MLops와 ML 모델러와 ML 엔지니어의 협업

우리는 유저에게 더 좋은 ML 서비스를
더 빠르게 제공하기 위해

Machine Learing

Deep Learning

Machine Learing

우리는 유저에게 더 좋은 ML 서비스를
더 빠르게 제공하기 위해

어떻게 하면 우리는 잘 협업할 수 있을까?

어떻게 하면 우리는 잘 협업할 수 있을까?
알려드리겠습니다

어떻게 하면 우리는 잘 협업할 수 있을까?

~~알려드리겠습니다~~

가 아니라 여러분과 같이
고민해보고 싶었습니다

특정 기술이 아닌 협업을 함께 고민하는
심플한 담론

특정 기술이 아닌 협업을 함께 고민하는 심플(?)한 담론

근데 이제 이런 것들을 곁들인

ML-Ops

DevOps

ML pipeline

agile

ML modeling

ML engineering

test for ML

CI/CD for ML

workflow management

저는 한때
ML modeling과
ML engineering을
동시에 해야만 한적이 있었습니다

그런데 요상하게 업무의 스위칭 비용이
너무나 높았습니다

그런데 요상하게 업무의 스위칭 비용이
너무나 높았습니다

왜 그럴까?

달라도 너무나 다른 그들의
도메인 지식, 업무방식, 평가방식

달라도 너무나 다른 그들의
도메인 지식, 업무방식, 평가방식

도메인 지식

미적분
선형대수
확률/통계
각종 논문
ML framework
programming
...

ML modeler

programming
infra structure
분산시스템
data engineering
ML framework
operations
...

ML engineer

달라도 너무나 다른 그들의
도메인 지식, 업무방식, 평가방식

Machine Learning

머쉰 르닝

학습

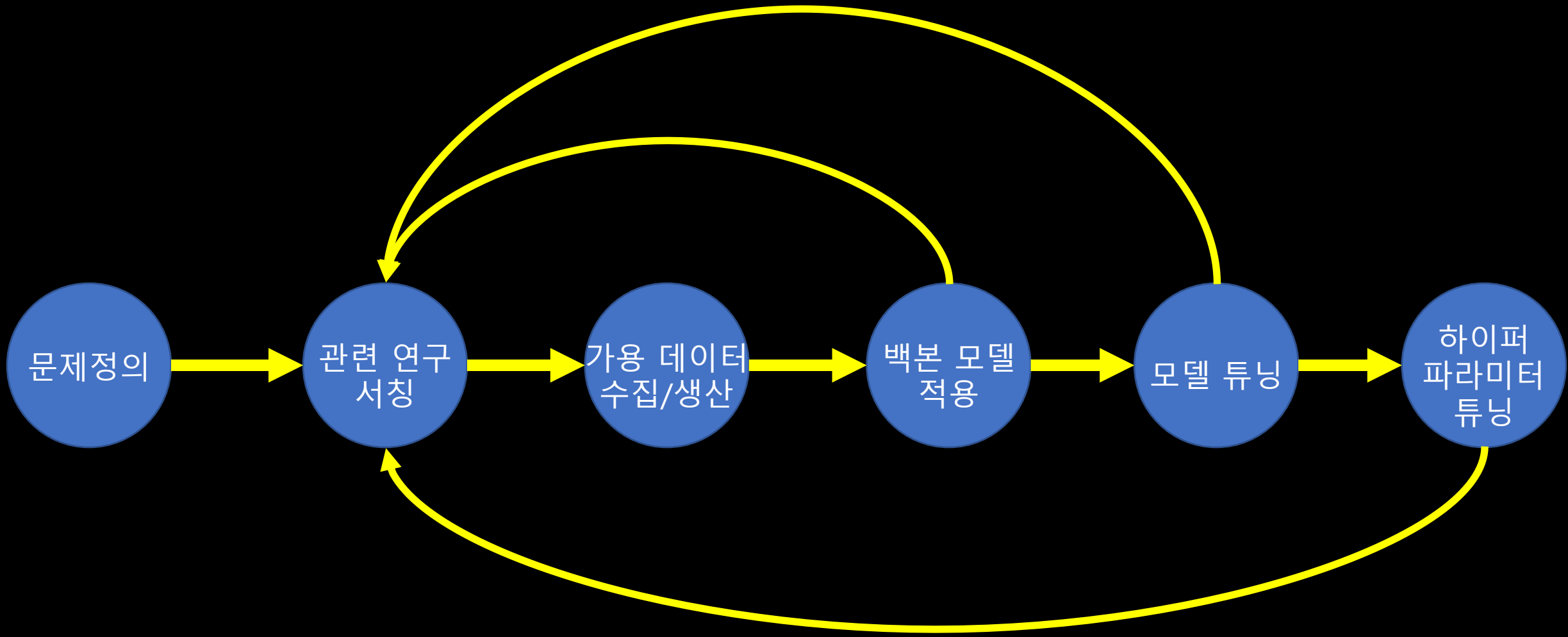
일단은.... 학습하는 놈이 있어야 하지 않을까?

