DEVIEW 2021

BERT로 만든 네이버 플레이스 비슷한 취향 유저 추천 시스템 (모델 개발부터 서빙까지)

주윤상 NAVER G플레이스AI개발
유원홍 NAVER G플레이스AI개발

CONTENTS

2. MLOps

2. MLOps

2.1 배경

배경 – 초기 상황

개발 리소스

2021년1월 ~ 4월 4 개월

실무 개발자 2 명

개발 요소

시스템 구축

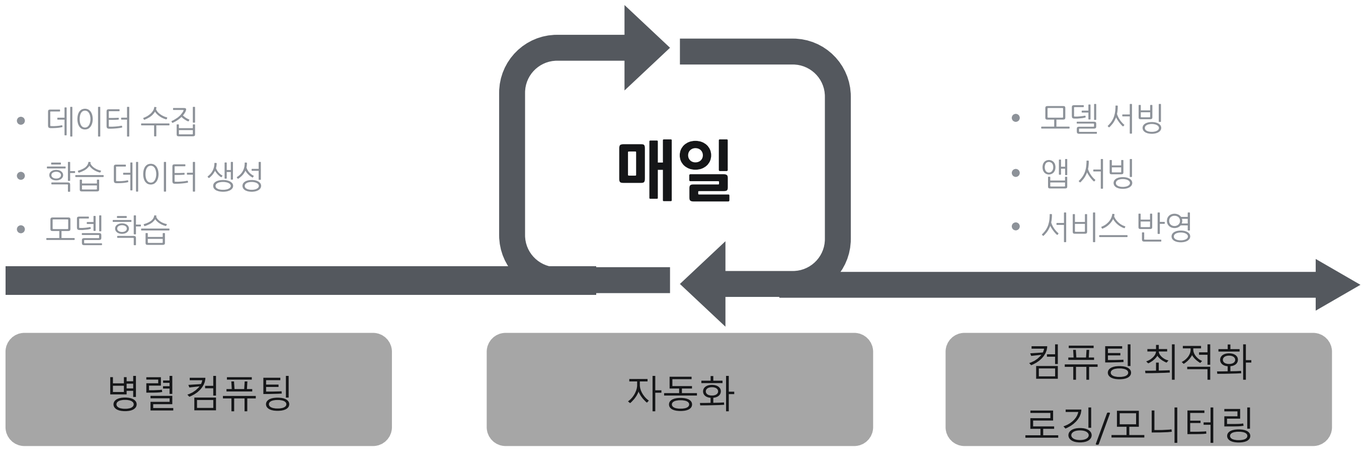
과제 개발

업무 전략

부서 역할의 당위성 검증 필요 > **효과**적인 전략 요구

적은 인원/기간 > **효율**적인 전략 요구

과제 요구 사항



매일 ML pipeline 수행

데이터 수집 > 데이터 정제 > 모델 학습 > 모델 배포 > 앱 배포 > 분석 > 서비스 반영

2. MLOps

2.2 부서 도입 MLOps

Ops (Operations)의 파생

DEVIEW 2021

DataOps

AIOps

DevOps

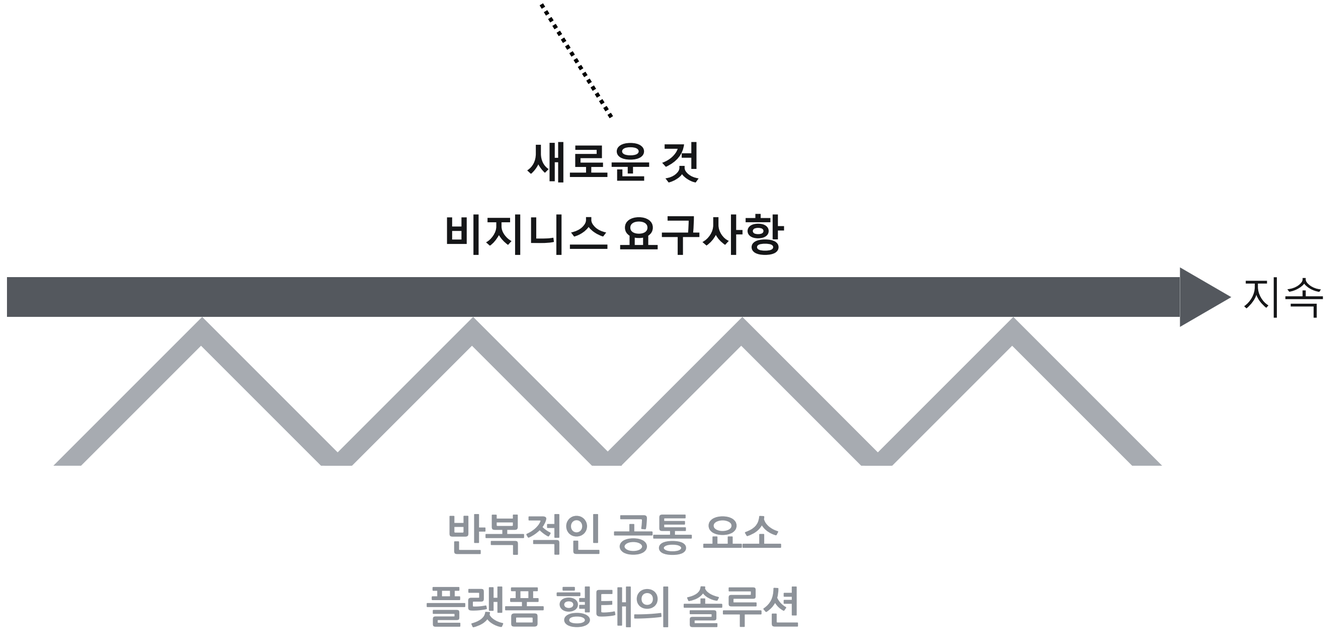
NoOps

MLOps

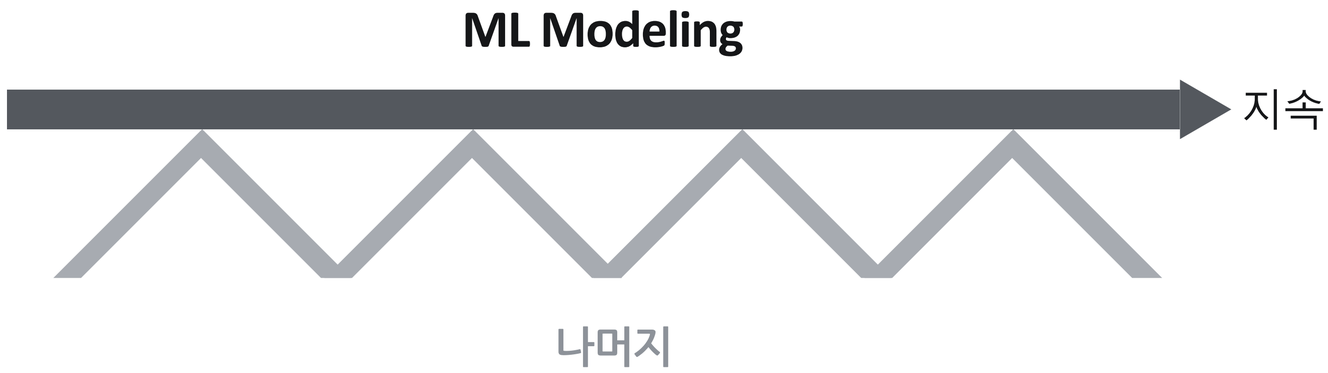
GitOps

Ops란?

Ops를 구별하는 기준: 새로운 것 또는 공통 요소를 무엇을 중심으로 볼 것인가?

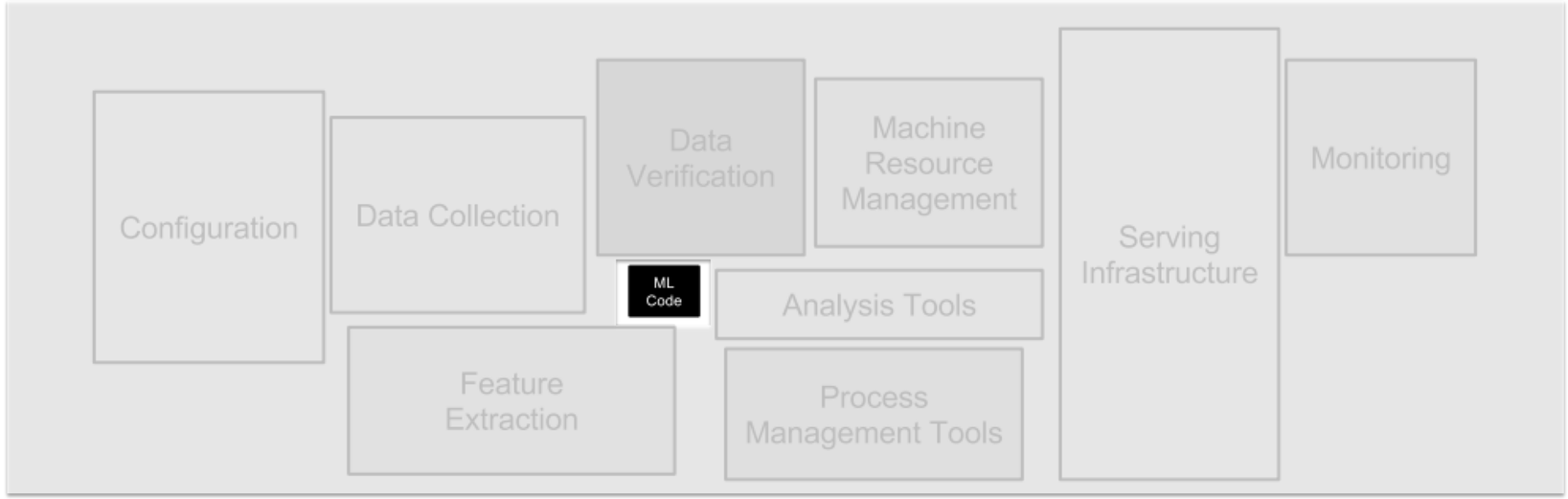


Ops란?



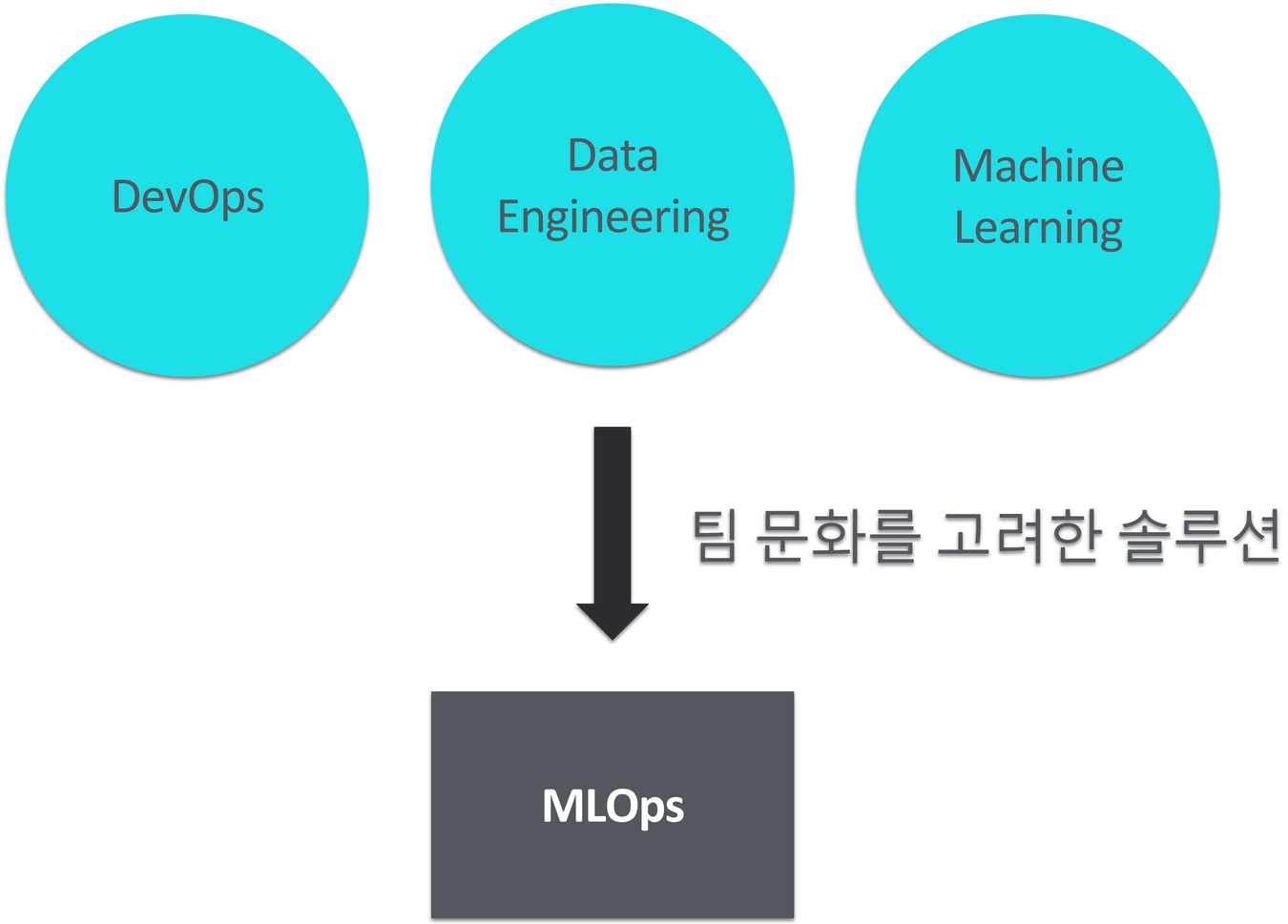
MLOps의 비중

핵심 ML code는 실제 ML 시스템에서 **작은 비중**을 차지한다.



Hidden technical debt in machine learning systems. In Advances in Neural Information Processing Systems, 2015.

