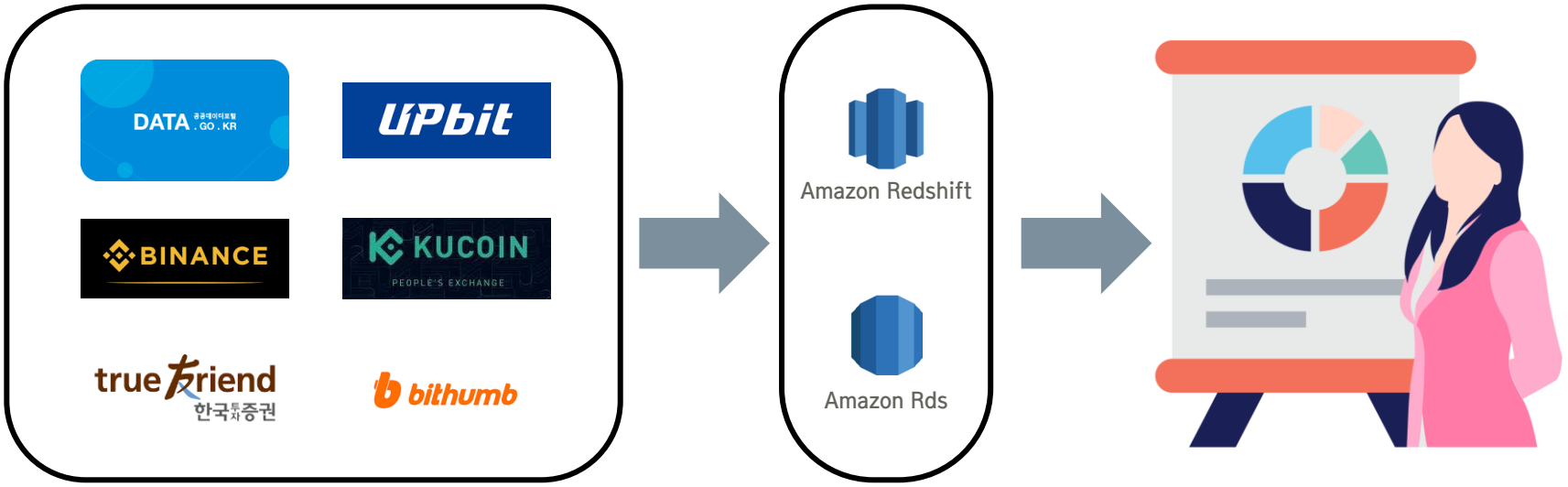
금융 데이터 수집을 통한
데이터 분석환경 구축

1팀 2조

김수민 김형준 이수영 조주혜 한기호

CONTENTS

01 프로젝트 목표



금융 데이터
통합

편리한 데이터
분석 가능

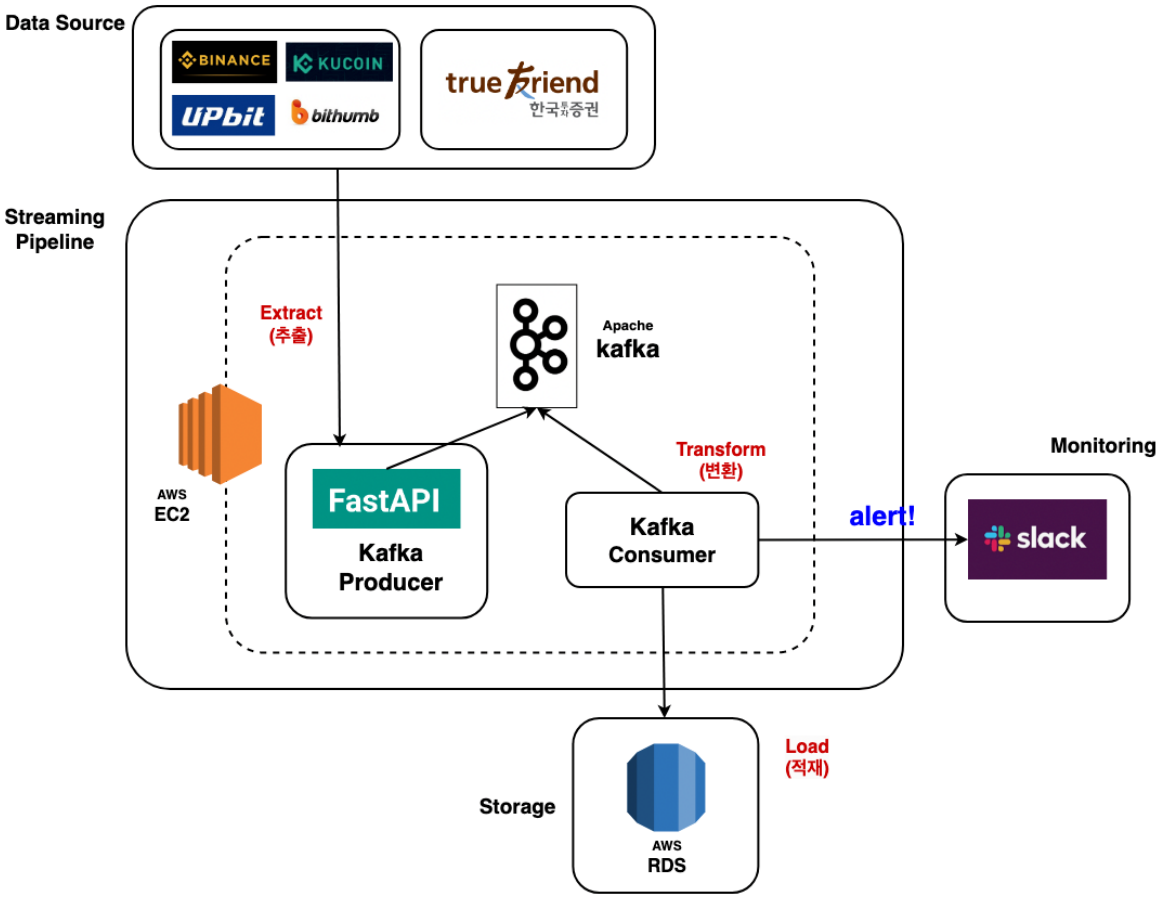
02 사용기술

스케쥴러	 Amazon MWAA
데이터 수집 도구	 AWS Lambda  FastAPI
실시간 데이터 처리	 kafka  Amazon EC2
데이터 전처리	 Amazon EMR (spark)
DB	 Amazon S3 (DataLake)  Amazon Redshift (DataWarehouse)  Amazon Rds (MySQL)
시각화 도구	 Amazon Quicksight
기타도구	 CLI  Slack  notion  Jira  Gather  Github  IAM