학습하는 놈

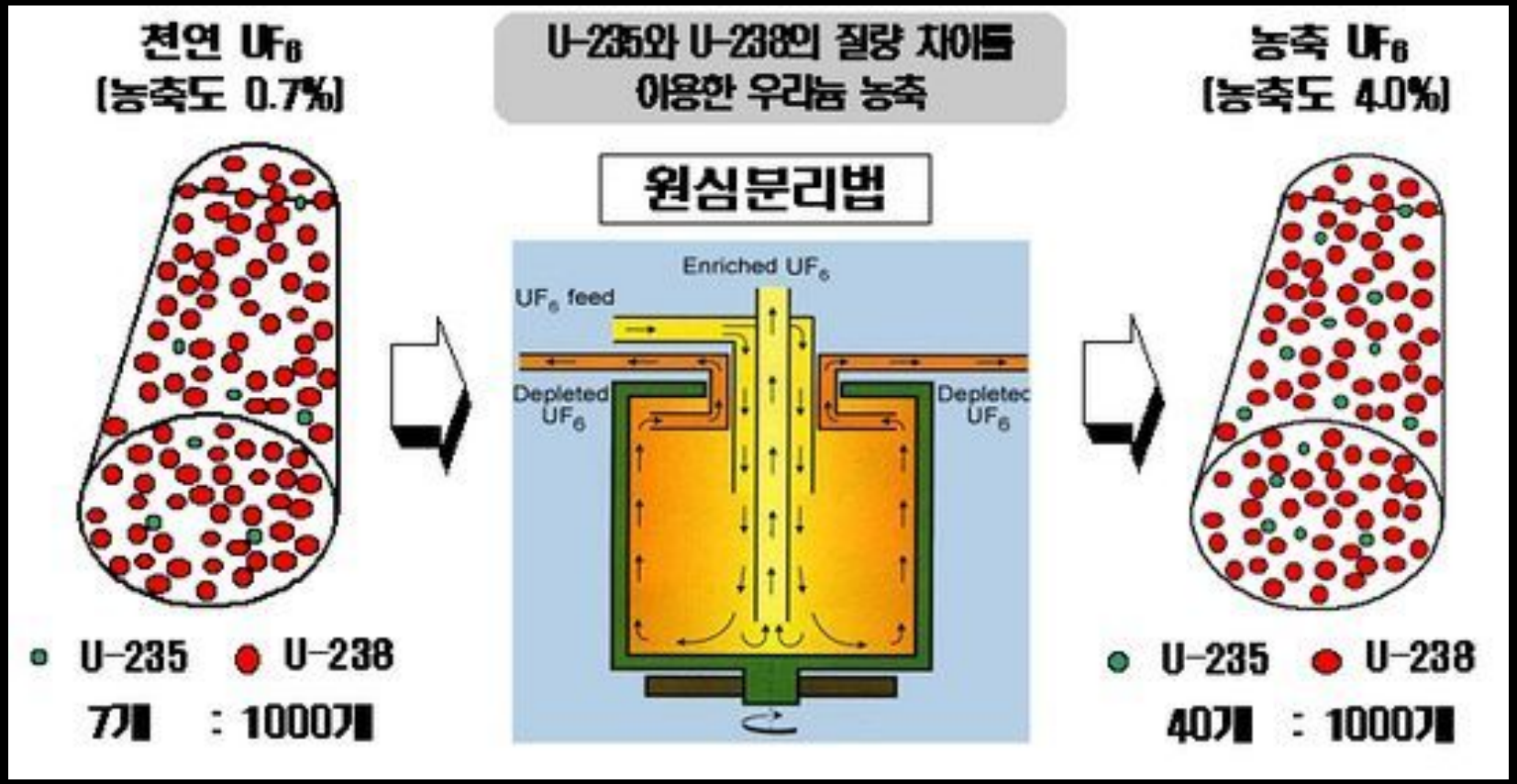
그거시 바로 **Model**

어떤 객체나 시스템, 현상, 개념을 설명하는 수단
잘 정의된 개념을 잘 정의된 구조로 잘 설명

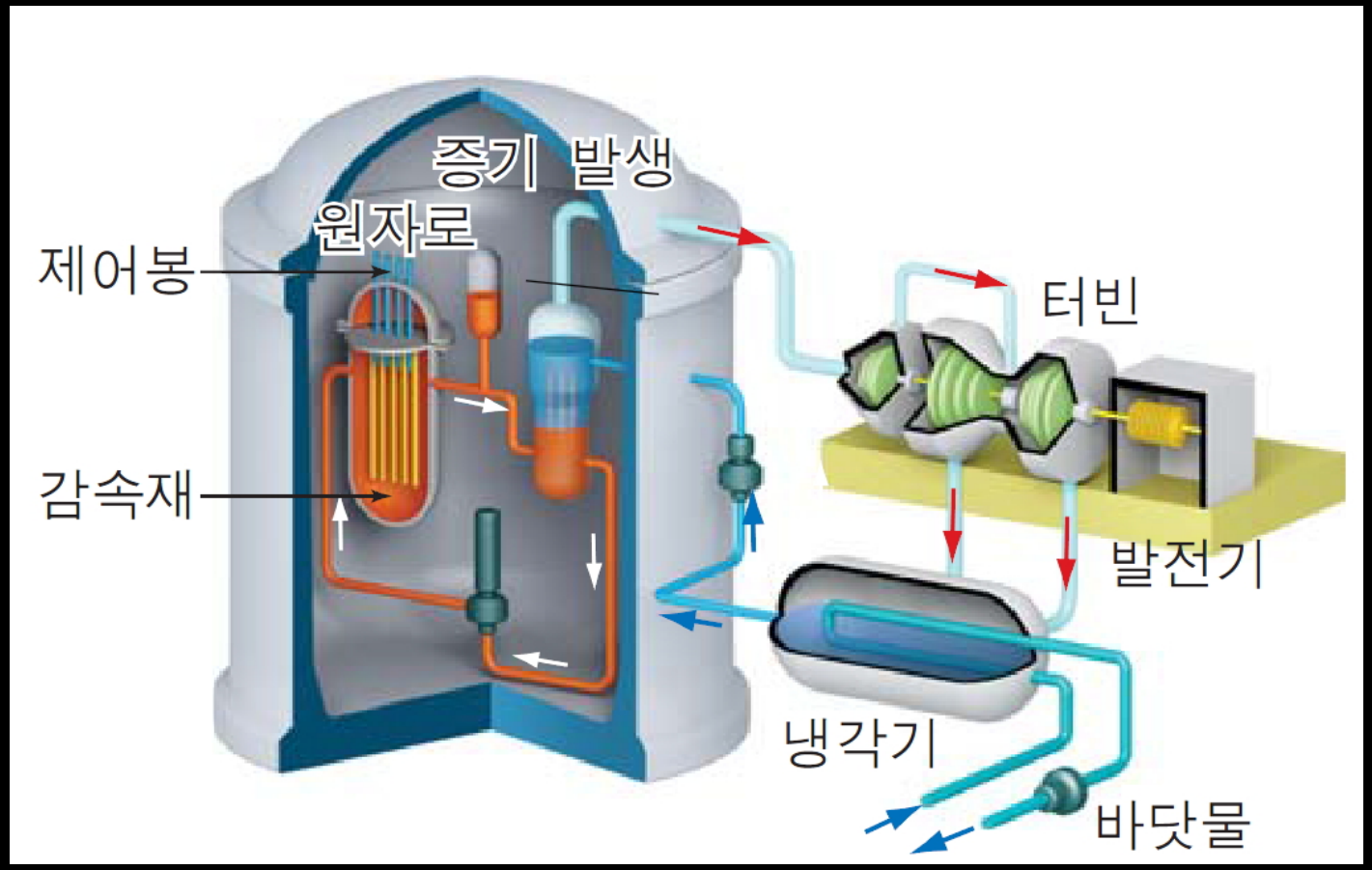
학습모델
학습을 통해 구조를 정의하여 데이터의 관계,
데이터를 통해 얻을 수 있는 정보를 설명