MLOps의 요소



활용 사내 플랫폼

hadoop – c3s

k8s – n2c

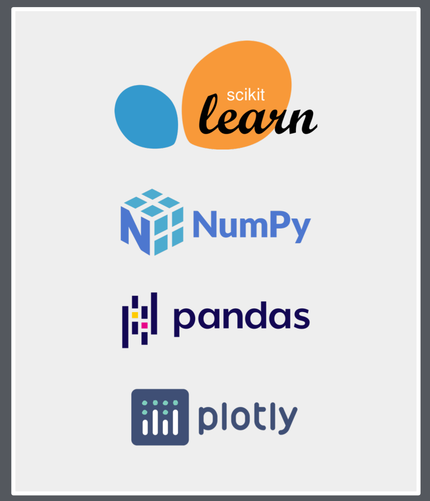
MLOps의 요소

DEVIEW 2021

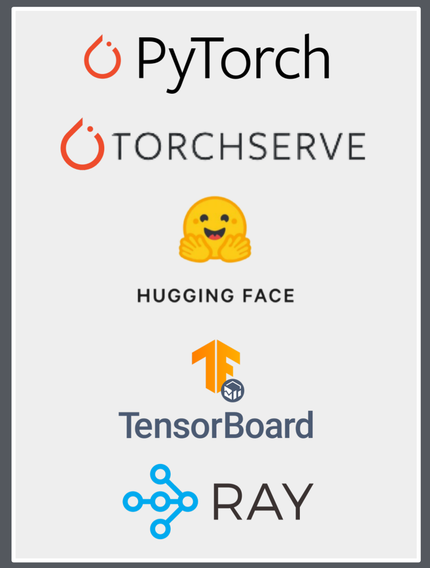
	관심 요소	플레이스AI 부서 - 부서 문화를 고려한 Ops		
		방향성	고려 요소	도입한 솔루션
DevOps	IT 개발/운영	cloud-native	container 운영 자동화 네트워크 구성 및 관리 observability	k8s (container orchestration) airflow (workflow engine) traefik (gateway) grafana (monitoring) ELK (logging)
Data Engineering	데이터	빅데이터 처리	데이터 수집/분석 환경 제공 online/offline storage	hadoop - hdfs, spark kafka - kafka broker cluster, kafka connect database - mongodb, postgresql, cassandra network file storage - ceph, nubes notebook - jupyter - papermill
ML (Machine Learning)	머신러닝	GPU 자원 추상화/최적화	GPU 자원 사용 model serving storage 연동	k8s (n2c GPU 전용 노드 운영) c3dls client, nsml client ray torchserve

부서 기술 스택

Language

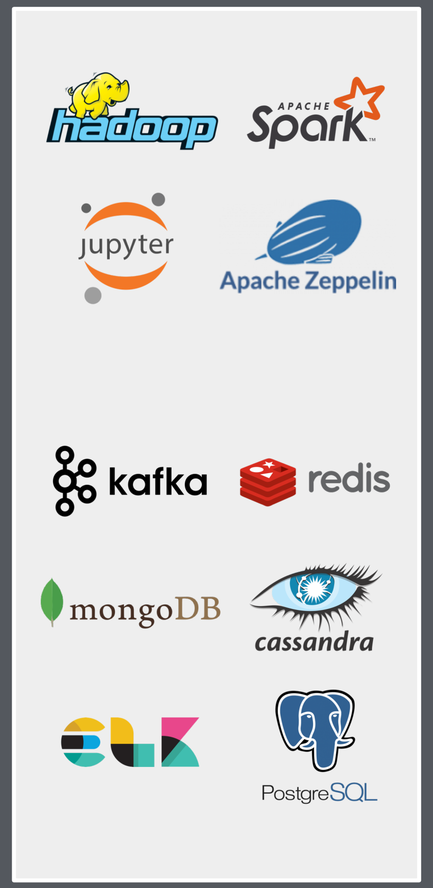


Data Analysis

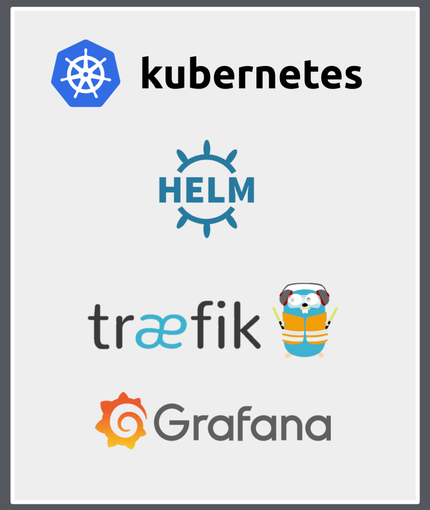


Machine Learning

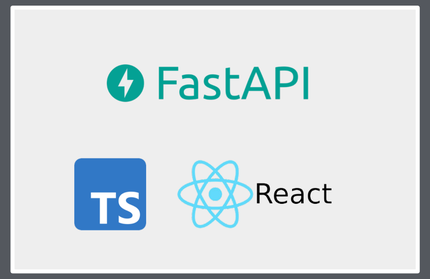
Workflow



Data Engineering



Cloud DevOps



Web Development

2. MLOps

2.3 개발 환경 구성

개발 환경 구성의 어려움

- 많은 환경 구성 요소
- K8S 클러스터내에서만 제공되는 요소

OS 패키지
cuda 10.1
spark 3.1.2
python 3.8.5

사내 플랫폼 클라이언트
c3s client
c3dls client
nsml client
nubes client

python 패키지
torch==1.7.0
torchvision==0.8.1
numpy==1.19.4
scipy==1.5.4
pandas==1.1.4
ploly==5.1.0
papermill==2.3.3
jupyterlab==2.2.9

Network Storage
cephfs
nubes

Secret 정보
kafka auth
database auth
c3s auth
k8s auth

개발자 Sandbox 환경

그림 출처
https://learn.vonage.com/blog/2017/12/12/voice-playground-testing-sandbox-nexmo-voice-apps/