03 구축과정

- 실시간처리

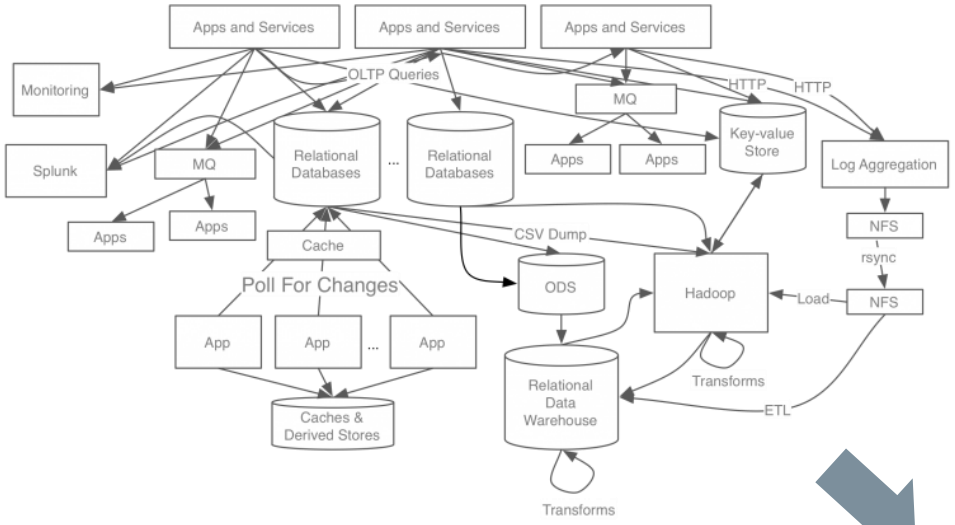
실시간처리 아키텍처



03 구축과정

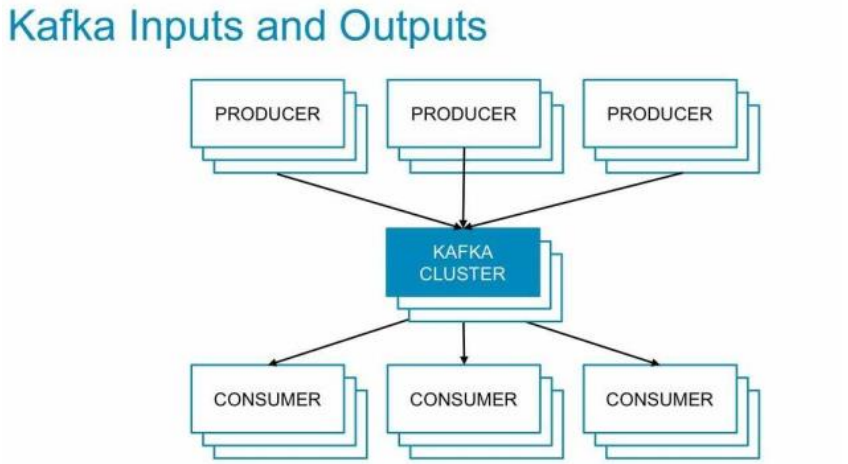
- 실시간처리

kafka 도입 이유



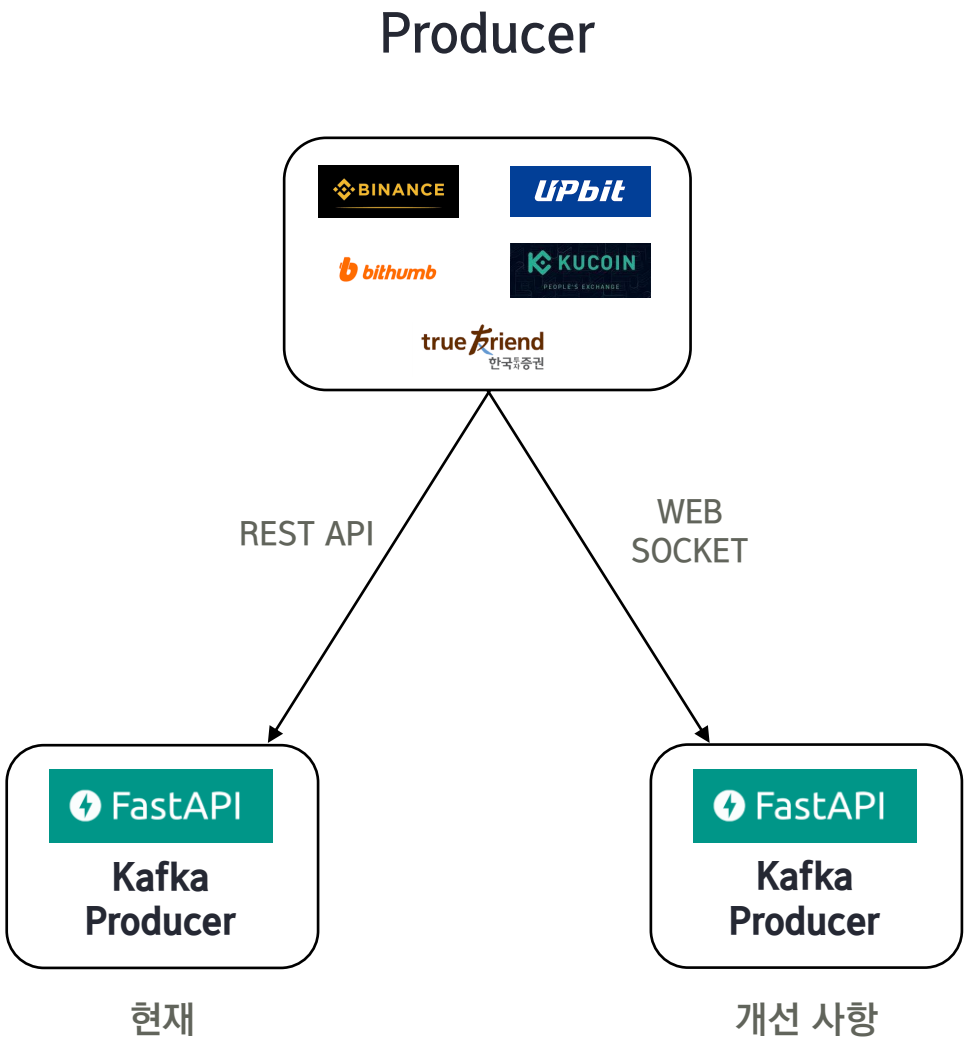
kafka 도입 전

kafka 도입 후



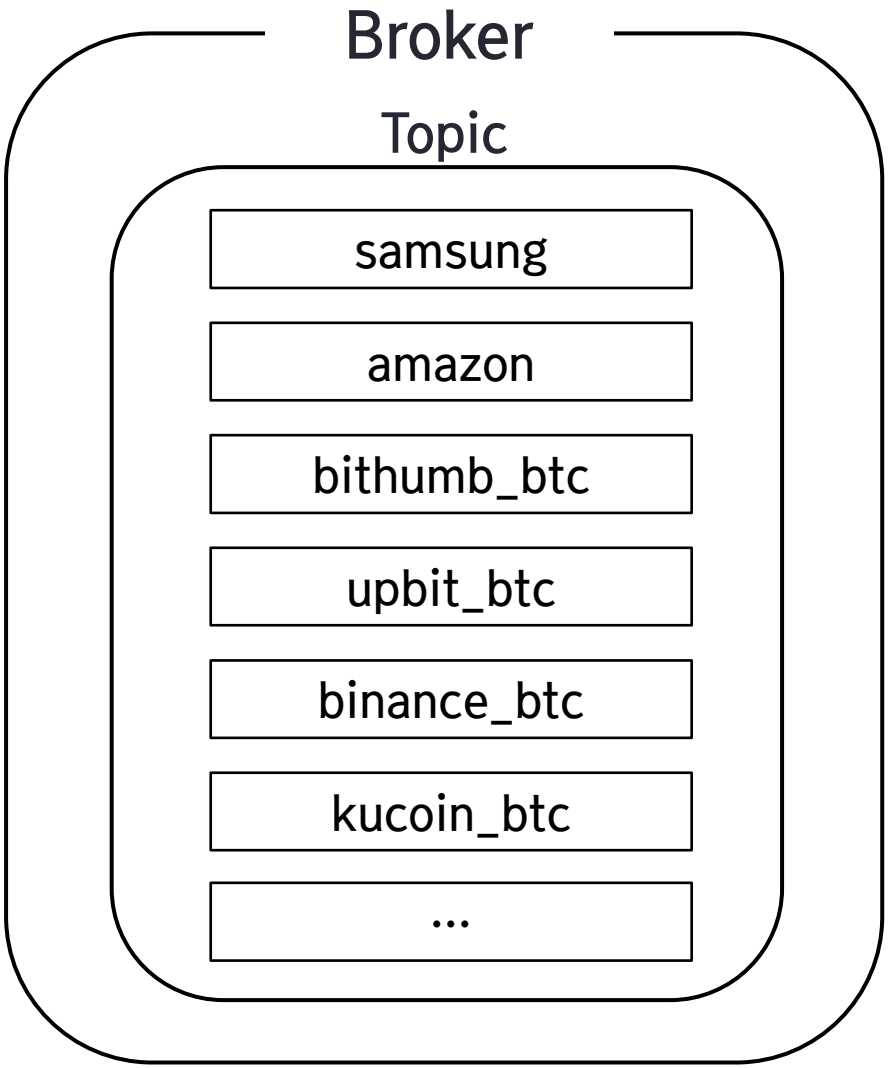
03 구축과정

- 실시간처리



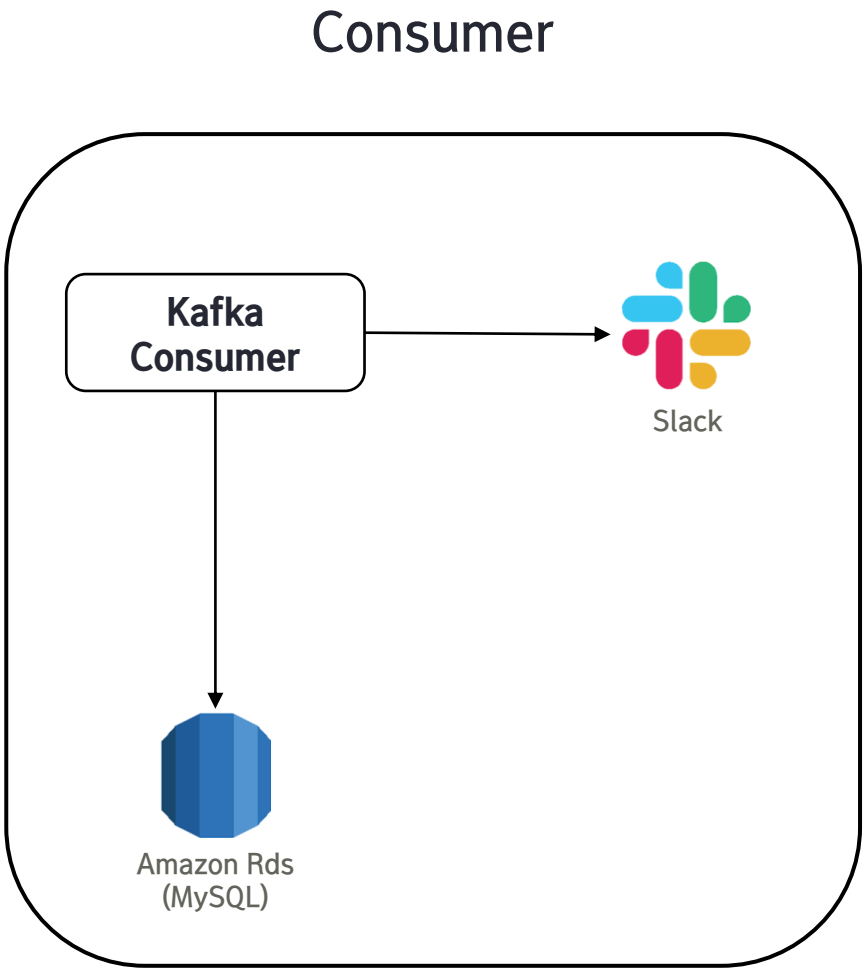
03 구축과정

- 실시간처리



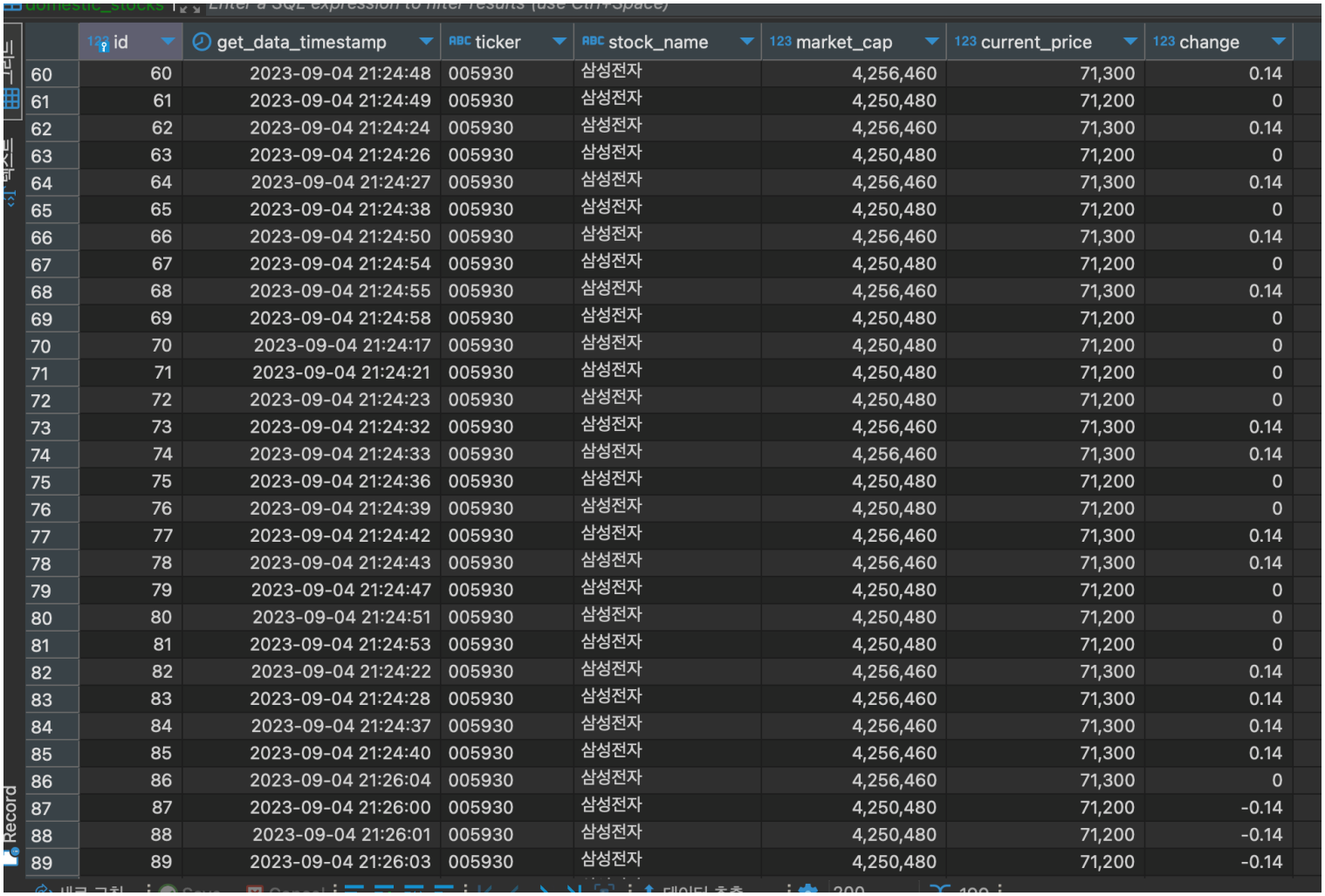
03 구축과정

- 실시간처리



03 구축과정

- 실시간처리

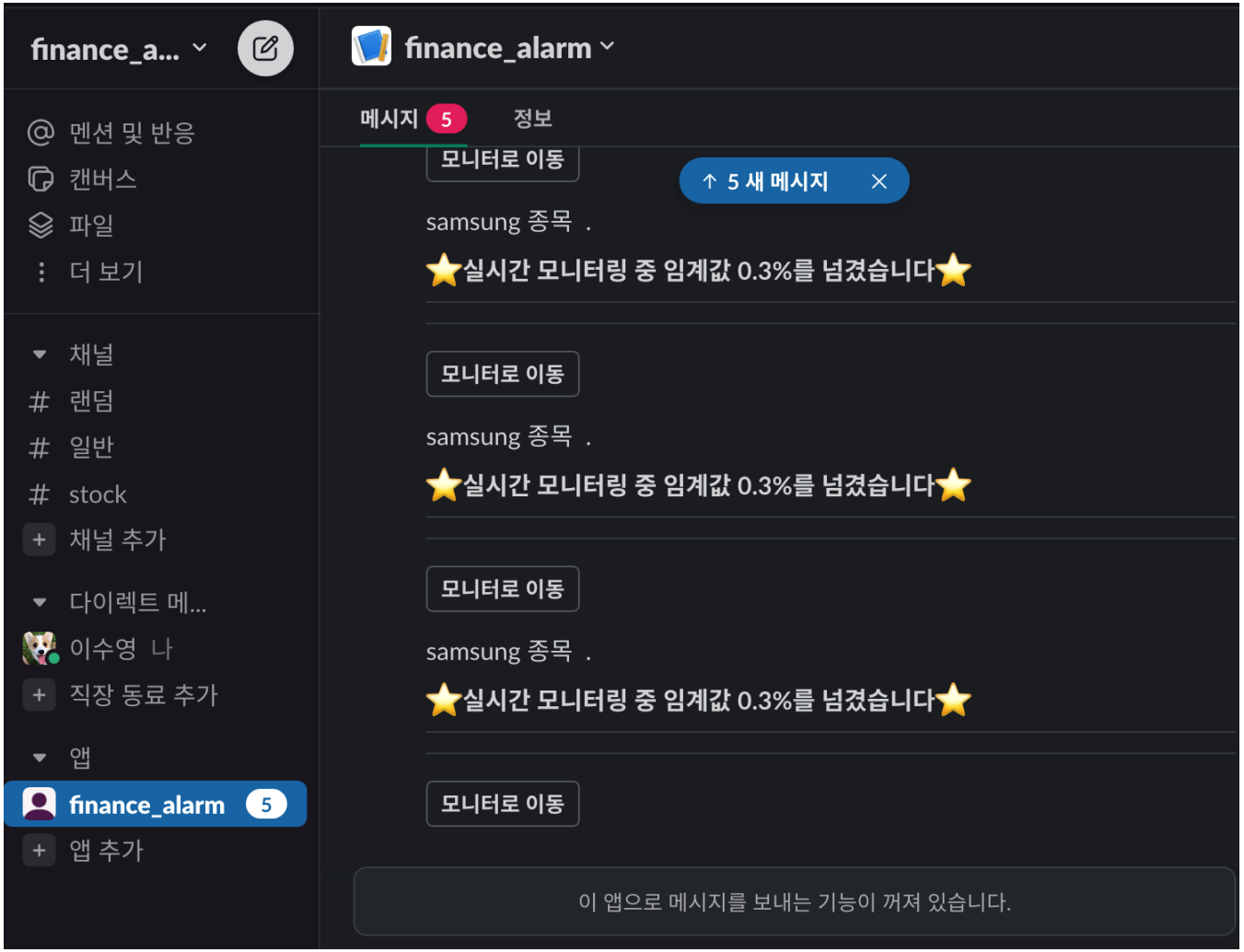


	id	get_data_timestamp	ticker	stock_name	market_cap	current_price	change
60	60	2023-09-04 21:24:48	005930	삼성전자	4,256,460	71,300	0.14
61	61	2023-09-04 21:24:49	005930	삼성전자	4,250,480	71,200	0
62	62	2023-09-04 21:24:24	005930	삼성전자	4,256,460	71,300	0.14
63	63	2023-09-04 21:24:26	005930	삼성전자	4,250,480	71,200	0
64	64	2023-09-04 21:24:27	005930	삼성전자	4,256,460	71,300	0.14
65	65	2023-09-04 21:24:38	005930	삼성전자	4,250,480	71,200	0
66	66	2023-09-04 21:24:50	005930	삼성전자	4,256,460	71,300	0.14
67	67	2023-09-04 21:24:54	005930	삼성전자	4,250,480	71,200	0
68	68	2023-09-04 21:24:55	005930	삼성전자	4,256,460	71,300	0.14
69	69	2023-09-04 21:24:58	005930	삼성전자	4,250,480	71,200	0
70	70	2023-09-04 21:24:17	005930	삼성전자	4,250,480	71,200	0
