

# 통합 데이터 웨어하우스 [멀티 계정]

☰ 태그	AWS	ETL	데이터엔지니어	진행중
📅 날짜	@2025년 2월 1일			

개요

역할

구성

사용기술

멀티 계정 파이프라인

어려운점

회고

## 개요

본격적인 Data warehouse 구축을 위해 분석용인 Snowflake 는 Batch 방식으로 변경, 점주앱을 위한 준실시간 DB 를 구축하는 프로젝트 입니다.

## 역할

저는 해당 프로젝트에서

- Snowflake 내부 파이프라인
- 외부계정 파이프라인 아키텍처 구성

해당 작업을 담당하고 있습니다.

## 구성

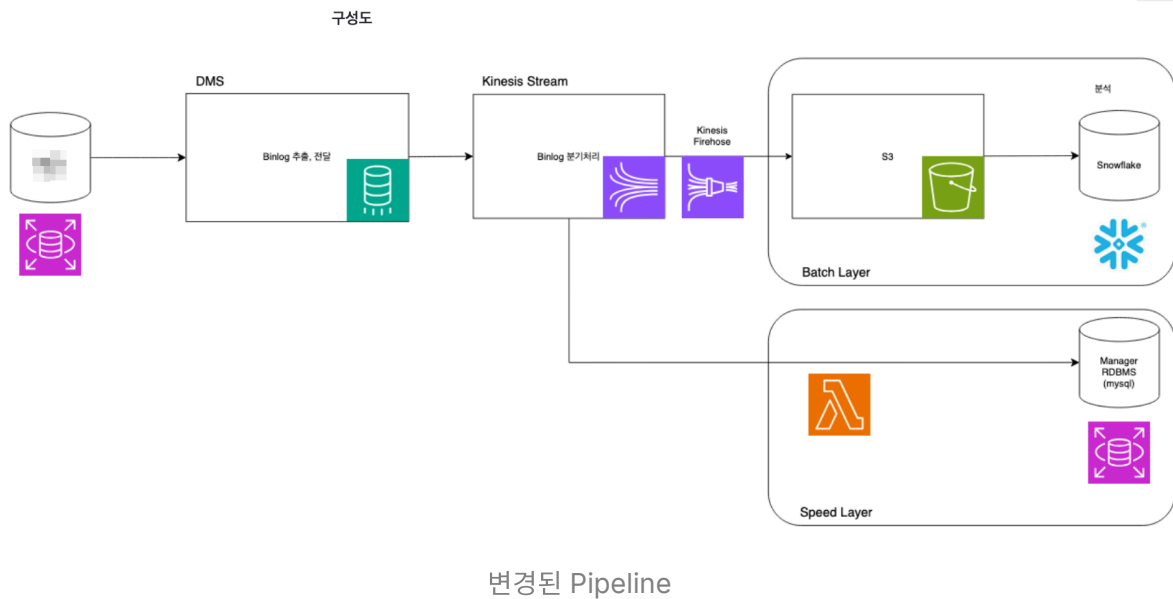
### 사용기술

- 서비스
  - AWS RDS
  - AWS DMS
  - AWS S3
  - AWS Lambda
  - AWS Kinesis DataStreams
  - Snowflake
  - Snowpark(Spark)
- 언어
  - Python
  - SQL
- AS-IS



이전 Pipeline

- To-Be



- 이전 방식

- 점주앱을 위해 이전 방식에서는 DMS → S3 → Snowflake[snowpipe] 방식을 사용하여 1분 간격으로 Snowflake 에 데이터를 최신화 하였습니다.



해당 방식으로 진행 시 브랜드 당 월 300 \$ 이상 발생 하였으며, 추가 브랜드마다 비용이 선형적으로 증가하여 지속적으로 운영하기는 힘들다고 판단하였습니다.

- 변경된 방식

- DMS → S3 를 가기전에 중간에 Kinesis Streams 를 두고 점주앱을 위한 데이터는 Speed Layer RDS 를 구성하여 바라보는 곳을 바꿨습니다.
- 또한 Snowflake 는 Kinesis 에서 Firehose 로 보내주는 데이터를 바라보도록 변경하였습니다.