Sandbox

Dev: 코드 실행 환경 구성

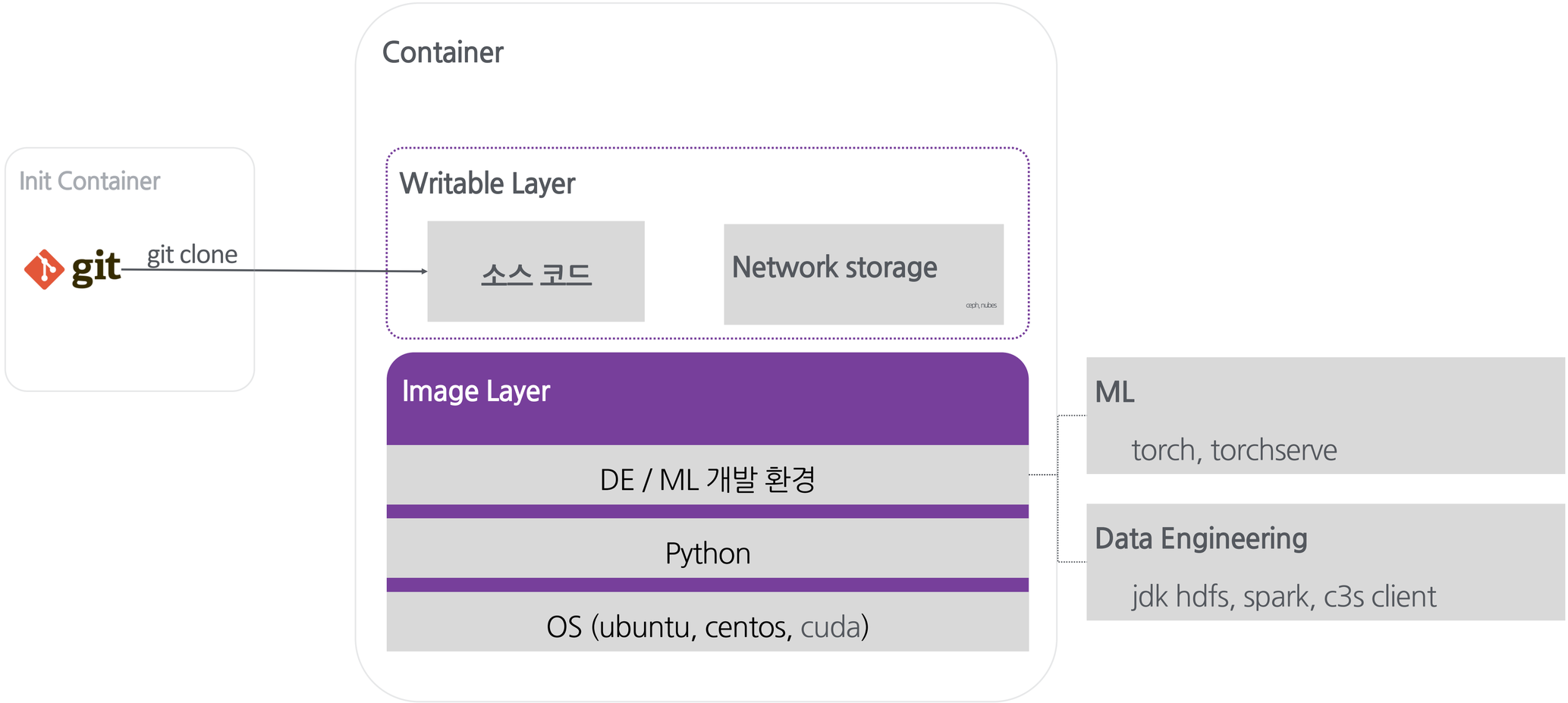
Data: 데이터 중심의 분석 환경 구성

Model: 모델링 중심의 실험 환경 구성

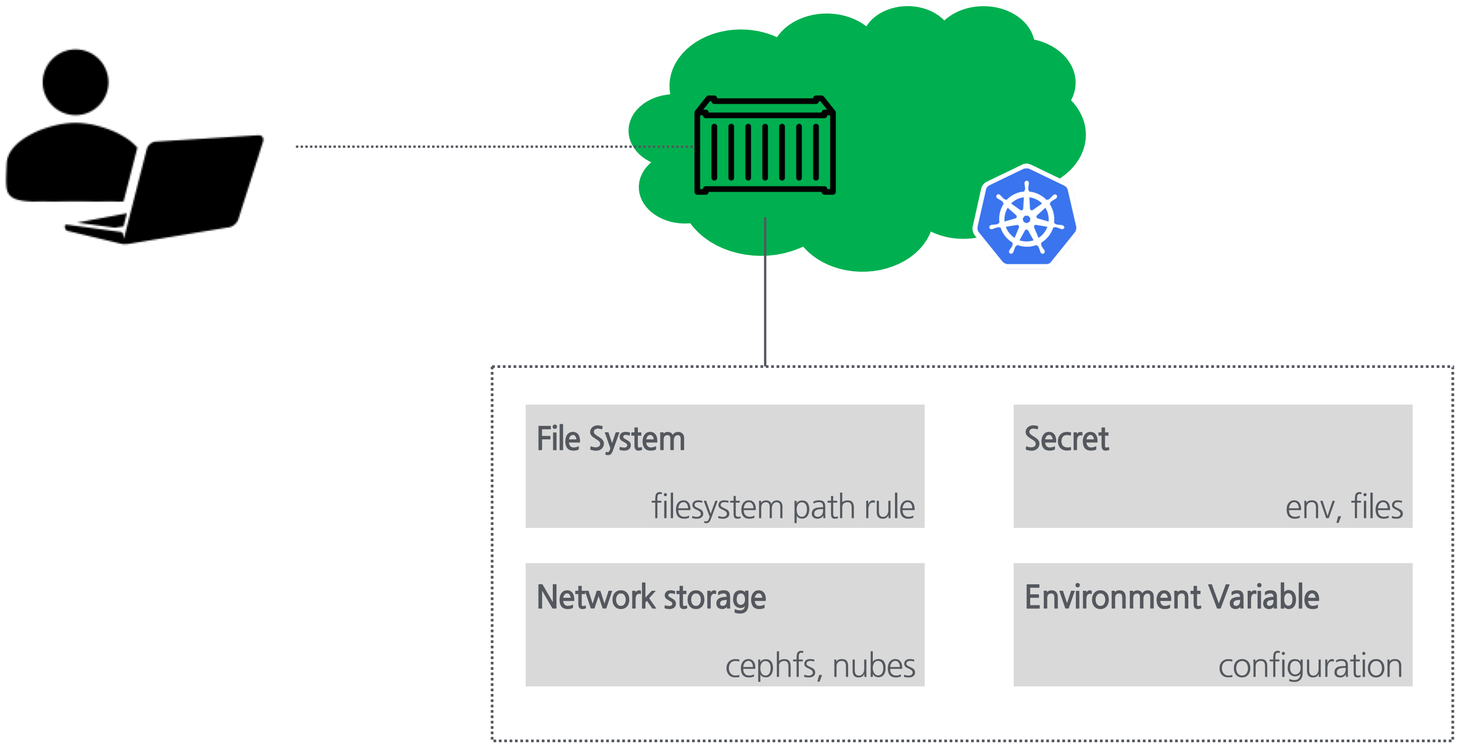
production 배포 환경과 유사하게

여러 프로젝트에서 반복할 수 있게

Container Layer



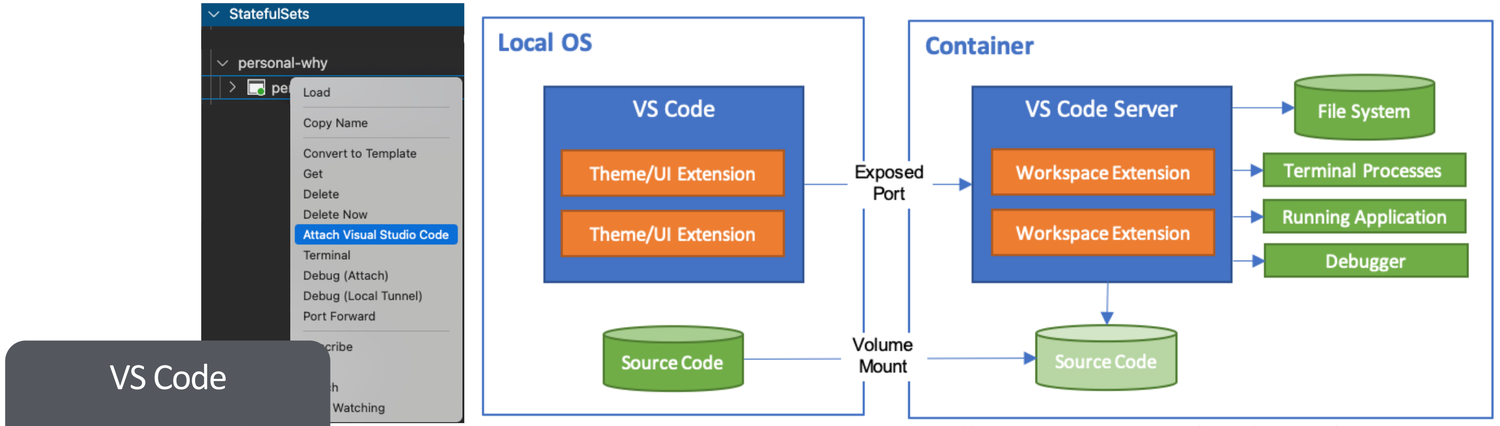
k8s-hosted sandbox



여러 환경 구성 요소를 빠르게 셋업하고 동기화

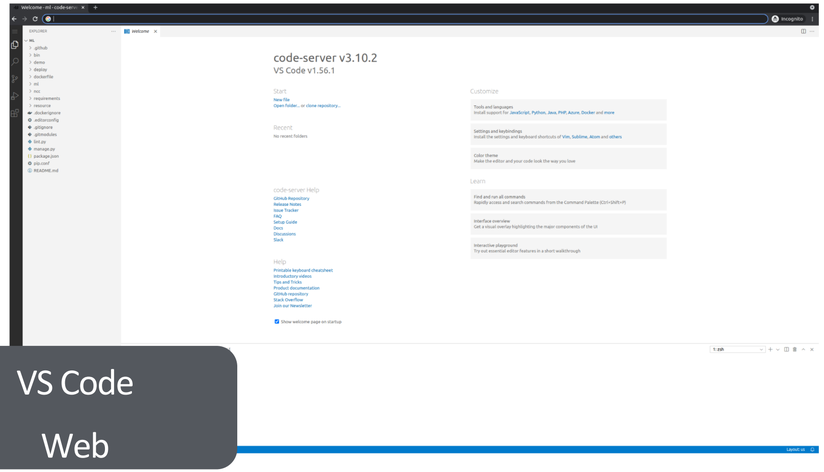
Production 배포 환경과 동일한 k8s 환경

k8s-hosted sandbox - editor

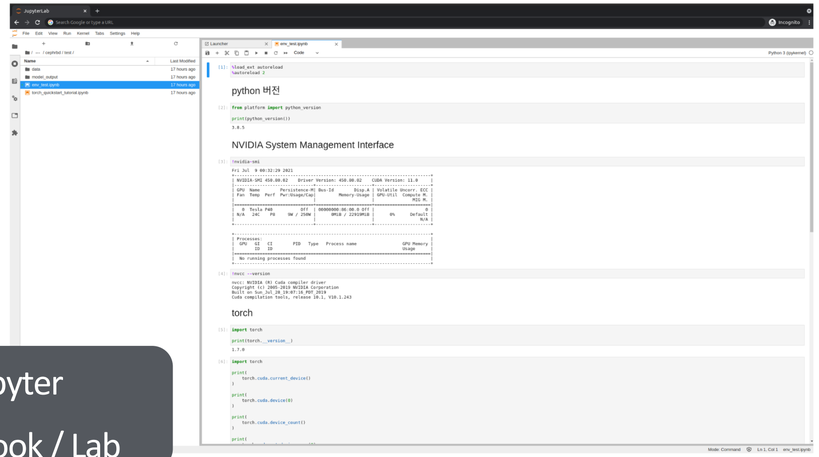


VS Code remote container

https://code.visualstudio.com/docs/remote/containers



VS Code Web



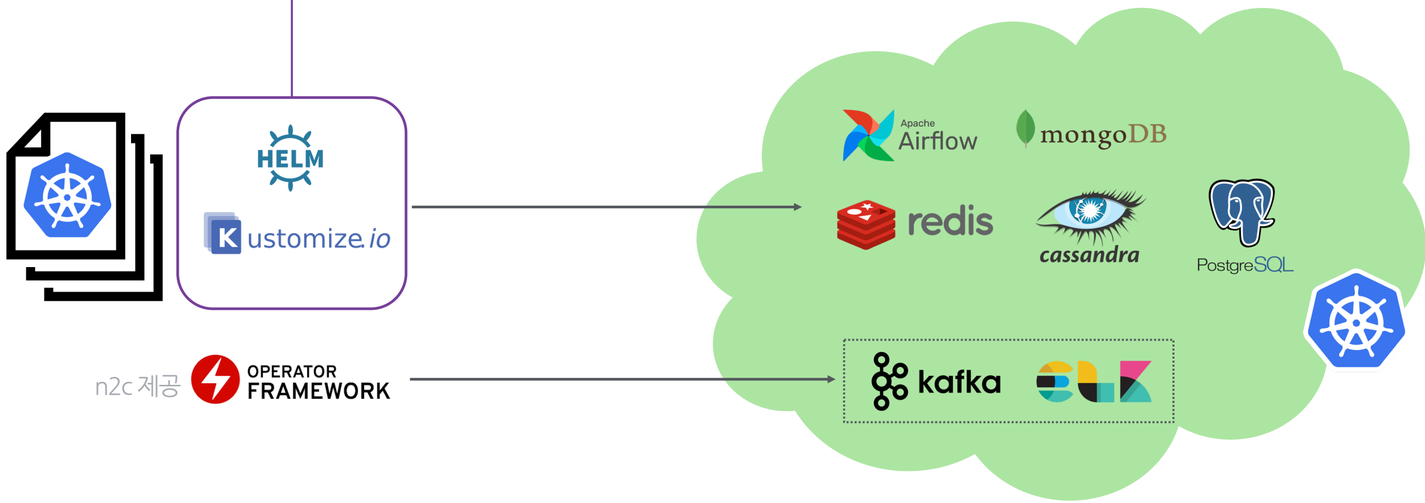
Jupyter Notebook / Lab

2. MLOps

2.4 Kubernetes Package

k8s package

Kubernetes Package 오픈소스를 kubernetes에 배포하여 빠르게 기반 플랫폼을 셋업

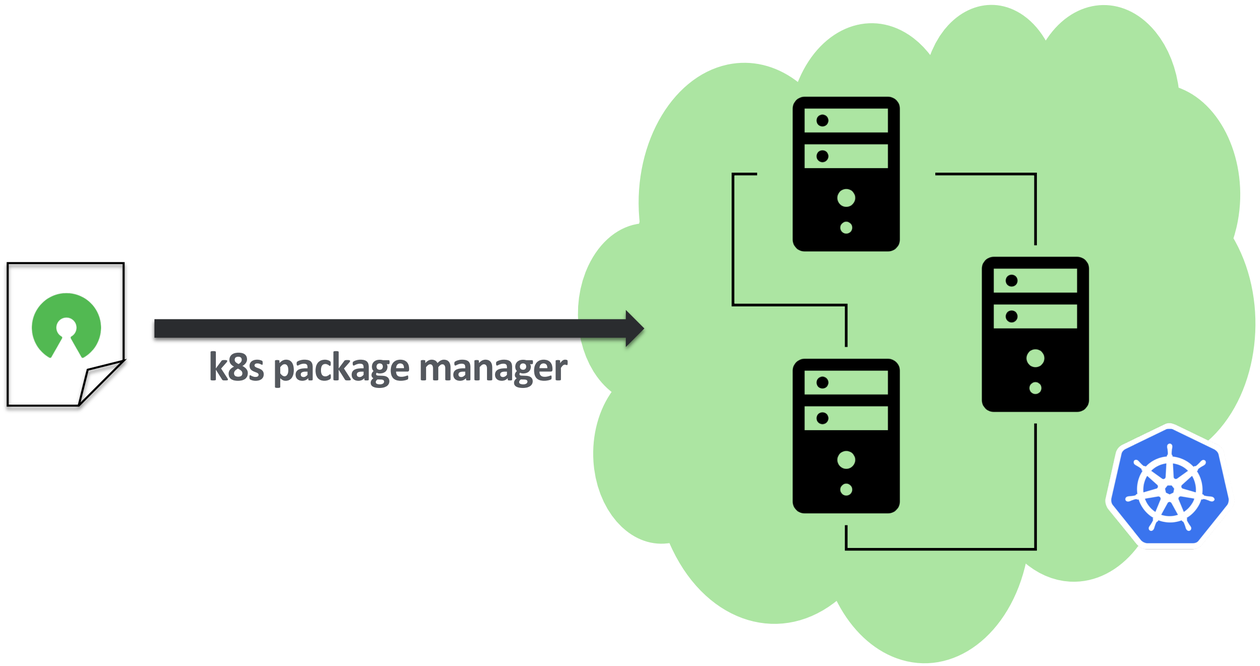


k8s package – Infrastructure As Code

DEVIEW 2021

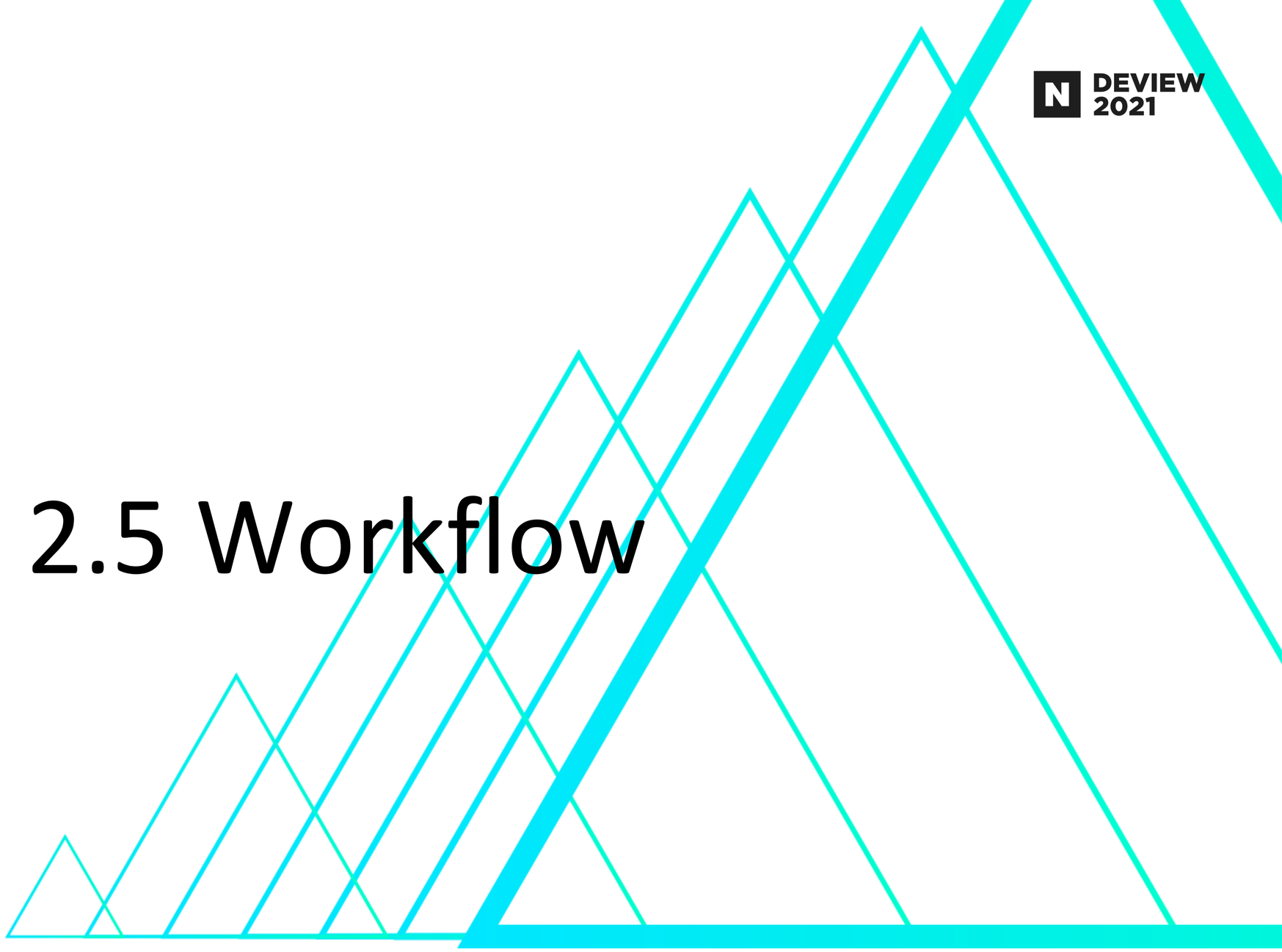
오픈 소스: https://artifacthub.io/

IaC 오픈소스 도입 효과: 양질의 기술 부채

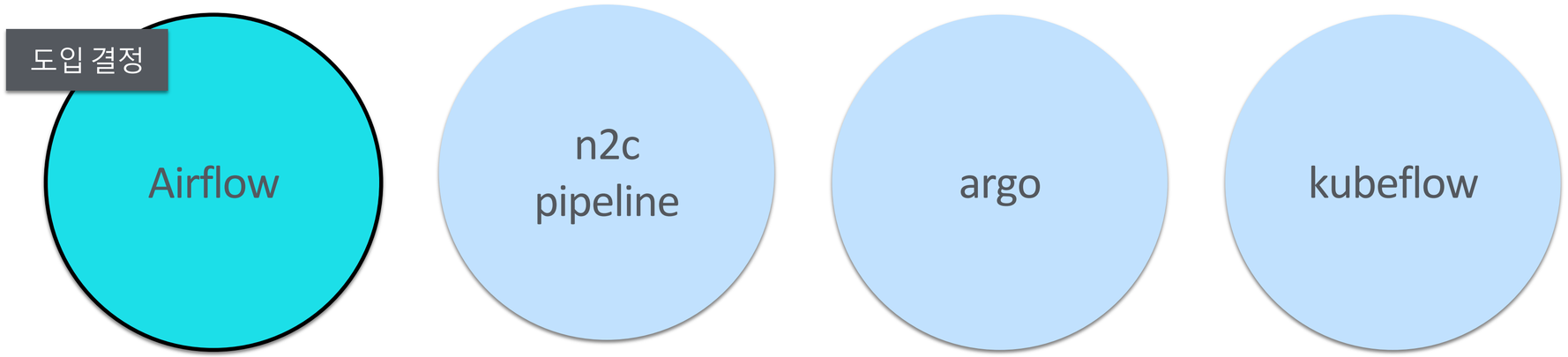


2. MLOps

2.5 Workflow



Workflow Engine 검토



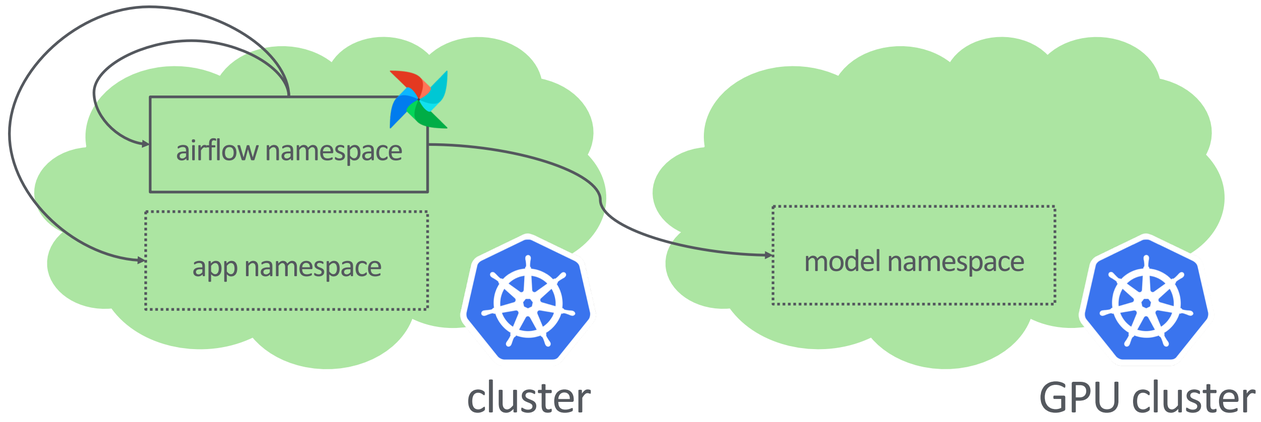
스케쥴링등 workflow 관련 기능이 많음

python 코드로 유연하게 workflow 개발

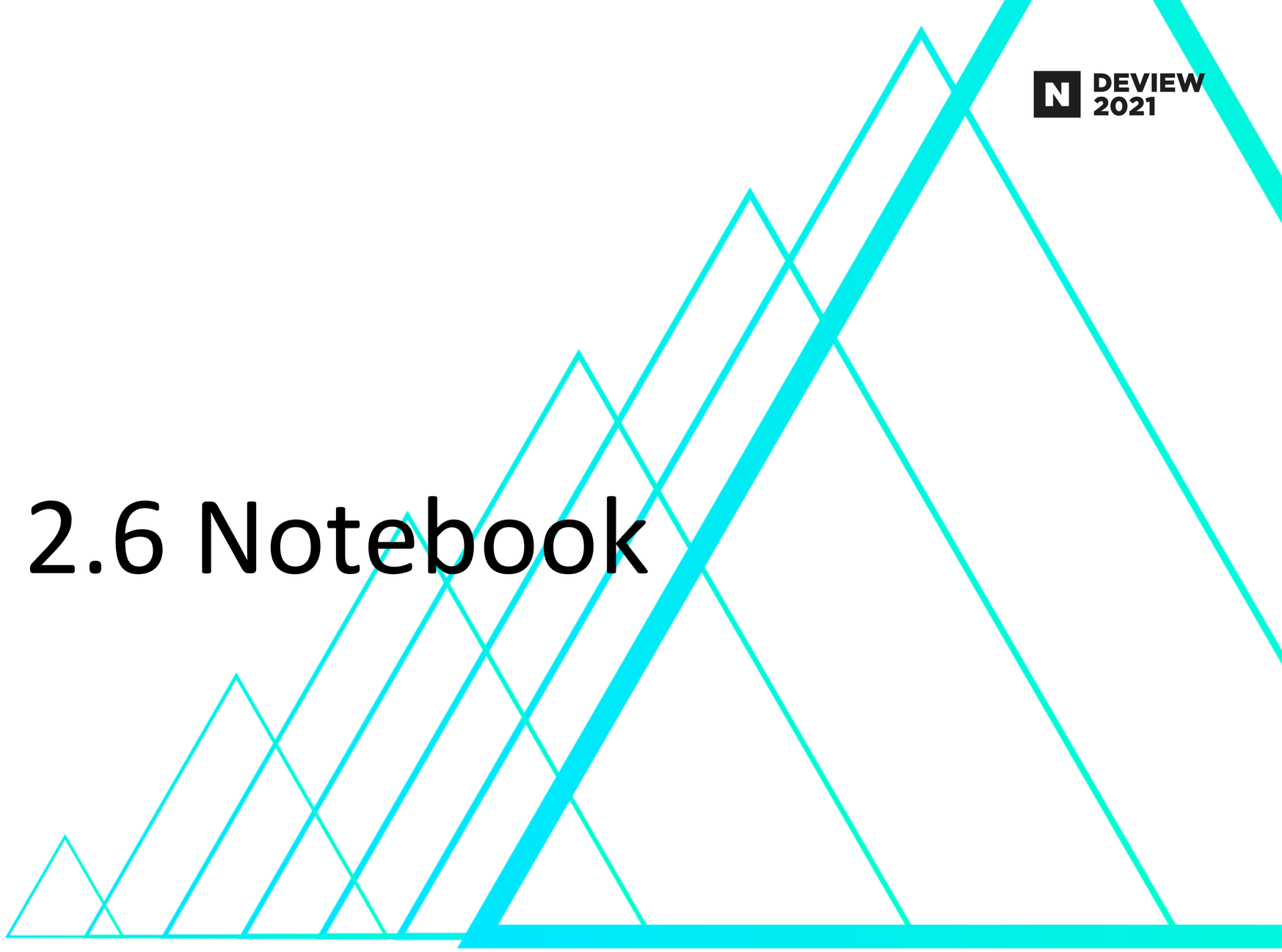
범용성이 가장 큼

Airflow - KubernetesPodOperator 활용

