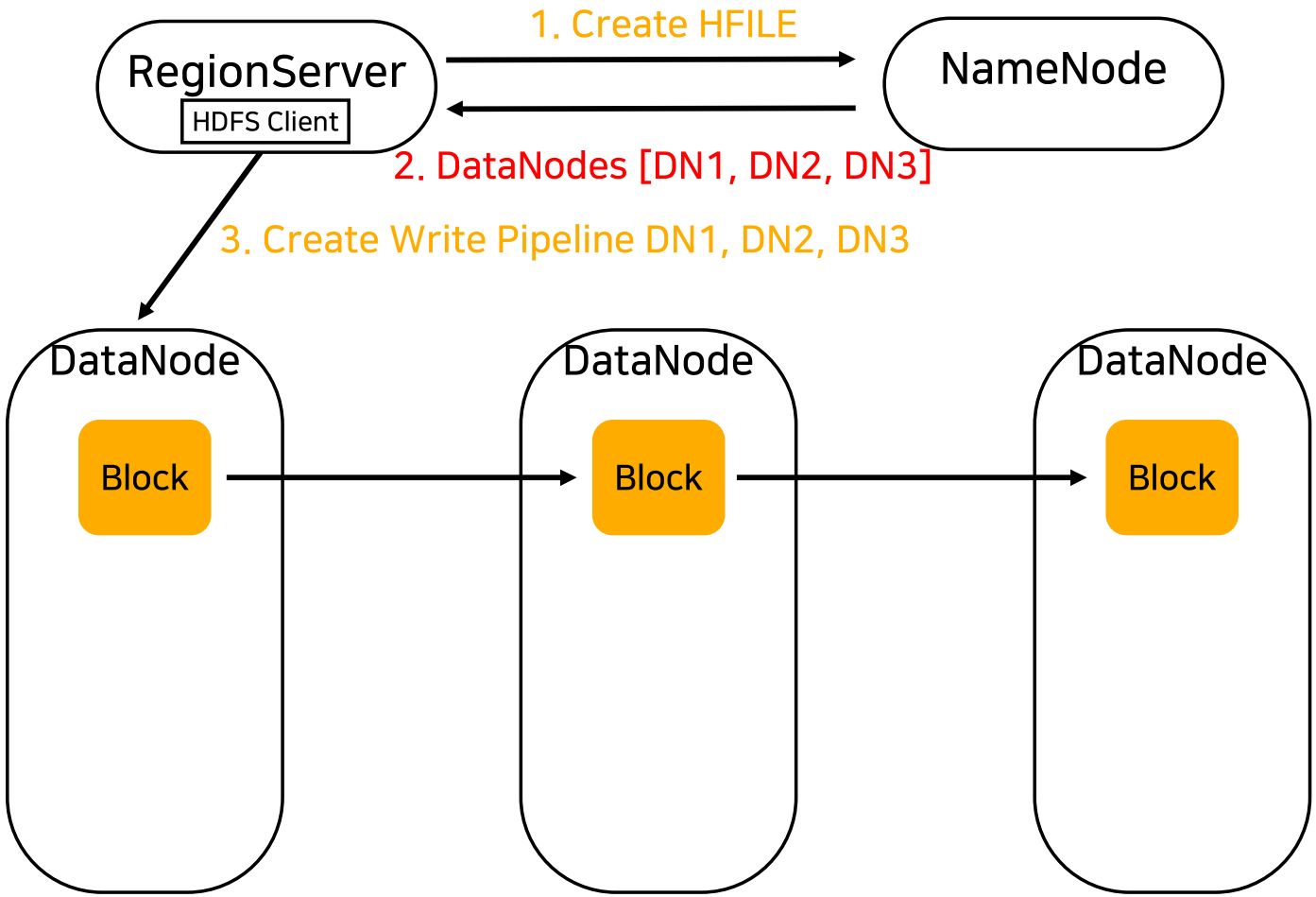
HDFS Write통해서 Locality를 어떻게?



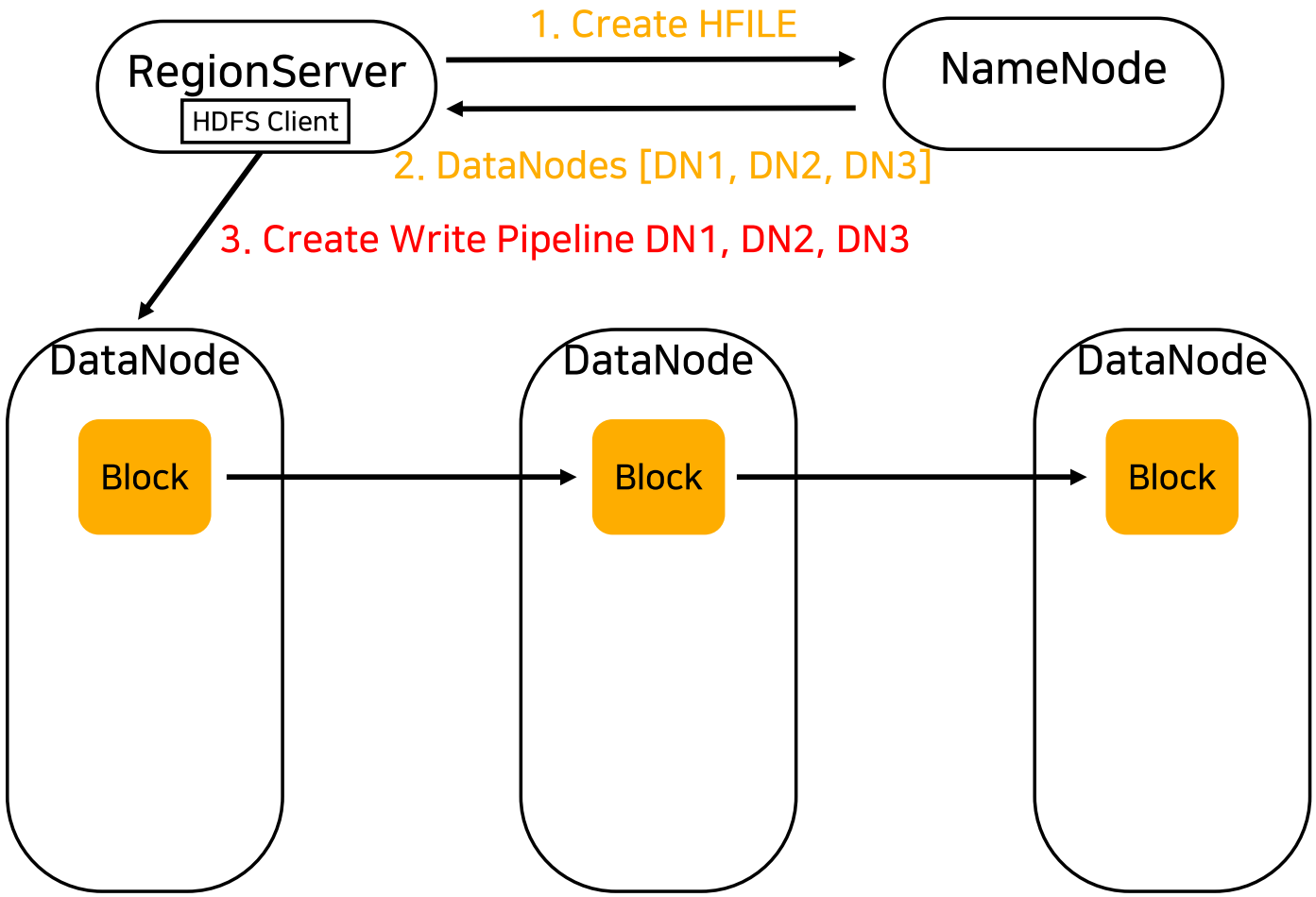
HDFS Write통해서 Locality를 어떻게?



HDFS Write통해서 Locality를 어떻게?



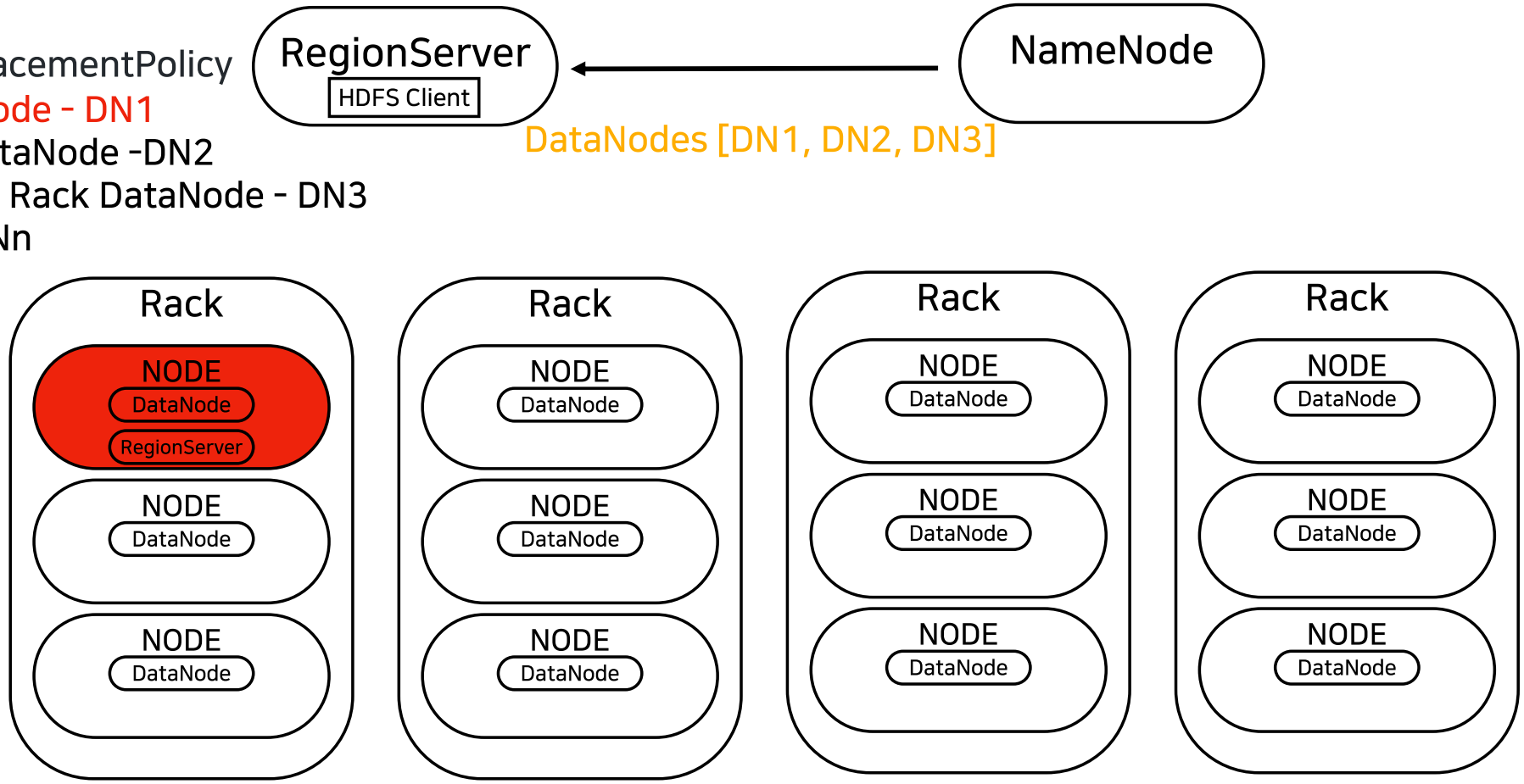
HDFS Write통해서 Locality를 어떻게?



정답은 DataNode 선정 방법!

Default BlockPlacementPolicy

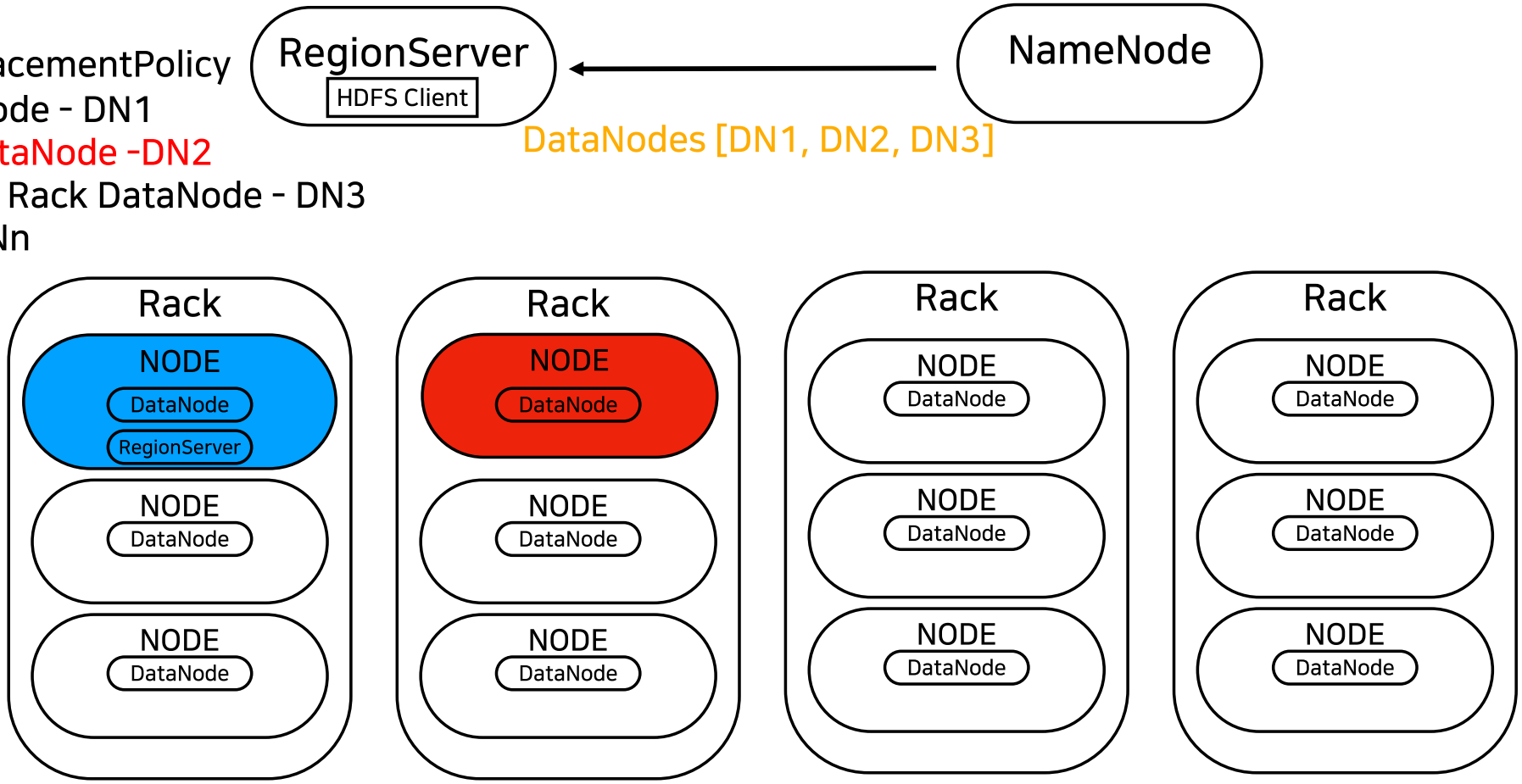
1. Local DataNode - DN1
2. 다른 Rack DataNode -DN2
3. DN2와 동일한 Rack DataNode - DN3
4. Random - DNn



정답은 DataNode 선정 방법!

Default BlockPlacementPolicy

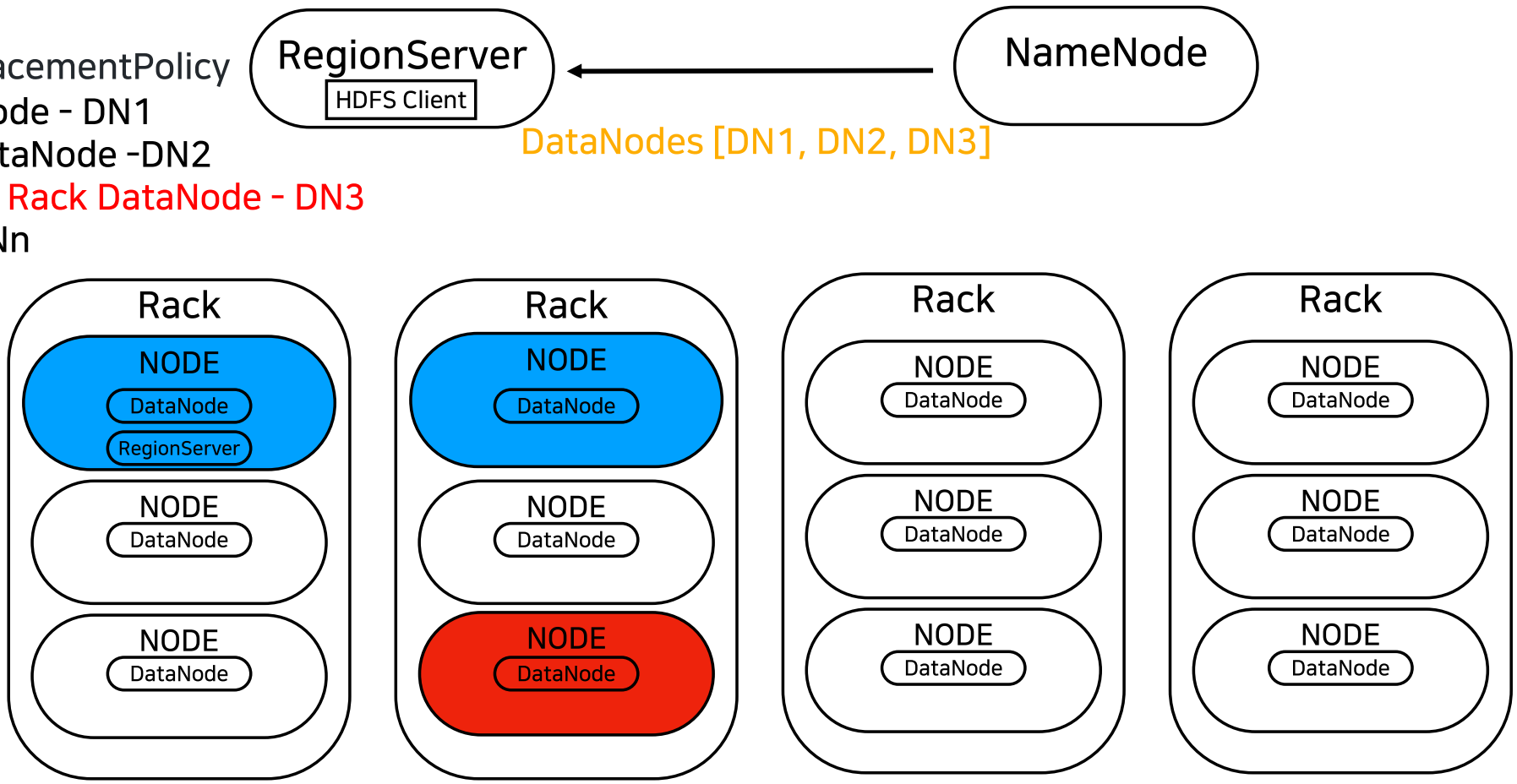
1. Local DataNode - DN1
2. 다른 Rack DataNode -DN2
3. DN2와 동일한 Rack DataNode - DN3
4. Random - DNn



정답은 DataNode 선정 방법!

Default BlockPlacementPolicy

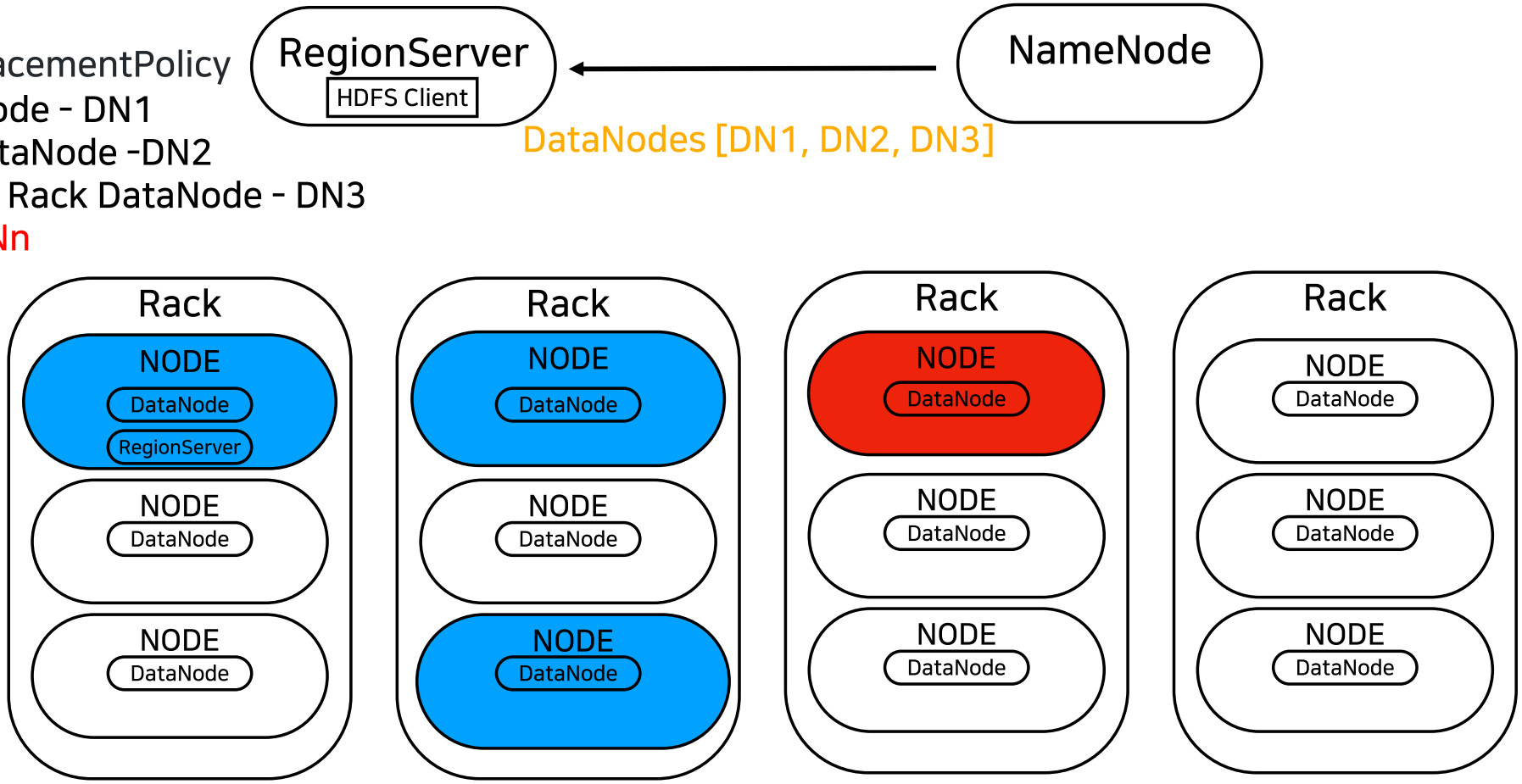
1. Local DataNode - DN1
2. 다른 Rack DataNode -DN2
3. DN2와 동일한 Rack DataNode - DN3
4. Random - DNn



정답은 DataNode 선정 방법!