문제정의
관련 연구 서칭
가용 데이터 수집/생산
백본 모델 적용
모델 튜닝
하이퍼 파라미터 튜닝

ML 서비스는 모델만 가지고
만들어 질 순 없습니다

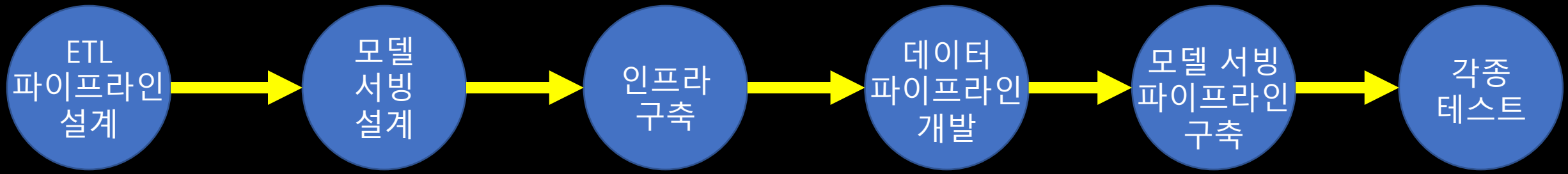


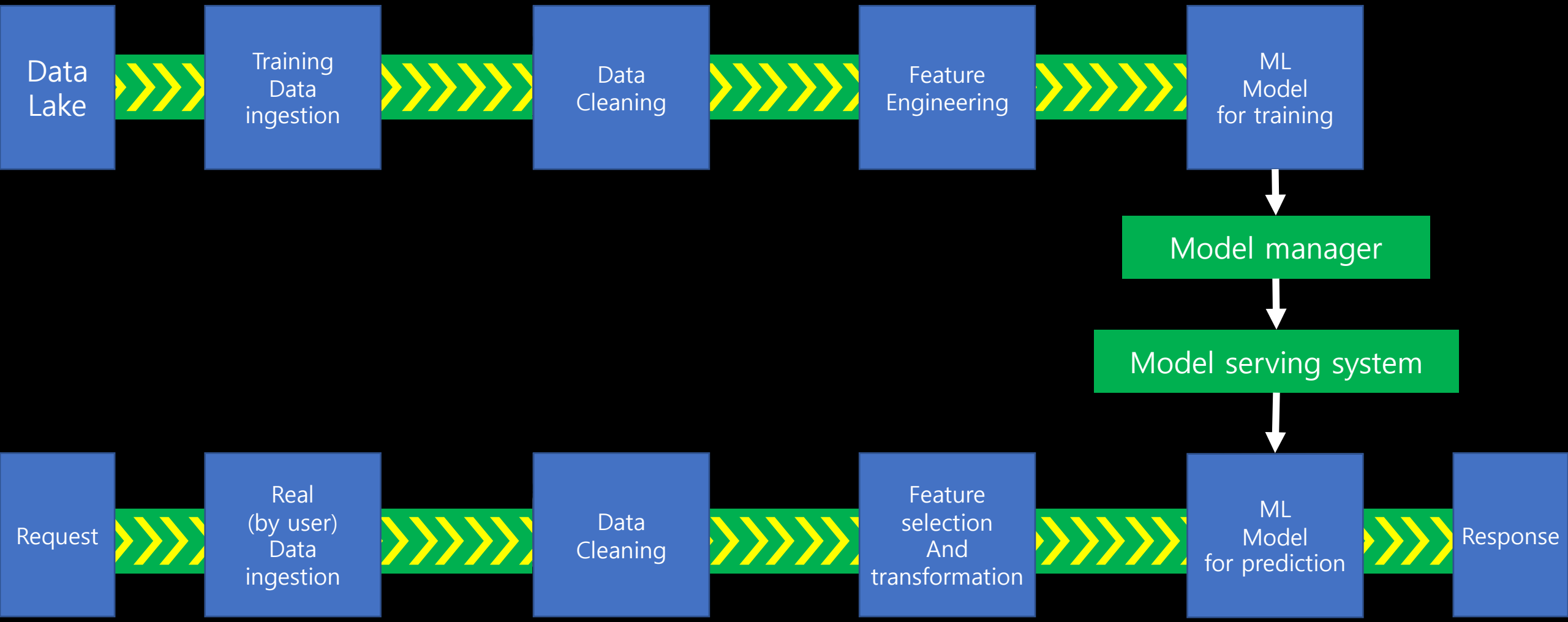
우라늄 농축기술만 있다고
전기 만드는거 아니잖아요?