각 Task를 여러 k8s cluster와 namespace scope에서 독립적으로 실행

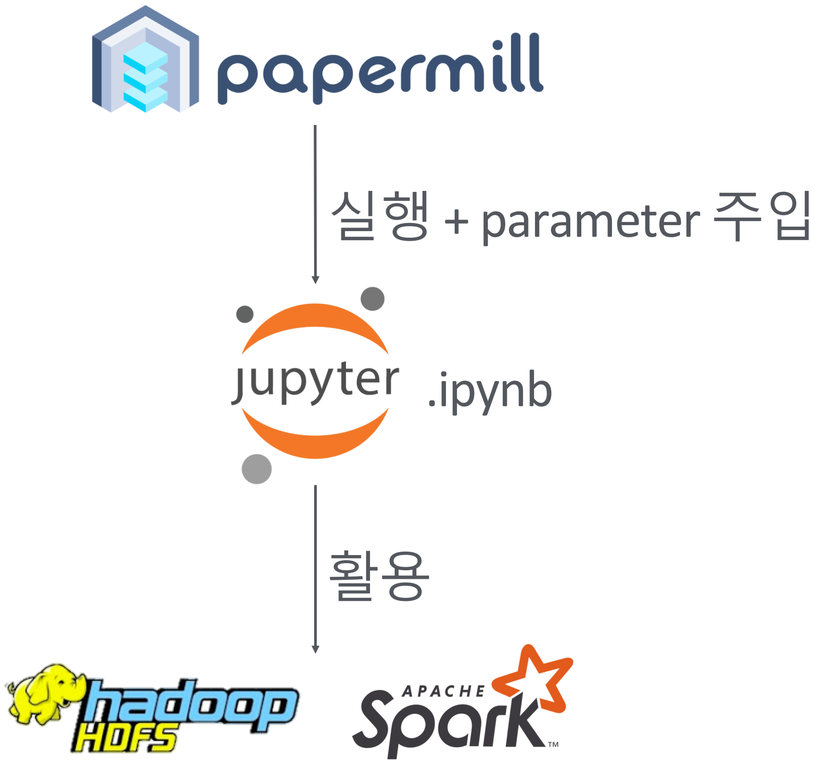


2.6 Notebook



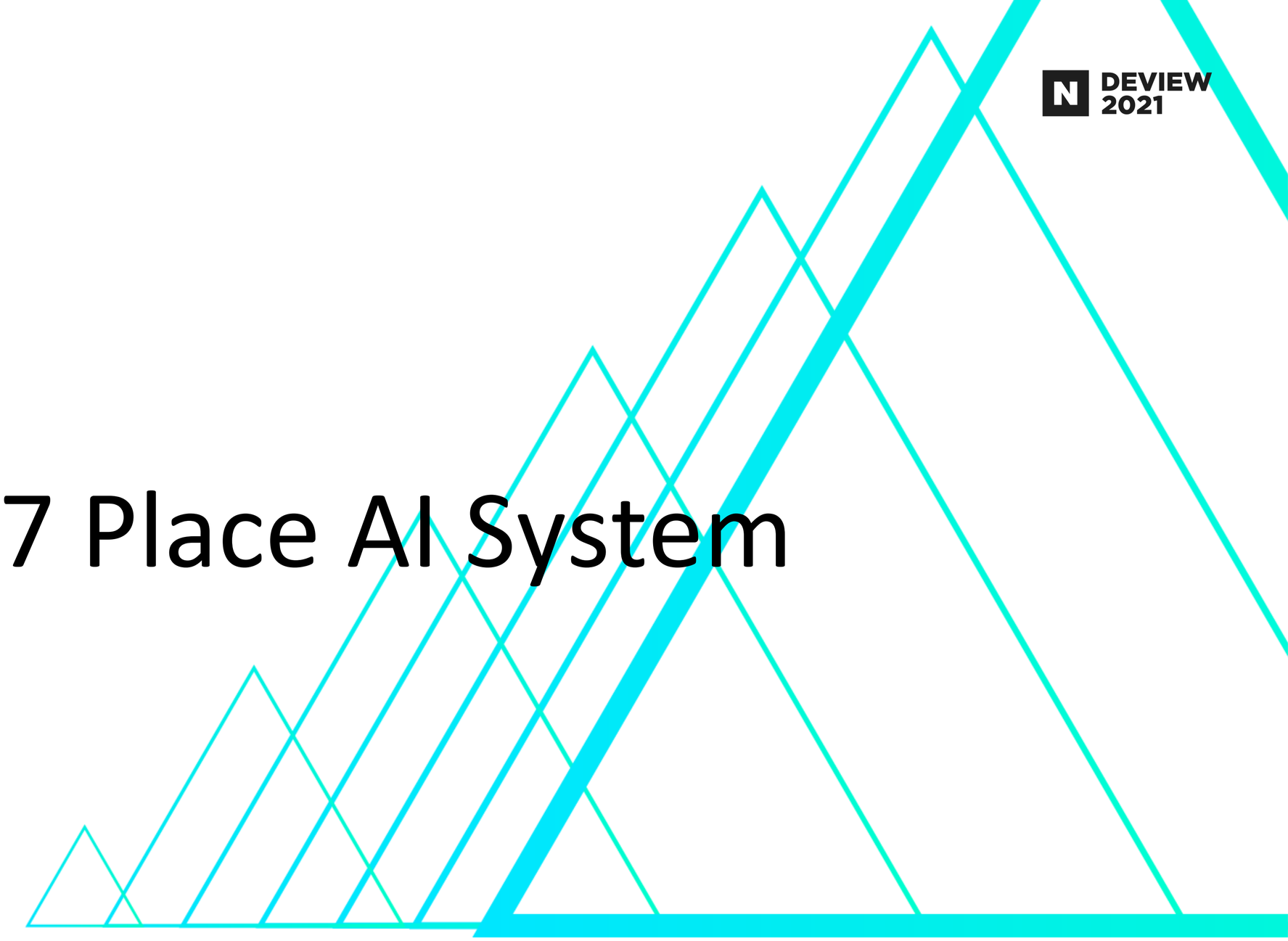
notebook-base code

notebook 형태로 코드 개발하고 papermill 도구를 활용하여 자동화



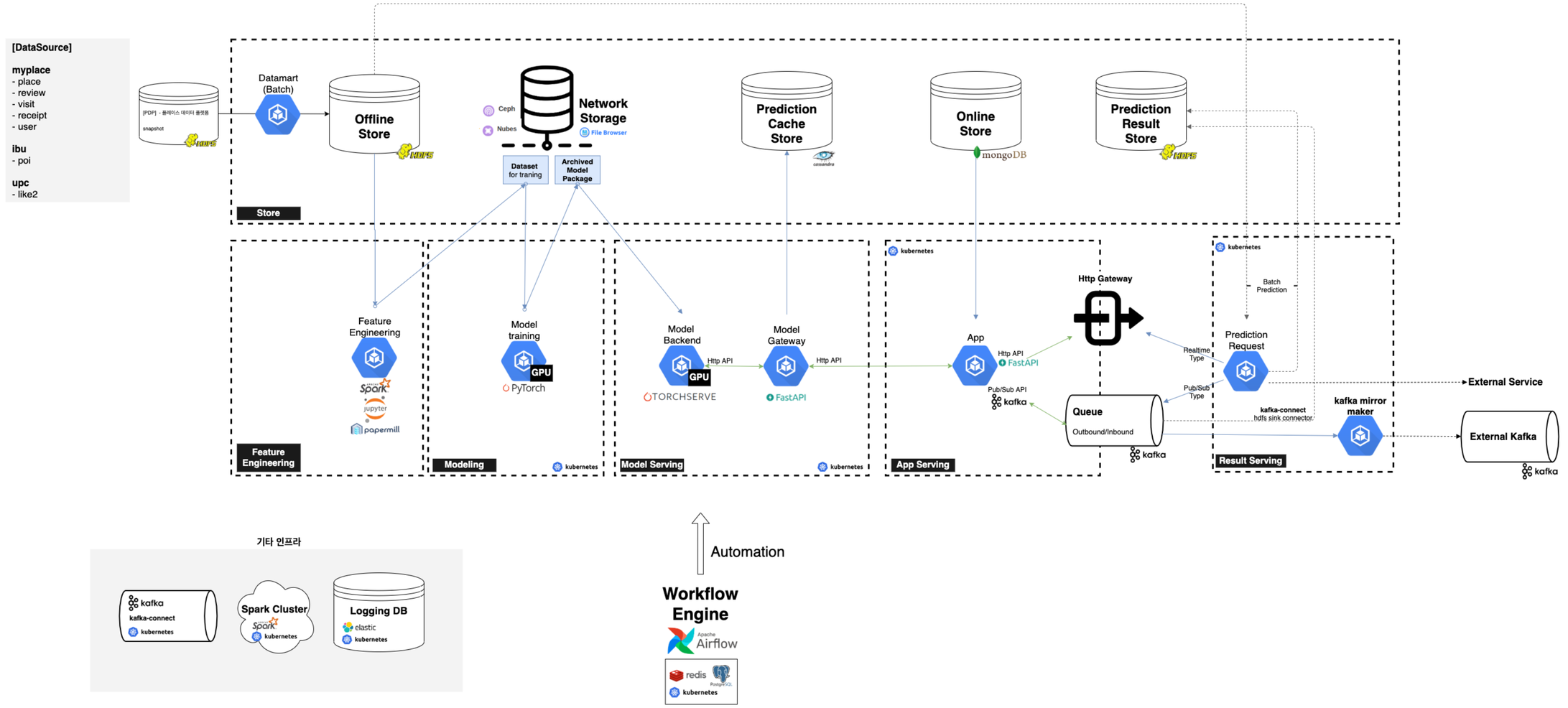
2. MLOps

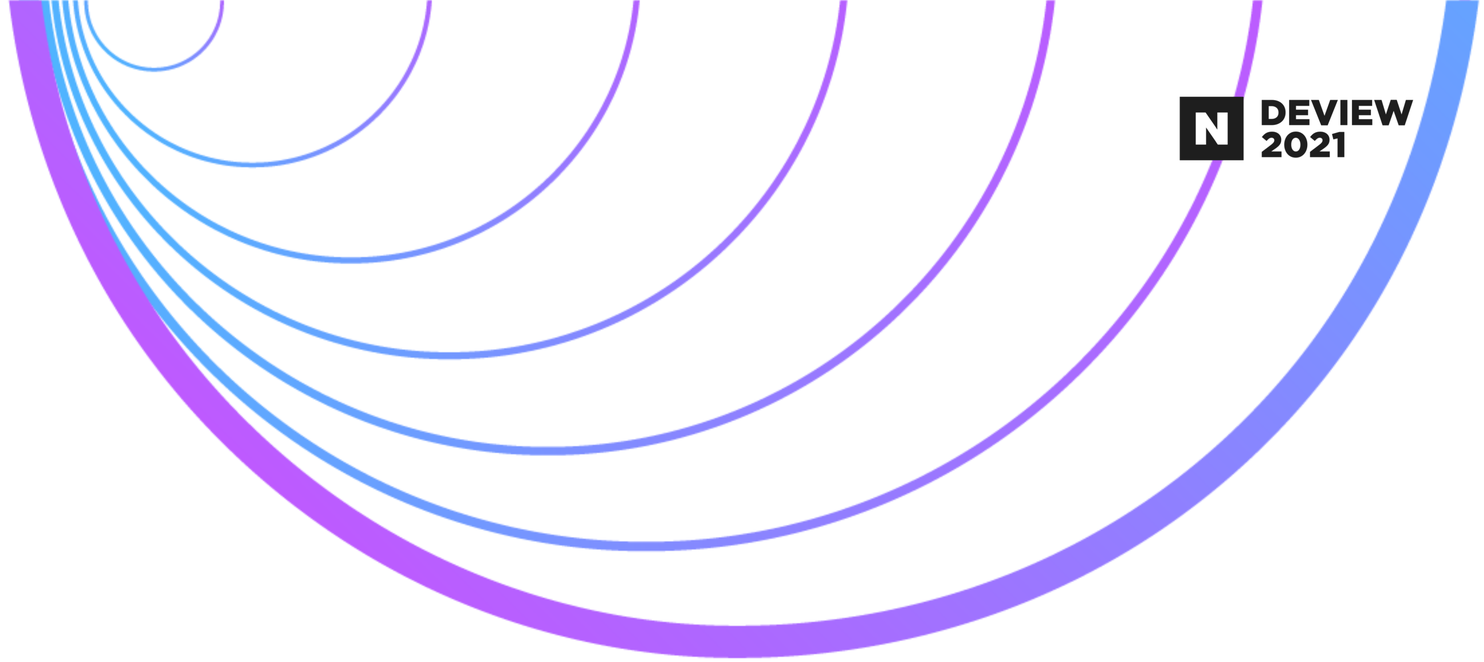
2.7 Place AI System



부서 시스템 구조

k8s 중심으로 부서 시스템 구축



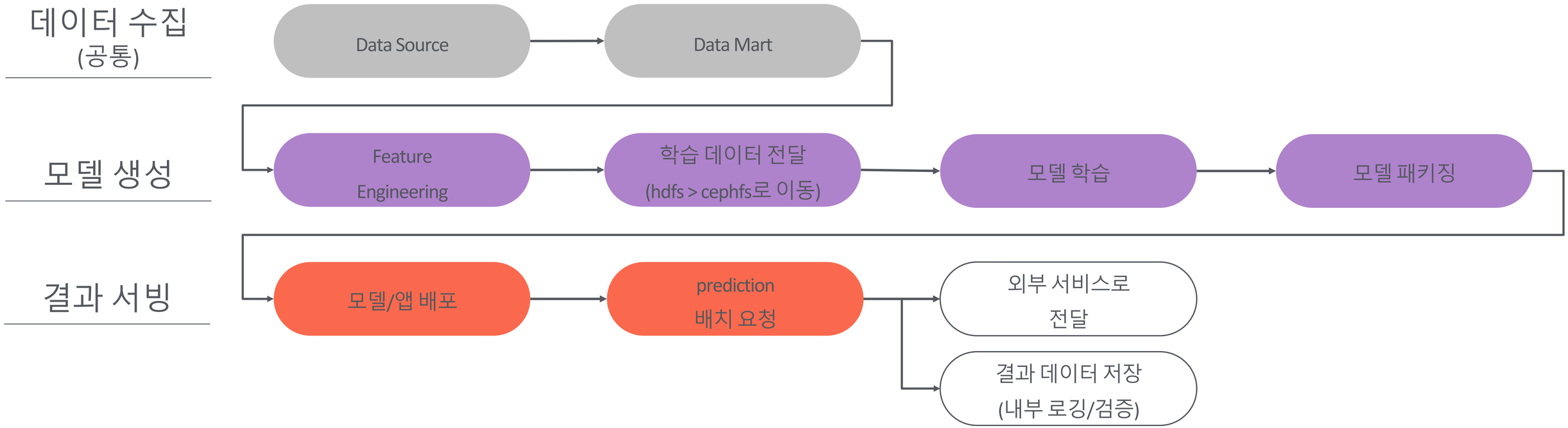


DEVIEW 2021

3. ML Pipeline

3.1 ML Pipeline 요약

ML Pipeline 요약



3.2 Data Source & Data Mart

Data Source & Data Mart

프로젝트 공통

1.datasource
- 원본 데이터

2.datamart
- 프로젝트 공통 ML 분석용 소스 데이터

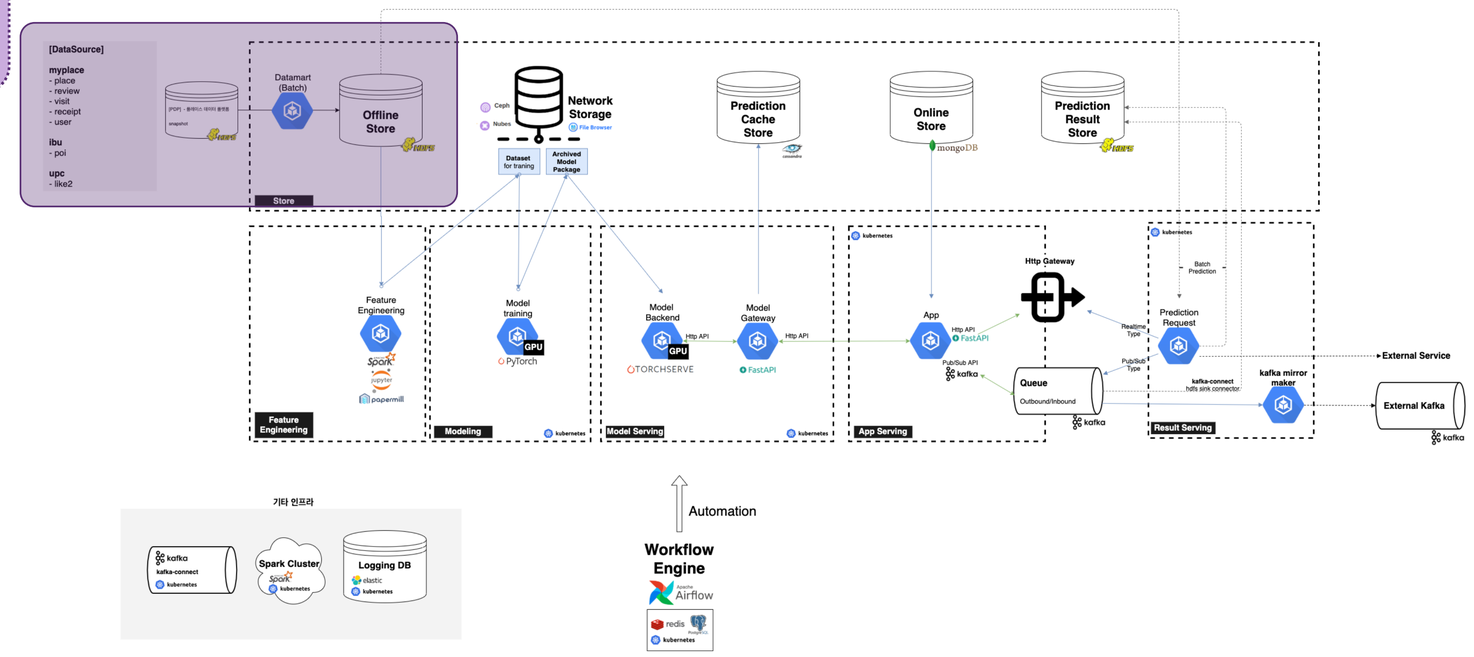
사용자 추천 과제

1.feature engineering

2.모델 학습

3.모델 / 앱 배포

4.전체 유저 분석 및 결과 전달



Data Source & Data Mart

분석 환경에 맞는 데이터 준비

정형화

- 원천 서비스 DB가 schema-less (mongodb)
- 데이터 소스는 data lake로 schema-on-read 정책
- schema를 확정하여 이후 분석 과정에서는 schema-aware한 작업을 수행

정제 (cleansing)

- 대상: 값 타입 혼용, 결측값, 이상값 등...

파일 포맷 변환

- 빠르게 처리 될 수 있도록 데이터 포맷 변경
- jsonline > parquet

유저 추천 과제에서 처리하는 데이터 규모: **억개** 단위, **테라 바이트** 단위 사이즈

3.3 Feature Engineering

Feature Engineering

프로젝트 공통

1.datasource

- 원본 데이터

2.datamart

- 프로젝트 공통 ML 분석용 소스 데이터