71	71	2023-09-04 21:24:21	005930	삼성전자	4,250,480	71,200	0
72	72	2023-09-04 21:24:23	005930	삼성전자	4,250,480	71,200	0
73	73	2023-09-04 21:24:32	005930	삼성전자	4,256,460	71,300	0.14
74	74	2023-09-04 21:24:33	005930	삼성전자	4,256,460	71,300	0.14
75	75	2023-09-04 21:24:36	005930	삼성전자	4,250,480	71,200	0
76	76	2023-09-04 21:24:39	005930	삼성전자	4,250,480	71,200	0
77	77	2023-09-04 21:24:42	005930	삼성전자	4,256,460	71,300	0.14
78	78	2023-09-04 21:24:43	005930	삼성전자	4,256,460	71,300	0.14
79	79	2023-09-04 21:24:47	005930	삼성전자	4,250,480	71,200	0
80	80	2023-09-04 21:24:51	005930	삼성전자	4,250,480	71,200	0
81	81	2023-09-04 21:24:53	005930	삼성전자	4,250,480	71,200	0
82	82	2023-09-04 21:24:22	005930	삼성전자	4,256,460	71,300	0.14
83	83	2023-09-04 21:24:28	005930	삼성전자	4,256,460	71,300	0.14
84	84	2023-09-04 21:24:37	005930	삼성전자	4,256,460	71,300	0.14
85	85	2023-09-04 21:24:40	005930	삼성전자	4,256,460	71,300	0.14
86	86	2023-09-04 21:26:04	005930	삼성전자	4,256,460	71,300	0
87	87	2023-09-04 21:26:00	005930	삼성전자	4,250,480	71,200	-0.14
88	88	2023-09-04 21:26:01	005930	삼성전자	4,250,480	71,200	-0.14
89	89	2023-09-04 21:26:03	005930	삼성전자	4,250,480	71,200	-0.14