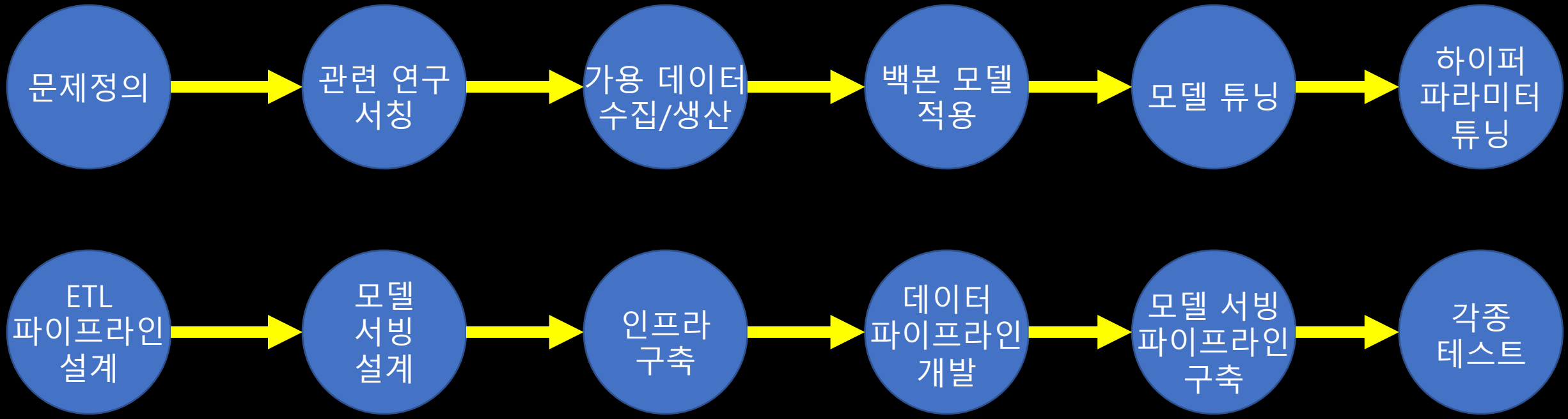
이런거 만드는 사람도
있어야 하지 않겠습니까?

ML engineer 등판


ETL
파이프라인
설계
모델
서빙
설계
인프라
구축
데이터
파이프라인
개발
모델 서빙
파이프라인
구축
각종
테스트


Data Lake
Training Data ingestion
Data Cleaning
Feature Engineering
ML Model for training
Model manager
Model serving system
Request
Real (by user) Data ingestion
Data Cleaning
Feature selection And transformation
ML Model for prediction
Response

문제는



업무 흐름과 방식이 너무나 다르다

과연 연구기반의 모델링 프로세스에서 에자일은 적합한 업무방식일까?

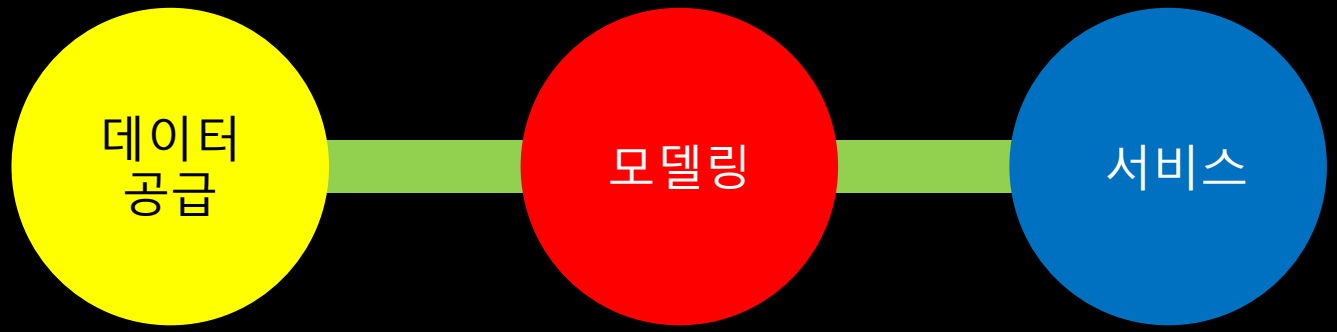
엔지니어는
하루치 작업을 하면
하루치 성과가 나오지만

연구자는
하루치 작업을 하면
성과가 나오는지 안 나오는지는
하늘만 알고계신다

그리고 상이한 업무방식을 어떻게
엮는 가장 좋은 방법은?