Kinesis 를 도입하게된 배경은 동일한 Binlog 를 분석용, 서비스용 DB 로 동시에 전송 가능하며, Lambda 또는 Flink 를 쉽게 연결 할 수 있어, 추후 요구사항에 따라 실시간 집계 또한 할 수 있게되어 여러 요구사항을 유연하게 처리할 수 있게 되는 장점이 있다고 생각하였습니다.



Kinesis 와 Speed Layer 를 도입하게 되면 여러 프랜차이즈를 DataWarehouse 에 추가하더라도 비용이 선형적으로 증가하지 않으며,

특히, 점주앱에서 사용하는 쿼리비용 또한 드라마틱하게 절감할 수 있게 될 것입니다.

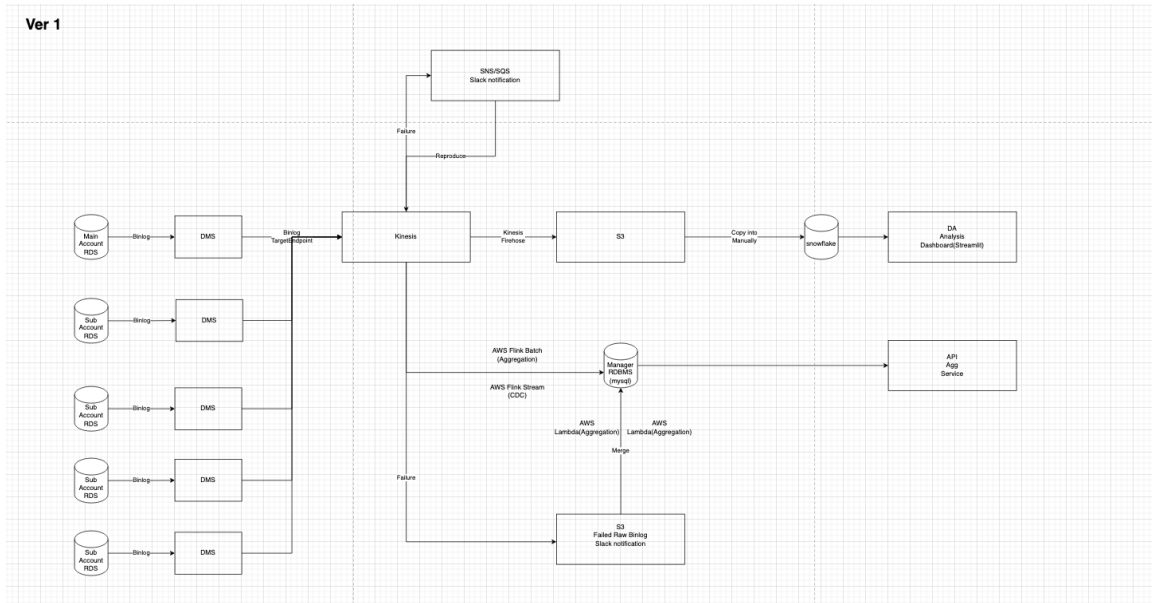
파이프라인중 Snowpipe 만 제거하여도 프랜차이즈당 월 200 \$ 를 절감할 수 있으며 20개 프랜차이즈를 DW 에 통합하게되면 월 2000 \$ 를 절감할 수 있게됩니다.