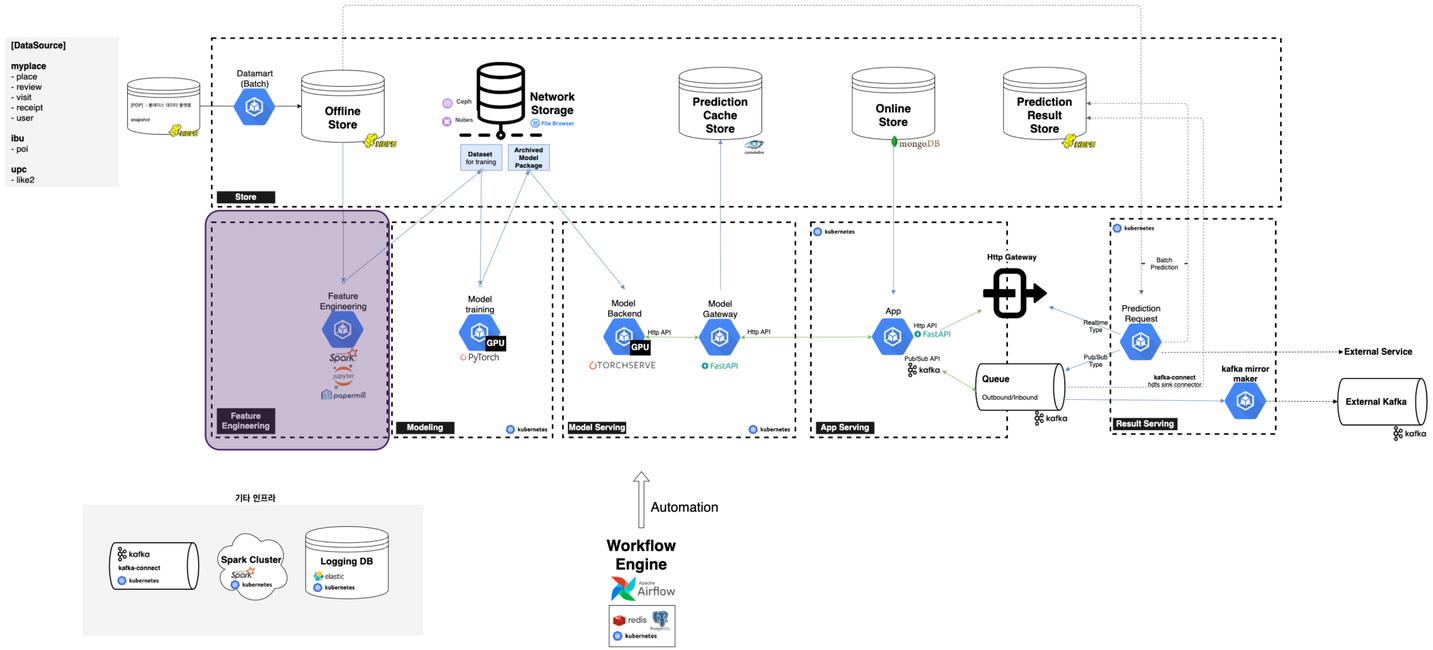
사용자 추천 과제

1.feature engineering

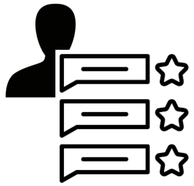
2.모델 학습

3.모델 / 앱 배포

4.전체 유저 분석 및 결과 전달

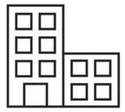


Feature Engineering



- **User Favor: 유저별 취향 목록**
  - `user_idno`
  - `review_list` (array)
  - `like_list` (array)

```
root
 |-- user_idno: string (nullable = true)
 |-- like_list: array (nullable = true)
 |    |-- element: struct (containsNull = false)
 |    |    |-- like_regymdt: timestamp (nullable = true)
 |    |    |-- like_displayid: string (nullable = true)
 |    |    |-- like_contentsid: string (nullable = true)
 |-- like_stat_list: array (nullable = true)
 |    |-- element: struct (containsNull = false)
 |    |    |-- like_displayid: string (nullable = true)
 |    |    |-- restaurant_like_displayid_count: long (nullable = false)
 |-- review_list: array (nullable = true)
 |    |-- element: struct (containsNull = false)
 |    |    |-- review_id: string (nullable = true)
 |    |    |-- review_rating: double (nullable = true)
 |    |    |-- review_visited_date: timestamp (nullable = true)
 |    |    |-- review_place_id: string (nullable = true)
```