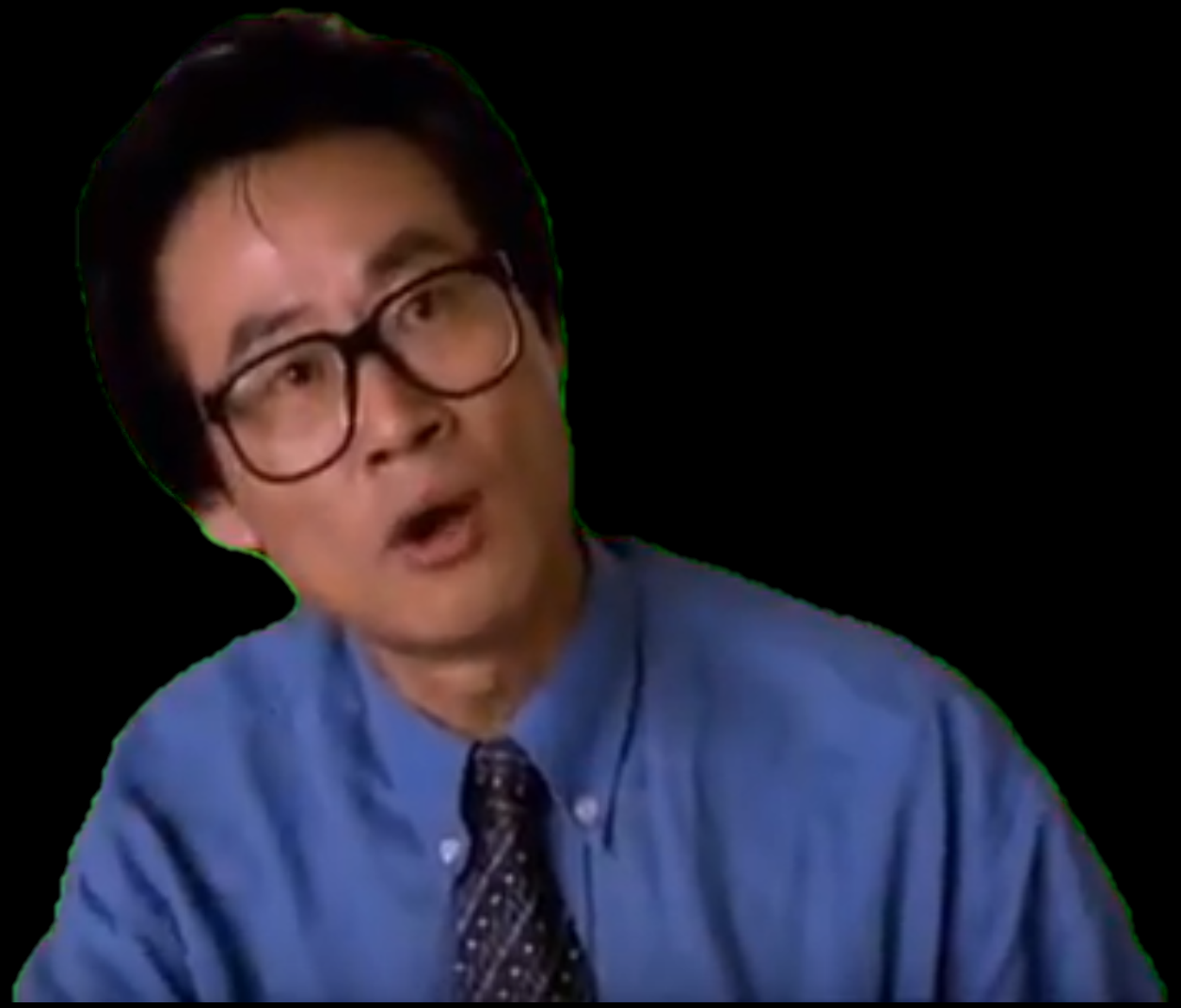
그리고 어떻게 하면 협업을 통해
빨리 시장에 서비스를 내놓을까?