Default BlockPlacementPolicy

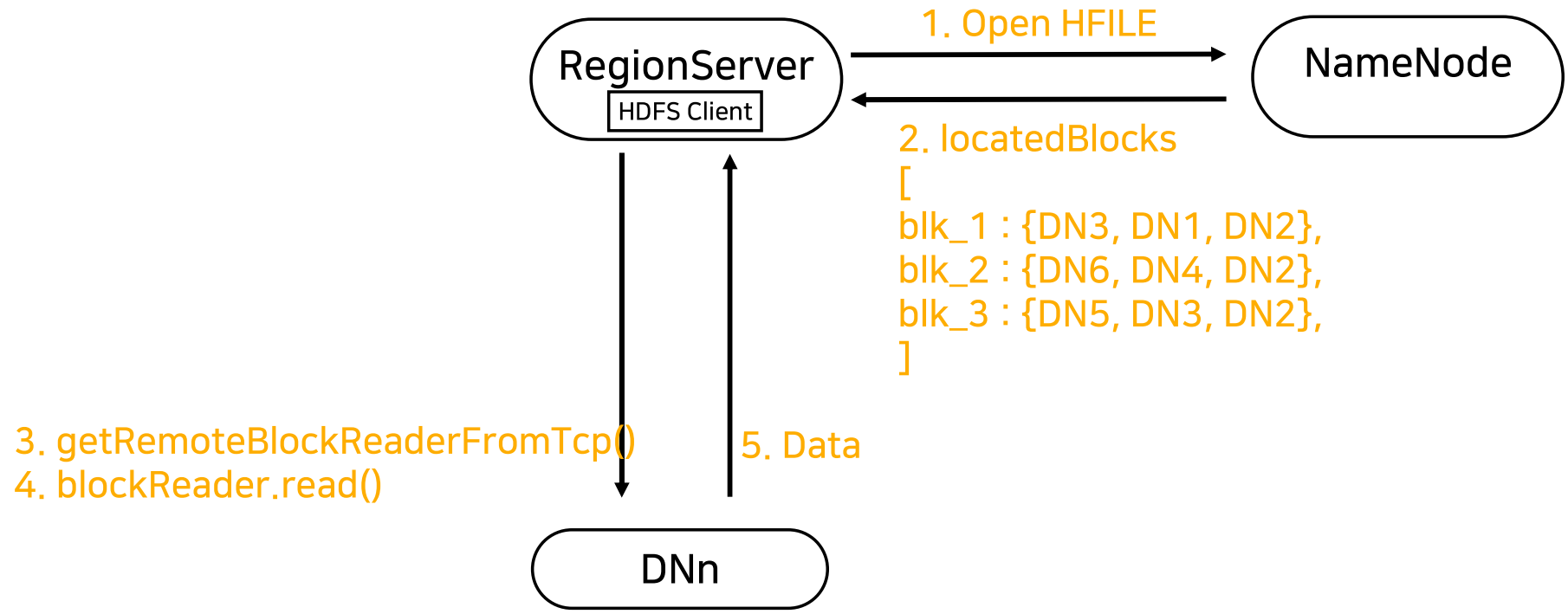
1. Local DataNode - DN1
2. 다른 Rack DataNode -DN2
3. DN2와 동일한 Rack DataNode - DN3
4. Random - DNn



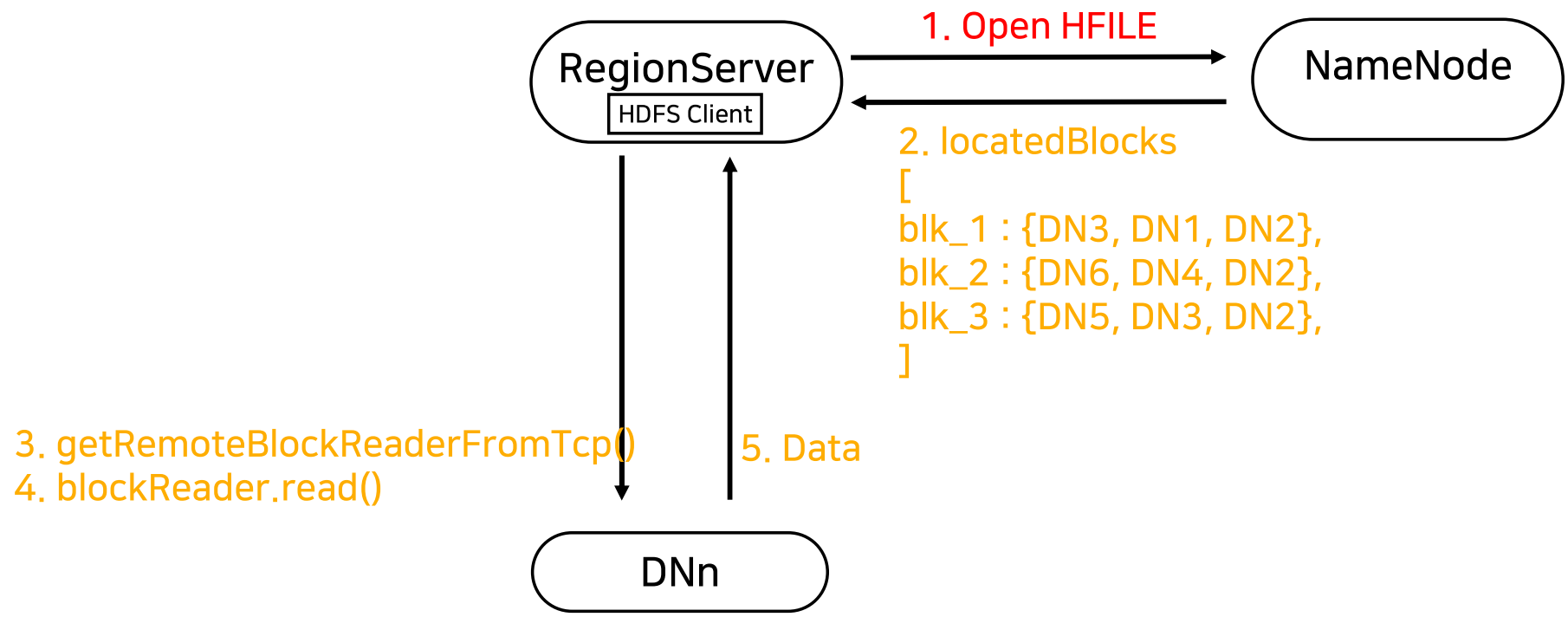
DEVIEW 2021

HBase Locality로 읽기 성능 높이는 방법

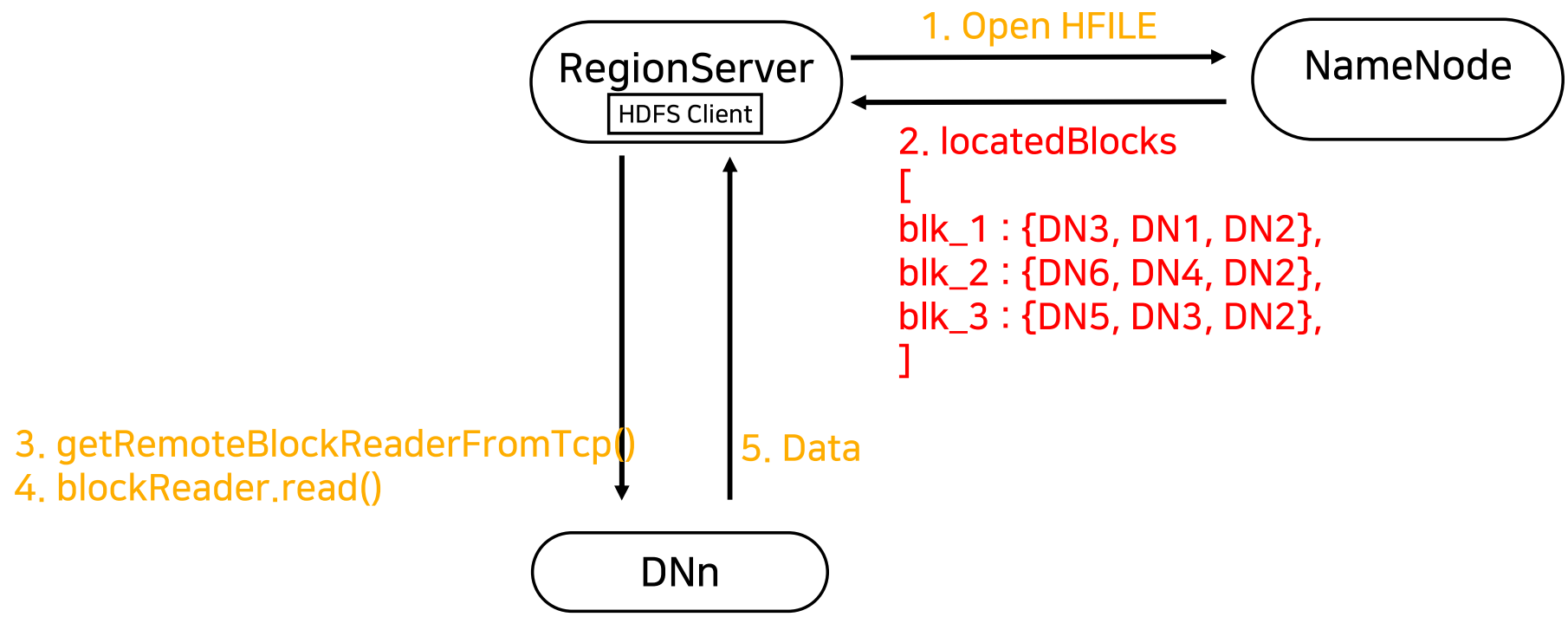
1. LocatedBlocks 순서



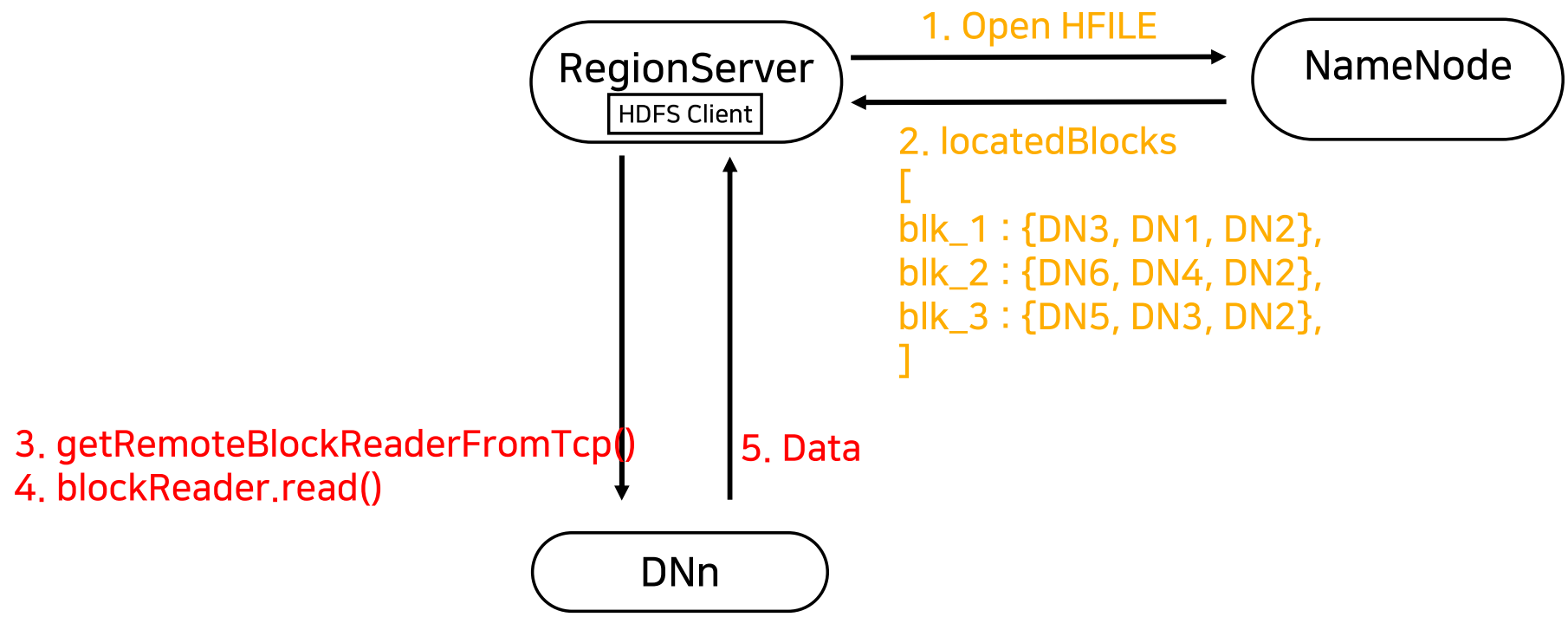
1. LocatedBlocks 순서



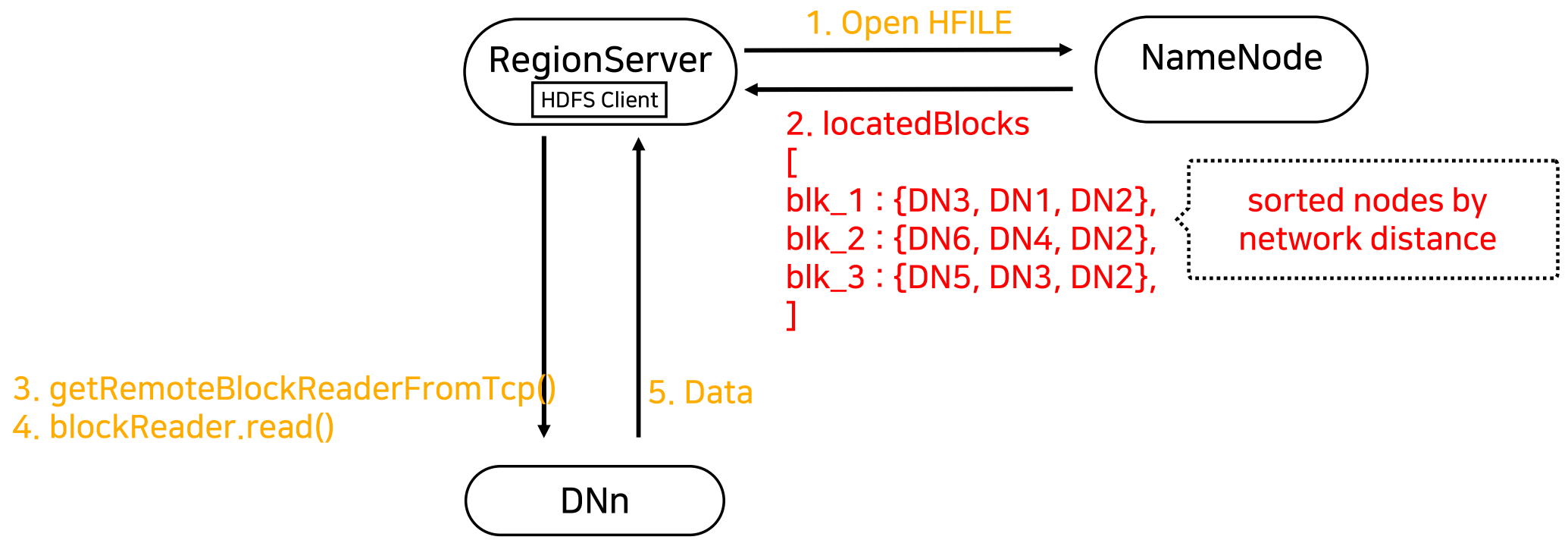
1. LocatedBlocks 순서



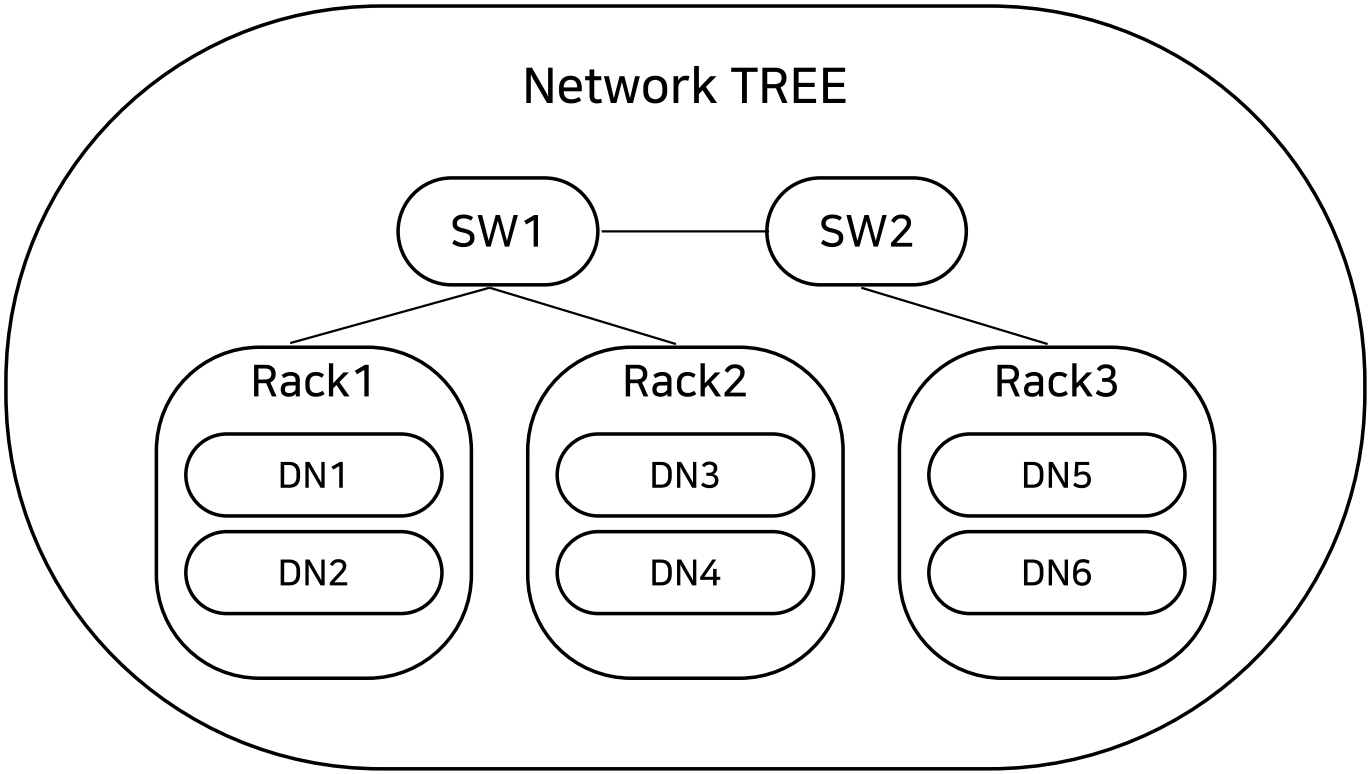
1. LocatedBlocks 순서



1. LocatedBlocks 순서

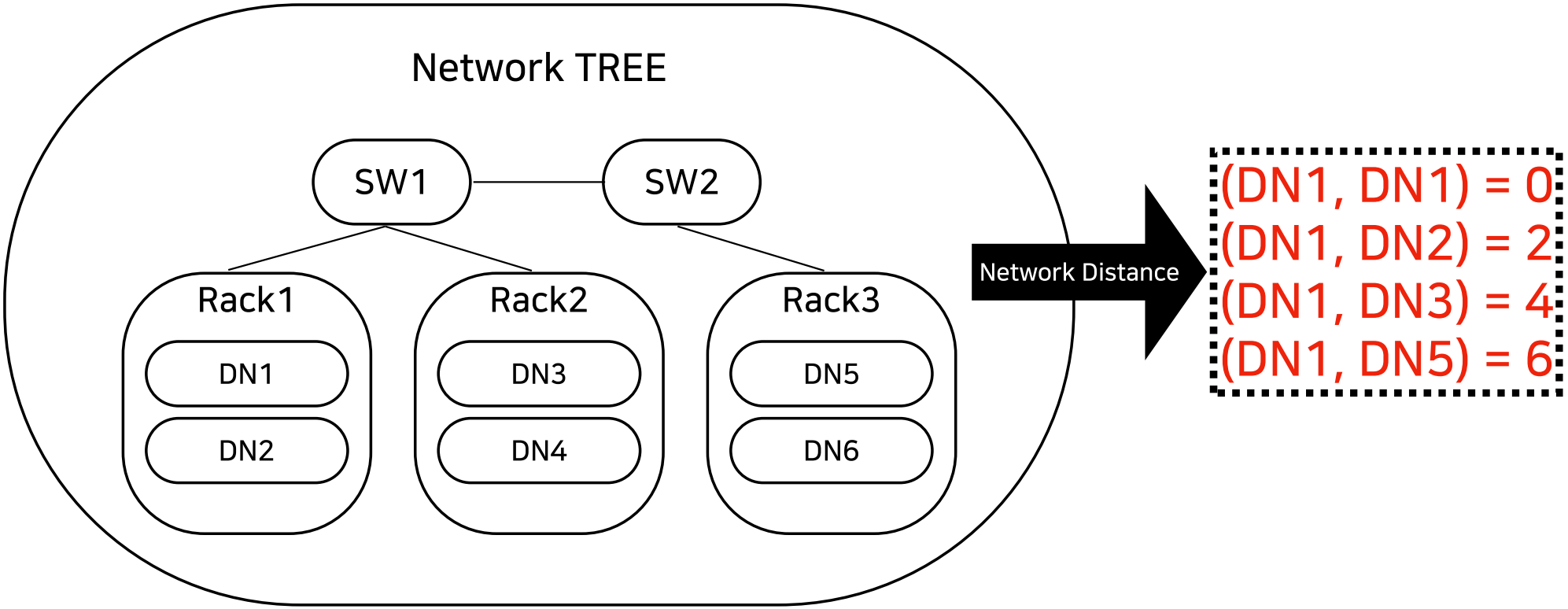


1. LocatedBlocks 순서

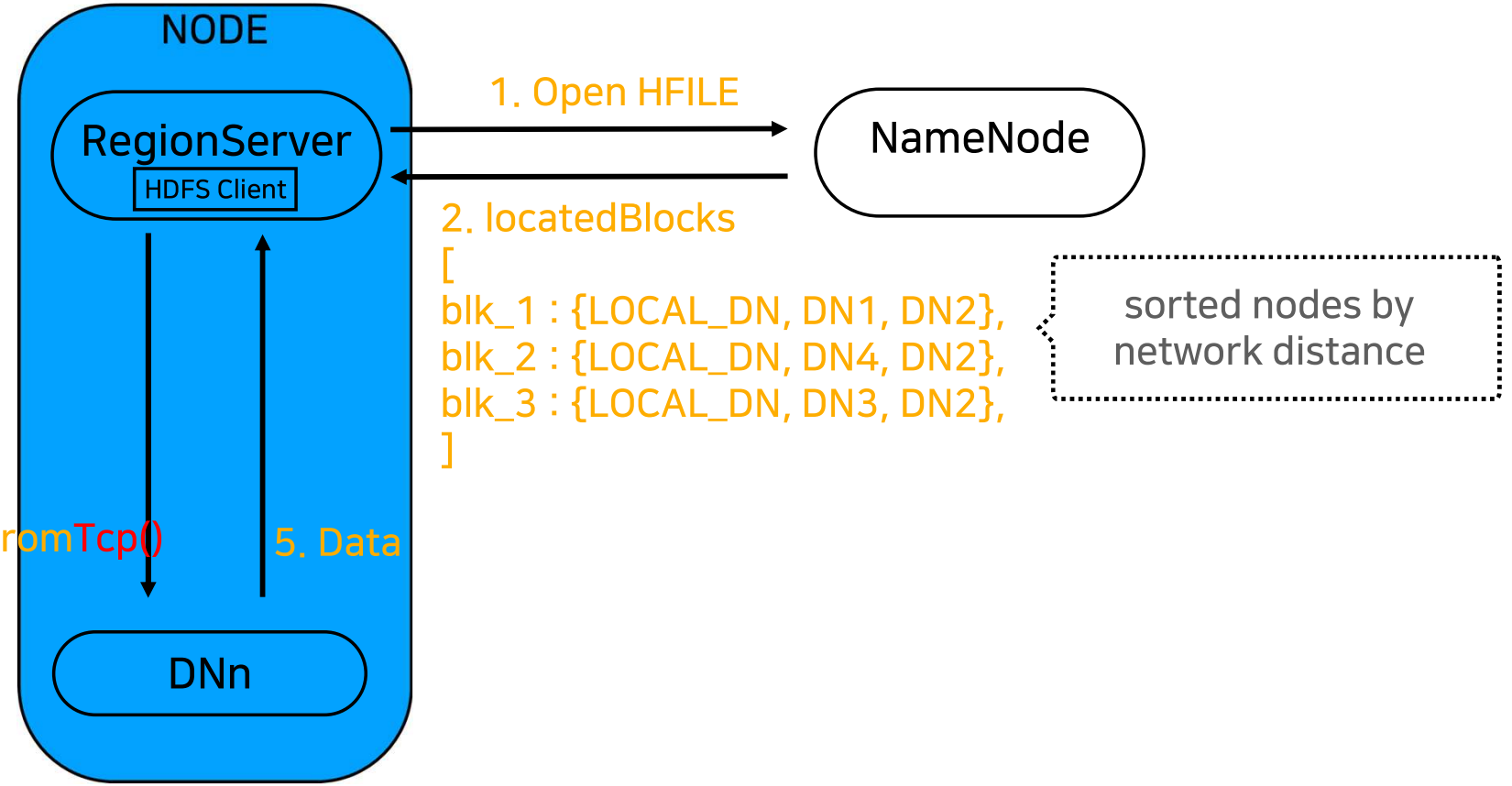


RACK INFO
DN1 /SW1/RACK1
DN2 /SW1/RACK1
DN3 /SW1/RACK2
DN4 /SW1/RACK2
DN5 /SW2/RACK3
DN6 /SW2/RACK3

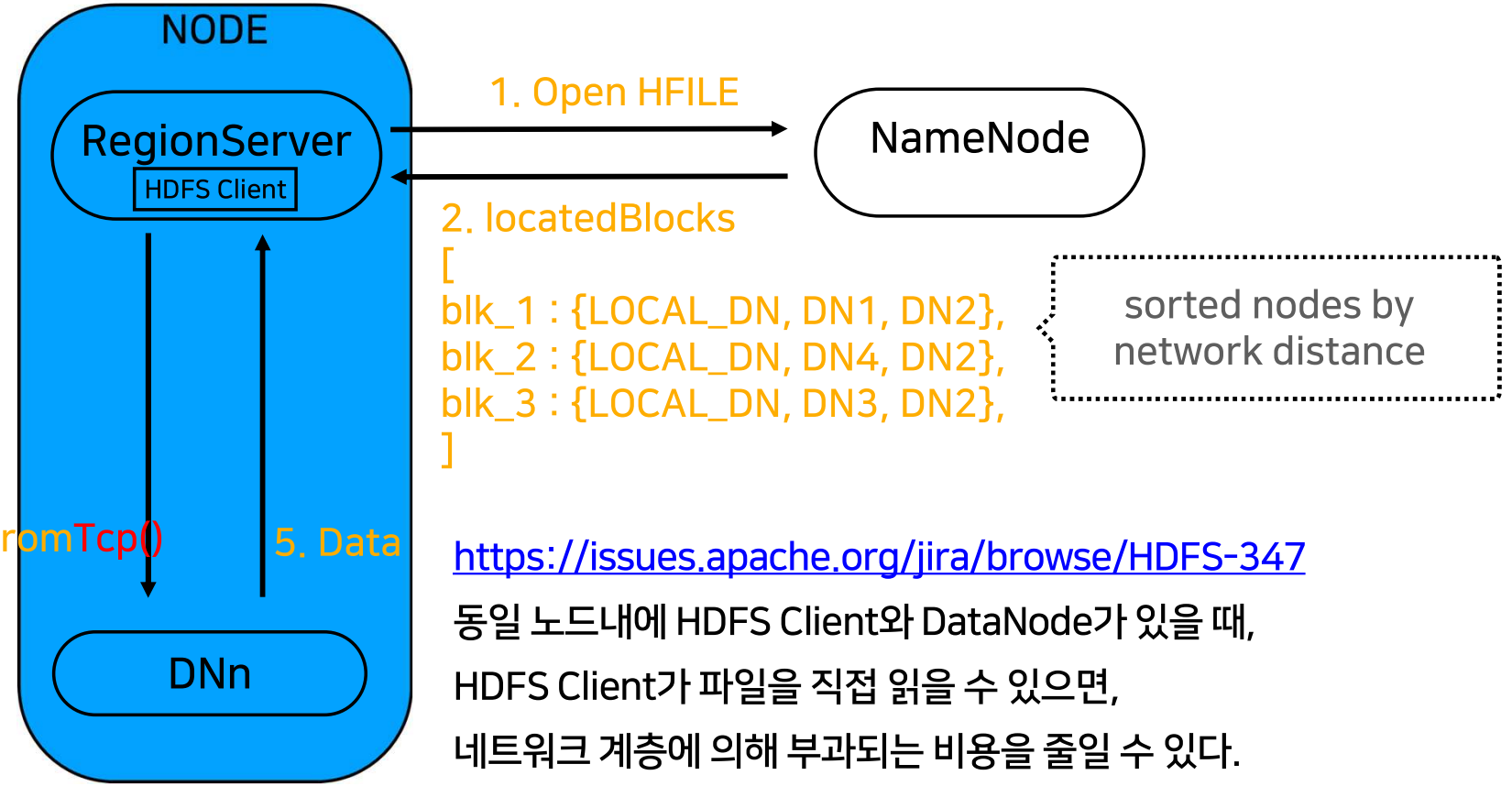
1. LocatedBlocks 순서



1. LocatedBlocks 순서



TCP 통신은 왜?