- **Place: 장소 정보**
  - `place_id`
  - `place_category_code_list`
  - `place_dongcode`

```
root
 |-- place_id: string (nullable = true)
 |-- place_category_code_list: array (nullable = true)
 |    |-- element: long (containsNull = true)
 |-- place_dongcode: string (nullable = true)
```

3.4 모델 학습

모델 학습

프로젝트 공통

1.datasource

- 원본 데이터

2.datamart

- 프로젝트 공통 ML 분석용 소스 데이터

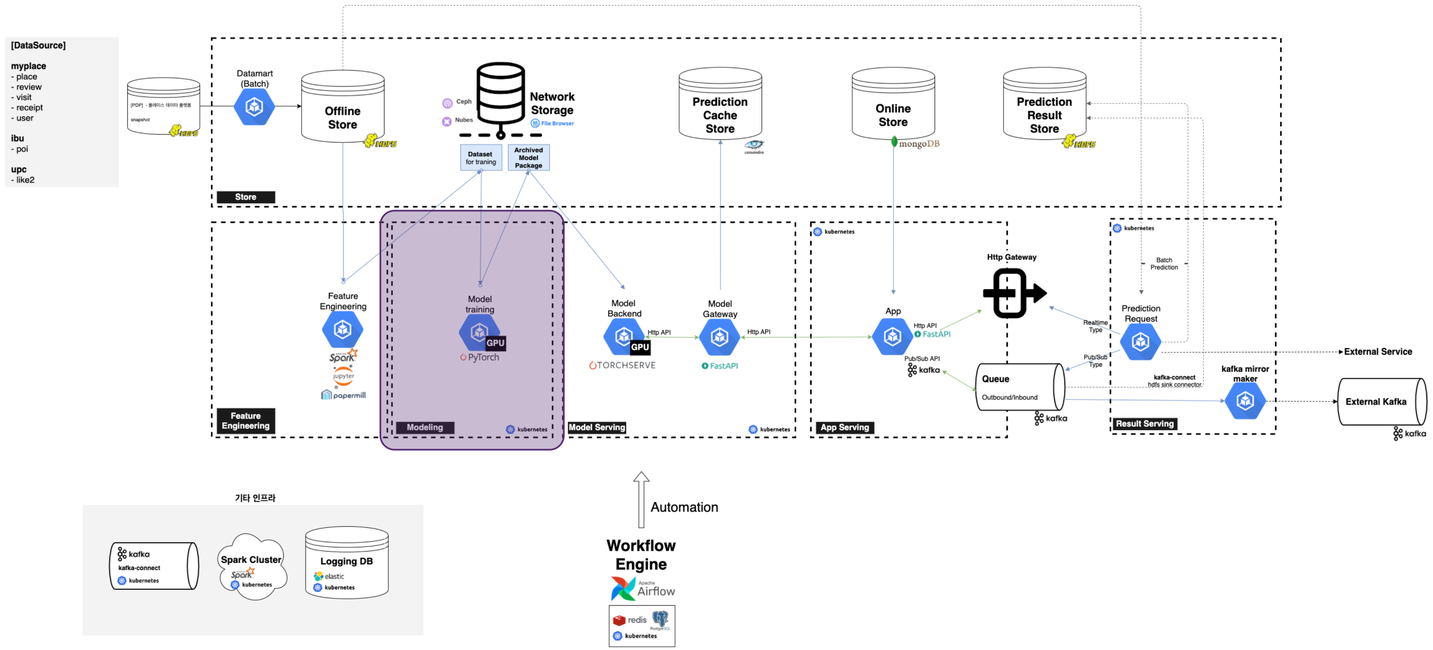
사용자 추천 과제

1.feature engineering

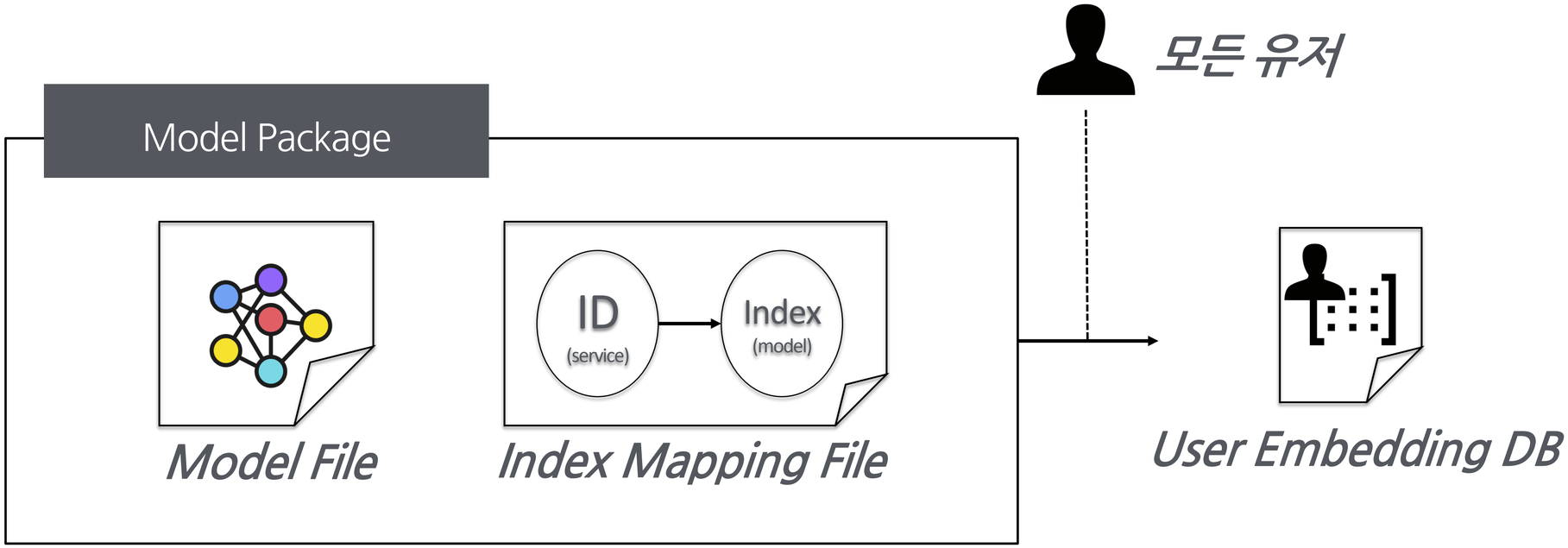
2.모델 학습

3.모델 / 앱 배포

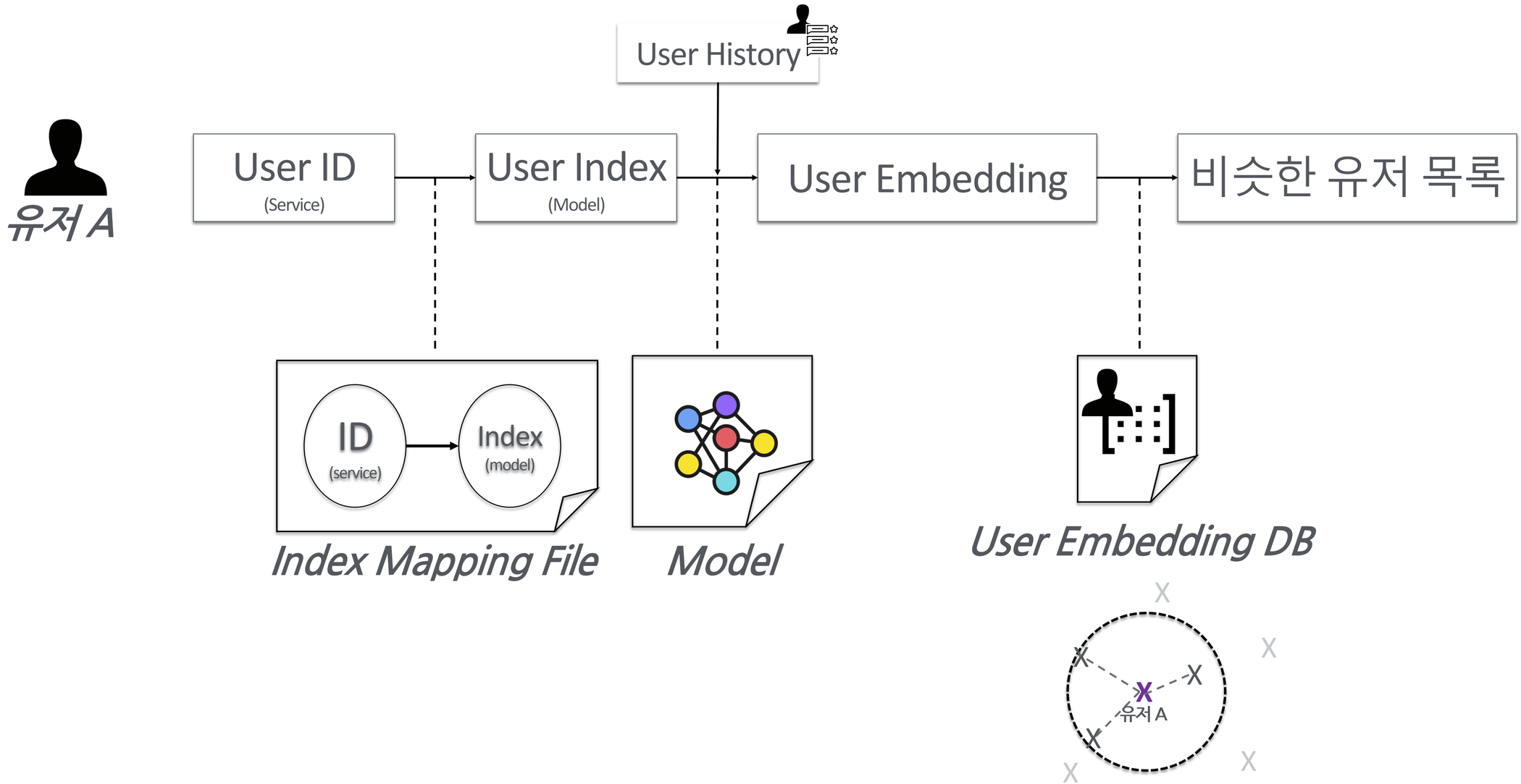
4.전체 유저 분석 및 결과 전달



모델 학습 – 결과 데이터


모든 유저
Model Package
ID
(service)
Index
(model)
Model File
Index Mapping File
User Embedding DB

모델 학습 – 처리 흐름


User History
유저 A
User ID
(Service)
User Index
(Model)
User Embedding
비슷한 유저 목록
ID
(service)
Index
(model)
Index Mapping File
Model
User Embedding DB
유저 A