Project with Data science



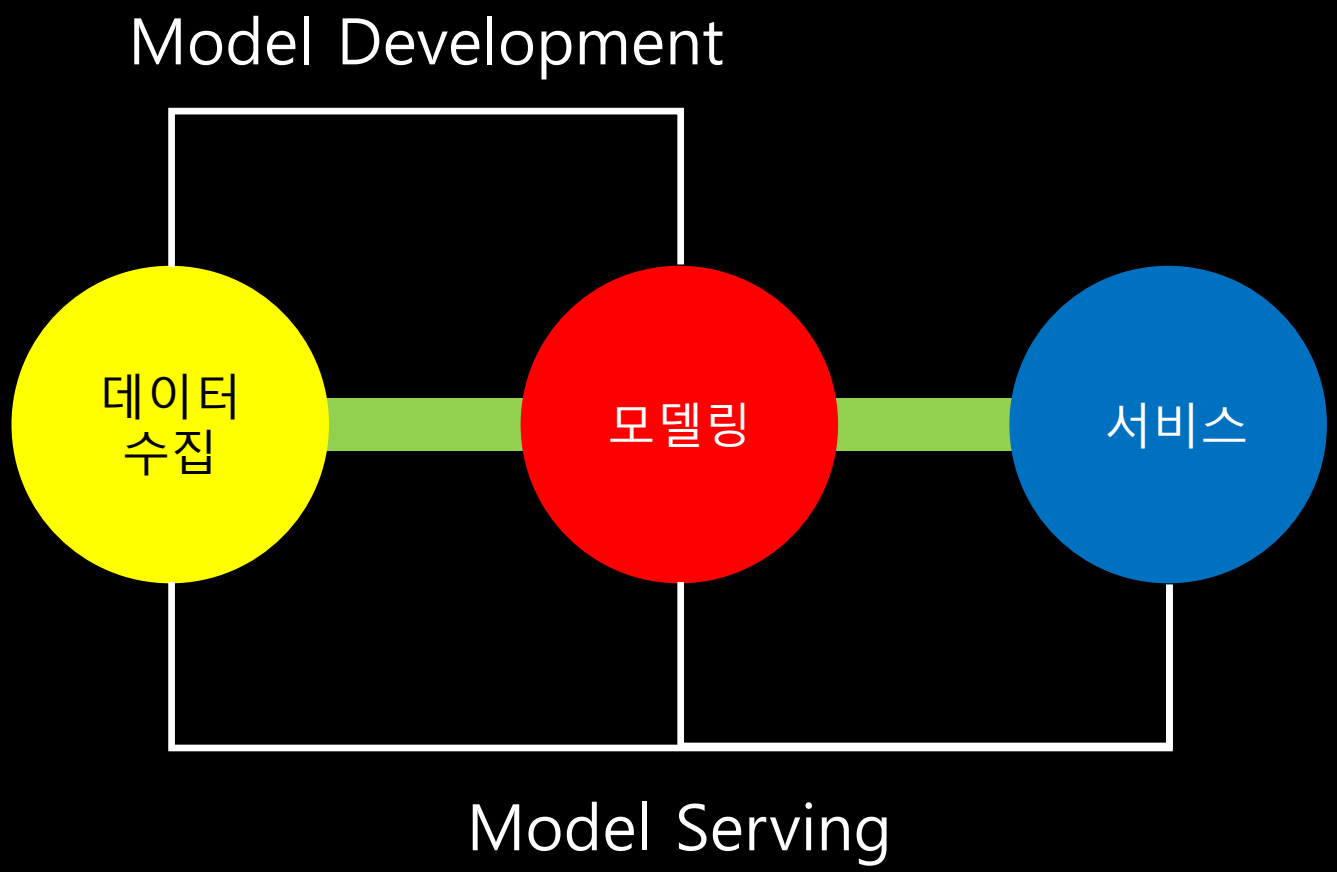
머리 가슴 배


백선생표 만능간장
얼마나
간단한지 보세요



이거 너무 간략한거 아니냐고

그게 실제론 말이죠



그게 실제론 말이죠

- Model Development
- Model Serving

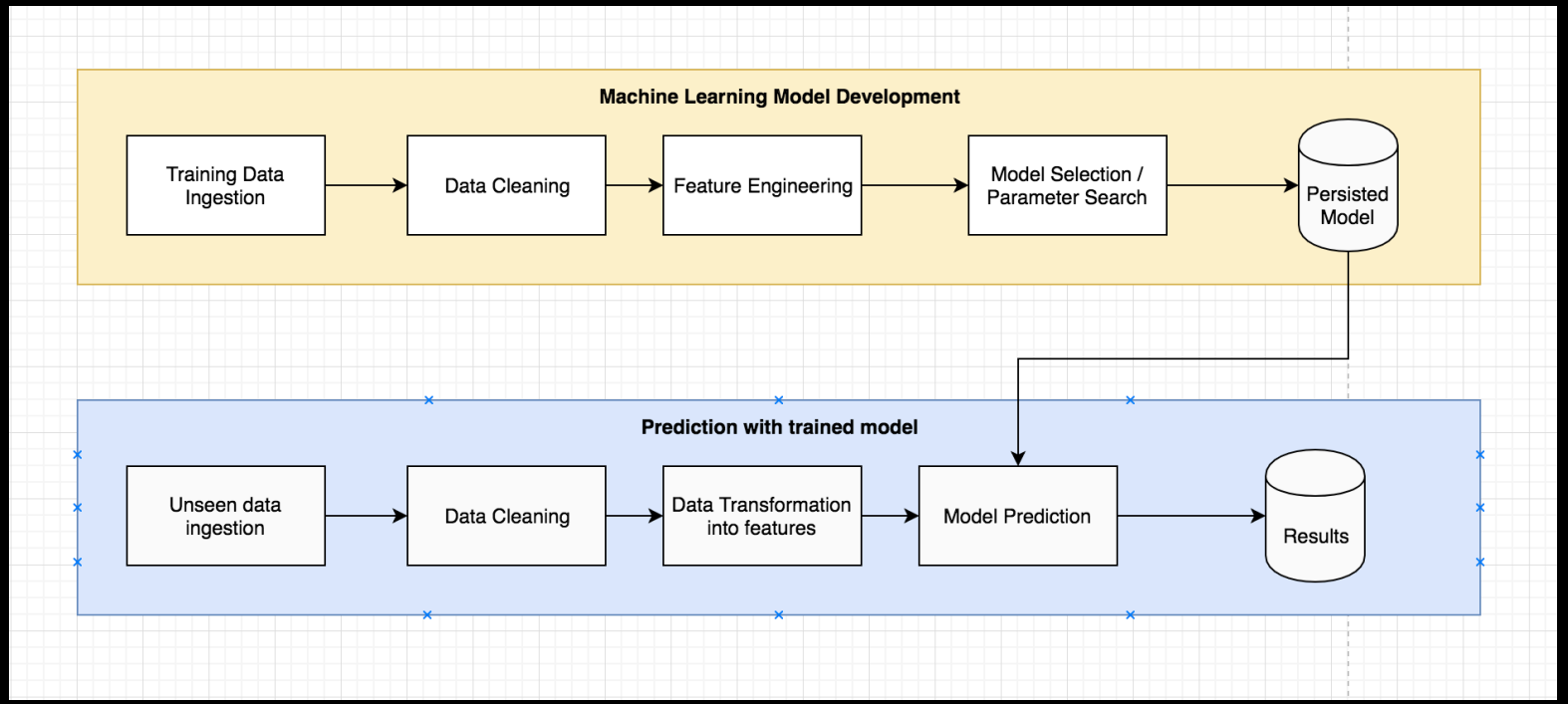


Image : https://axsauze.github.io/scalable-data-science

작은 조직의 프로젝트라면

신경써야 할 모델의 수가
그다지 많지 않습니다.