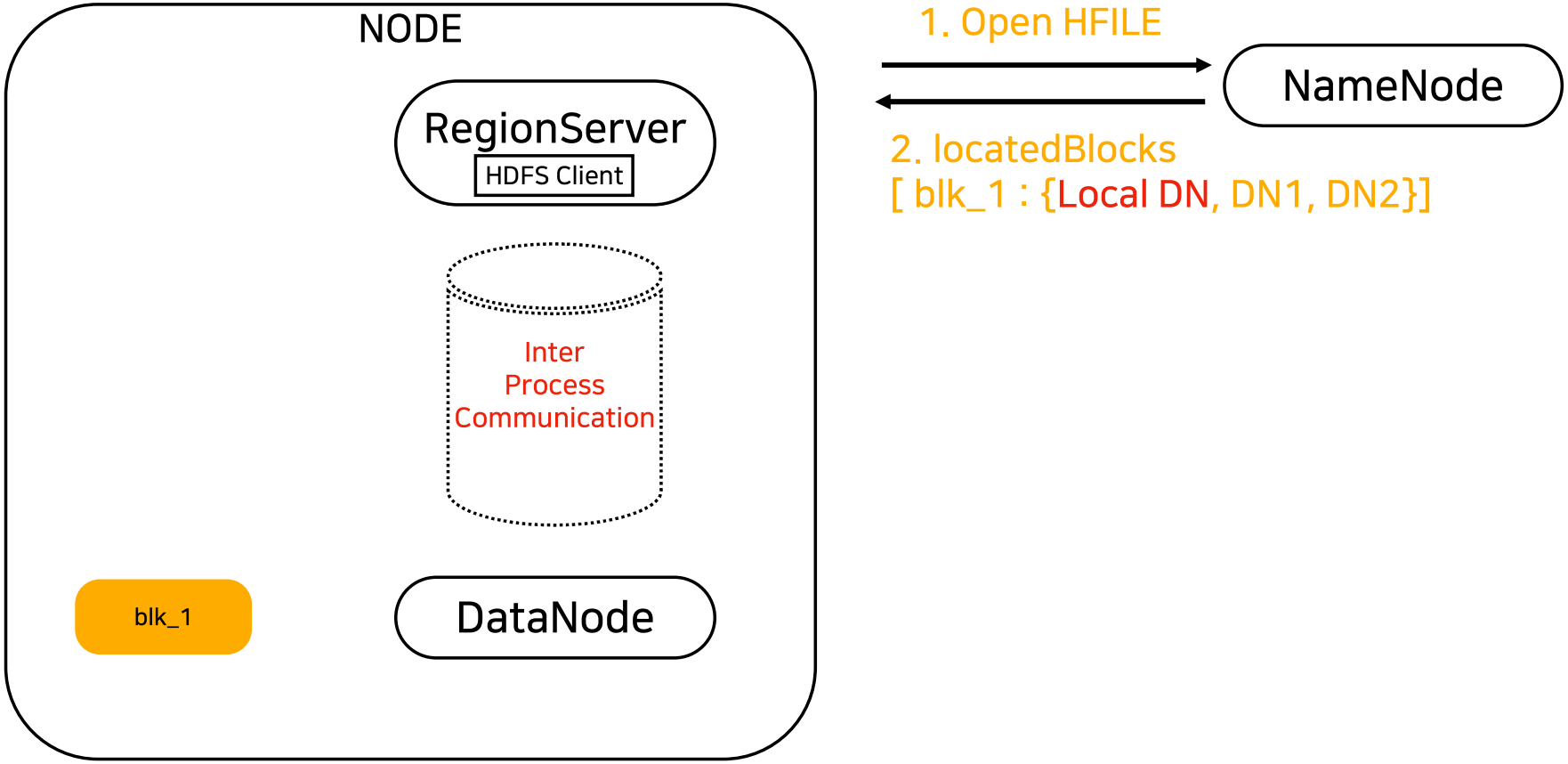
https://issues.apache.org/jira/browse/HDFS-347

동일 노드내에 HDFS Client와 DataNode가 있을 때,

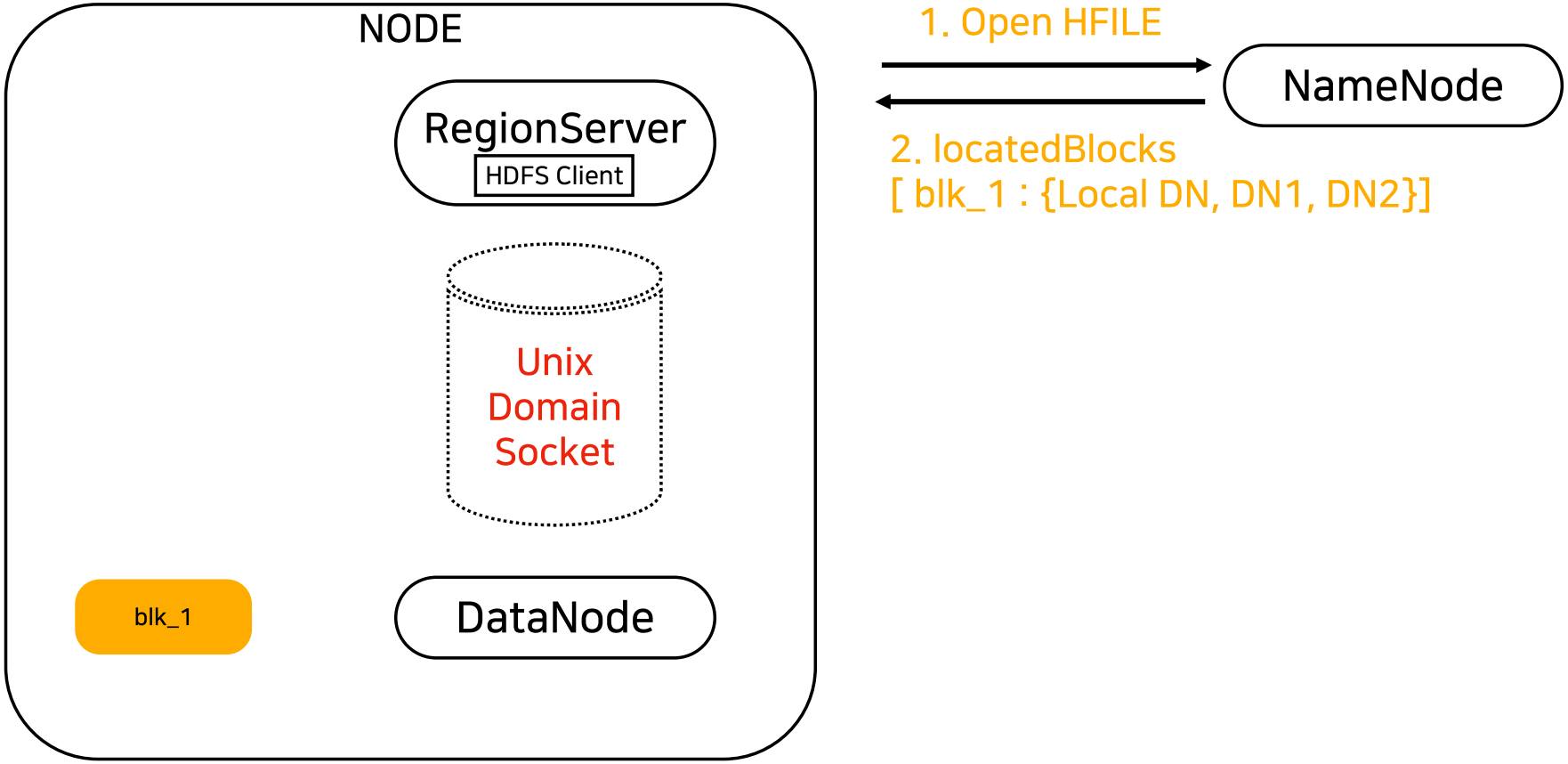
HDFS Client가 파일을 직접 읽을 수 있으면,

네트워크 계층에 의해 부과되는 비용을 줄일 수 있다.

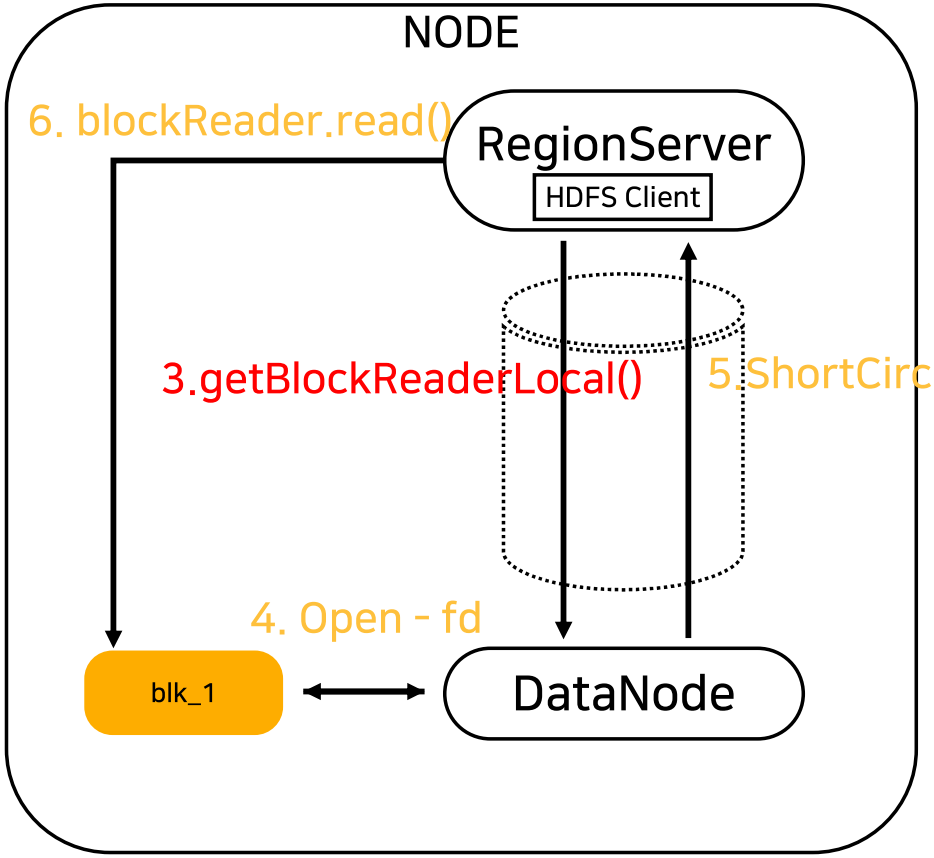
2. HDFS Short-Circuit Local Reads

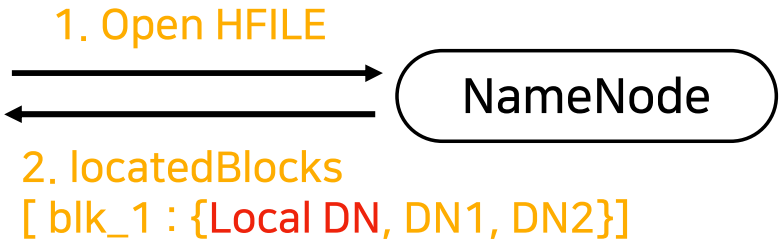


2. HDFS Short-Circuit Local Reads



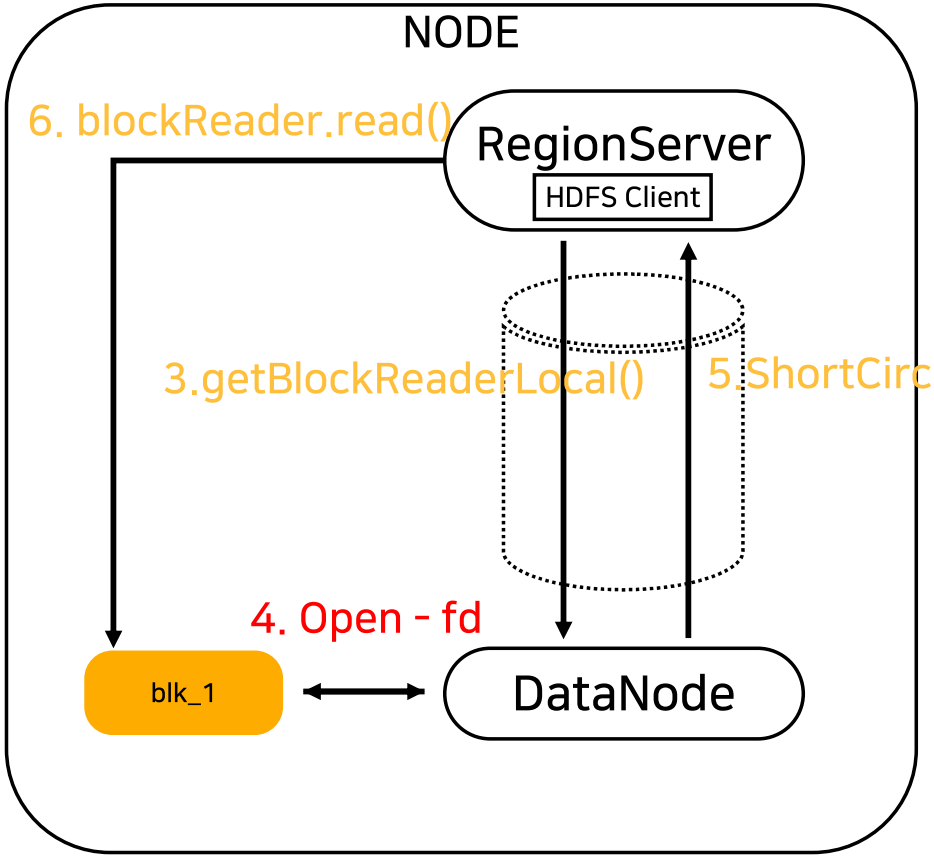
2. HDFS Short-Circuit Local Reads

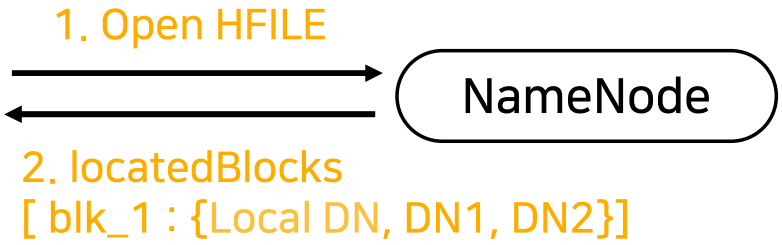





```
public class ShortCircuitReplica{
  private final FileInputStream dataStream
  private final Slot slot
  ...
}

public FileInputStream(FileDescriptor fd) {
}
```

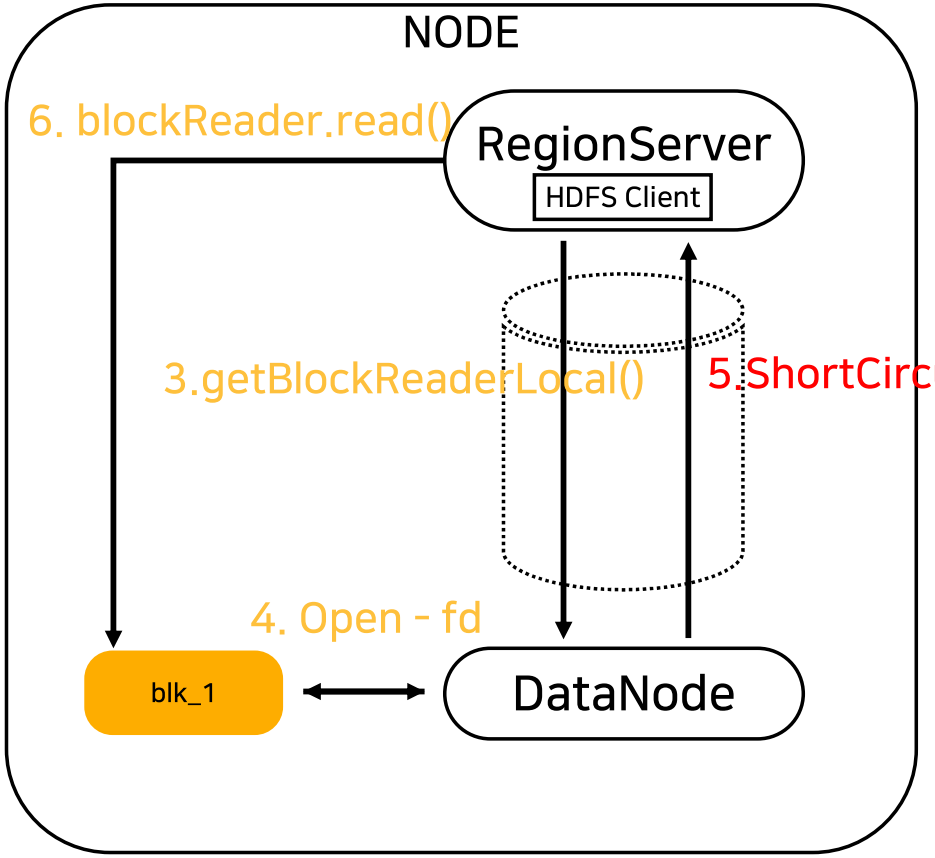
2. HDFS Short-Circuit Local Reads

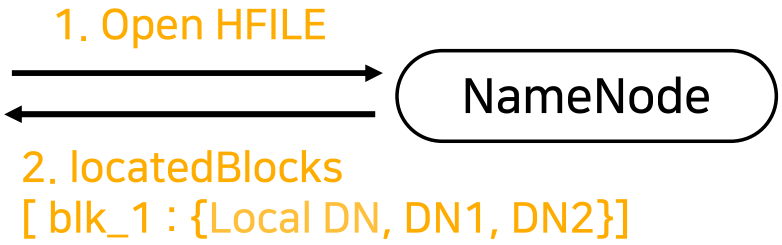





```
public class ShortCircuitReplica{
  private final FileInputStream dataStream
  private final Slot slot
  ...
}

public FileInputStream(FileDescriptor fd) {
}
```

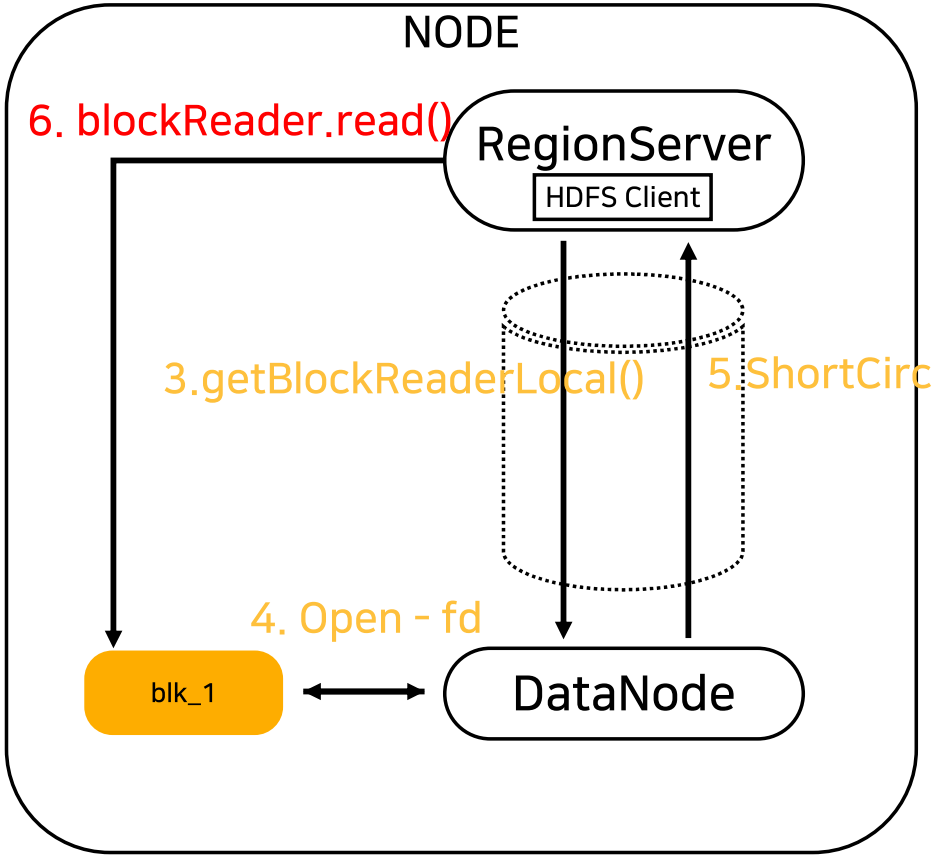
2. HDFS Short-Circuit Local Reads

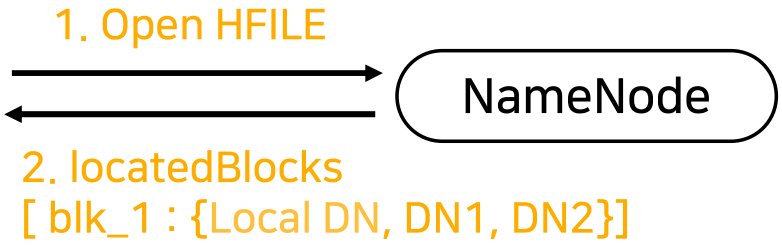





```
public class ShortCircuitReplica{
  private final FileInputStream dataStream
  private final Slot slot
  ...
}

public FileInputStream(FileDescriptor fd) {
}
```

2. HDFS Short-Circuit Local Reads

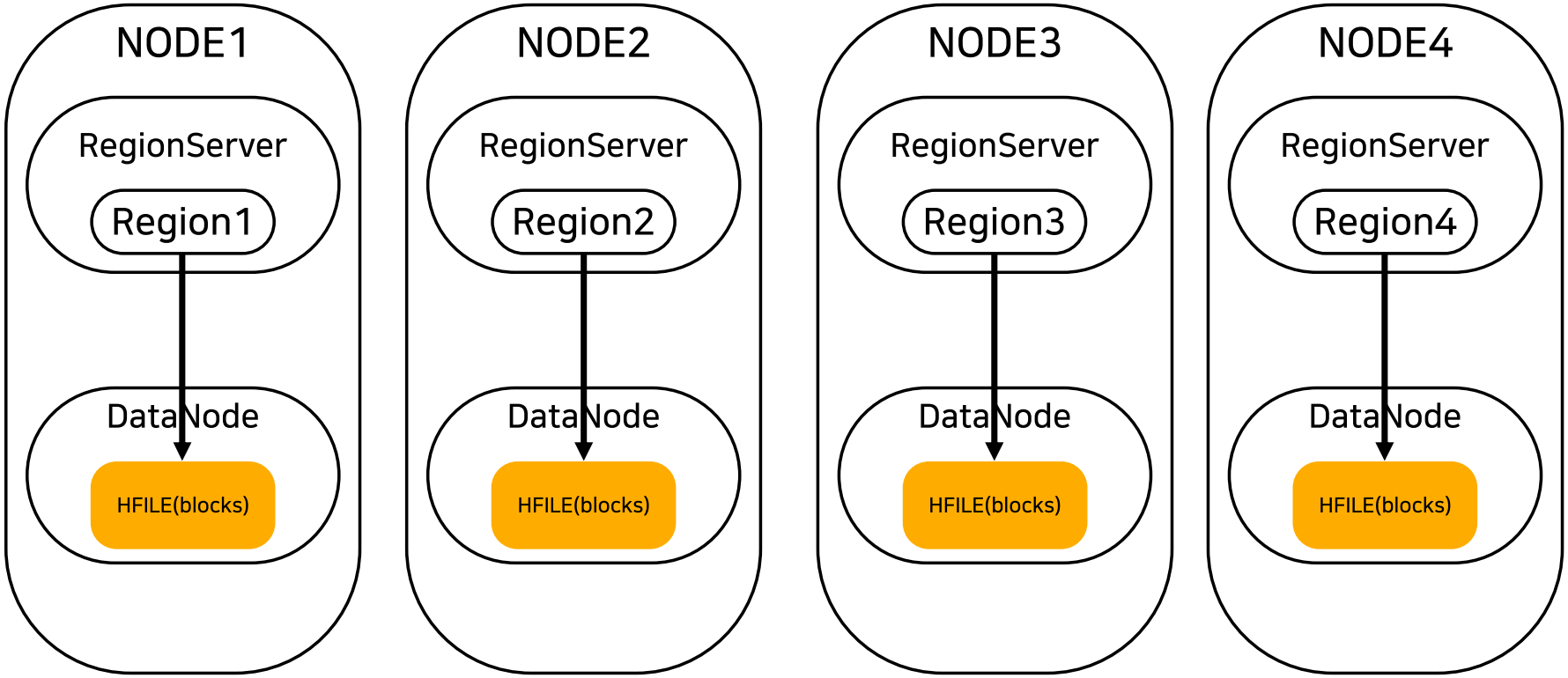




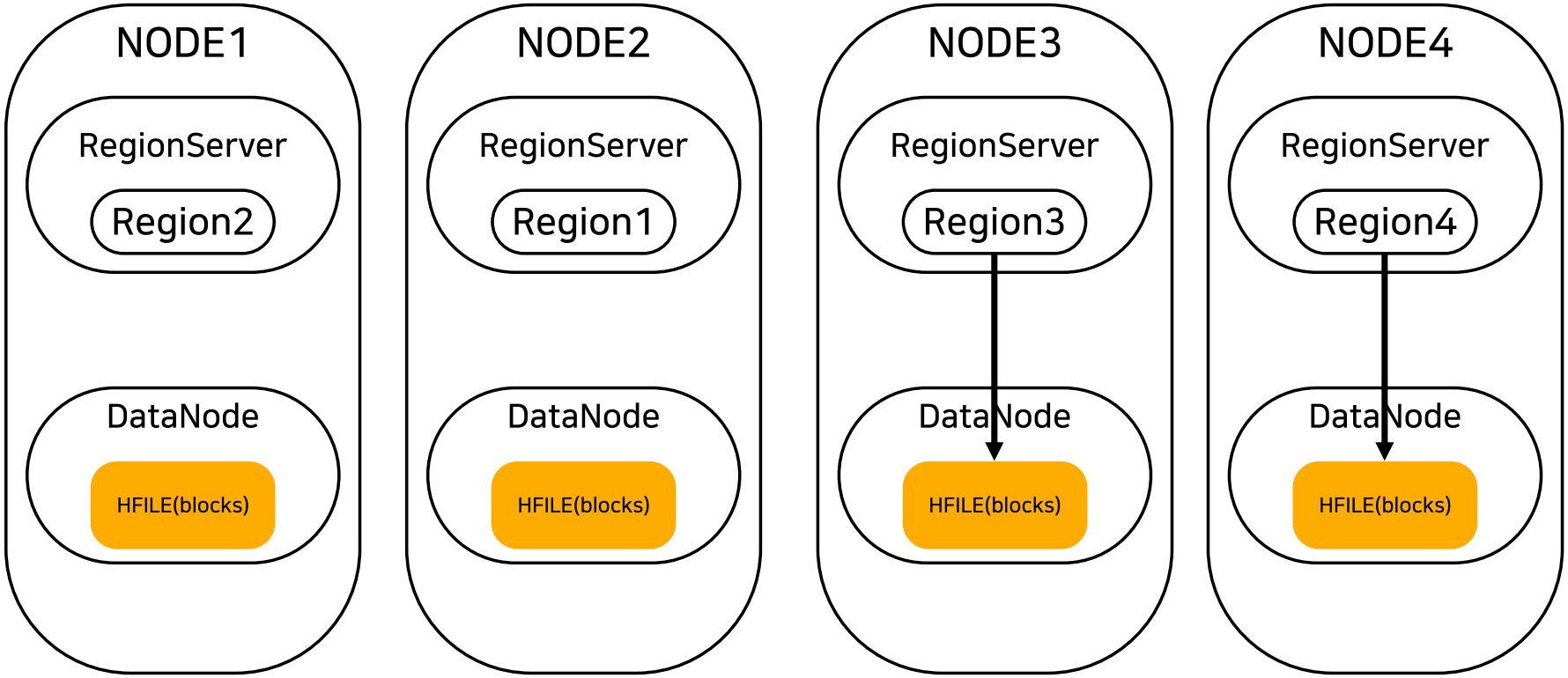

```
public class ShortCircuitReplica{
  private final FileInputStream dataStream
  private final Slot slot
  ...
}

public FileInputStream(FileDescriptor fd) {
}
```