삼성전자 테이블

03 구축과정

- 실시간처리

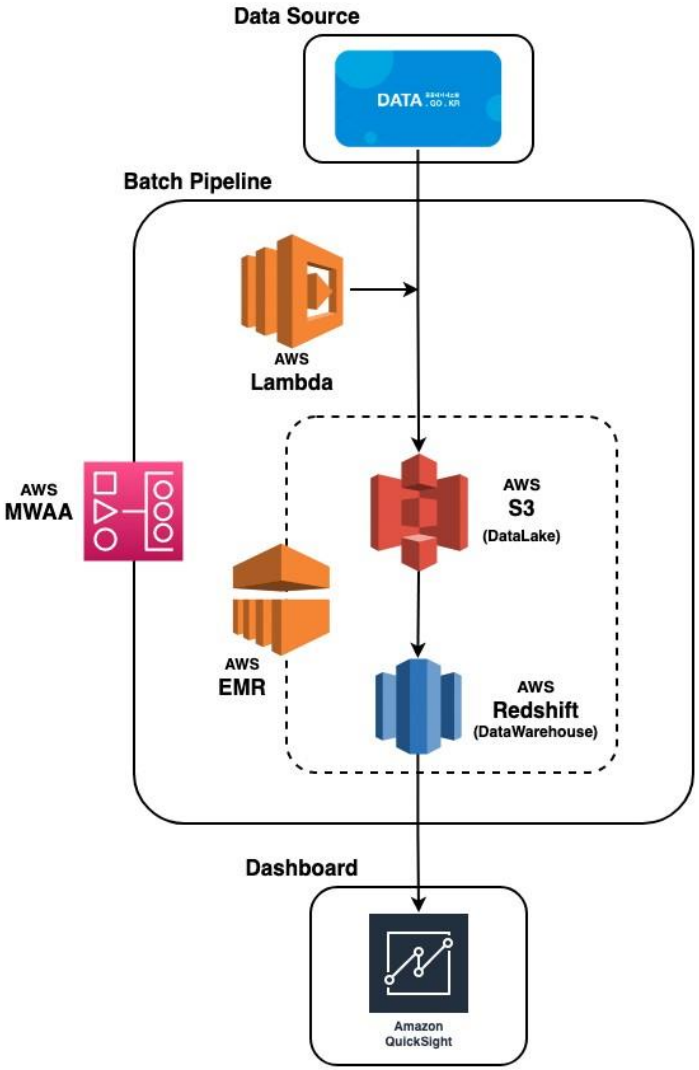


Slack에 변동률 알림

03 구축과정

- 배치처리

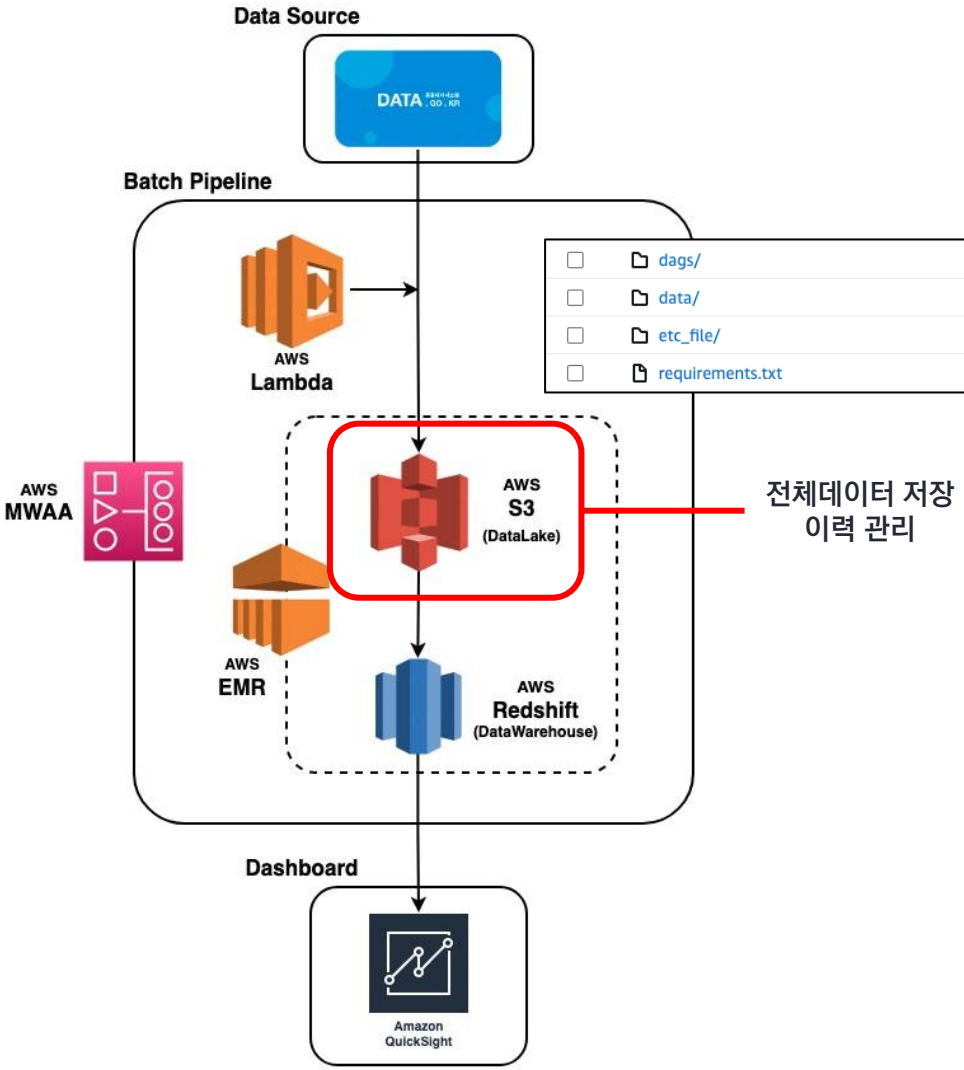
배치처리 아키텍처



03 구축과정

- 배치처리

배치처리 아키텍처



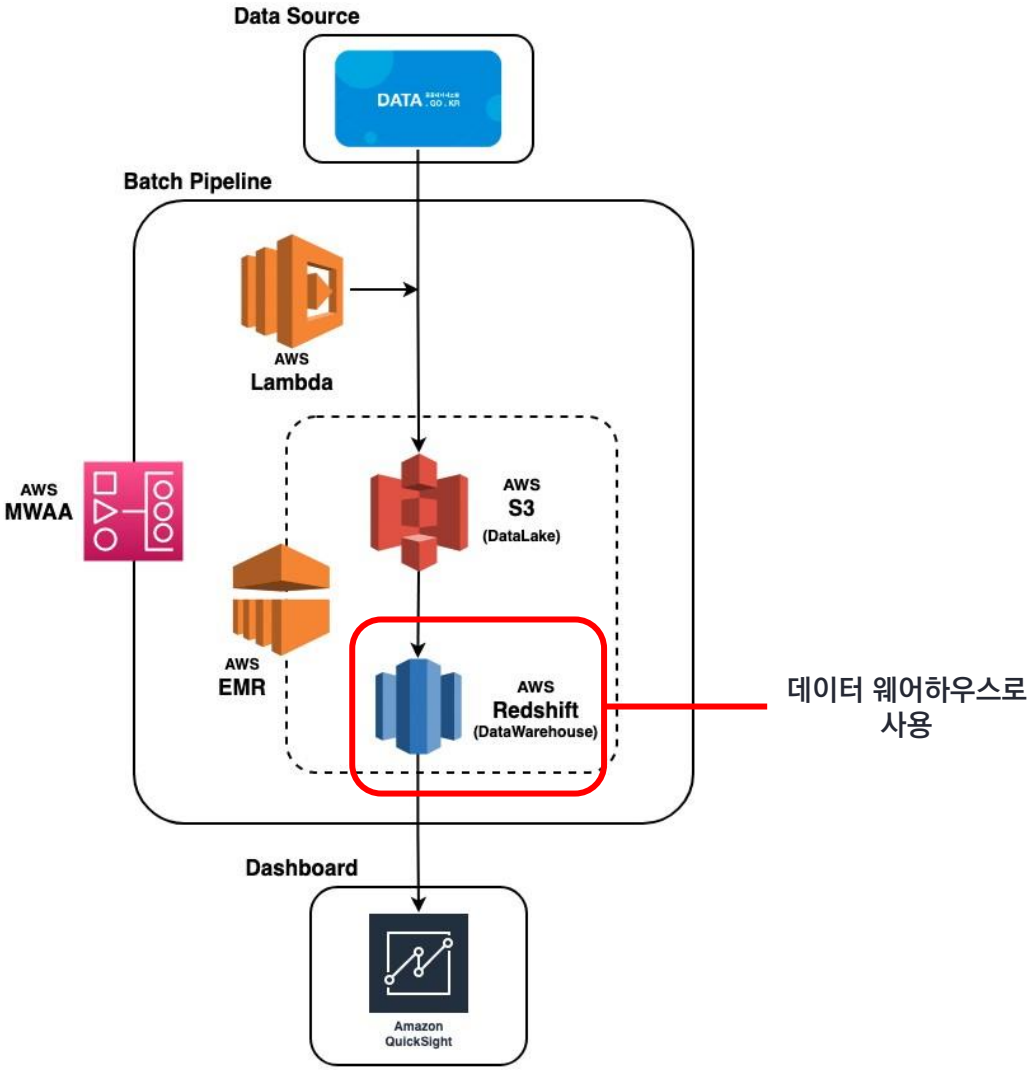
☐	dags/	폴더
☐	data/	폴더
☐	etc_file/	폴더
☐	requirements.txt	txt