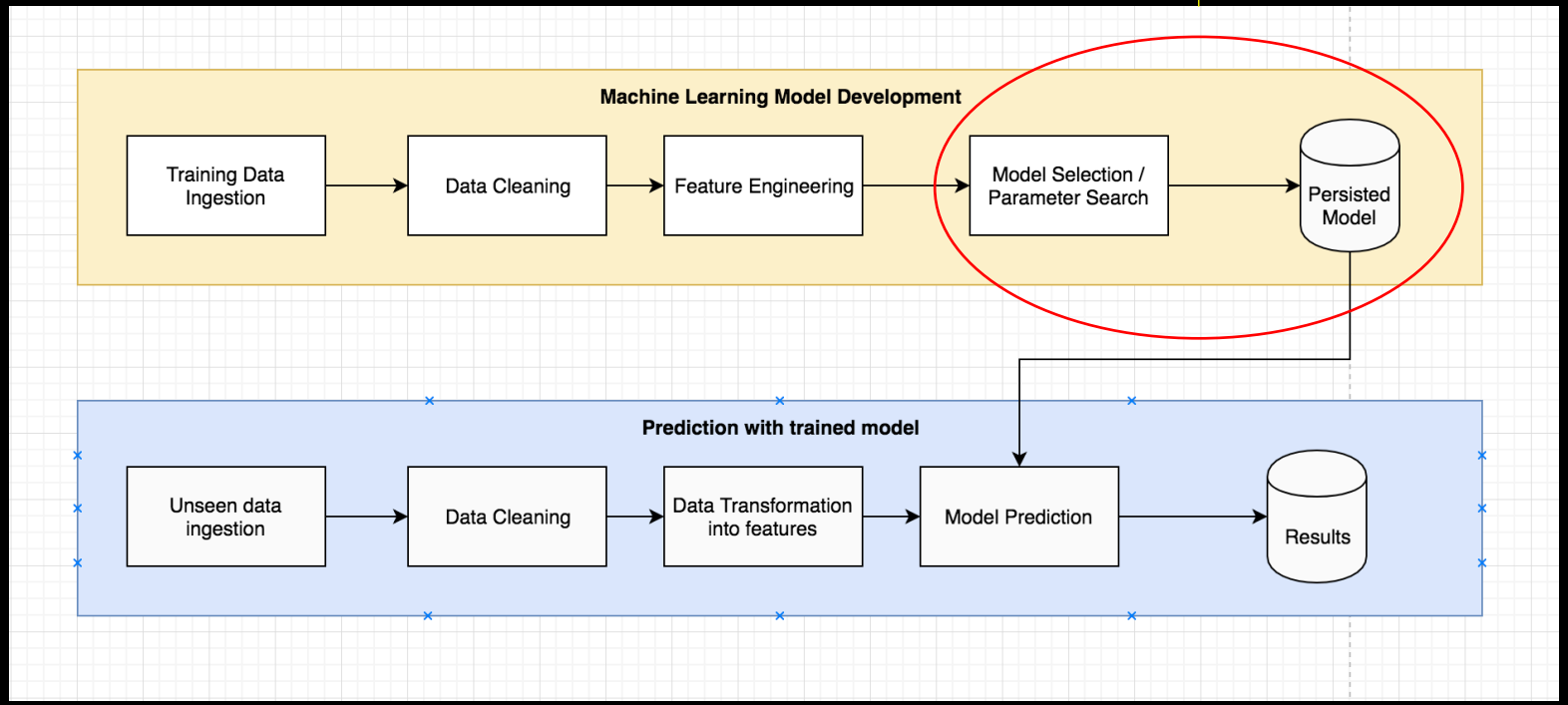


Image : https://axsauze.github.io/scalable-data-science

작은 조직의 프로젝트라면

데이터 사이언티스트(모델러)의
수가 적기 때문에 모델링
협업에 특별히 문제가 없습니다.

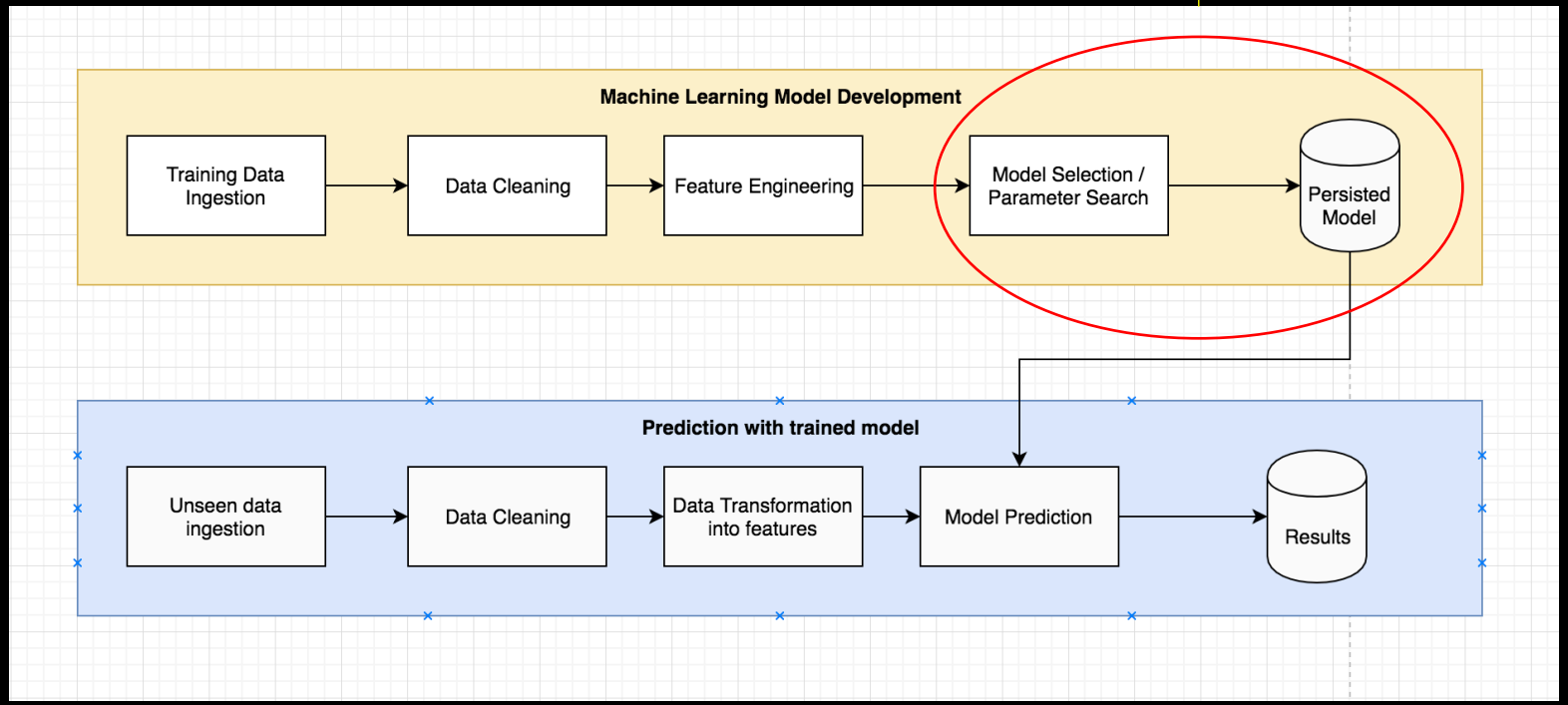


Image : https://axsauze.github.io/scalable-data-science

작은 조직의 프로젝트라면

워크플로우의 수가 적기 때문에
각 단계를 각각 별도의 방법으로
트레킹을 해도 무방하다.