네이버 데이터 저장소에서 HBase Locality를 지키기 위한 운영

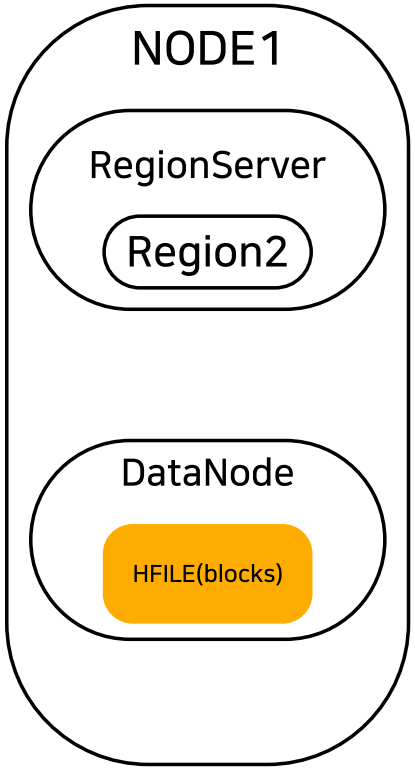
1. 롤링 리스타트 후


NODE1
RegionServer
Region1
DataNode
HFILE(blocks)
NODE2
RegionServer
Region2
DataNode
HFILE(blocks)
NODE3
RegionServer
Region3
DataNode
HFILE(blocks)
NODE4
RegionServer
Region4
DataNode
HFILE(blocks)

1. 롤링 리스타트 후


NODE1
RegionServer
Region2
DataNode
HFILE(blocks)
NODE2
RegionServer
Region1
DataNode
HFILE(blocks)
NODE3
RegionServer
Region3
DataNode
HFILE(blocks)
NODE4
RegionServer
Region4
DataNode
HFILE(blocks)

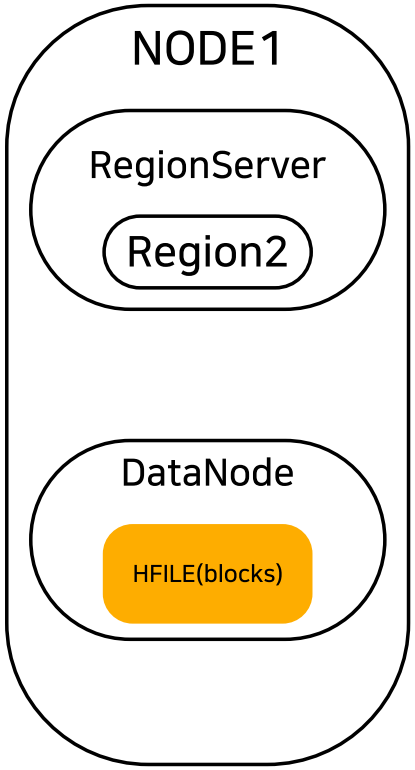
1. 롤링 리스타트 후

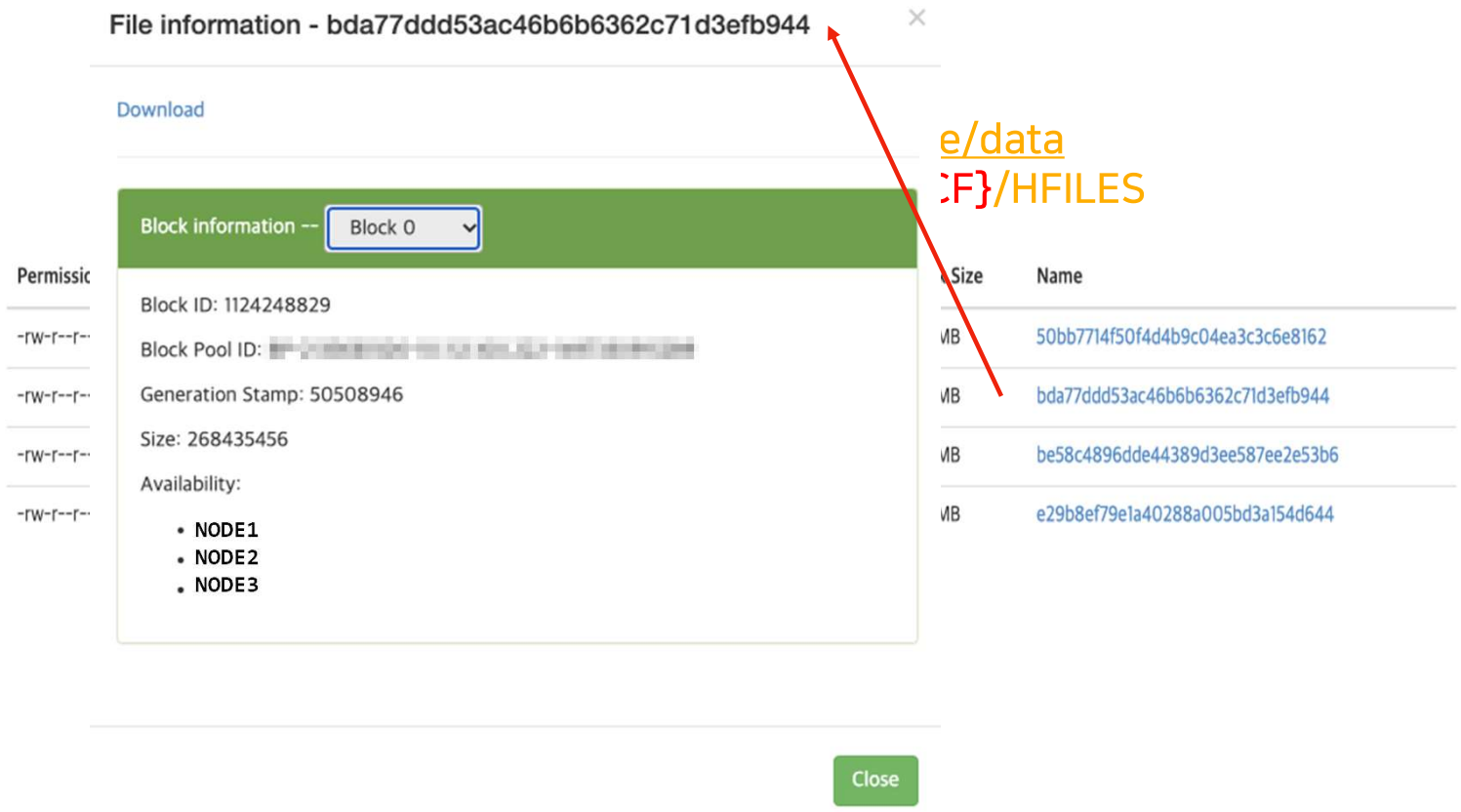


HDFS://{HDFS_NAMESPACE}/HBase/data
/{NameSpace}/{Table}/{Region}/{CF}/HFILES

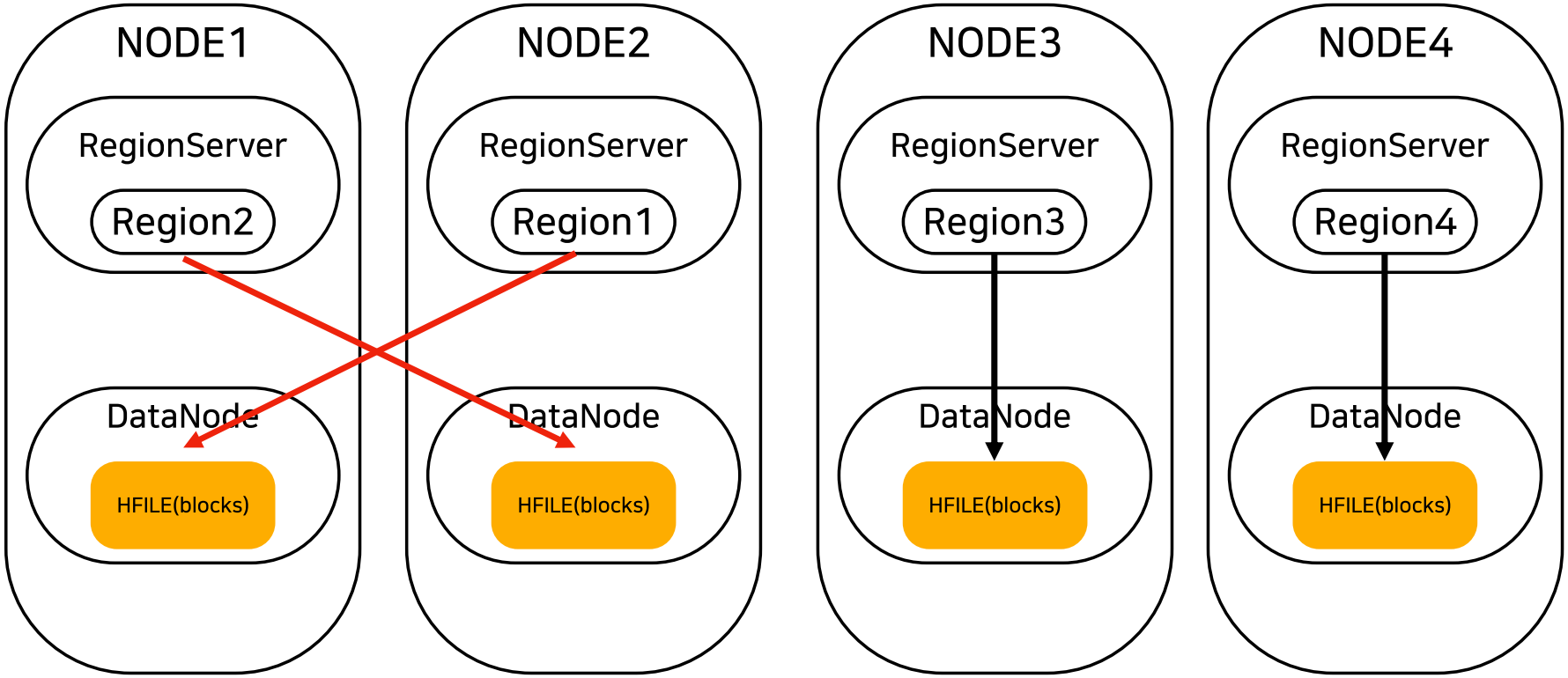
Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	[redacted]	[redacted]	202.33 MB	2021. 4. 14. 오후 2:08:39	3	256 MB	50bb7714f50f4d4b9c04ea3c3c6e8162
-rw-r--r--	[redacted]	[redacted]	103.79 GB	2021. 4. 14. 오전 1:10:45	3	256 MB	bda77ddd53ac46b6b6362c71d3efb944
-rw-r--r--	[redacted]	[redacted]	1.41 GB	2021. 4. 14. 오후 12:18:45	3	256 MB	be58c4896dde44389d3ee587ee2e53b6
-rw-r--r--	[redacted]	[redacted]	67.37 MB	2021. 4. 14. 오후 2:46:09	3	256 MB	e29b8ef79e1a40288a005bd3a154d644

1. 롤링 리스타트 후

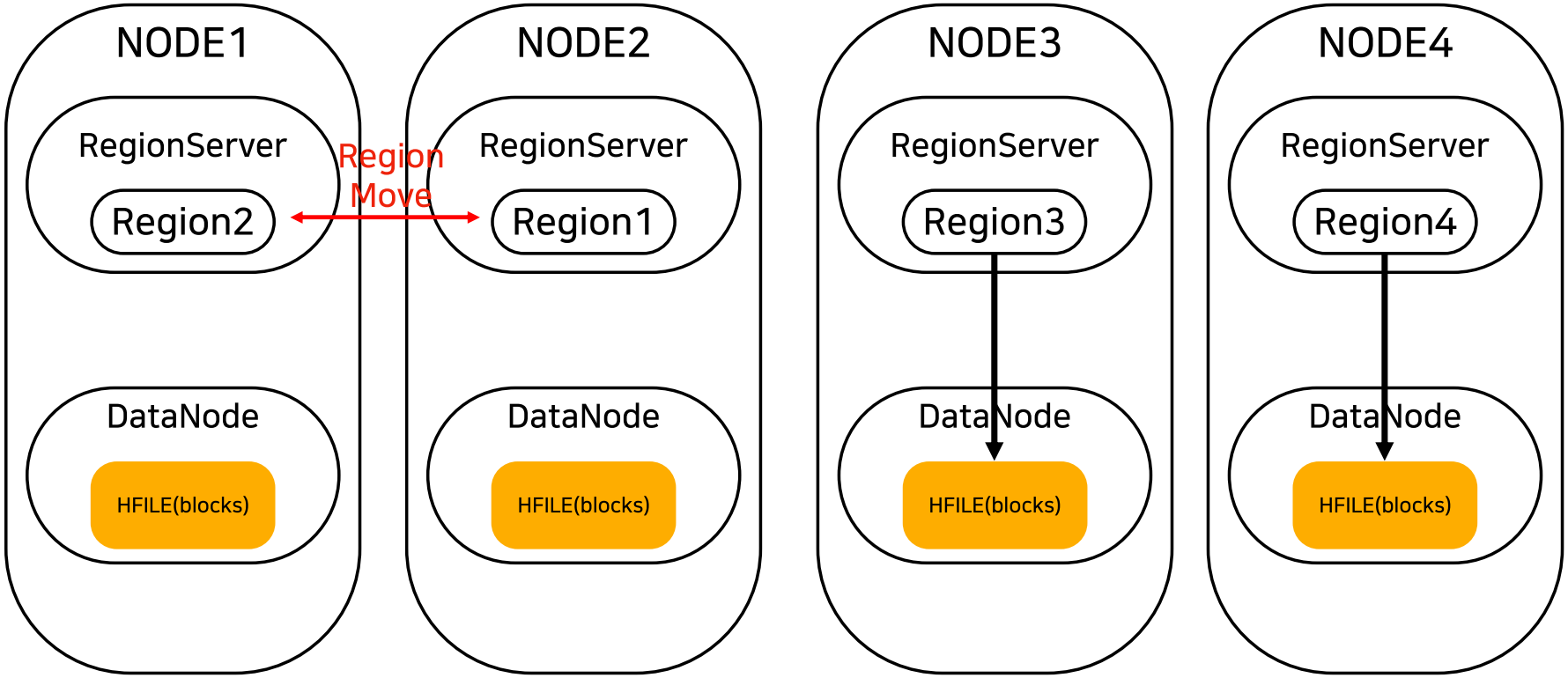

NODE1
RegionServer
Region2
DataNode
HFILE(blocks)


File information - bda77ddd53ac46b6b6362c71d3efb944
Download
Block information --
Block 0
Block ID: 1124248829
Block Pool ID:
Generation Stamp: 50508946
Size: 268435456
Availability:
NODE1
NODE2
NODE3
Close
Permissic
Size
Name
50bb7714f50f4d4b9c04ea3c3c6e8162
bda77ddd53ac46b6b6362c71d3efb944
be58c4896dde44389d3ee587ee2e53b6
e29b8ef79e1a40288a005bd3a154d644
e/data
F}/HFILES

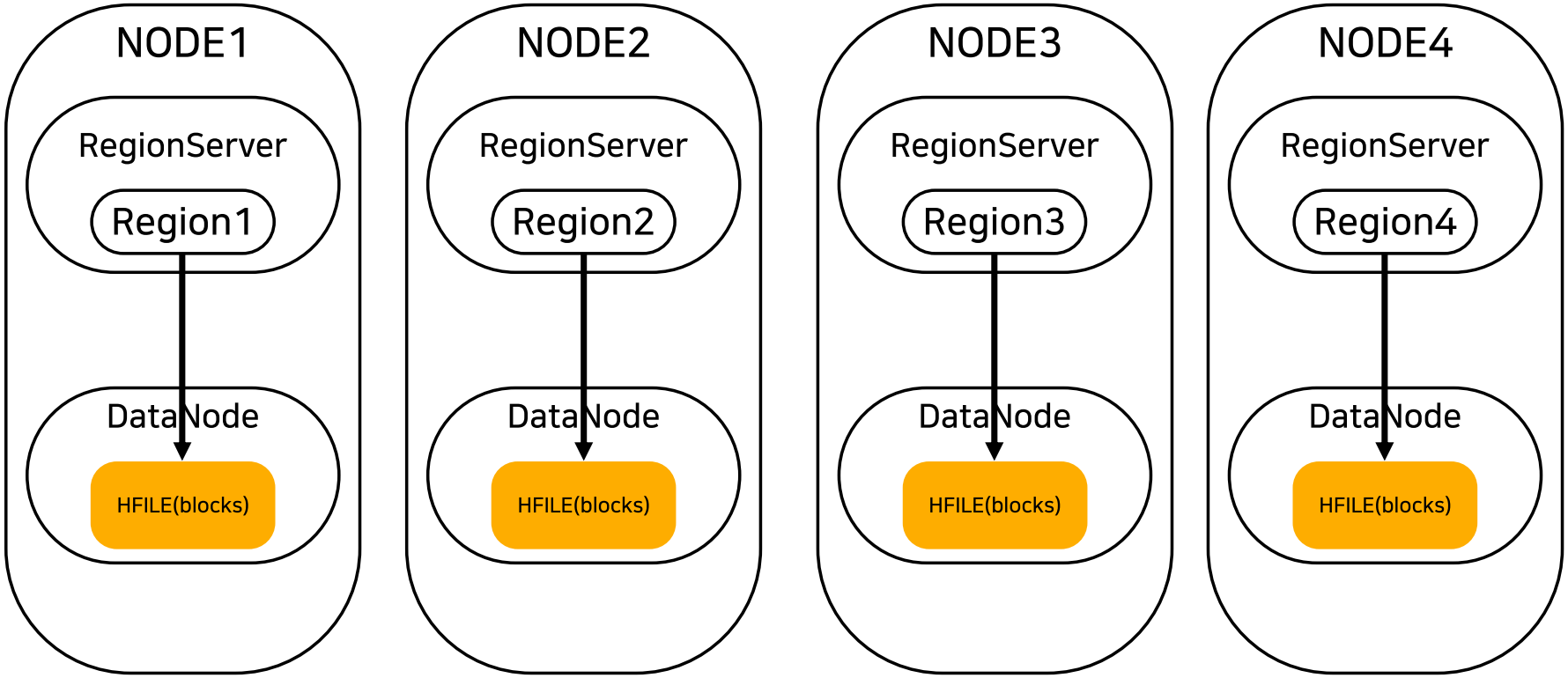
1. 롤링 리스타트 후


NODE1
RegionServer
Region2
DataNode
HFILE(blocks)
NODE2
RegionServer
Region1
DataNode
HFILE(blocks)
NODE3
RegionServer
Region3
DataNode
HFILE(blocks)
NODE4
RegionServer
Region4
DataNode
HFILE(blocks)

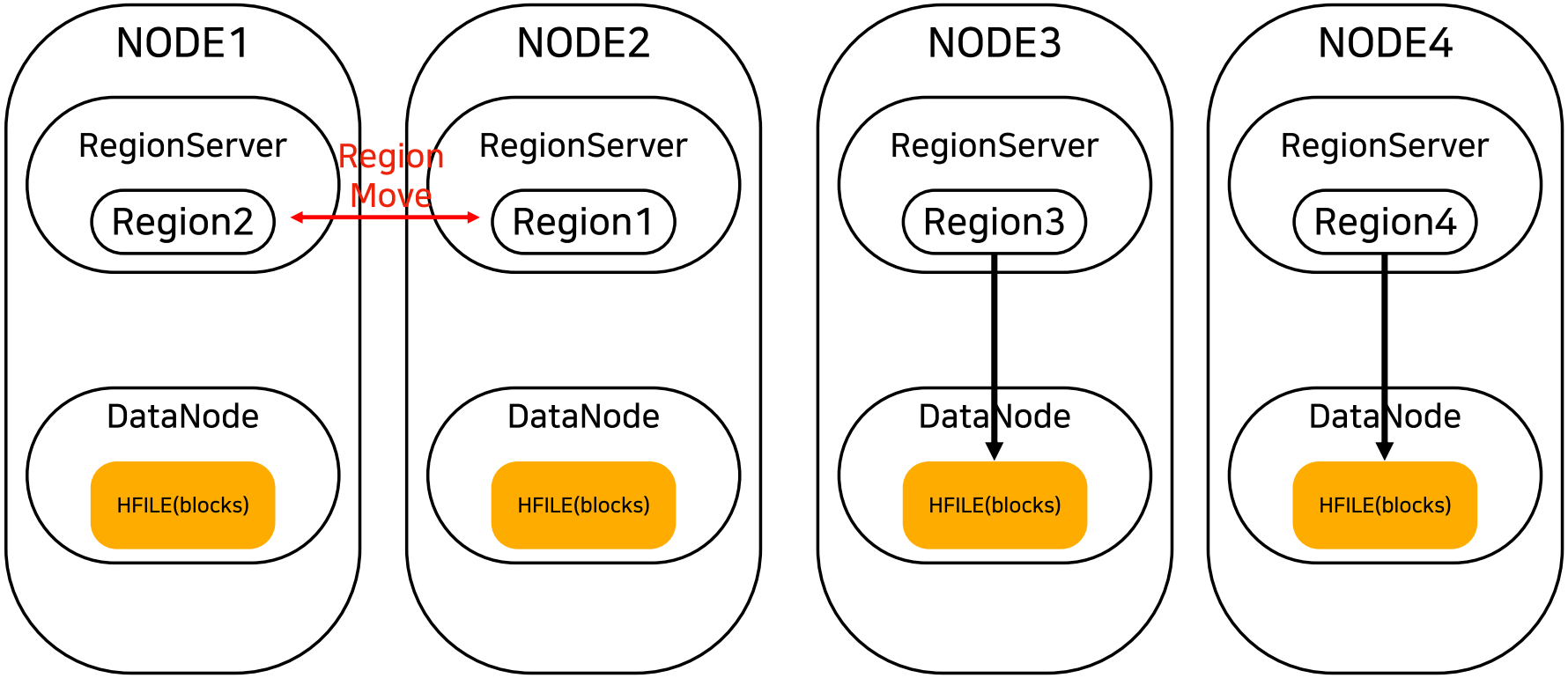
1. 롤링 리스타트 후


NODE1
RegionServer
Region2
DataNode
HFILE(blocks)
Region Move
NODE2
RegionServer
Region1
DataNode
HFILE(blocks)
NODE3
RegionServer
Region3
DataNode
HFILE(blocks)
NODE4
RegionServer
Region4
DataNode
HFILE(blocks)

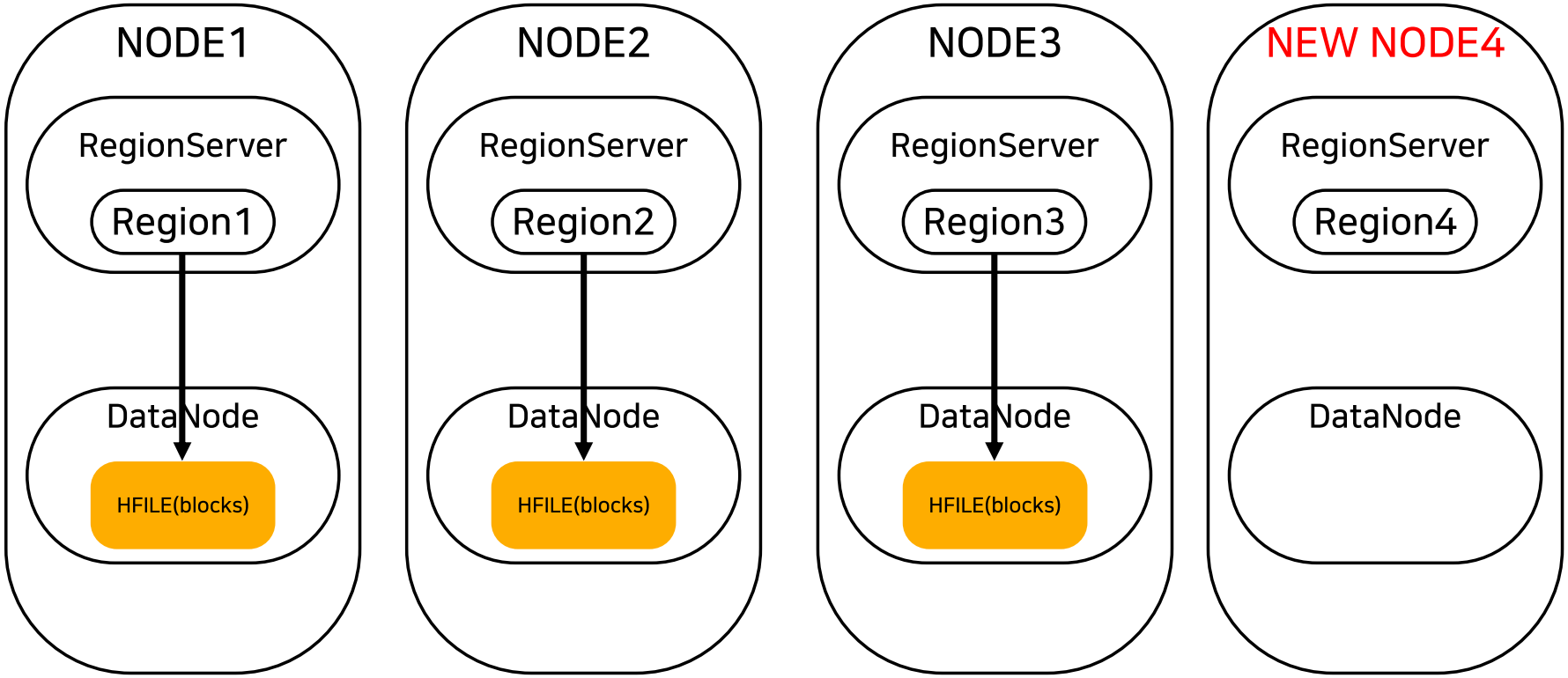
1. 롤링 리스타트 후


NODE1
RegionServer
Region1
DataNode
HFILE(blocks)
NODE2
RegionServer
Region2
DataNode
HFILE(blocks)
NODE3
RegionServer
Region3
DataNode
HFILE(blocks)
NODE4
RegionServer
Region4
DataNode
HFILE(blocks)