전체데이터 저장
이력 관리

03 구축과정

- 배치처리

배치처리 아키텍처



03 구축과정

- 배치처리

스케쥴러

Amazon EC2

장점
- 비용 효율성
- 유연성

단점
- 초기세팅 필요
- 관리 어려움

Amazon EKS

장점
- 뛰어난 보안성
- 높은 이용률

단점
- 러닝커브가 있을 수 있음

Amazon MWAA

장점
- 편리한 관리 기능

단점
- DAG 수 제한
- 리소스 대비 많은 비용 부과

03 구축과정

– 배치처리

스케쥴러

Amazon EC2

장점
- 비용 효율성
- 유연성

단점
- 초기세팅 필요
- 관리 어려움

Amazon EKS

장점
- 뛰어난 보안성
- 높은 이용률

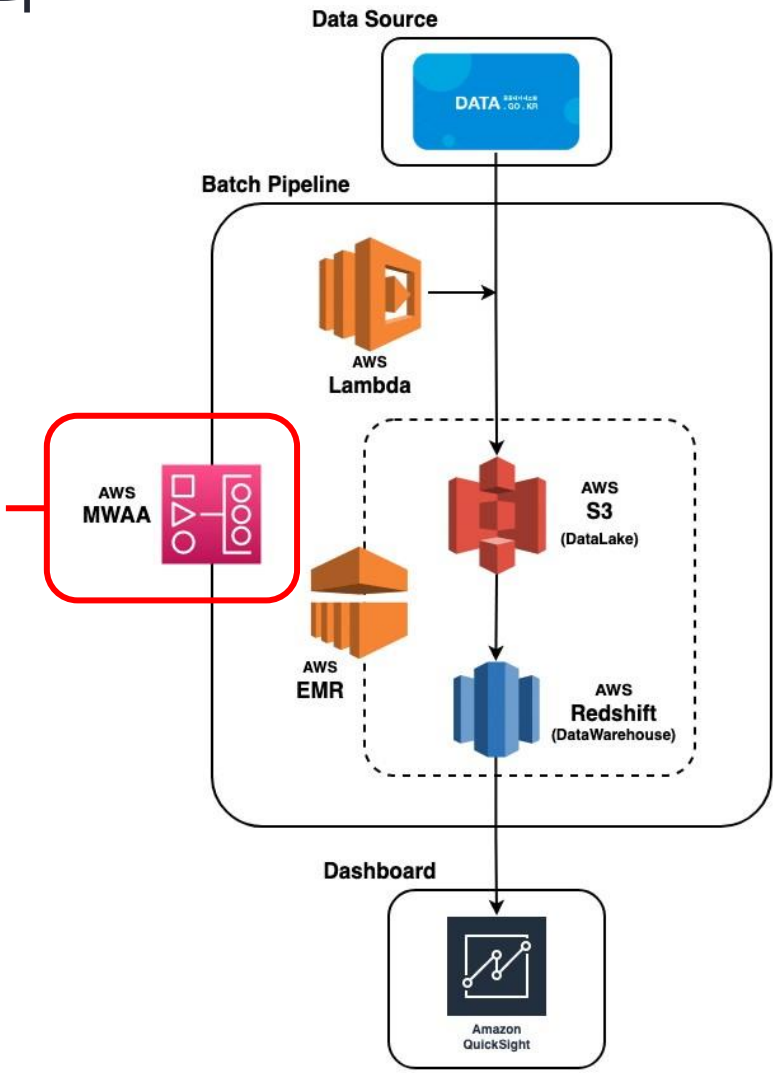
단점
- 러닝커브가 있을 수 있음

Amazon MWAA

장점
- 편리한 관리 기능

단점
- DAG 수 제한
- 리소스 대비 많은 비용 부과

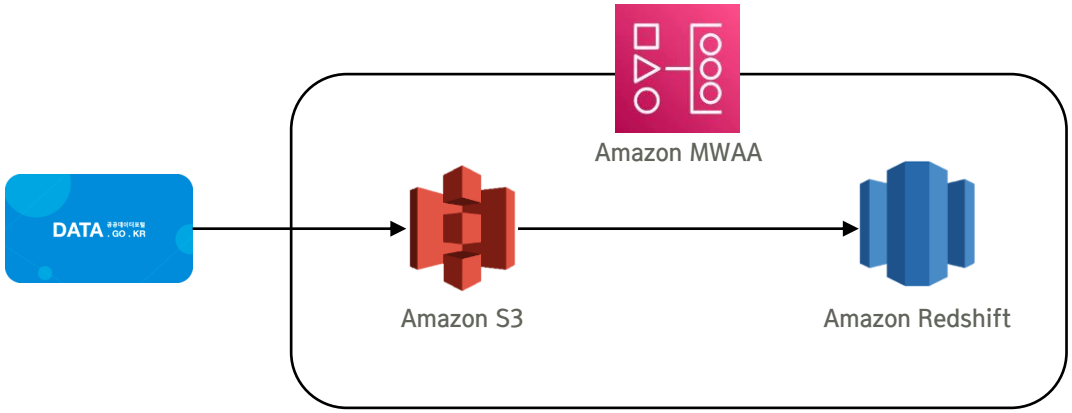
MWAA를 이용해
스케쥴링



03 구축과정

- 배치처리

초기 데이터 적재 방법

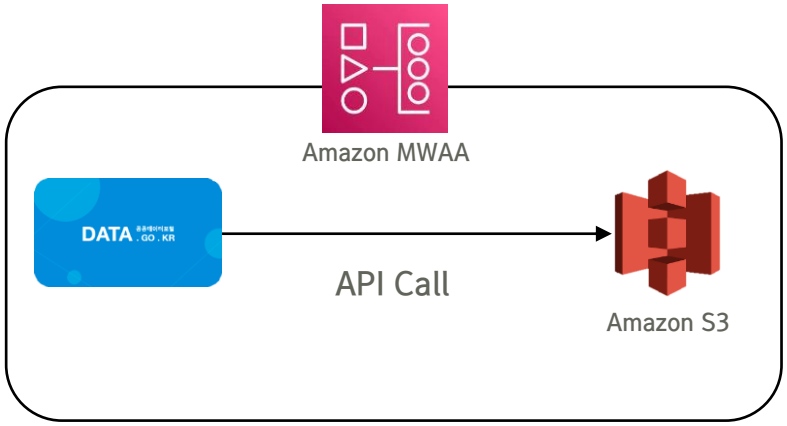


매일 full refresh

03 구축과정

– 배치처리

데이터 적재

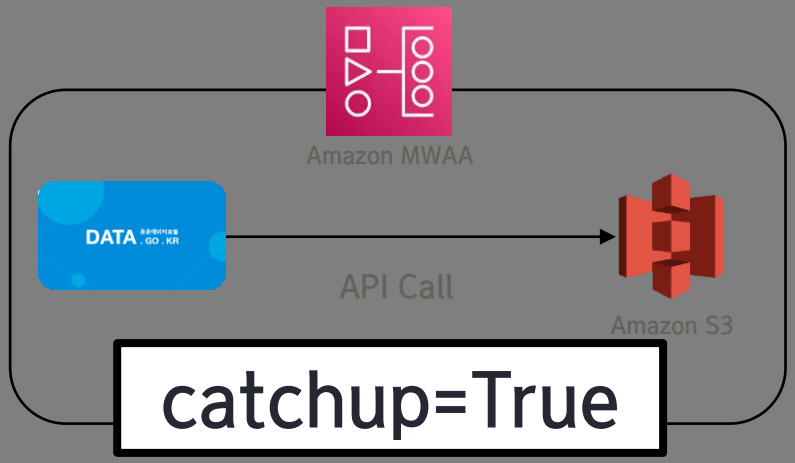


데이터가 많은 경우
문제가 생기게 됨!

03 구축과정

- 배치처리

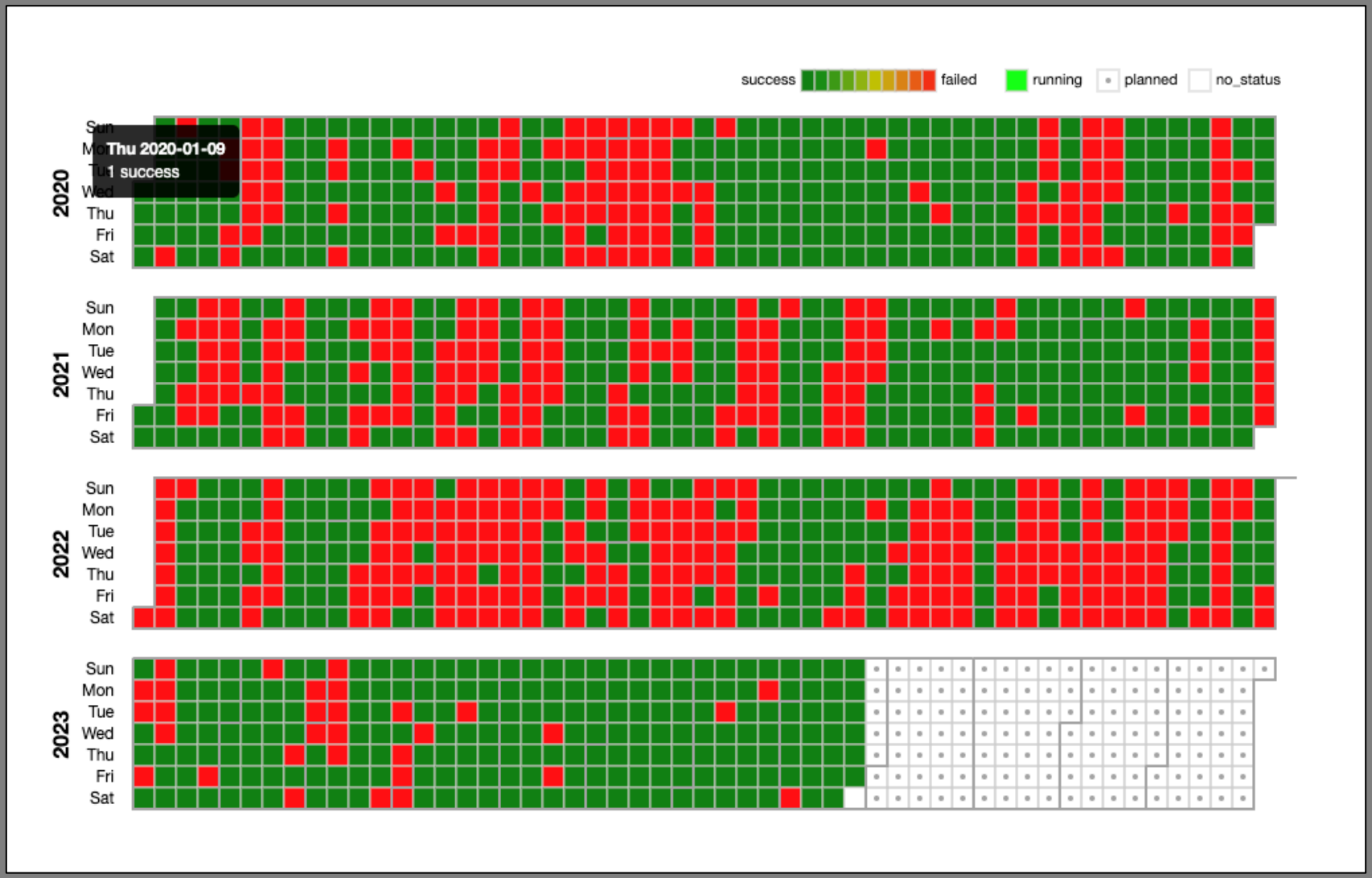
데이터 적재



데이터가 많은 경우
문제가 생기게 됨!

03 구축과정

- 배치처리


success
failed
running
planned
no_status
Thu 2020-01-09
1 success
2020
2021
2022
2023
Sun
Mon
Tue
Wed
Thu
Fri
Sat