## 멀티 계정 파이프라인

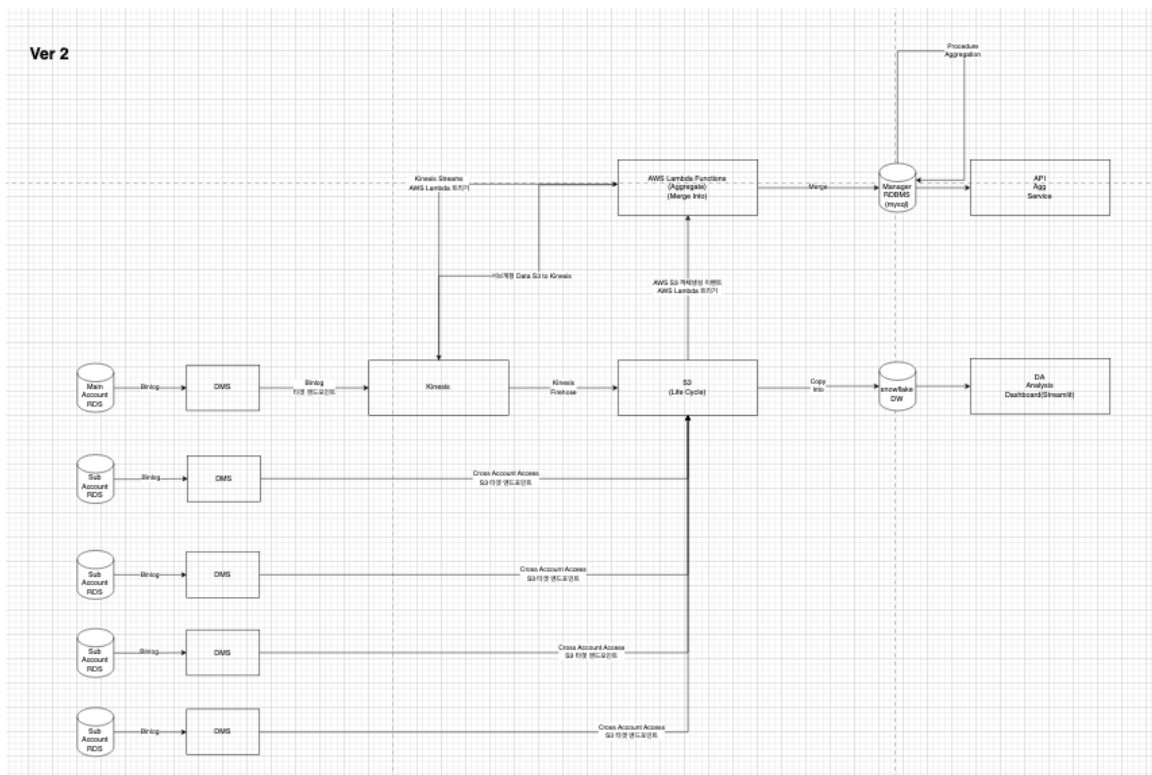
10개 이상의 프랜차이즈의 DB 는 AWS 의 하나의 계정이 아닌 다양한 계정에 있으며 이를 메인 계정의 Kinesis, RDS, S3 에 적재하는 작업을 진행하려고 하고 있으며,

두 개의 버전으로 준비하고 있습니다.

- Ver 1



## • Ver 2



Ver 1 의 경우에는

1. 각 계정의 DMS 를 생성 동일
2. DMS 에서 메인 계정의 Kinesis Streams 를 타겟으로 데이터를 보냄
3. Kinesis Streams 에 지정된 방식으로 S3, RDS 로 데이터 전송

Ver 2 의 경우에는

1. 각 계정의 DMS 를 생성 동일
2. DMS 에서 메인 계정의 S3 로 데이터 전송
3. S3 에 객체생성알림의 트리거를 AWS Lambda 로 하여 Kinesis 로 데이터 전송
4. 이후 점주앱 RDS 로 실시간 데이터 Merge 작업
5. Snowflake 는 매일 자정 프랜차이즈별로 S3에 저장되어있는 테이블 데이터를 가져오는 ETL 작업 시작

현재 주문 APP 에서 발생하는 다양한 데이터를 **Kafka 같은 메시지 큐, 통합 주문 API Gateway 같은 중간 서버**를 두어 Datawarehouse 와 운영 DB 에 **동시에 데이터를 보낼 수 있는 방법도** 고려하고 있습니다.

## 어려운점

해당 프로젝트에서 가장 어려운점은 해당 프로젝트를 진행하며 많은 서비스를 사용하고 있는데 권한을 최소로 가져가기 위해 너무 많은 시간을 소모하는 것입니다. 하나의 서비스에도 수십개의 권한이 있는데 이 권한을 구분하여 설정하는 것에 시간을 많이 소모하고 있으며,

🥲

추가적으로 멀티계정의 데이터를 메인계정으로 통합하는 과정에서 PassRole 이나 AssumeRole 같은 권한의 이해도 부족함을 깨달았으며, 많이 부딪히며 배우고 있습니다.

## 회고



이 때 Kinesis 는 모든 방식을 커버가능한 서비스로 어떠한 방식으로 결정되더라도 유연하게 대처할 수 있을것이라 생각합니다.