2. 장비 수리 후


NODE1
RegionServer
Region2
DataNode
HFILE(blocks)
Region Move
NODE2
RegionServer
Region1
DataNode
HFILE(blocks)
NODE3
RegionServer
Region3
DataNode
HFILE(blocks)
NODE4
RegionServer
Region4
DataNode
HFILE(blocks)

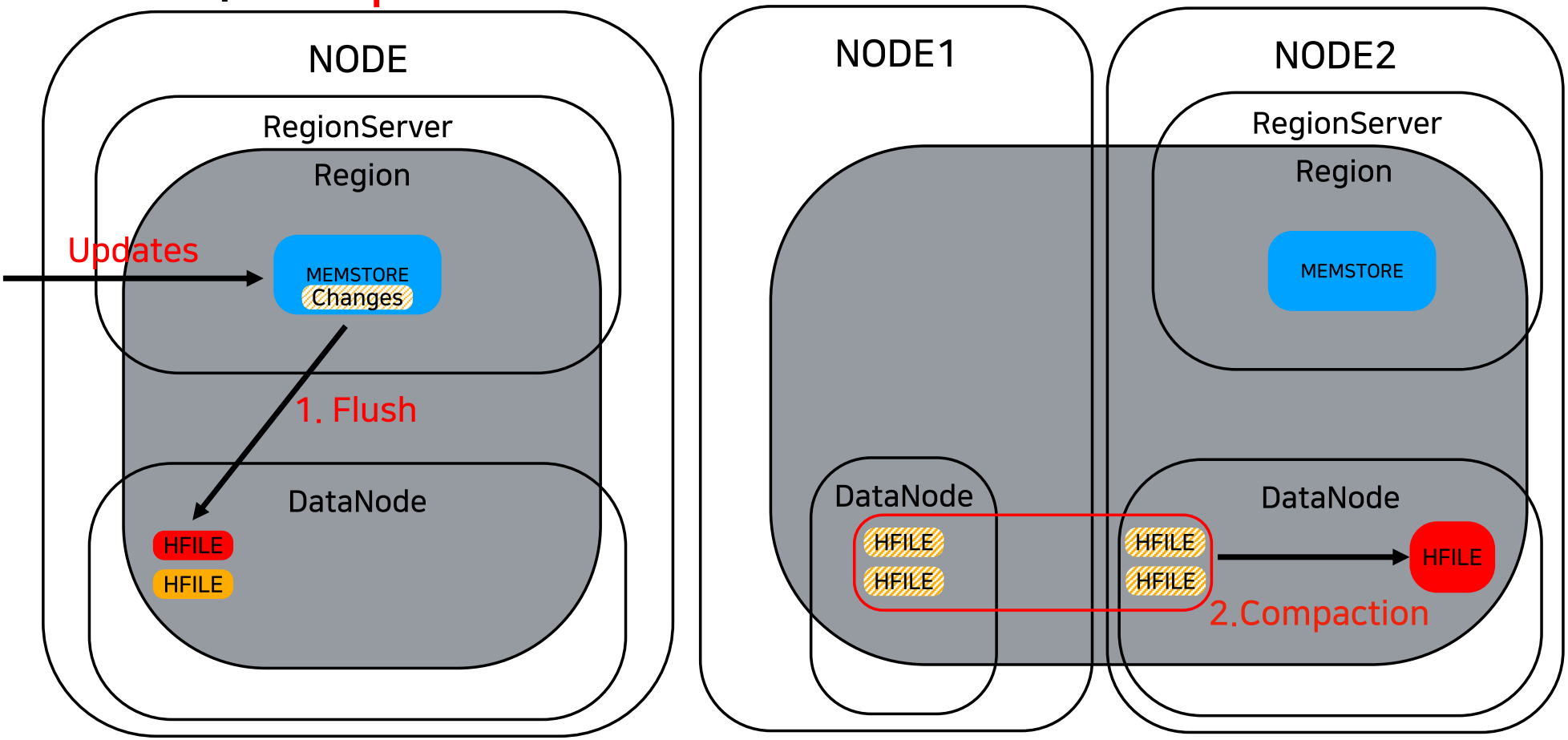
3. 새로운 장비 추가 후


NODE1
RegionServer
Region1
DataNode
HFILE(blocks)
NODE2
RegionServer
Region2
DataNode
HFILE(blocks)
NODE3
RegionServer
Region3
DataNode
HFILE(blocks)
NEW NODE4
RegionServer
Region4
DataNode

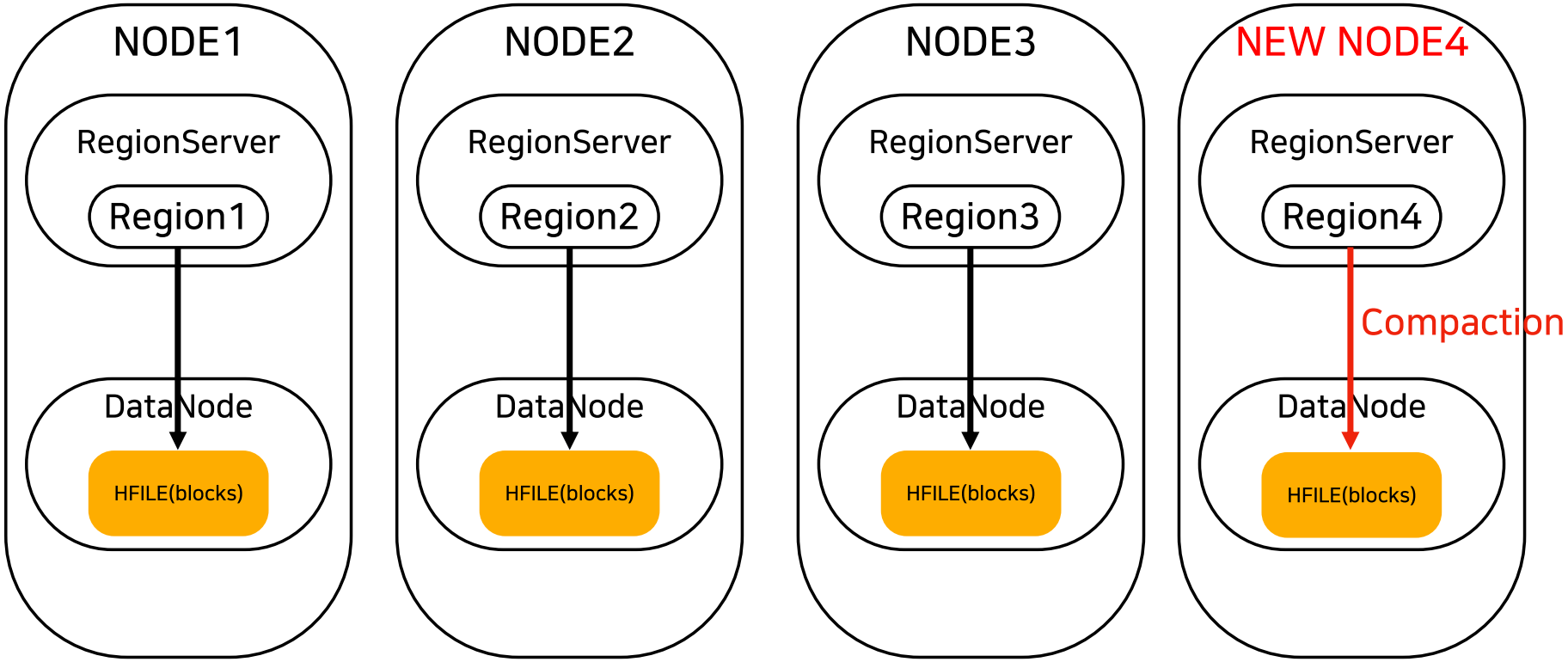
HBase에서 Locality를 올리는 방법

DEVIEW 2021

Flush와 Compaction



3. 새로운 장비 추가 후


NODE1
RegionServer
Region1
DataNode
HFILE(blocks)
NODE2
RegionServer
Region2
DataNode
HFILE(blocks)
NODE3
RegionServer
Region3
DataNode
HFILE(blocks)
NEW NODE4
RegionServer
Region4
Compaction
DataNode
HFILE(blocks)

클러스터 규모와 비례하는 디스크 장애

- 수천대의 클러스터
- 장비 한대에서 사용하는 DISK는 12개
- 클러스터 규모와 증가하는 DISK 장애율

Cluster	장비 수 (대)
Cluster A	xxxx
Cluster B	xxx
Cluster C	xxx
…	…
총	xxxx대

해당 년. 월	장애 발생 횟수 (건)
2021. 4	151
2021. 5	210
2021. 6	102
2021. 7	126
2021. 8	249
2021. 9	168

클러스터 규모와 비례하는 디스크 장애

DEVIEW 2021

- 한국 클러스터만 약 6000대
- 장비 한대에서 사용하는 DISK는 12개
- 클러스터 규모와 증가하는 DISK 장애율

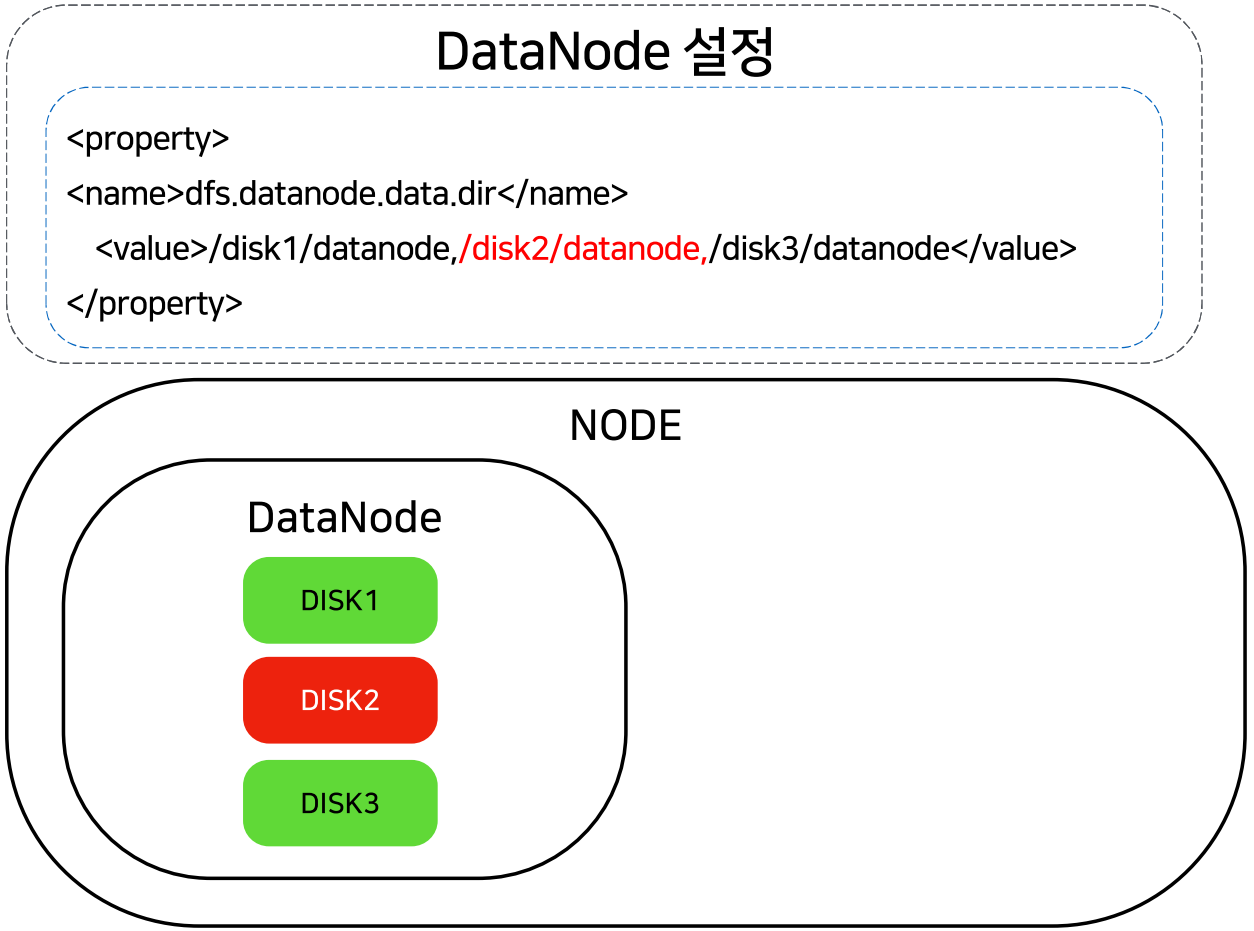
클러스터 규모와 반비례 하는 HBase Locality!

DataNode Hot Swap Drive

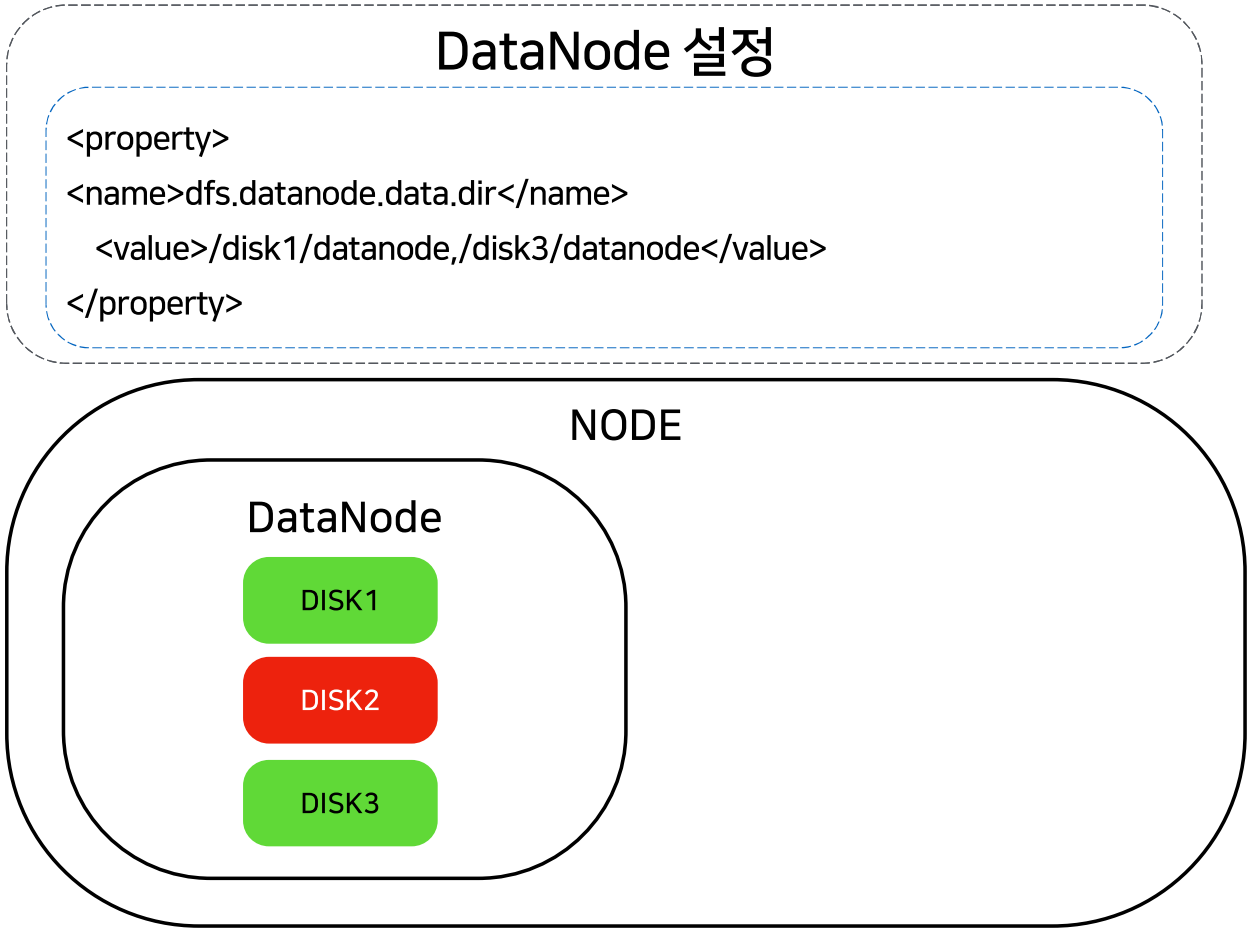
DEVIEW 2021

- DataNode Damon을 종료하지 않고 디스크 추가/제거 가능
- Hadoop 2.7.0 부터 지원
- 12개의 디스크 중 고장난 디스크만 교체 & 투입 가능
- HBase Locality를 최대한 유지하면서 교체 가능