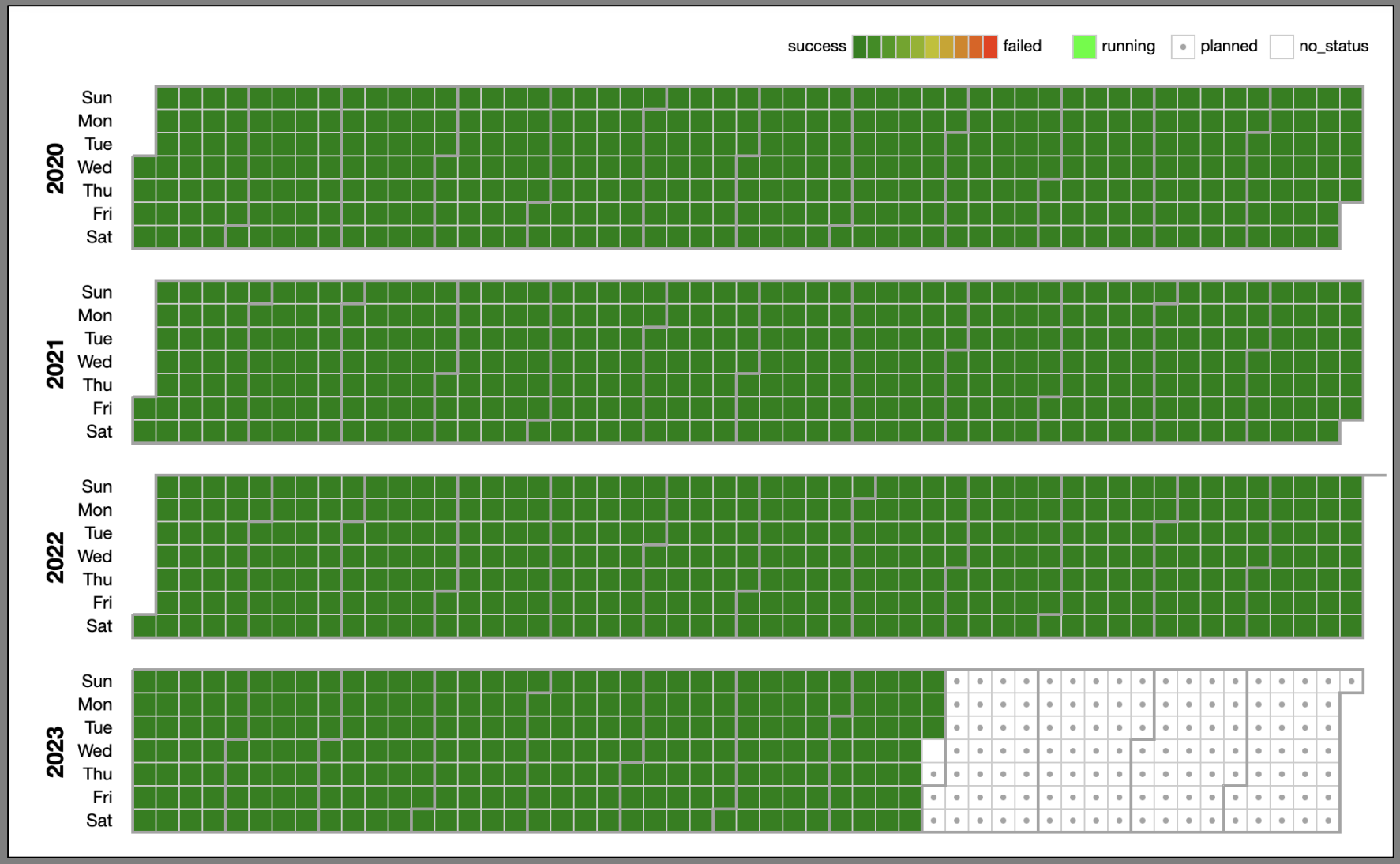
03 구축과정

- 배치처리


success
failed
running
planned
no_status
Thu 2020-01-09
1 success
2020
2021
2022
2023
Sun
Mon
Tue
Wed
Thu
Fri
Sat
max_active_runs=1

03 구축과정

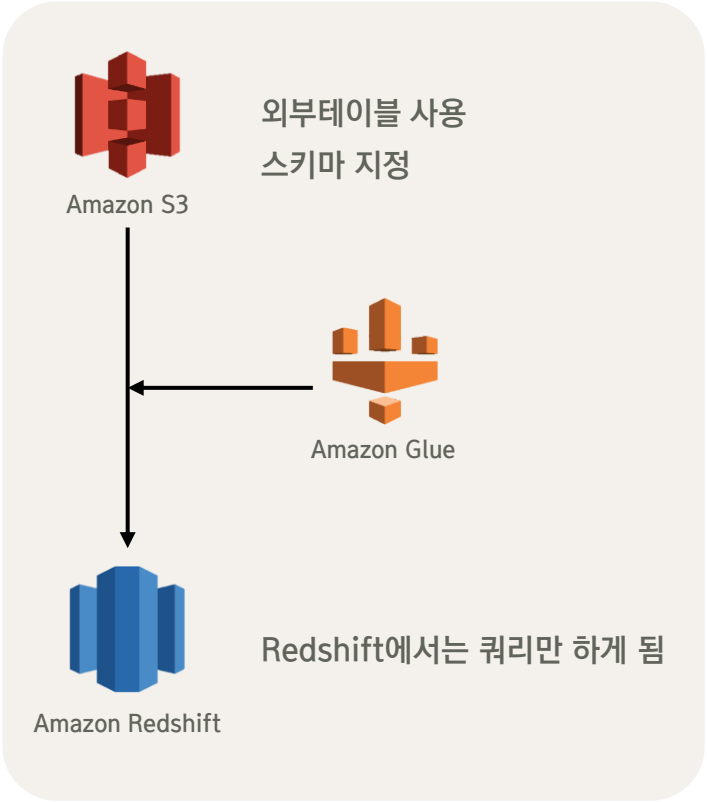
- 배치처리


success
failed
running
planned
no_status
2020
2021
2022
2023
Sun
Mon
Tue
Wed
Thu
Fri
Sat

03 구축과정

- 배치처리

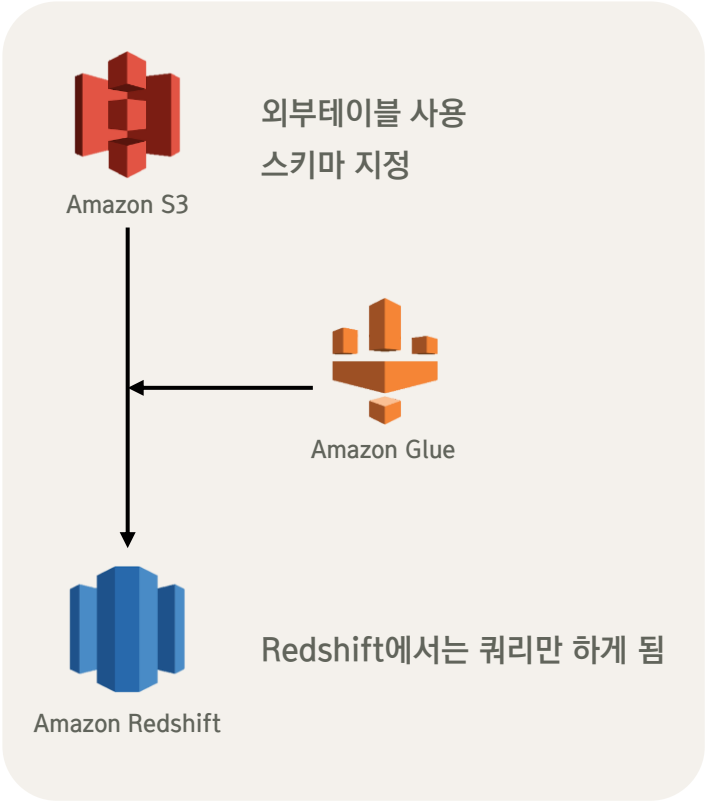
외부테이블 사용과 파티셔닝



03 구축과정

- 배치처리

외부테이블 사용과 파티셔닝



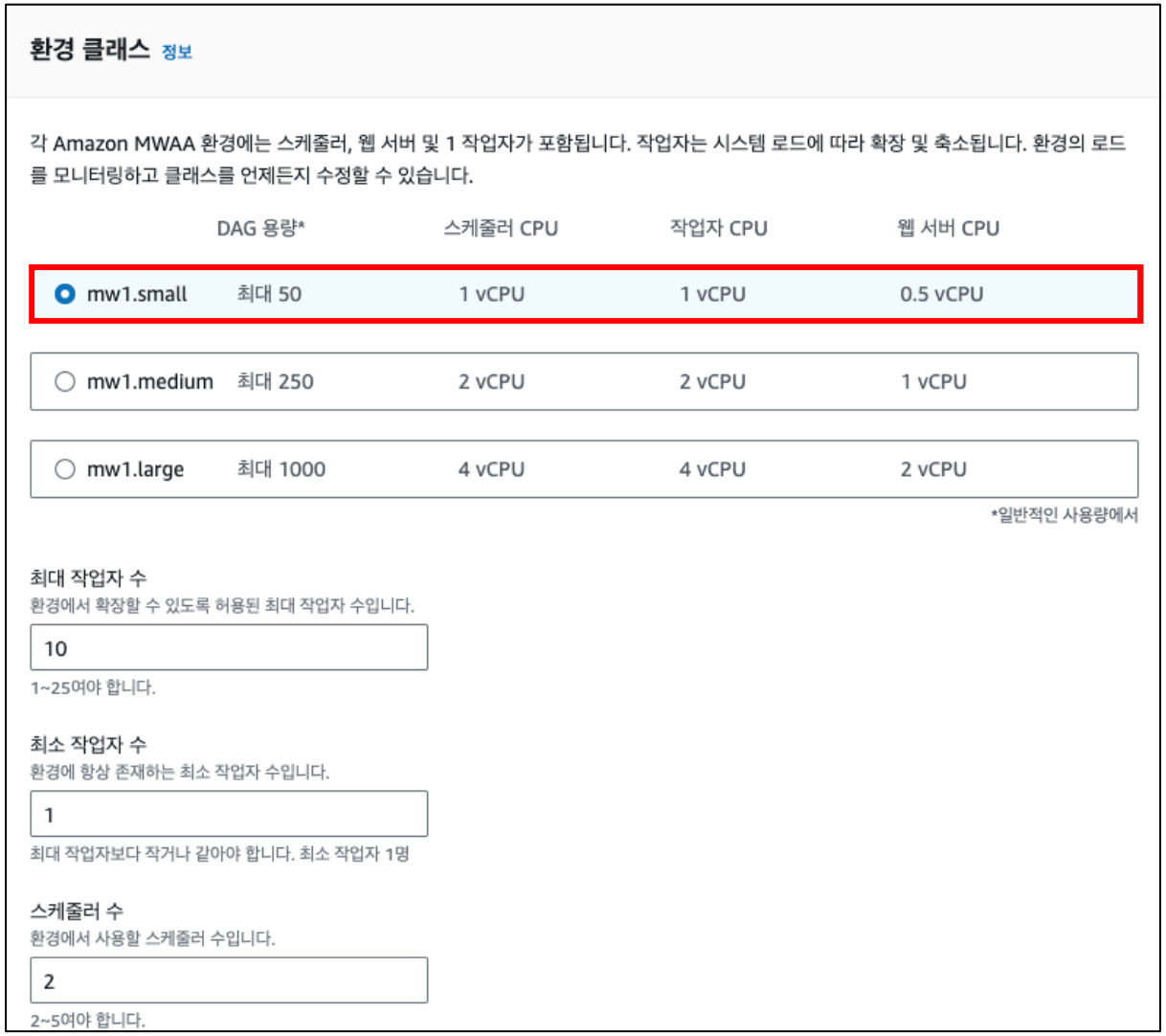
- dt=2022-11/
- dt=2022-12/
- dt=2023-01/
- dt=2023-02/
- dt=2023-03/
- dt=2023-04/
- dt=2023-05/
- dt=2023-06/
- dt=2023-07/
- dt=2023-08/