3.5 모델 / 앱 배포

모델 / 앱 배포

프로젝트 공통

1.datasource

- 원본 데이터

2.datamart

- 프로젝트 공통 ML 분석용 소스 데이터

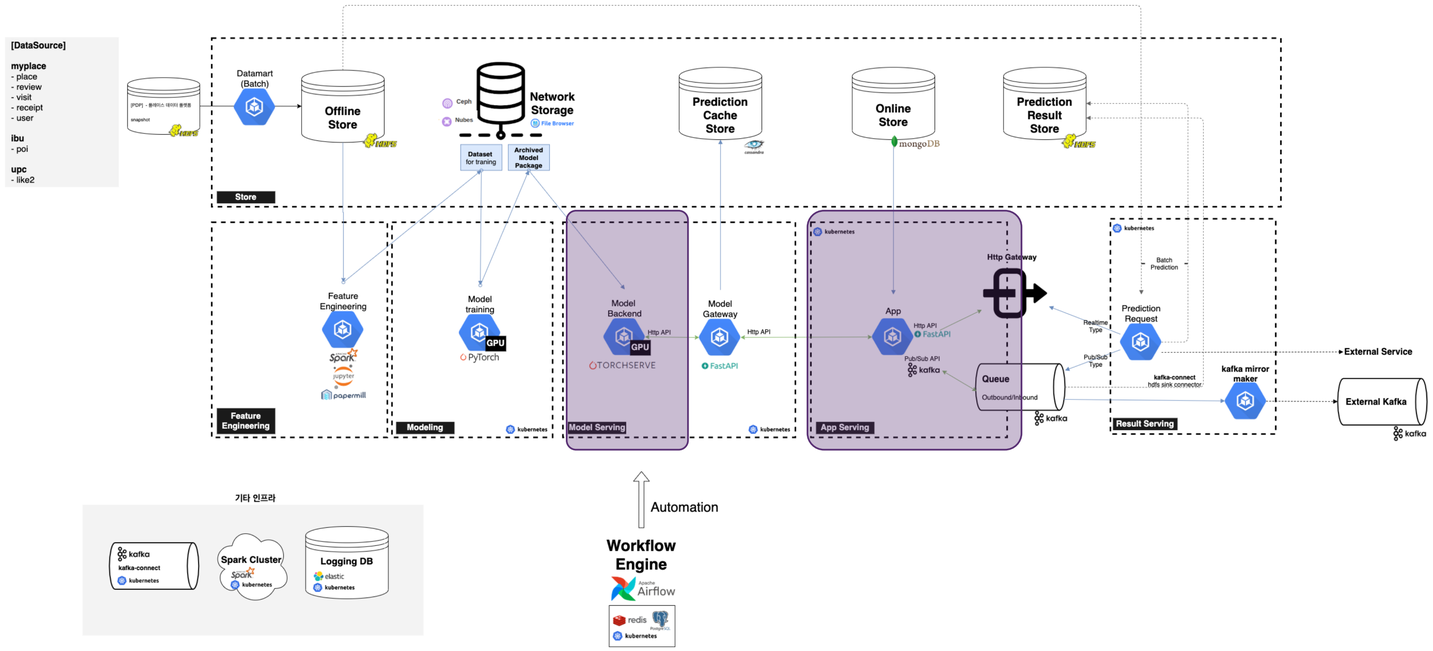
사용자 추천 과제

1.feature engineering

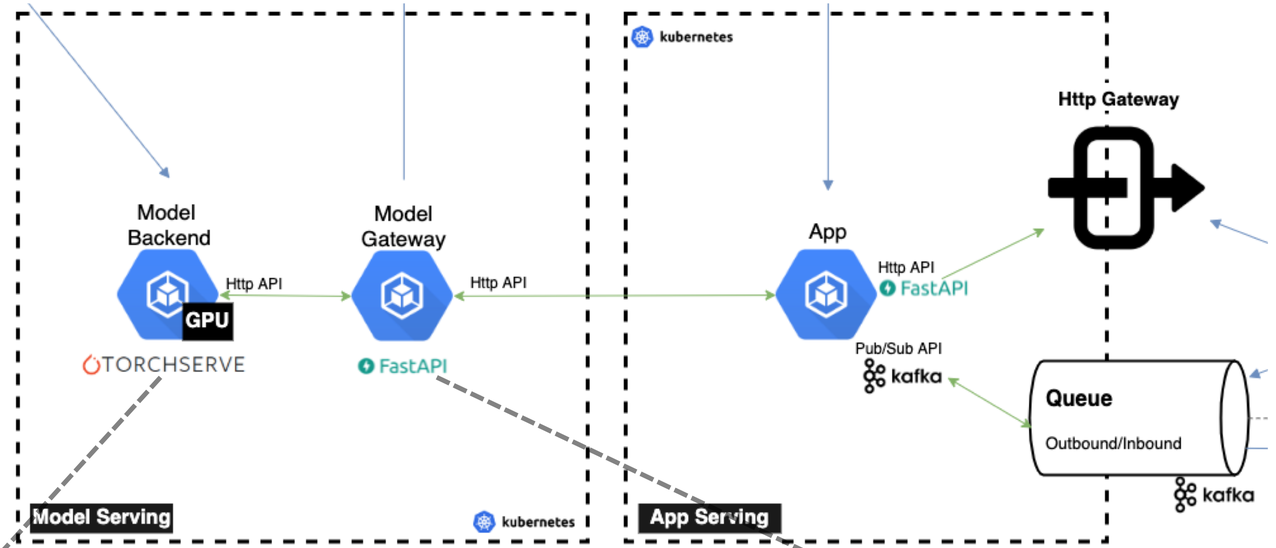
2.모델 학습

3.모델 / 앱 배포

4.전체 유저 분석 및 결과 전달



모델 / 앱 배포



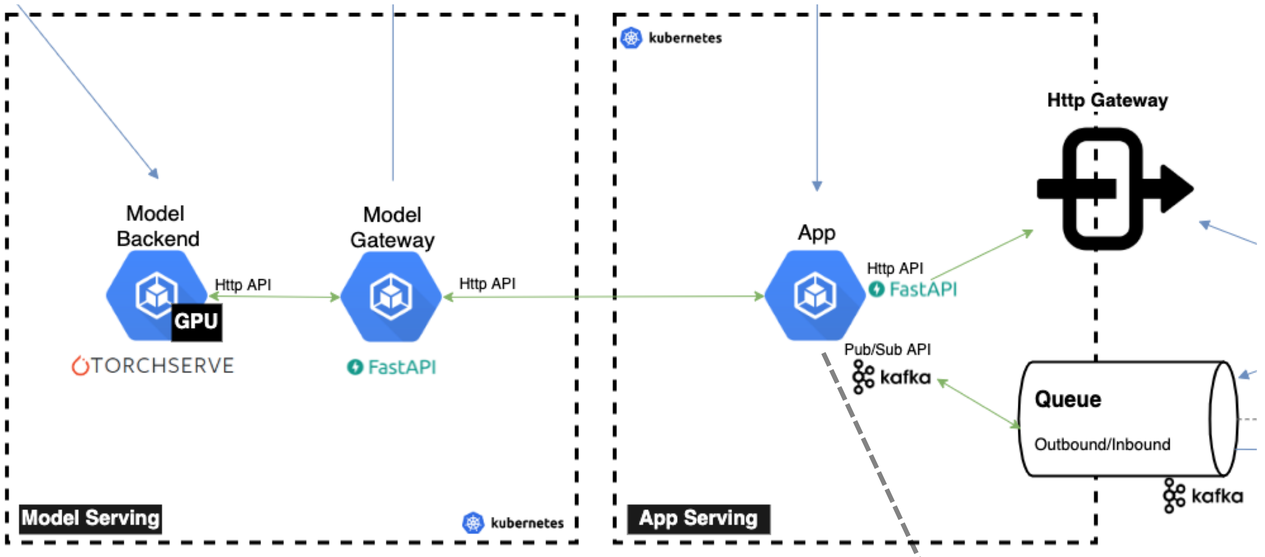
Model Backend

- **GPU** intensive
- tensor input > tensor output

Model Gateway

- **CPU** intensive
- 데이터 포맷 전처리/후처리
- model prediction output store (same input > same output)
- network entrypoint (사내 gpu 배포 환경 파편화 대응, 버전 관리)
- model backend 공유 (같은 model을 여러 app에서 사용)
- 모델 통합 API 제공 + 문서 (swagger)

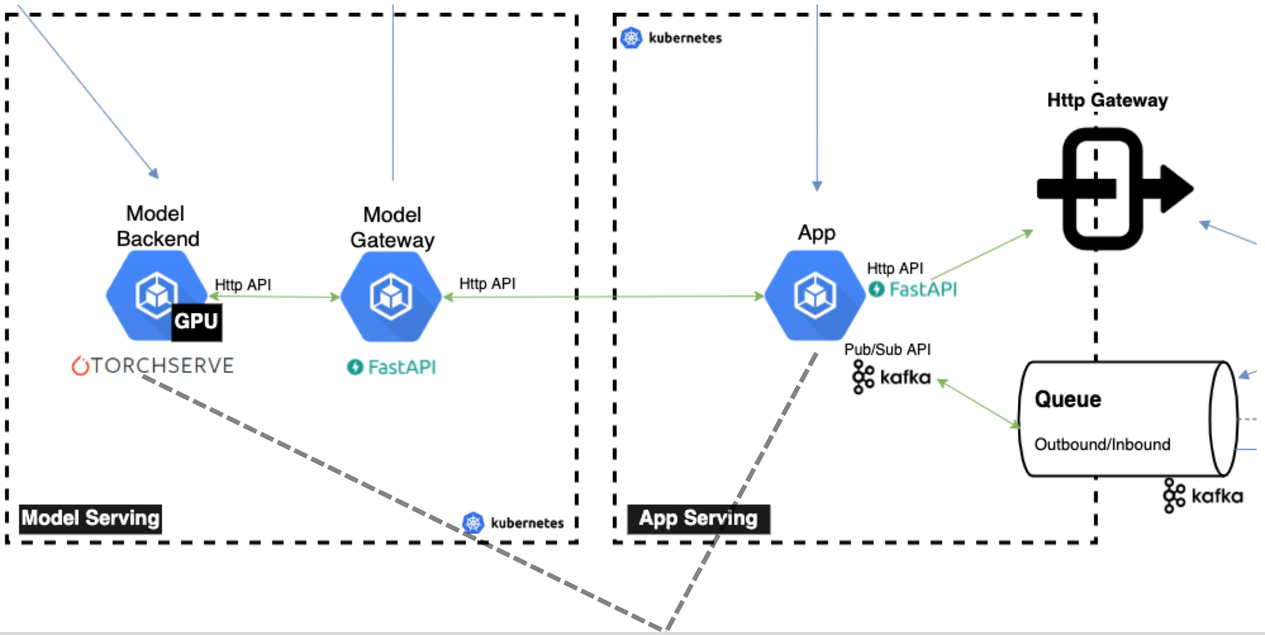
모델 / 앱 배포



App Serving

- 비지니스 로직 처리
- Web API
- Pub/Sub API (kafka producing / consuming)

모델 / 앱 배포



airflow dag - blue-green deployment

2개 이상의 helm chart를 배포

- (n2c pipeline의 blue-green 배포 기능은 1개의 helm chart)