DataNode Hot Swap Drive



DataNode Hot Swap Drive



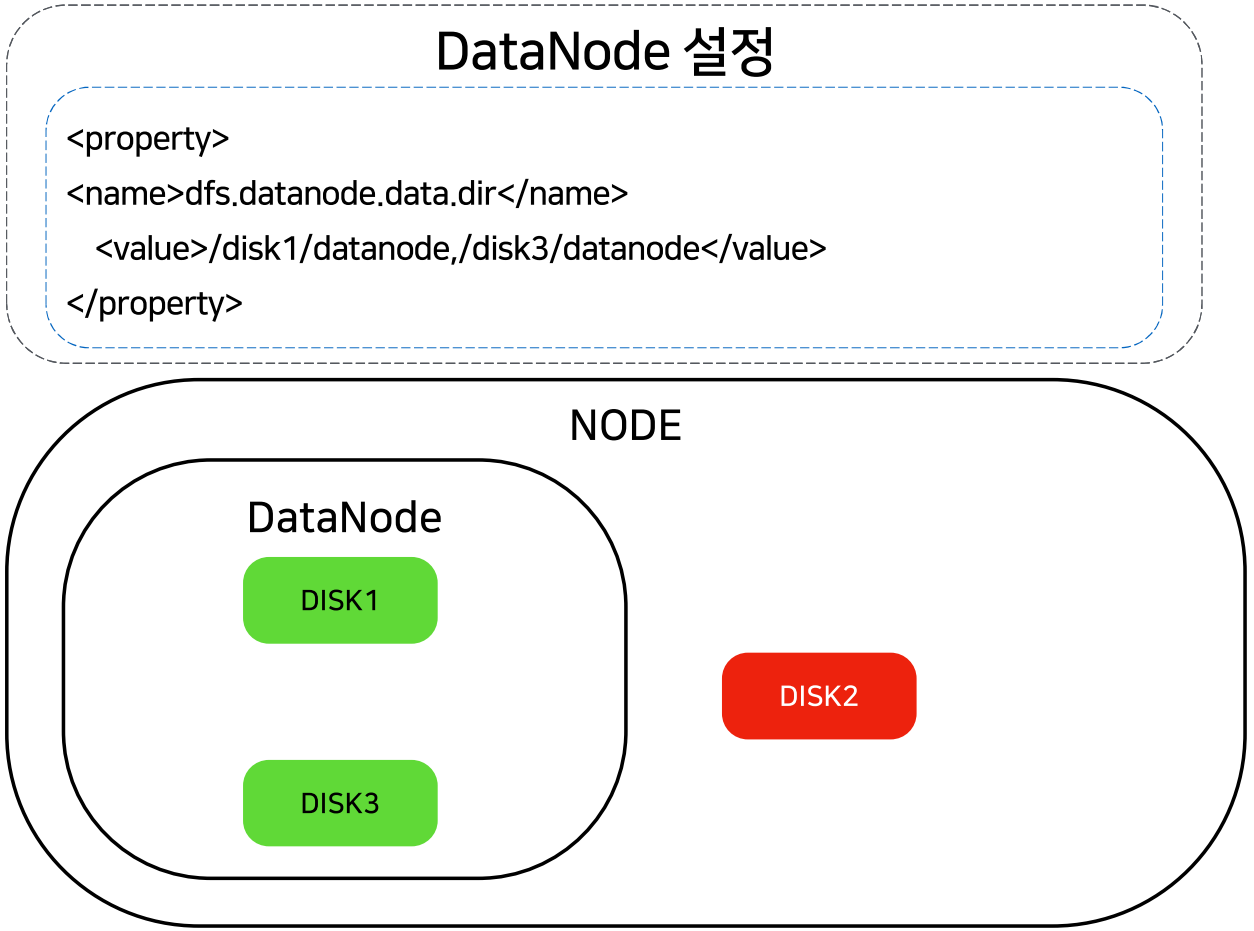
DataNode Hot Swap Drive

DEVIEW 2021

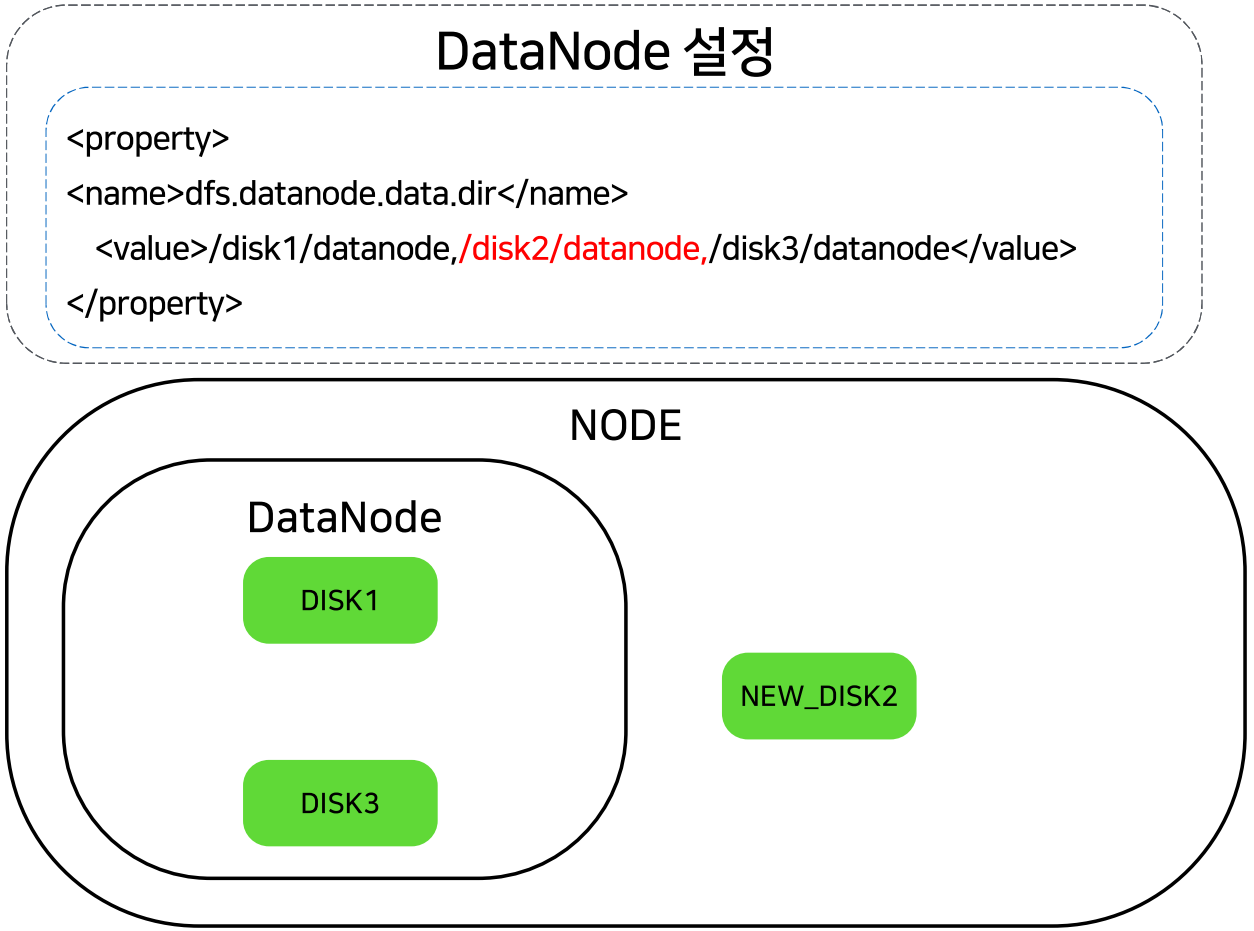
Reconfig

hdfs dfsadmin -reconfig datanode ${DATANODE}:${PORT} start

DataNode Hot Swap Drive



DataNode Hot Swap Drive



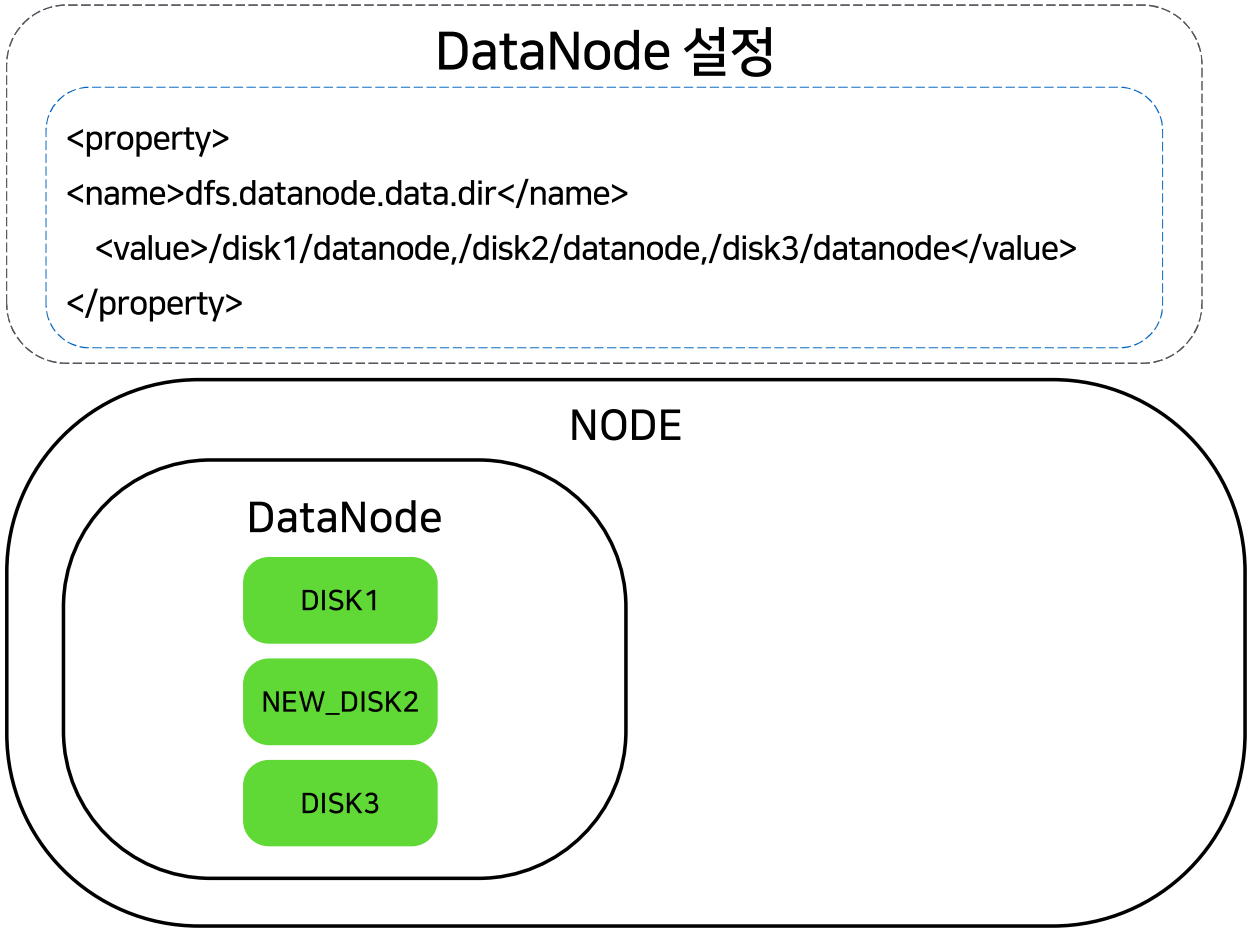
DataNode Hot Swap Drive

DEVIEW 2021

Reconfig

hdfs dfsadmin -reconfig datanode ${DATANODE}:${PORT} start

DataNode Hot Swap Drive



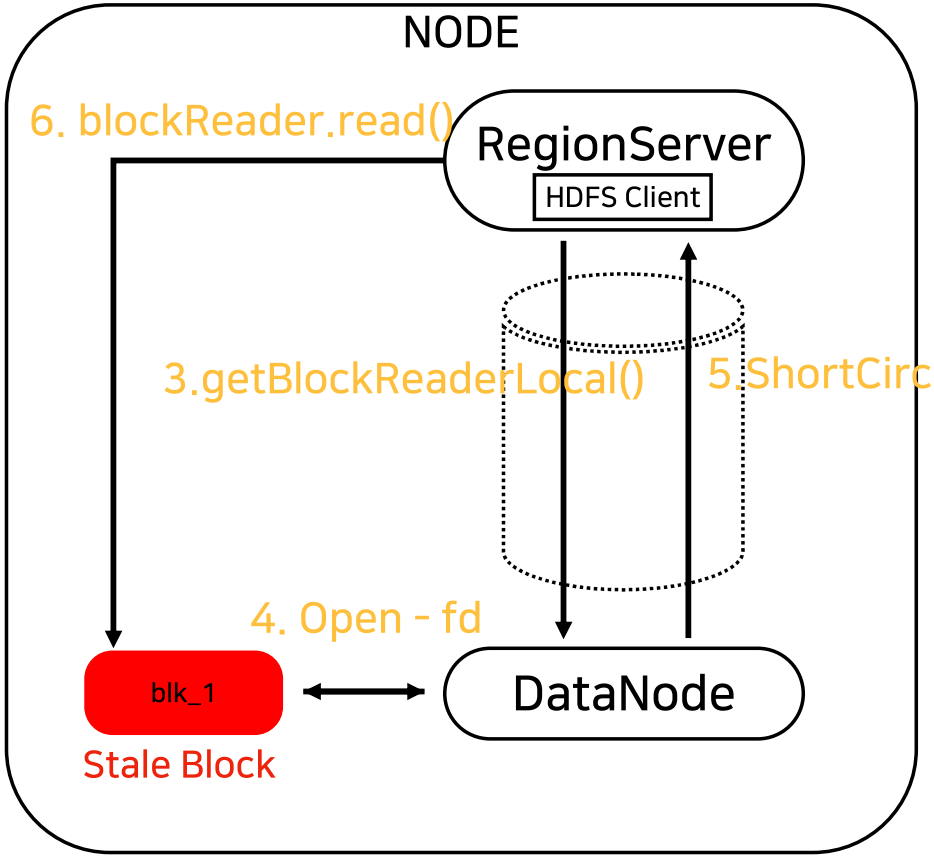
언마운트 불능 현상1

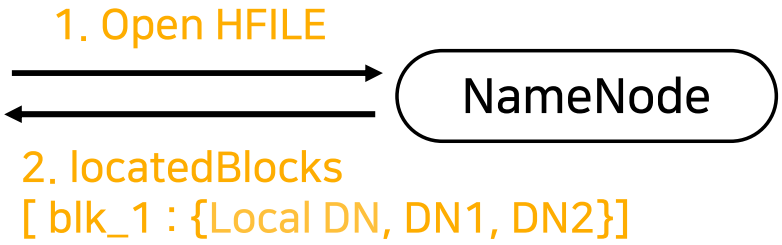
DEVIEW 2021

- 간헐적으로 디스크가 언마운트 되지 않는 현상 발생
  - -f 옵션을 주고 강제로 언마운트함
  - 이 경우, OS에서는 계속 오류가 발생하고 있는 것으로 인지
  - 혹은 FD를 물고있는 프로세스 종료 (ex: RegionServer)

- 원인은 HDFS의 short-circuit이 HBase에서 예기치 못한 동작을 초래한 것
  - 디스크 장애 시, HDFS에서는 즉시 해당 디스크의 FD를 닫음
  - 하지만, lazy 방식으로 FD를 관리하는 HBase에서는 열어두고 있음
  - 결과적으로 해당 디스크가 사용 중(busy)이므로 언마운트가 제대로 되지 않는 현상이 발생

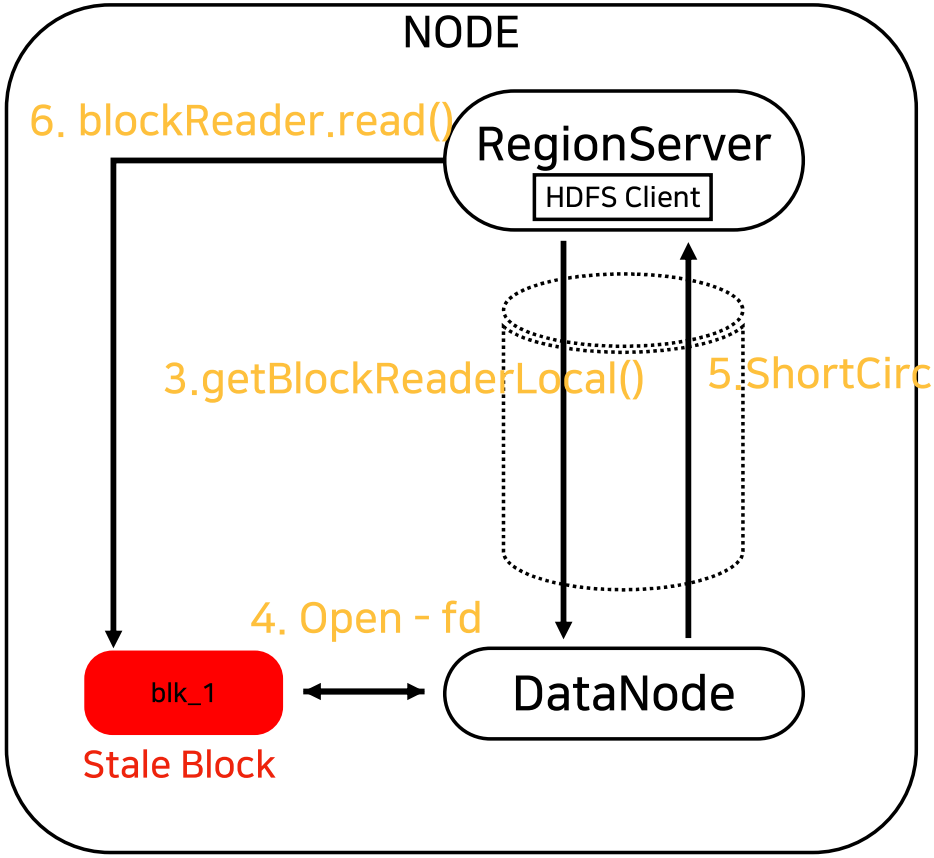
Block이 유효하지 않을 때

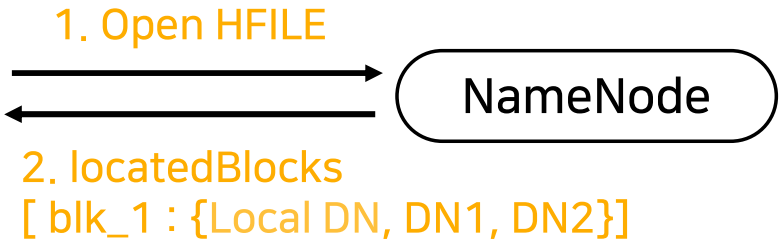





```
public class ShortCircuitReplica{
  private final FileInputStream dataStream
  private final Slot slot
  ...
}

public FileInputStream(FileDescriptor fd) {
}
```

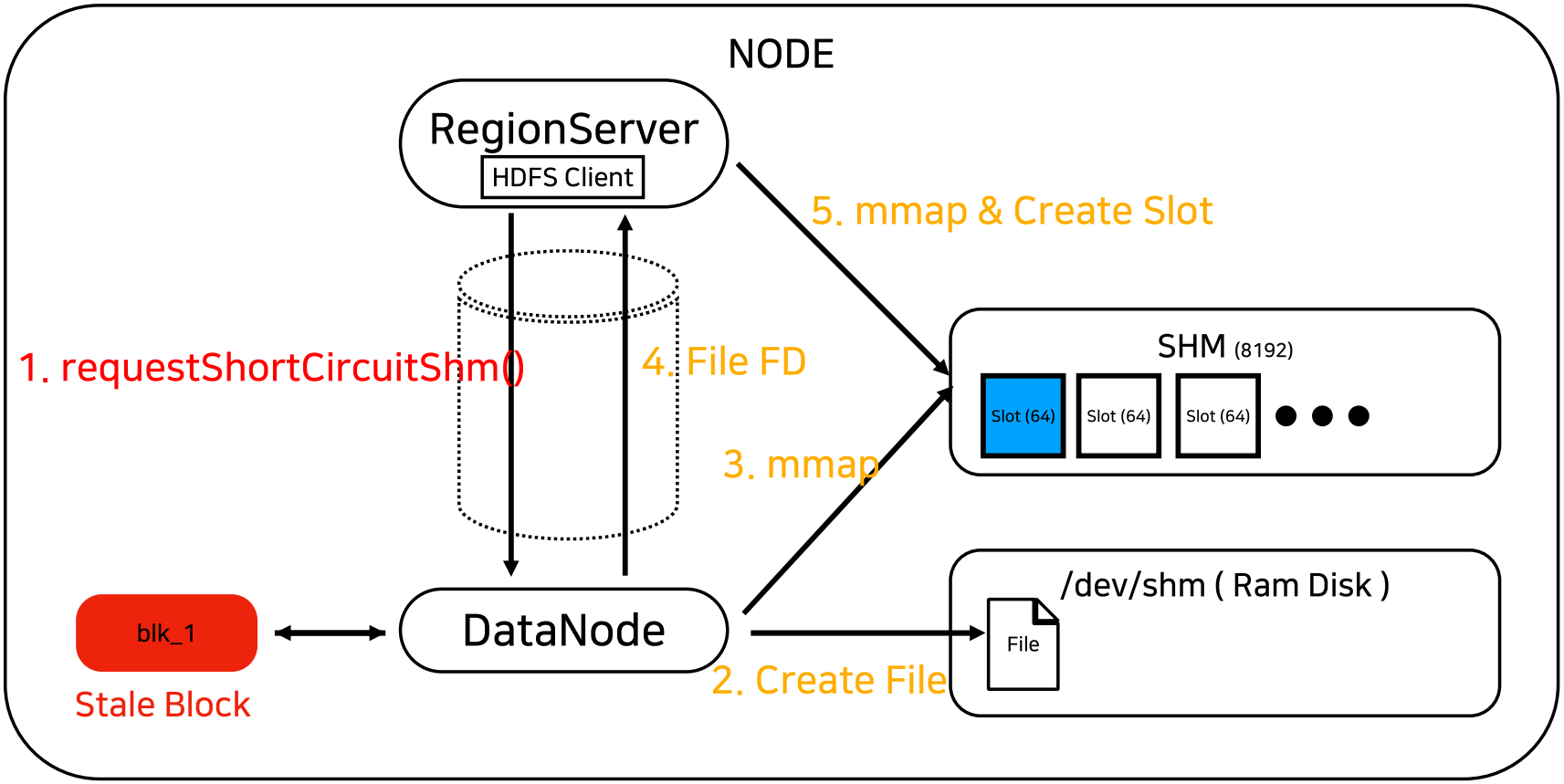
Block이 유효하지 않을 때



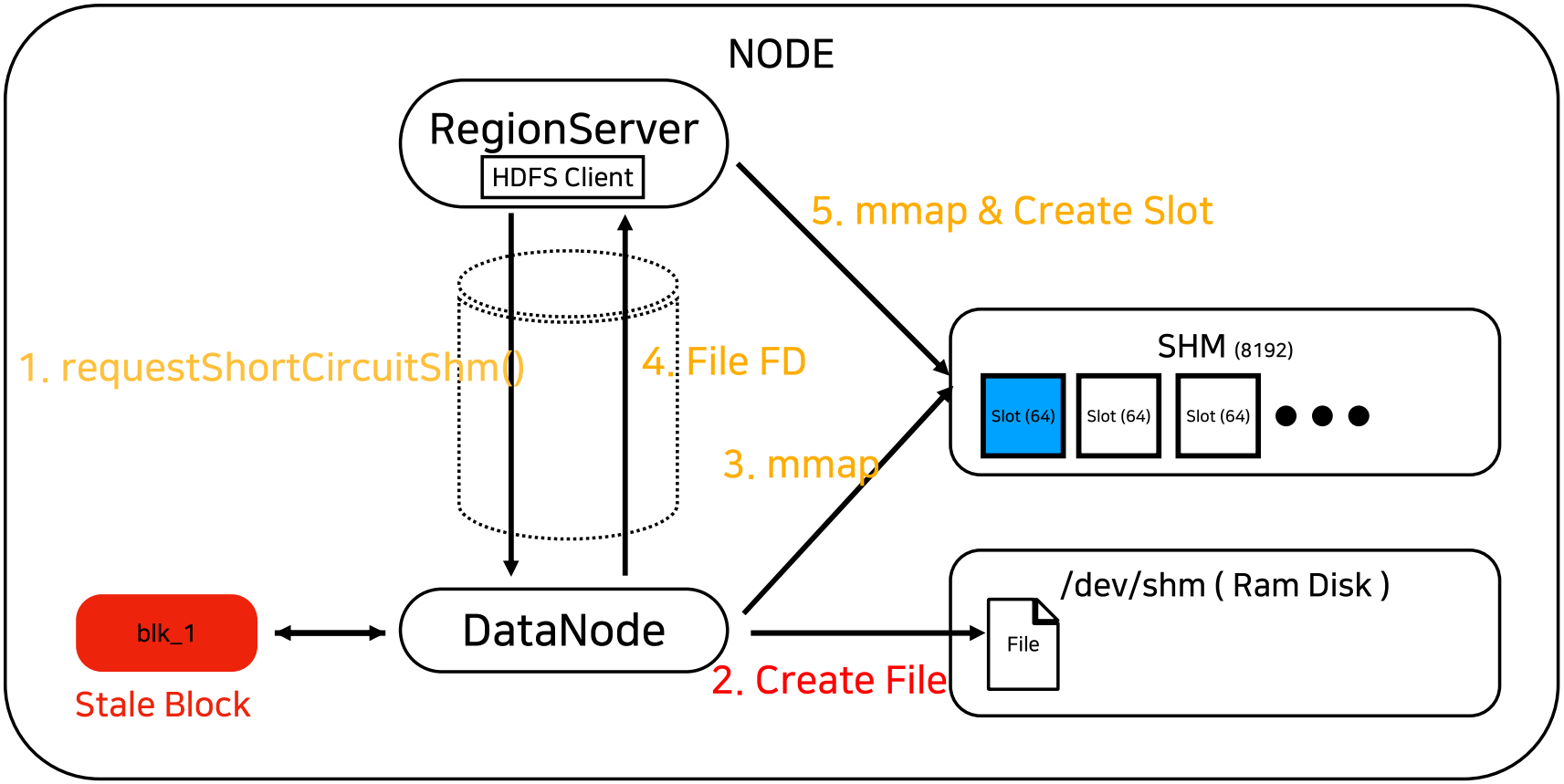



```
public class ShortCircuitReplica{
  private final FileInputStream dataStream
  private final Slot slot
  ...
}

public FileInputStream(FileDescriptor fd) {
}
```