날짜 파티셔닝
→ 최적화

03 구축과정

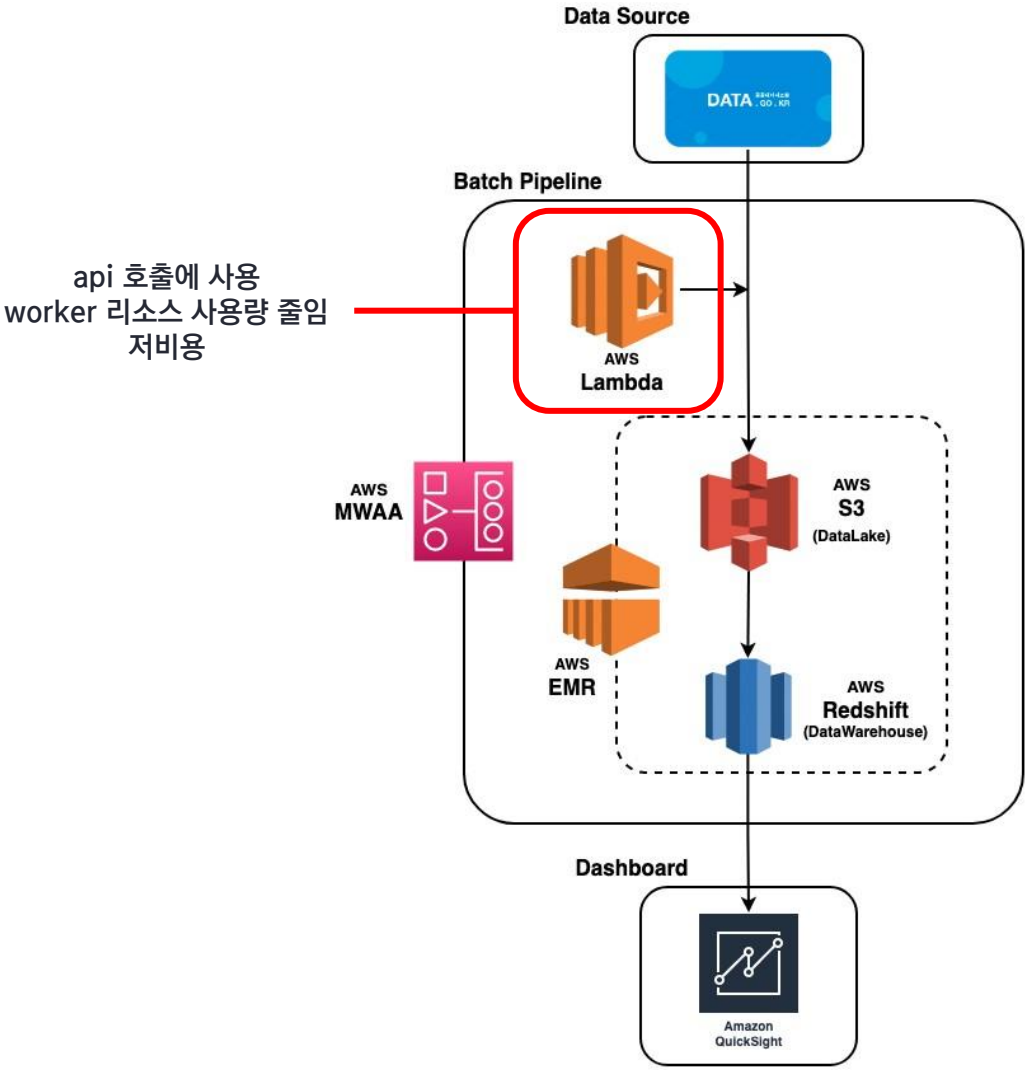
- 배치처리

MWAA 리소스 제한



03 구축과정

- 배치처리


Data Source
DATA
Batch Pipeline
api 호출에 사용
worker 리소스 사용량 줄임
저비용
AWS
Lambda
AWS
MWAA
AWS
S3
(DataLake)
AWS
EMR
AWS
Redshift
(DataWarehouse)
Dashboard
Amazon
QuickSight

03 구축과정

- 배치처리

api
worker

03 구축과정

- 배치처리

전처리 과정

Amazon EMR
(spark)

특징

- 대규모 데이터 처리에 적합
- Cluster 관리필요
- SparkStreaming 지원 O

Amazon Glue
(spark)

특징

- 스크립트 실행에 중점을 둠
- Cluster관리 불필요
- SparkStreaming 지원 X

03 구축과정

- 배치처리

전처리 과정

Amazon EMR
(spark)

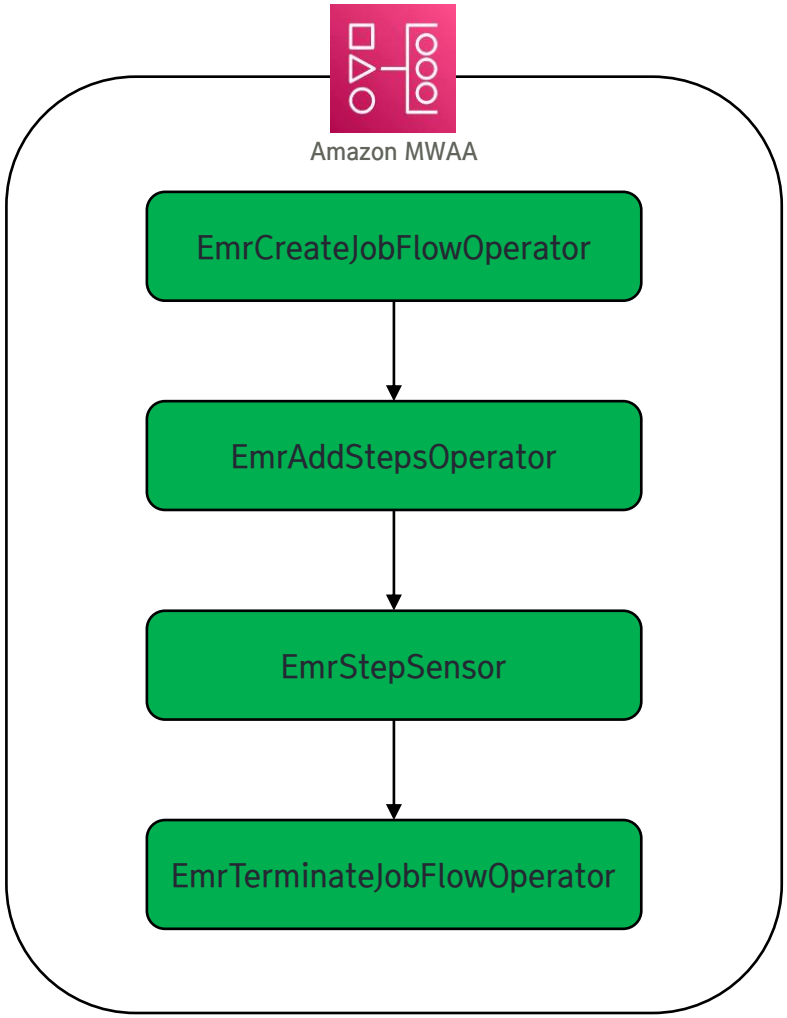
특징

- 대규모 데이터 처리에 적합
- Cluster 관리필요
- SparkStreaming 지원 O

Amazon Glue
(spark)

특징

- 스크립트 실행에 중점을 둠
- Cluster관리 불필요
- SparkStreaming 지원 X



03 구축과정

- 배치처리

전처리 과정

Amazon EMR
(spark)

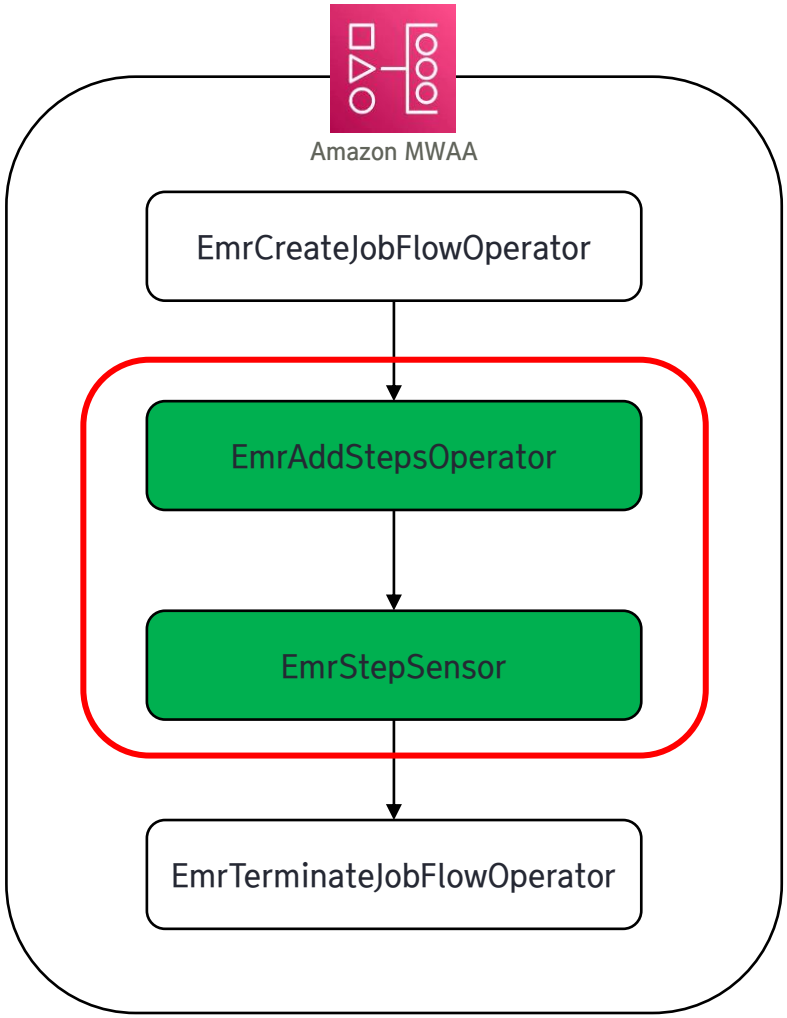
특징

- 대규모 데이터 처리에 적합
- Cluster 관리필요
- SparkStreaming 지원 O

Amazon Glue
(spark)