배포 순서

- 신규 모델 배포 > 신규 앱 배포 > LoadBalancer 업데이트 > 이전 앱 제거 > 이전 모델 제거

3.6 결과 전달

전체 유저 분석 및 결과 전달

DEVIEW 2021

프로젝트 공통

1.datasource

- 원본 데이터

2.datamart

- 프로젝트 공통 ML 분석용 소스 데이터

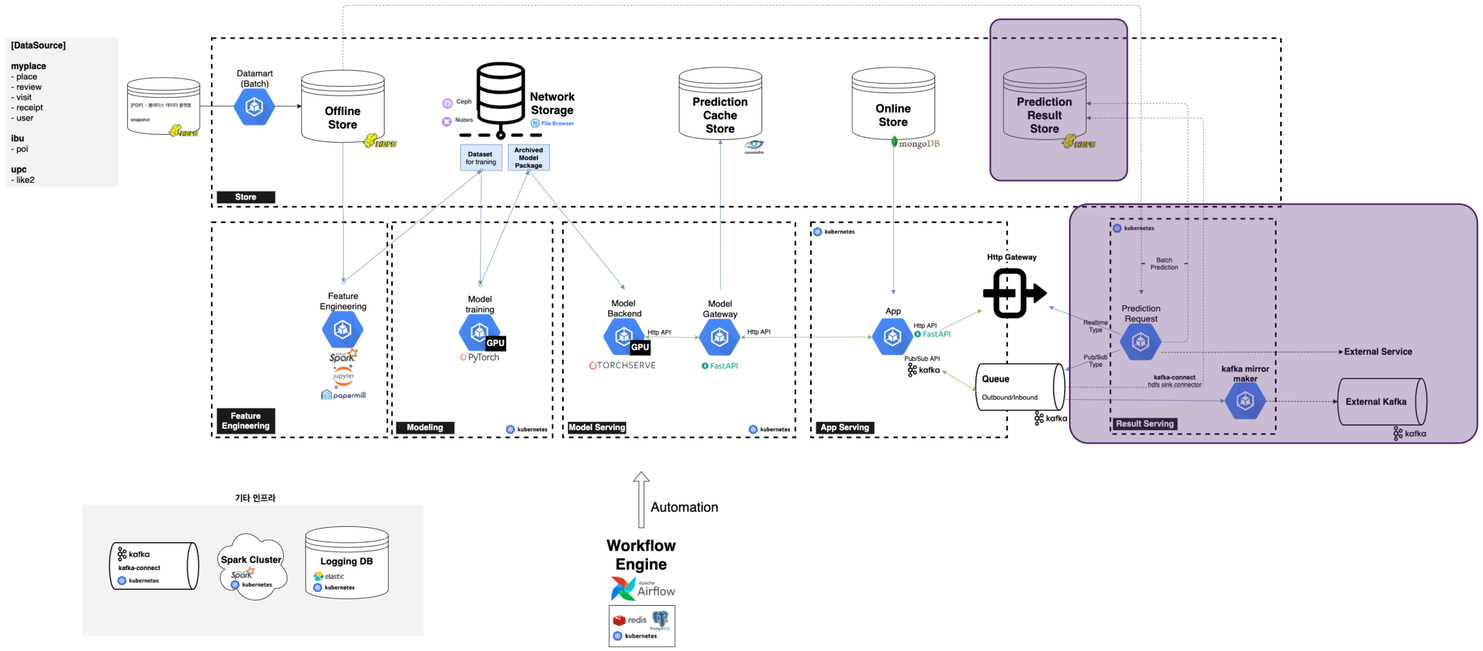
사용자 추천 과제

1.feature engineering

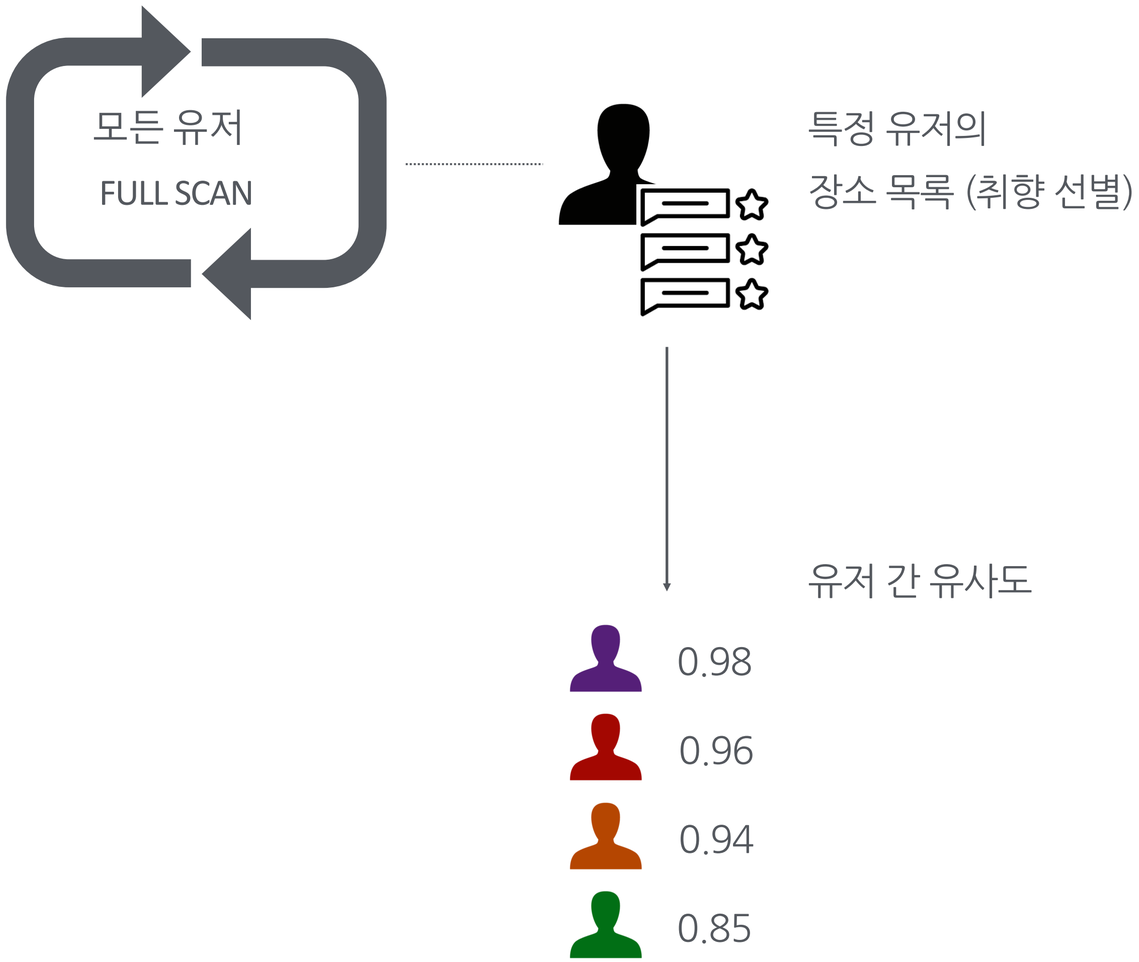
2.모델 학습

3.모델 / 앱 배포

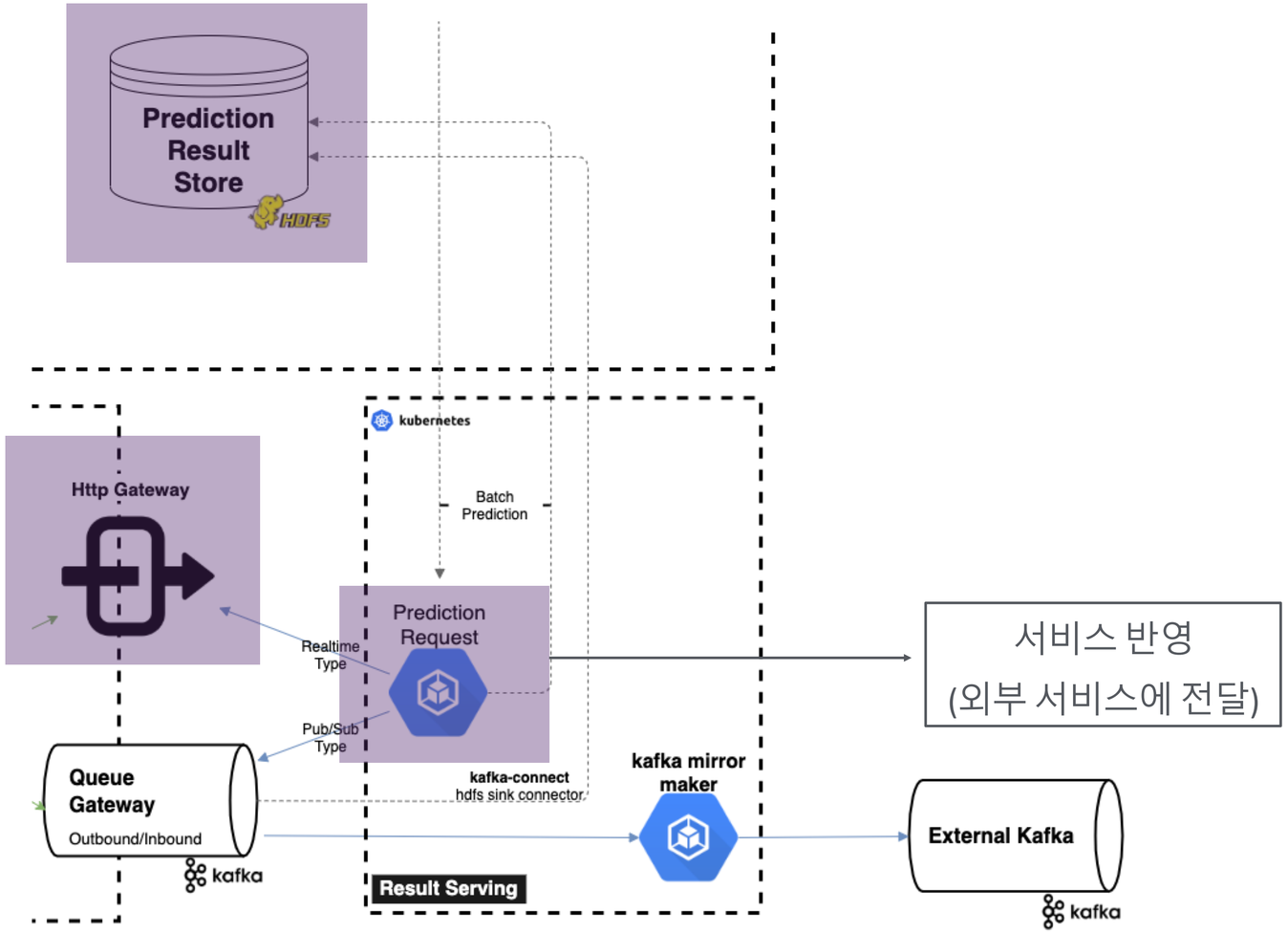
4.전체 유저 분석 및 결과 전달



전체 유저 분석 및 결과 전달



전체 유저 분석 및 결과 전달



전체 유저 분석 및 결과 전달

Apache zeppelin notebook으로 기획자 / 개발자간 결과 공유

SQL / pyspark 쿼리 sample code 가이드

SQL

Took 0 sec. Last updated by anonymous at June 16 2021, 12:56:52 PM.

```
%sql
SELECT idno, payload.* FROM result WHERE idno = quer_idno
```

```
%pyspark

target_user_review_df = target_user_df.select(
        F.explode("review_list").alias('review_item')
    ).select(
        F.col('review_item.*')
    )

z.show(target_user_review_df)
```

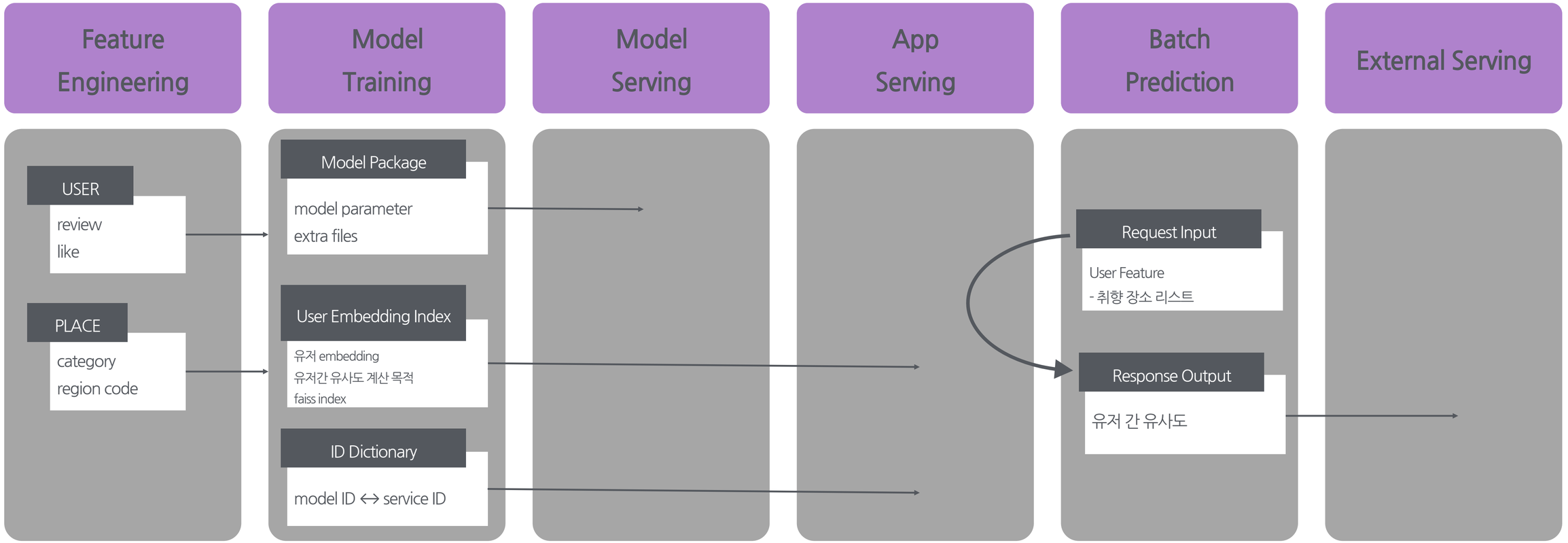
SPARK JOB FINISHED

settings

review_index_partition_by_user	review_id	review_rating	review_body	review_image_url	review_visited_date	review_place_id	review_media_classification_list
3	[redacted]	5.0		null	2020-09-01 09:00:00.0	[redacted]	WrappedArray()
4		4.5		null	2020-08-12 09:00:00.0		WrappedArray()
5		4.5		null	2020-07-12 09:00:00.0		WrappedArray()
6		4.5		null	2020-07-12 09:00:00.0		WrappedArray()
7		4.5		null	2020-06-14 19:44:03.0		WrappedArray()
8		4.5		null	2020-06-11 15:38:00.0		WrappedArray()

3.7 ML Pipeline 요약

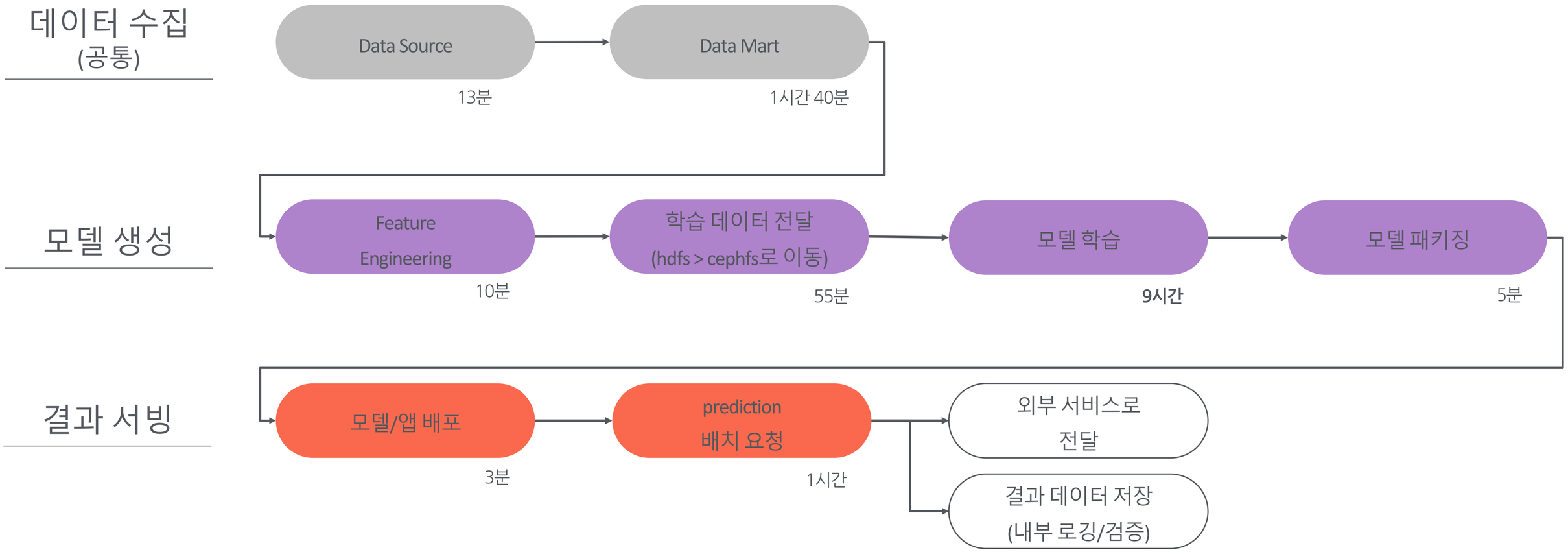
Data Flow


Feature Engineering
Model Training
Model Serving
App Serving
Batch Prediction
External Serving
USER
review
like
PLACE
category
region code
Model Package
model parameter
extra files
User Embedding Index
유저 embedding
유저간 유사도 계산 목적
faiss index
ID Dictionary
model ID ↔ service ID
Request Input
User Feature
- 취향 장소 리스트
Response Output
유저 간 유사도

총 소요 시간

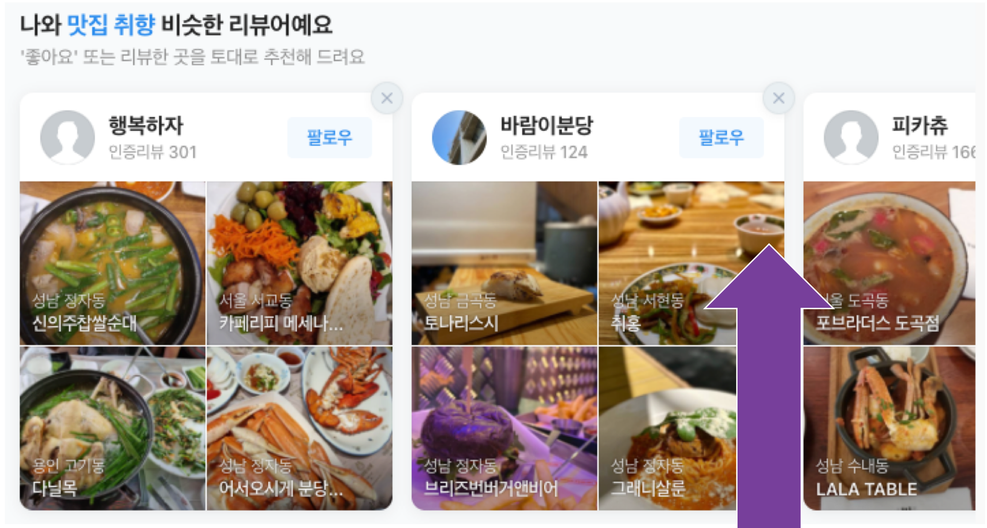
총 소요시간: 13시간

(2021-06-13일 기준)



마치며

마치며


나와 맛집 취향 비슷한 리뷰어예요
'좋아요' 또는 리뷰한 곳을 토대로 추천해 드려요
행복하자
인증리뷰 301
팔로우
바람이분당
인증리뷰 124
팔로우
피카츄
성남 정자동
신의주찹쌀순대
서울 서교동
카페리피 메세나...
성남 금곡동
토나리스시
성남 서현동
취홍
서울 도곡동
포브라더스 도곡점
용인 고기동
다닐목
성남 정자동
어서오시게 분당...
성남 정자동
브리즈번버거앤비어
성남 정자동
그래니살룬
성남 수내동
LALA TABLE

ML Modeling
지속
ML Ops

Thank You