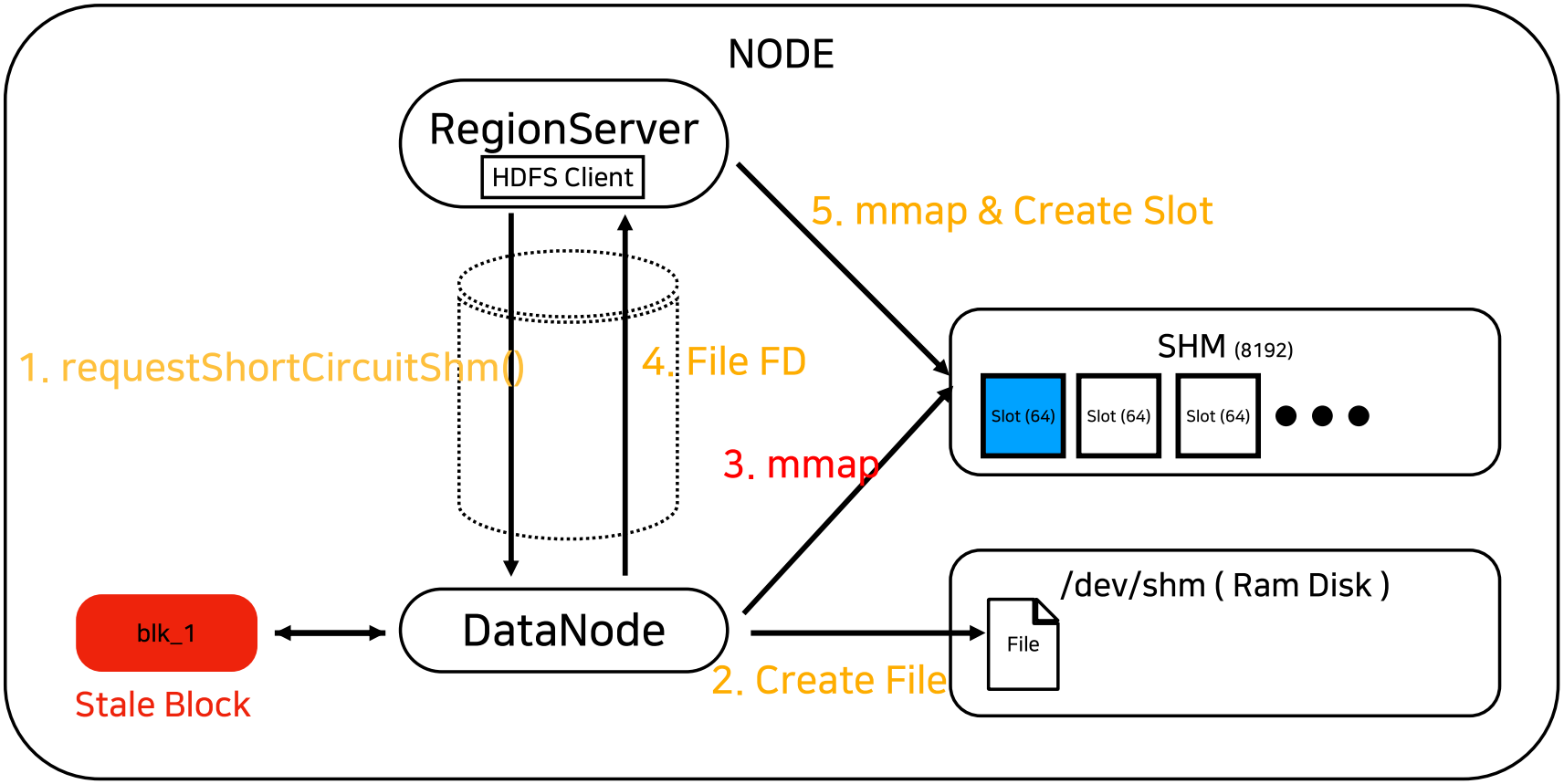
Block이 유효하지 않을 때



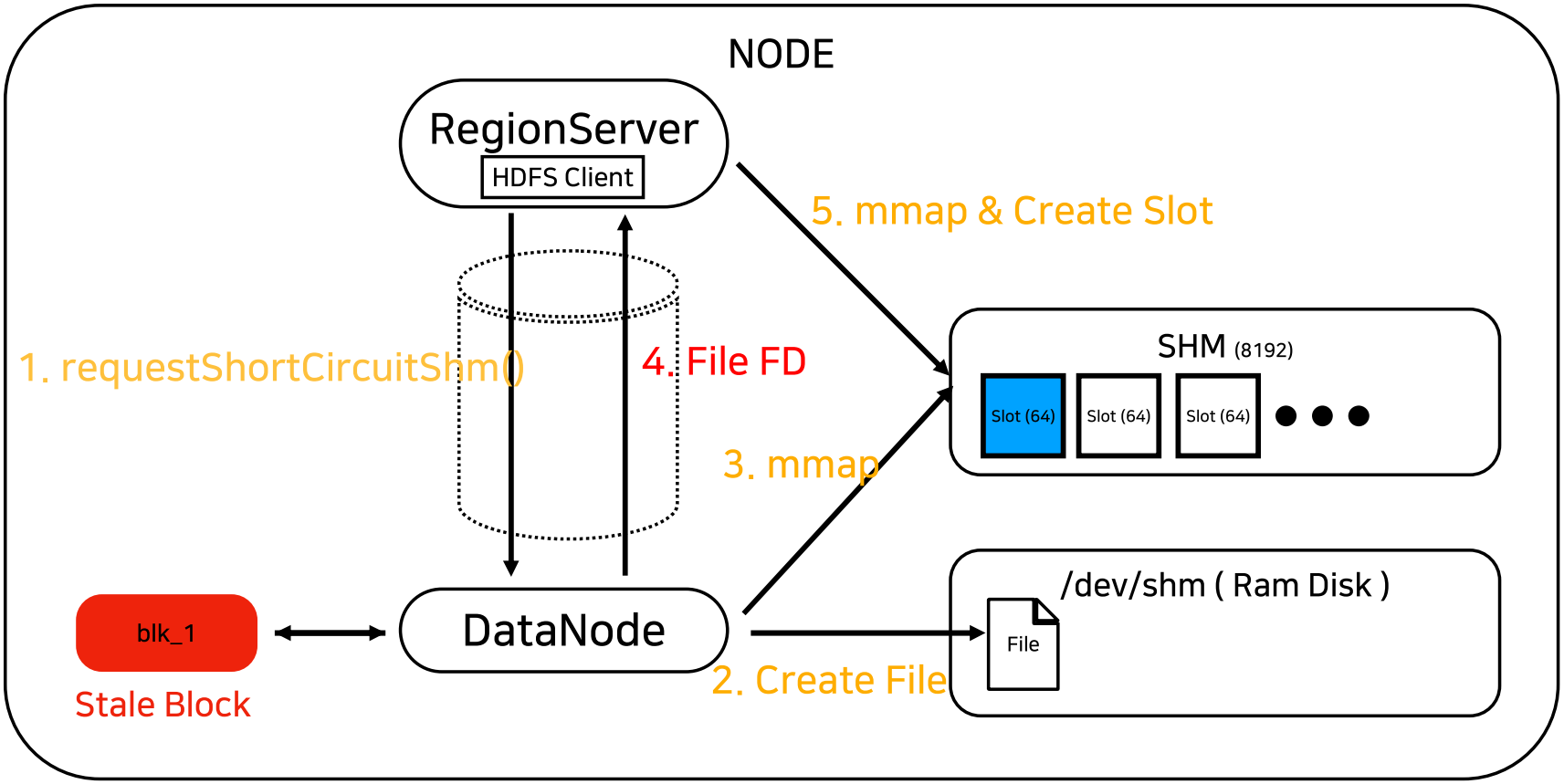
Block이 유효하지 않을 때



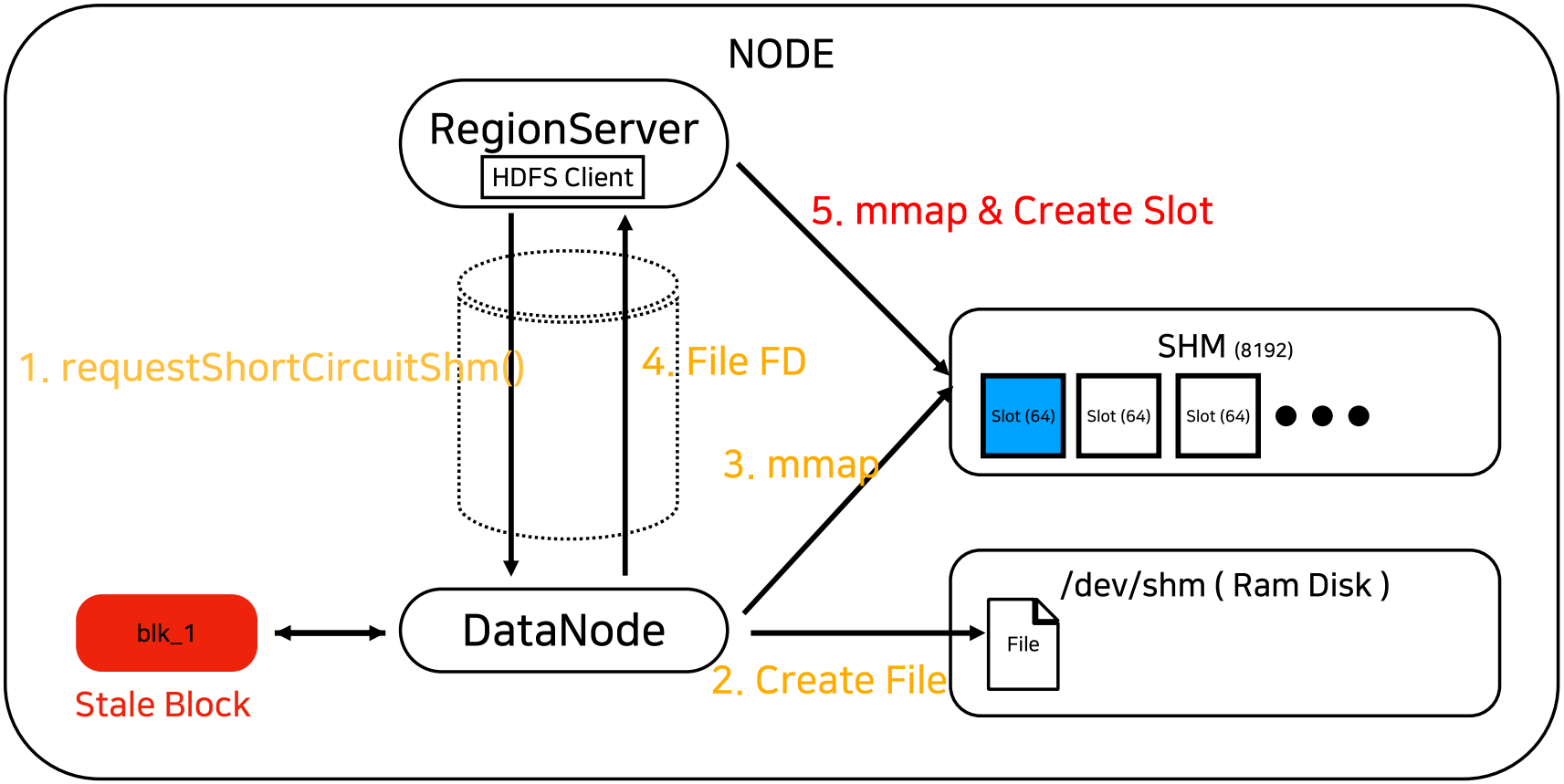
Block이 유효하지 않을 때



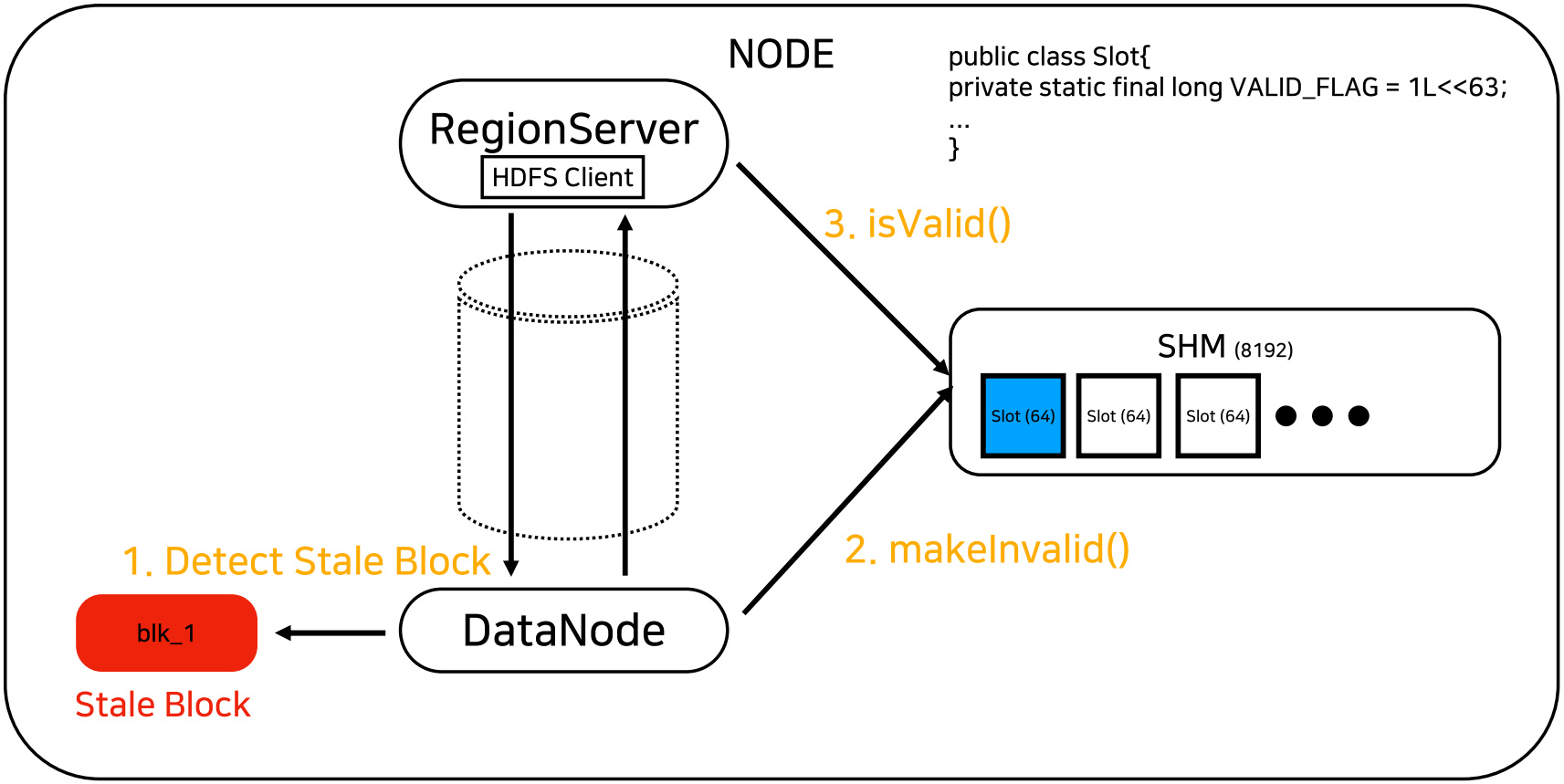
Block이 유효하지 않을 때



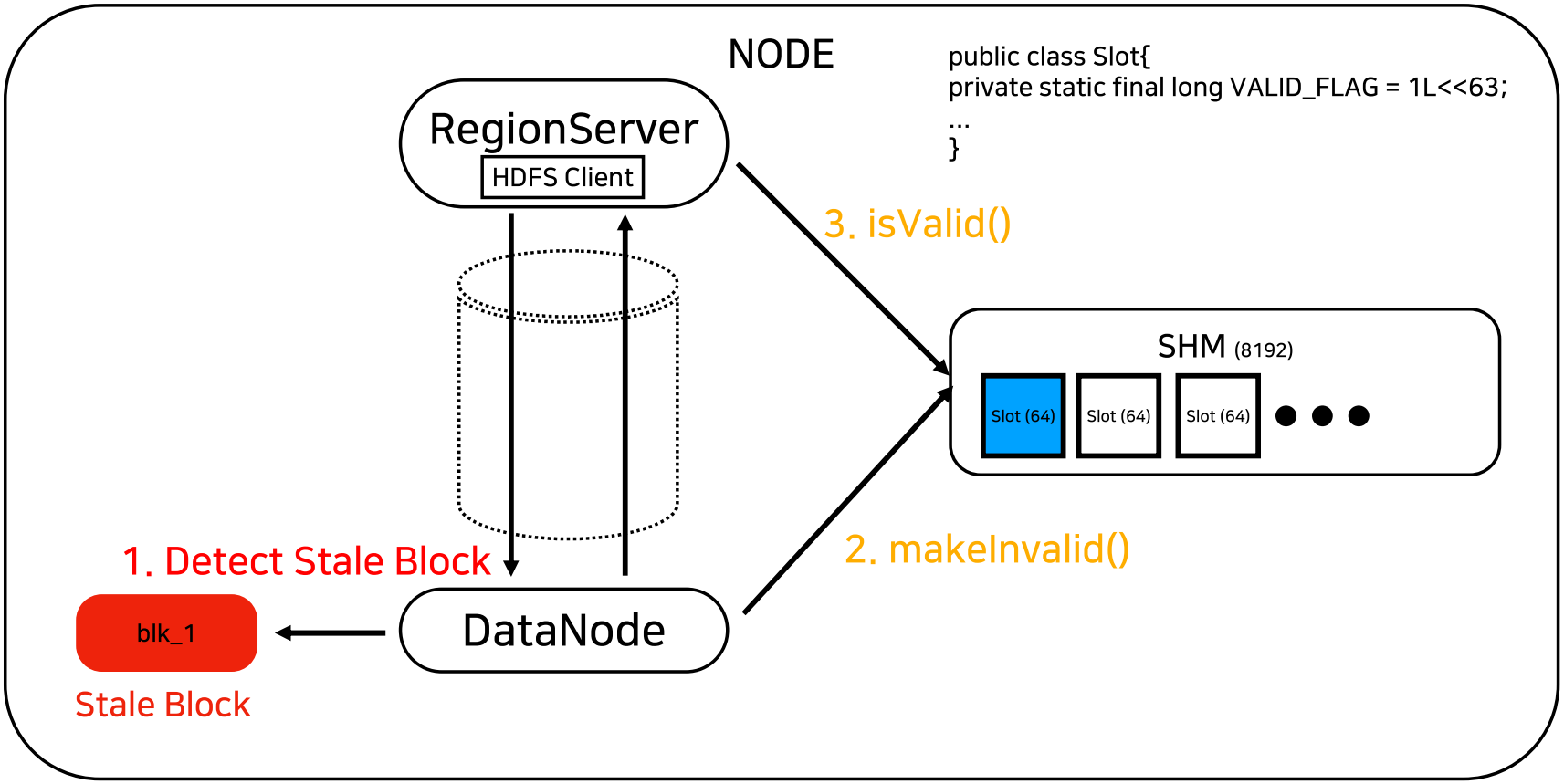
Block이 유효하지 않을 때



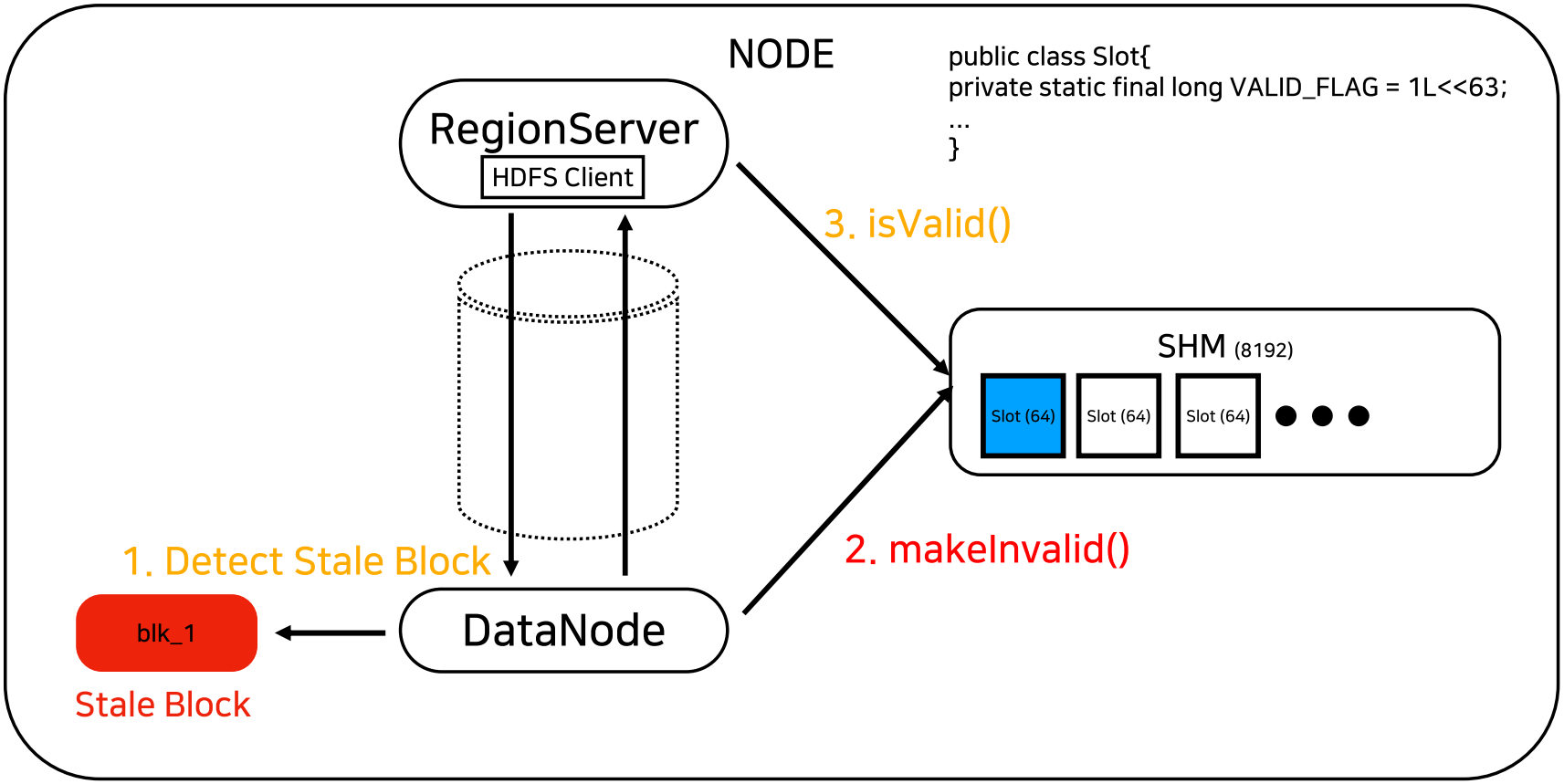
Block이 유효하지 않을 때



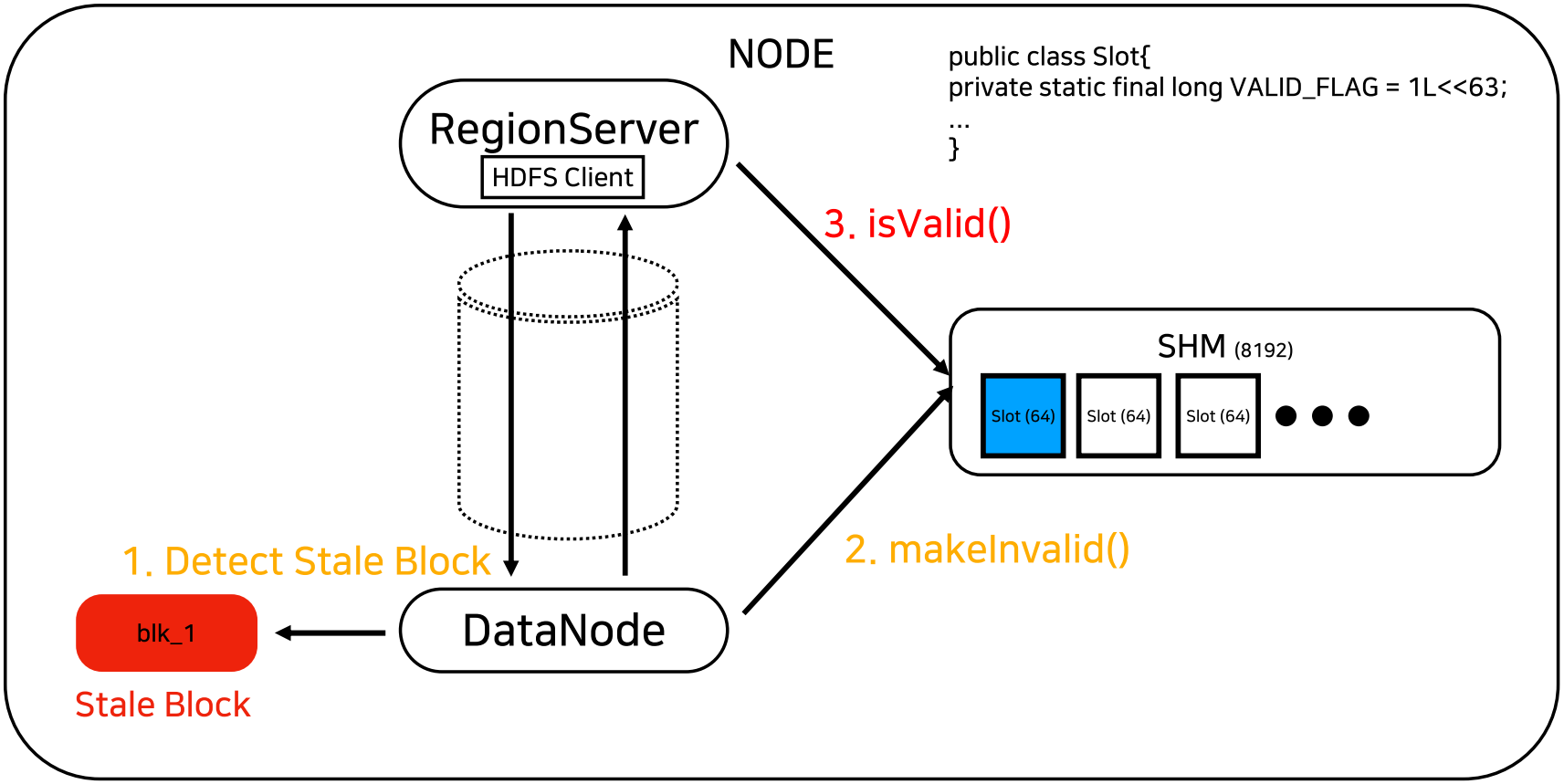
Block이 유효하지 않을 때



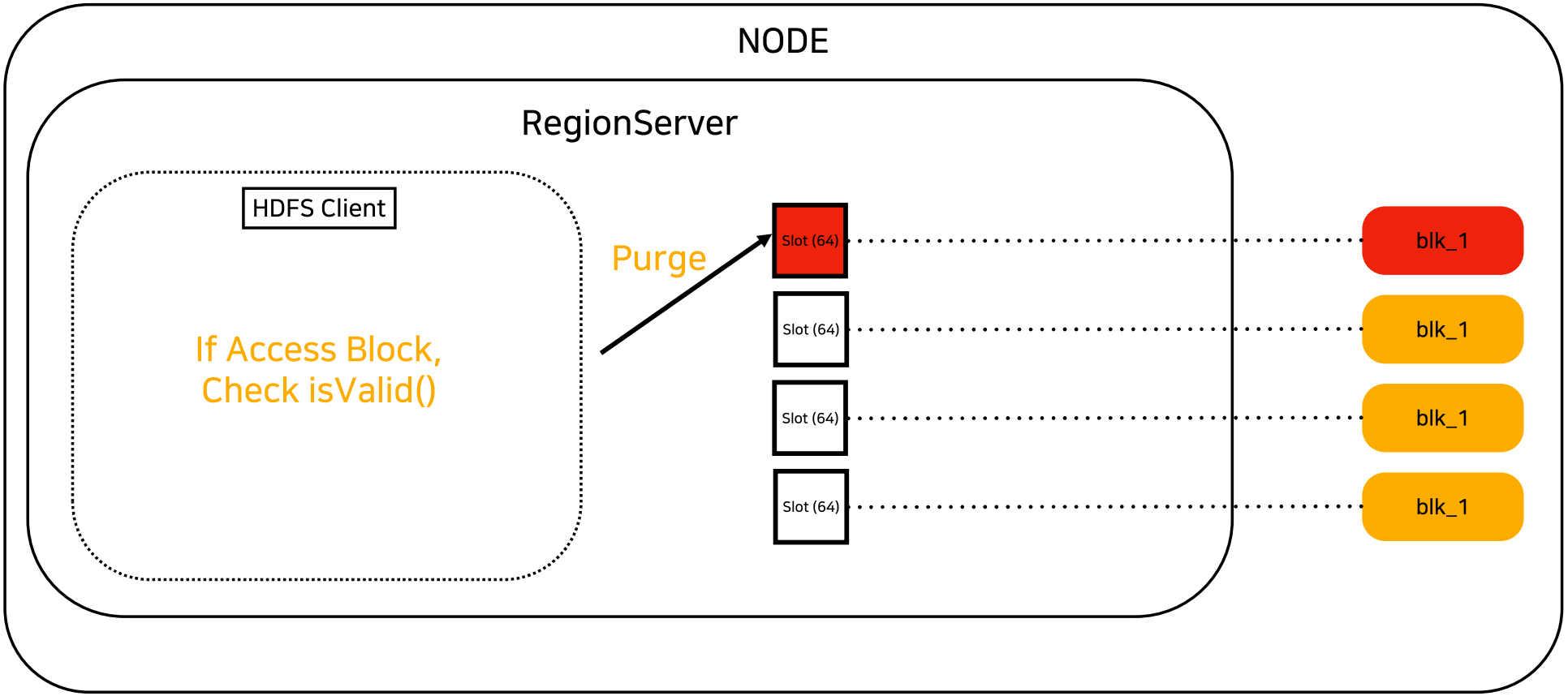
Block이 유효하지 않을 때



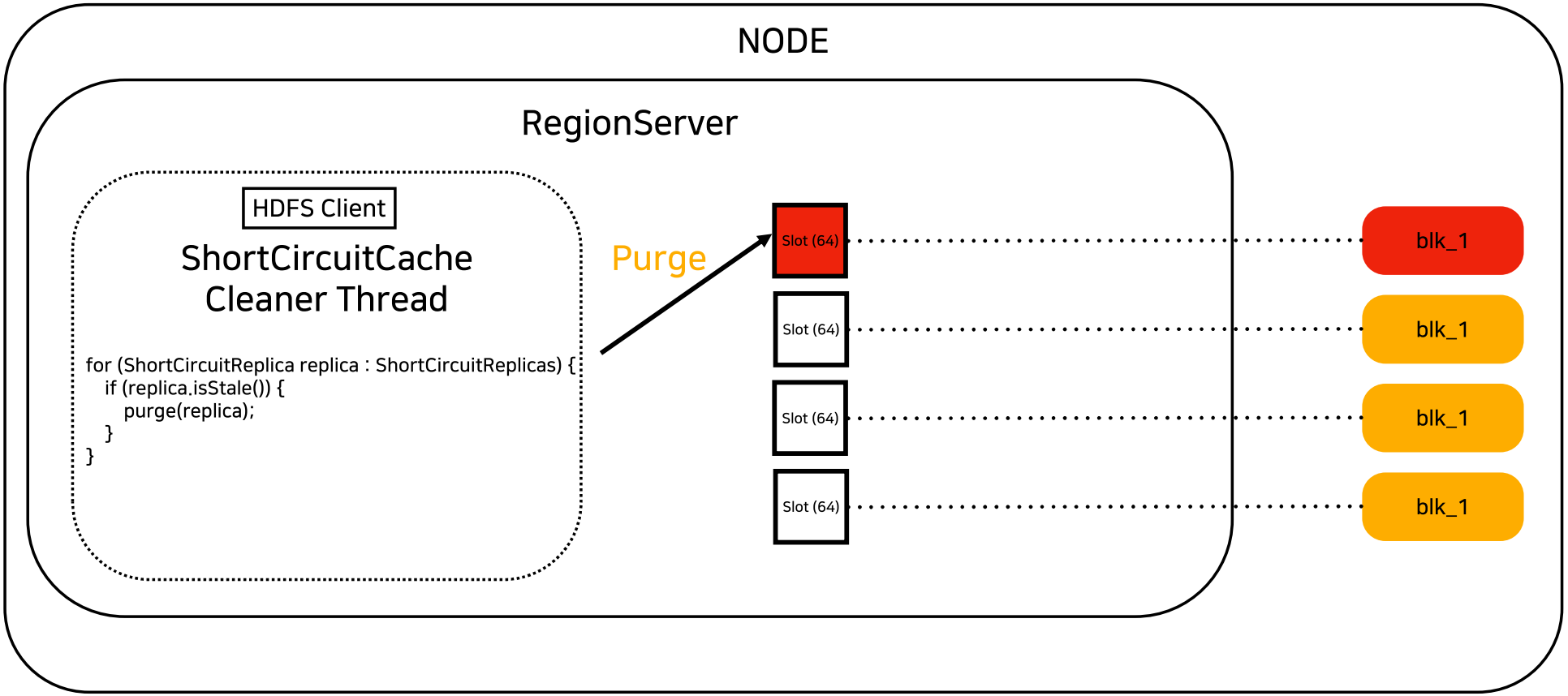
Block이 유효하지 않을 때



Lazy Purge


NODE
RegionServer
HDFS Client
If Access Block,
Check isValid()
Purge
Slot (64)
Slot (64)
Slot (64)
Slot (64)
blk_1
blk_1
blk_1
blk_1