특징

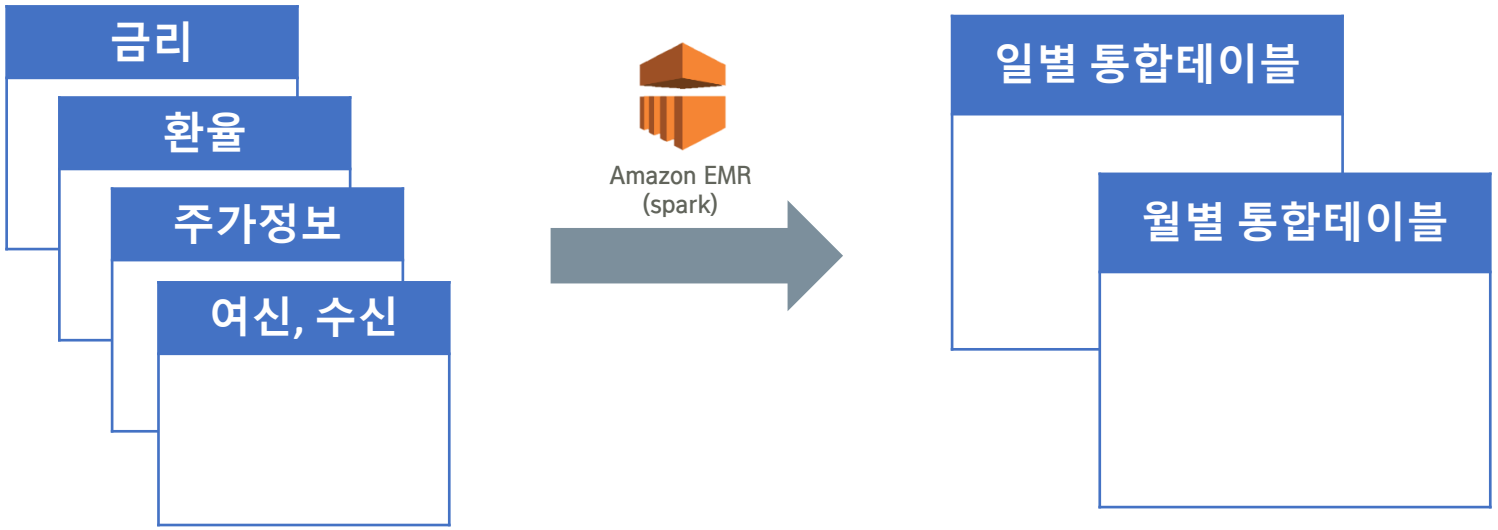
- 스크립트 실행에 중점을 둠
- Cluster관리 불필요
- SparkStreaming 지원 X



03 구축과정

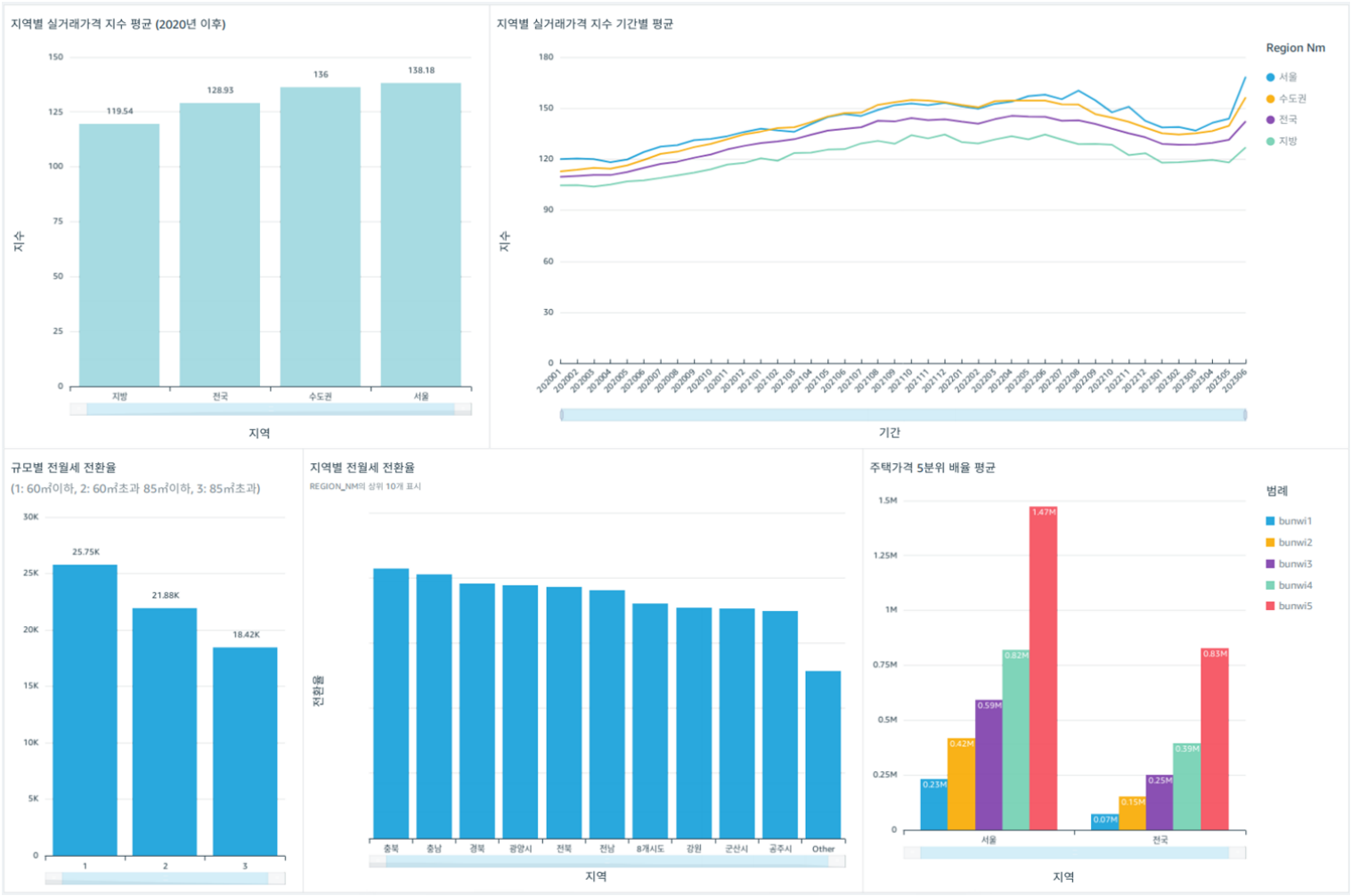
- 배치처리

전처리 과정



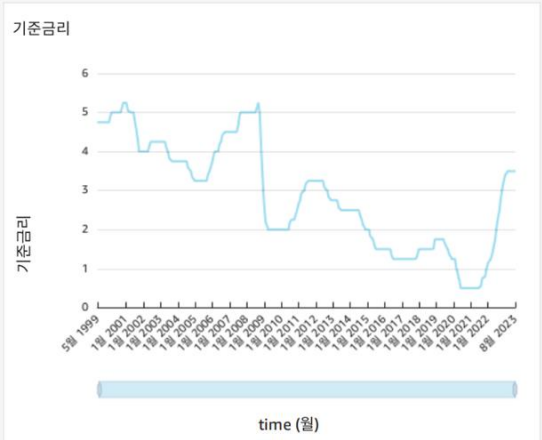
04 최종 시각화

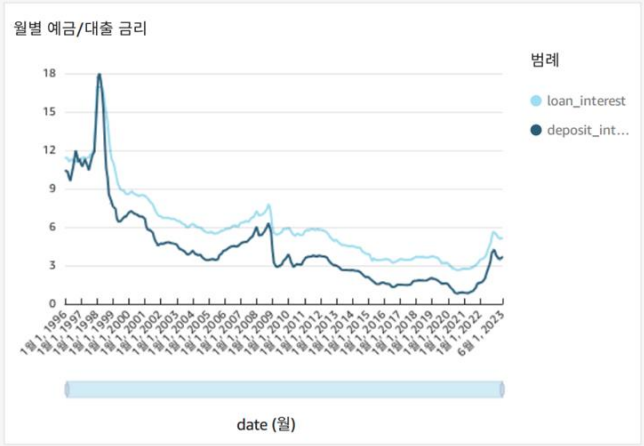
부동산 데이터 시각화

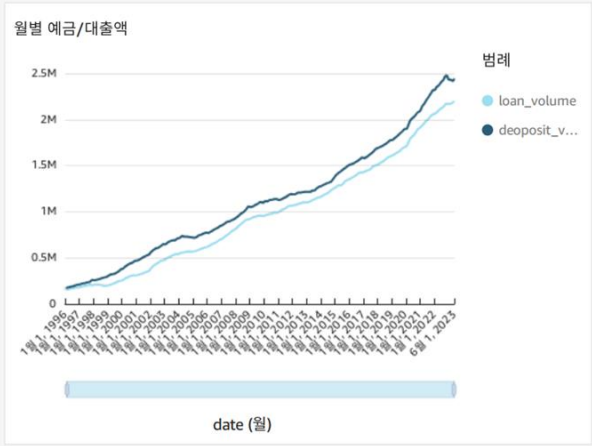


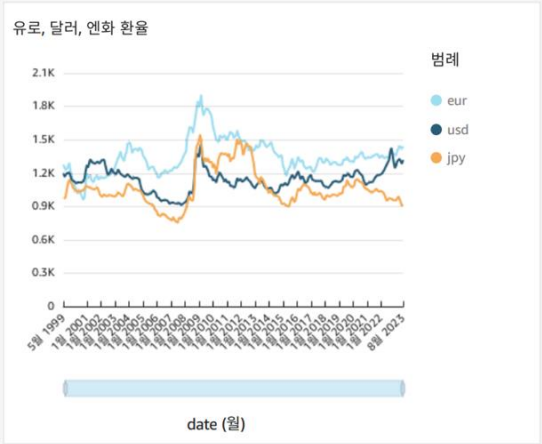
04 최종 시각화

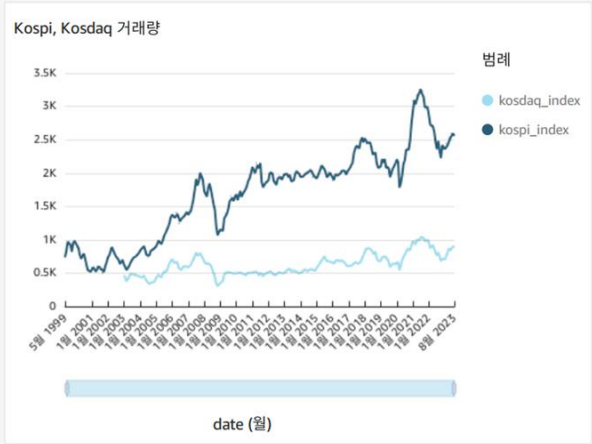
지표 데이터 시각화











Q & A