Aggressive Purge



Aggressive Purge

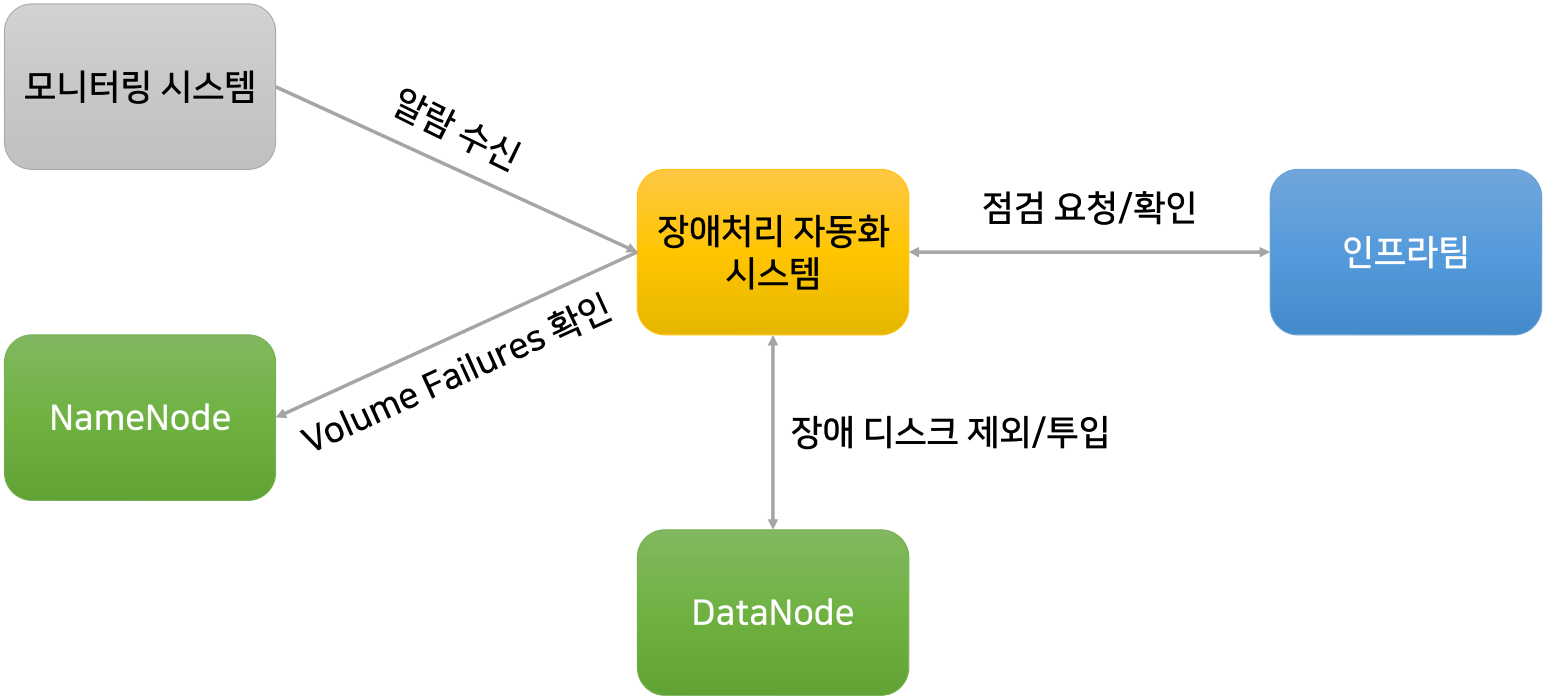
https://issues.apache.org/jira/browse/HDFS-12204

언마운트 불능 현상2

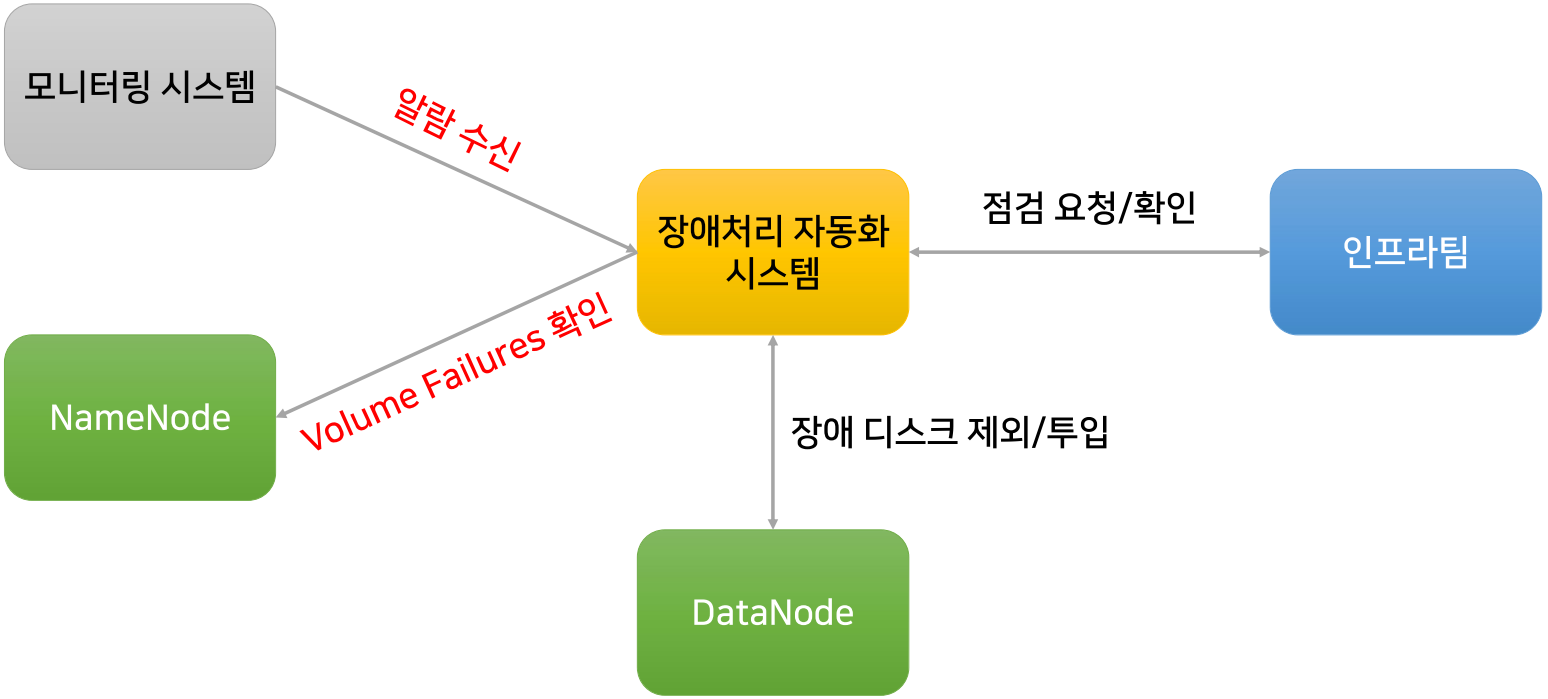
DEVIEW 2021

- HADOOP 3.0.0에서 발생
- Disk가 고장으로 메타데이터를 읽을 수 없을 때 볼륨 레퍼런스를 릴리즈 하지 않는 버그 (HDFS-15963)

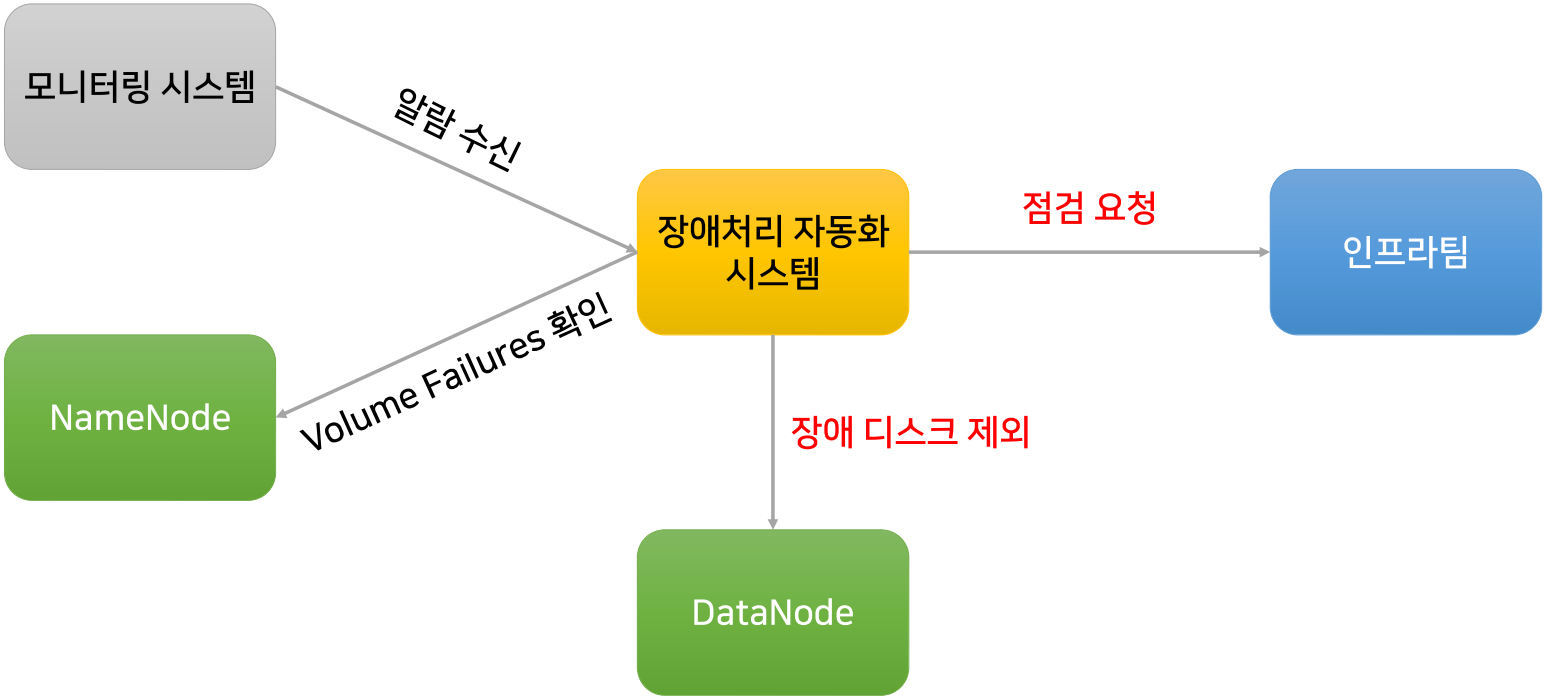
디스크 장애 처리 자동화


모니터링 시스템
알람 수신
장애처리 자동화 시스템
점검 요청/확인
인프라팀
NameNode
Volume Failures 확인
장애 디스크 제외/투입
DataNode

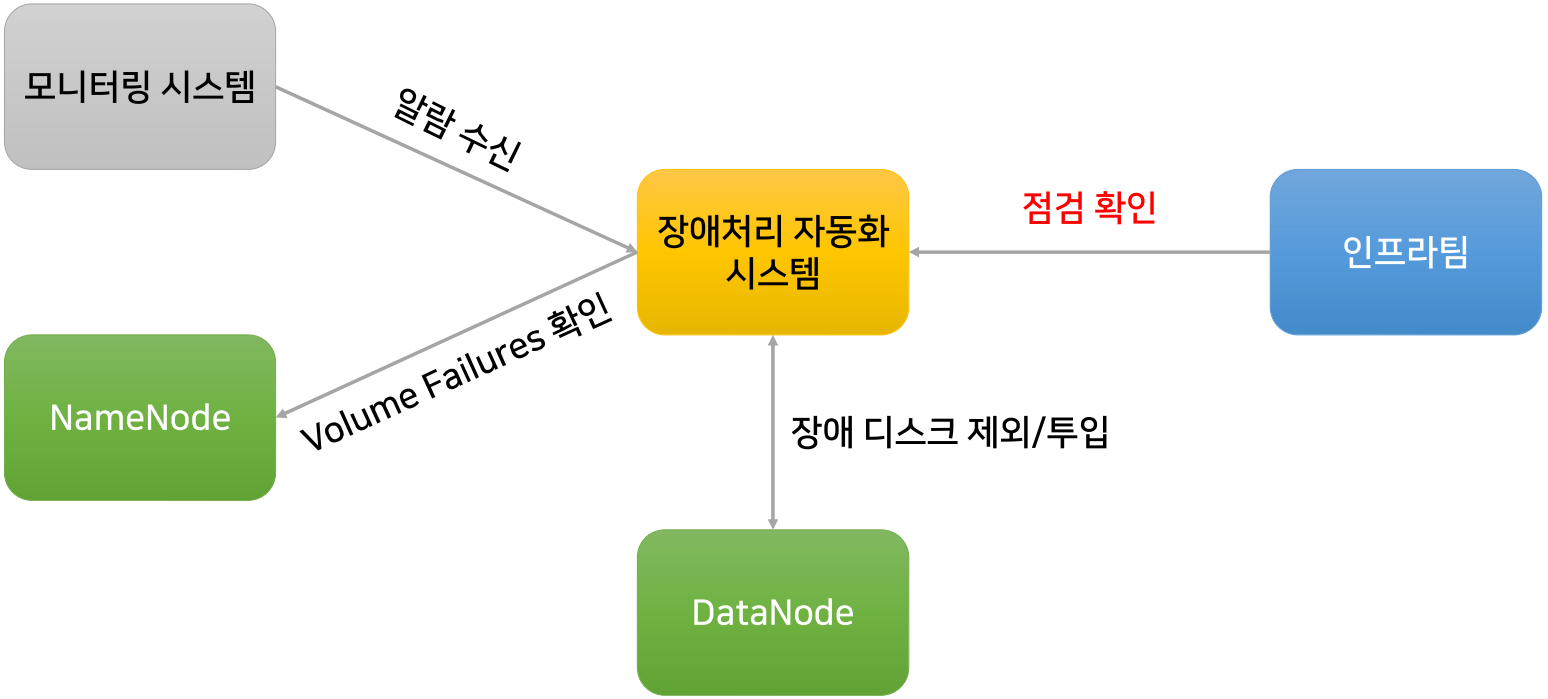
디스크 장애 처리 자동화


모니터링 시스템
알람 수신
장애처리 자동화 시스템
점검 요청/확인
인프라팀
NameNode
Volume Failures 확인
장애 디스크 제외/투입
DataNode

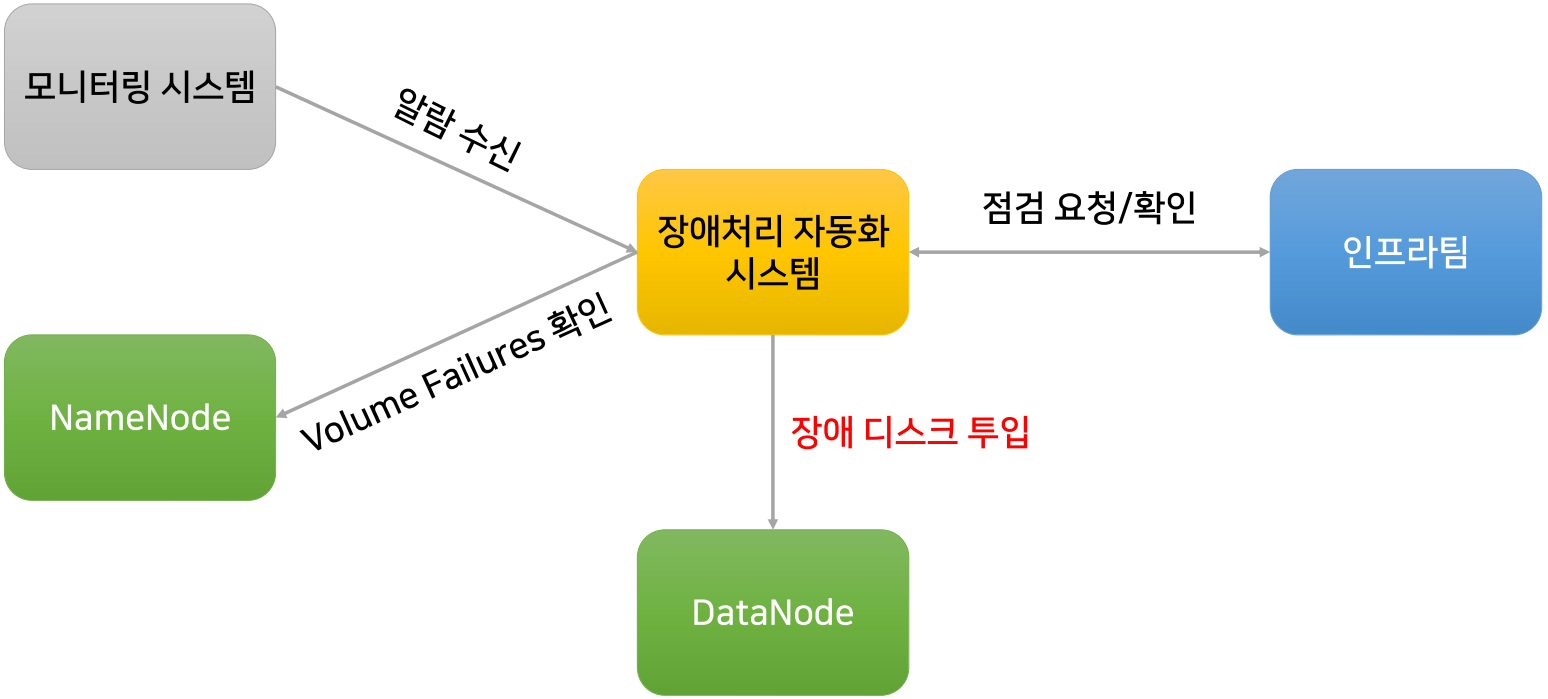
디스크 장애 처리 자동화


모니터링 시스템
알람 수신
장애처리 자동화 시스템
점검 요청
인프라팀
NameNode
Volume Failures 확인
장애 디스크 제외
DataNode

디스크 장애 처리 자동화


모니터링 시스템
알람 수신
장애처리 자동화 시스템
점검 확인
인프라팀
NameNode
Volume Failures 확인
장애 디스크 제외/투입
DataNode

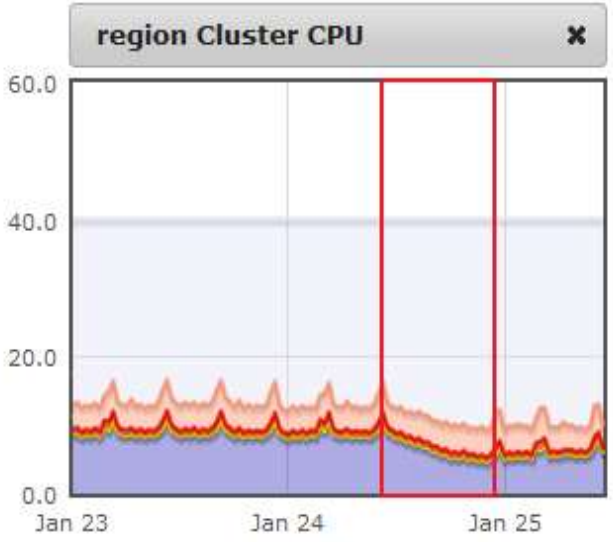
디스크 장애 처리 자동화


모니터링 시스템
알람 수신
장애처리 자동화 시스템
점검 요청/확인
인프라팀
NameNode
Volume Failures 확인
장애 디스크 투입
DataNode

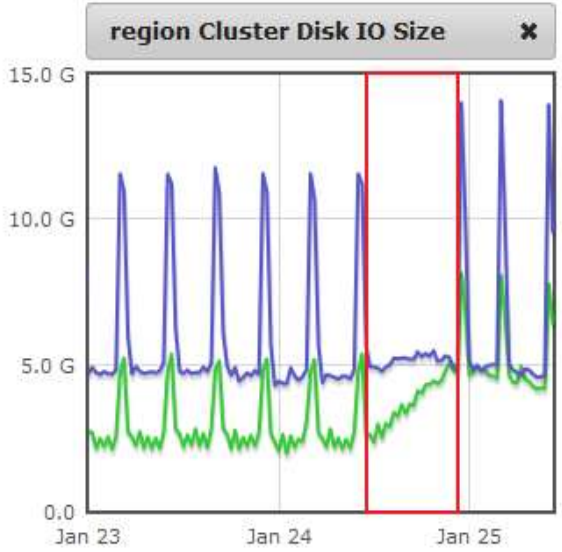
네이버 데이터 저장소에서 효과

네이버 데이터 저장소에서 효과

HBase Regionserver 백여대
01/24 11:00 경 ShortCircuit 설정 적용 (Rolling restart)



파란색 user cpu가 40% 줄어듬

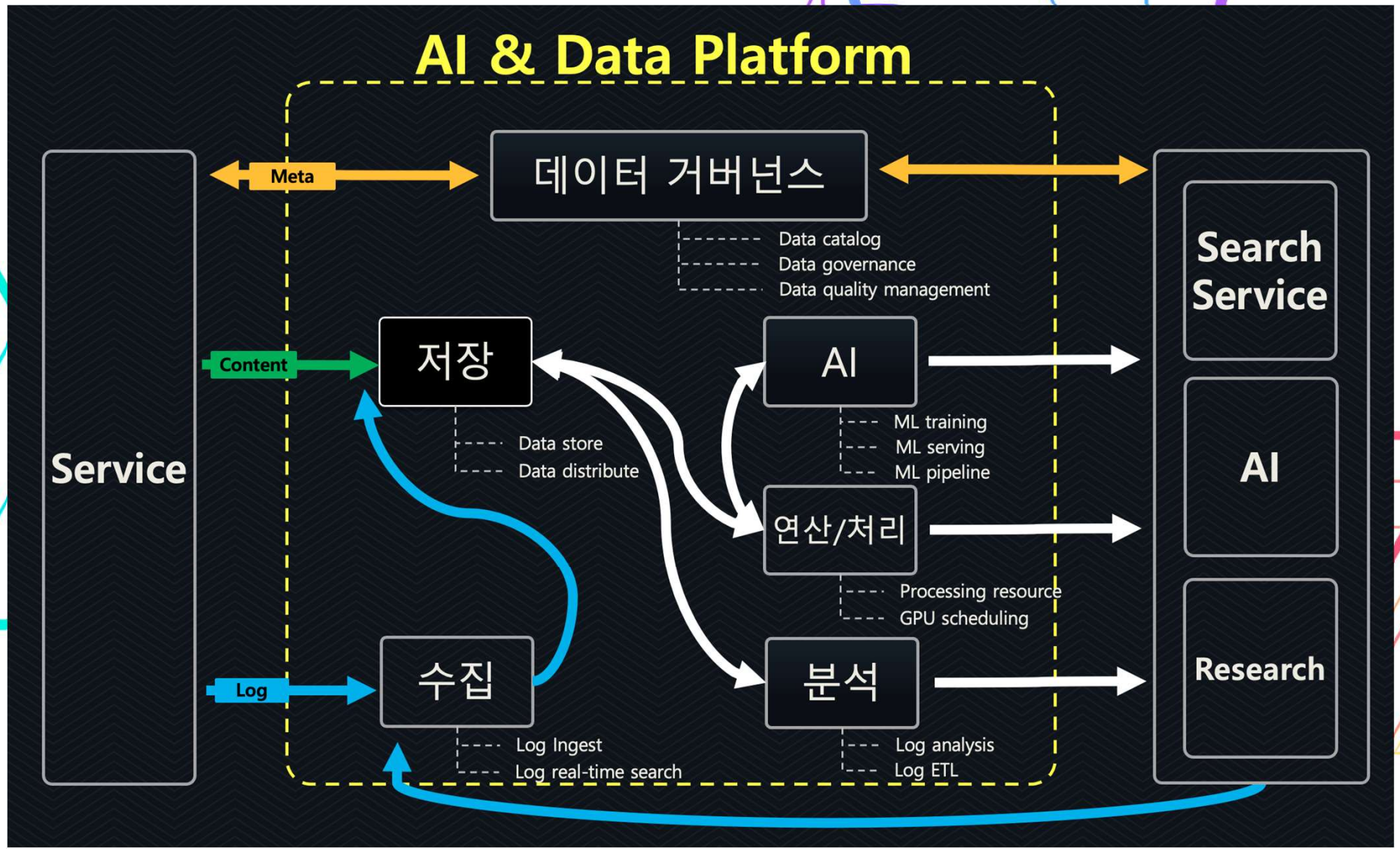


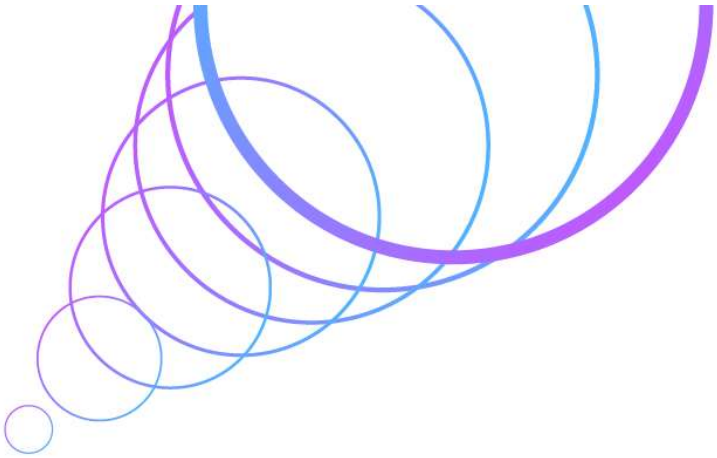
녹색 Disk Read가 2G/s 늘어남

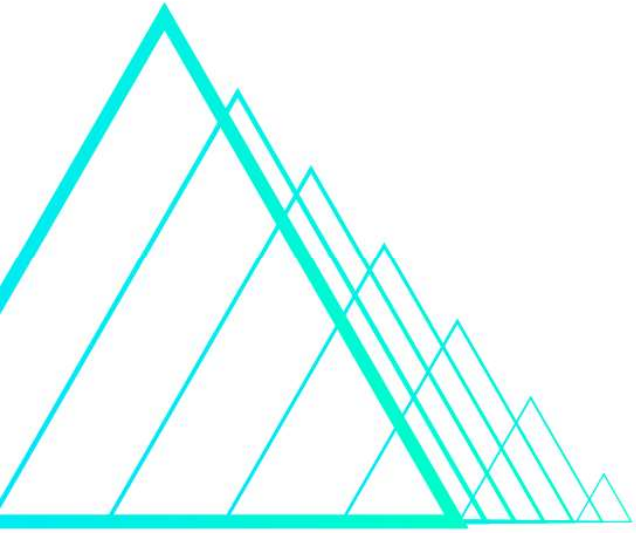
We're hiring

AI & Data Platform 소개
https://naver-career.gitbook.io/kr/service/search/ai-and-data-platform

경력 등록
https://d2.naver.com/news/7591059


AI & Data Platform
Service
Meta
데이터 거버넌스
Data catalog
Data governance
Data quality management
Content
저장
Data store
Data distribute
AI
ML training
ML serving
ML pipeline
연산/처리
Processing resource
GPU scheduling
분석
Log analysis
Log ETL
Log
수집
Log Ingest
Log real-time search
Search Service
AI
Research





Thank You