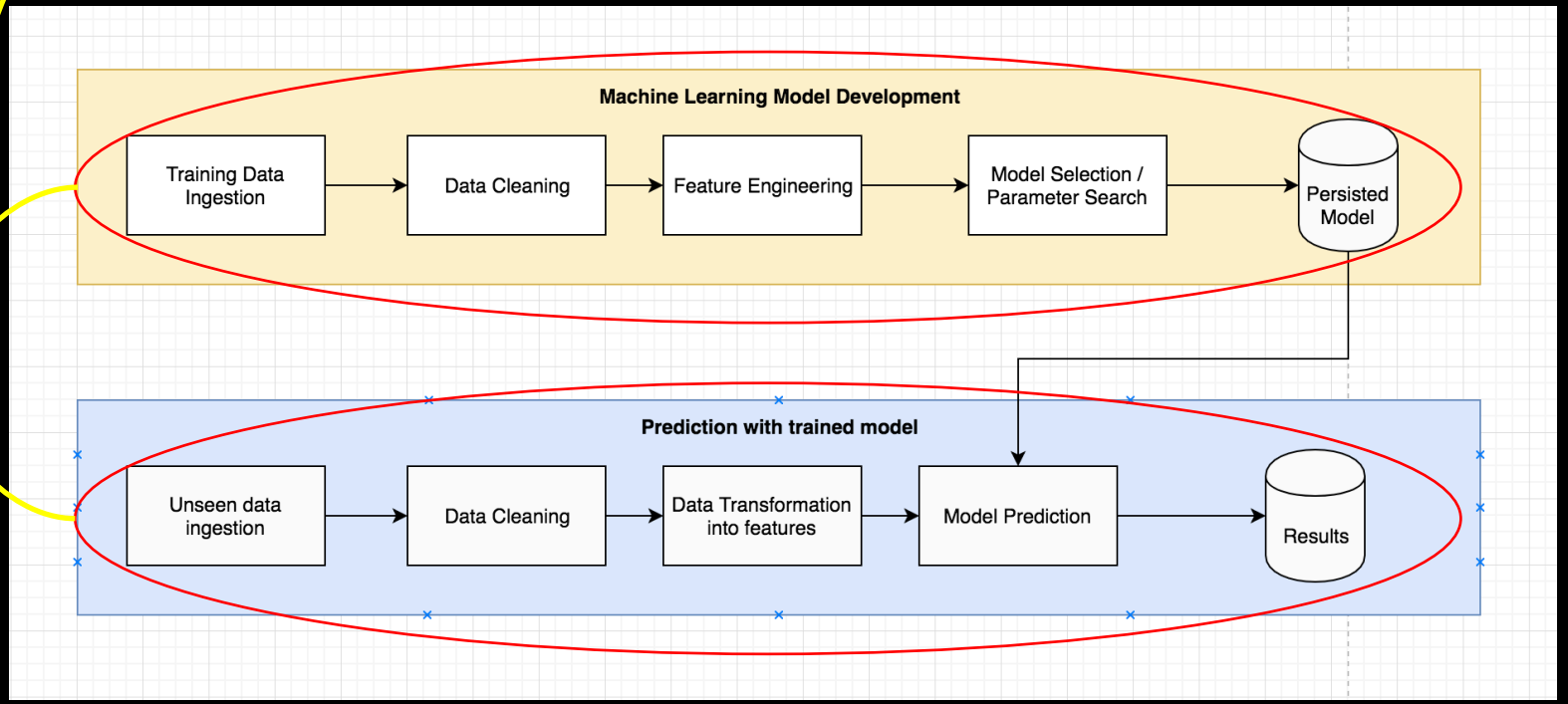


Image : https://axsauze.github.io/scalable-data-science

하지만

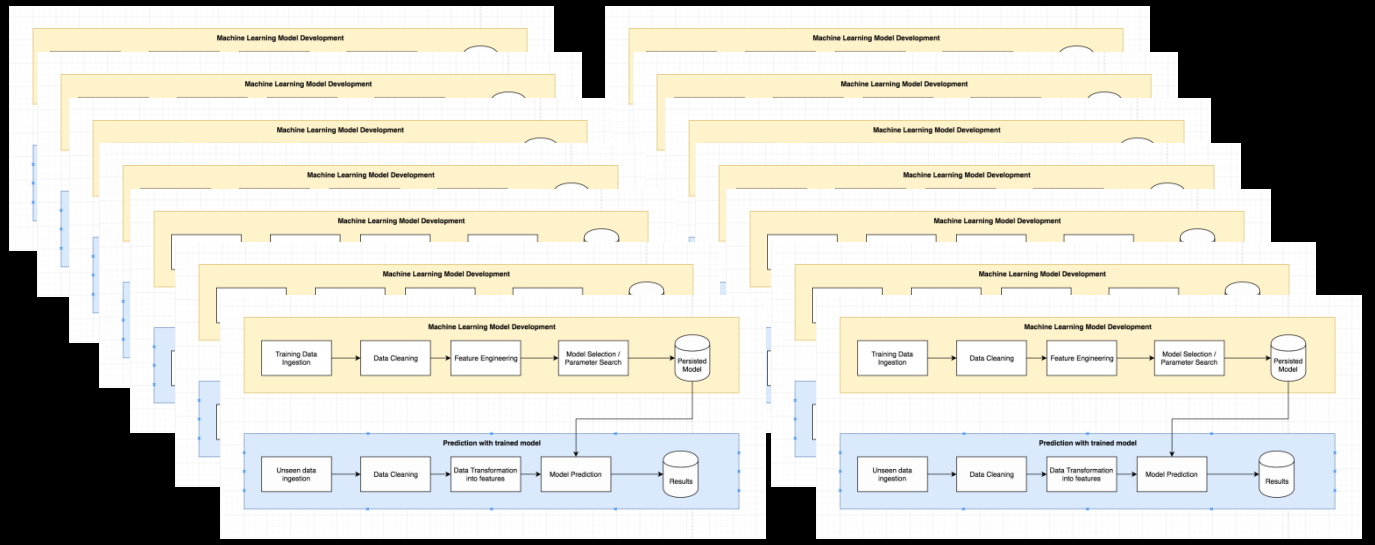


Image : https://axsauze.github.io/scalable-data-science

규모가 커지면 이야기가 달라진다

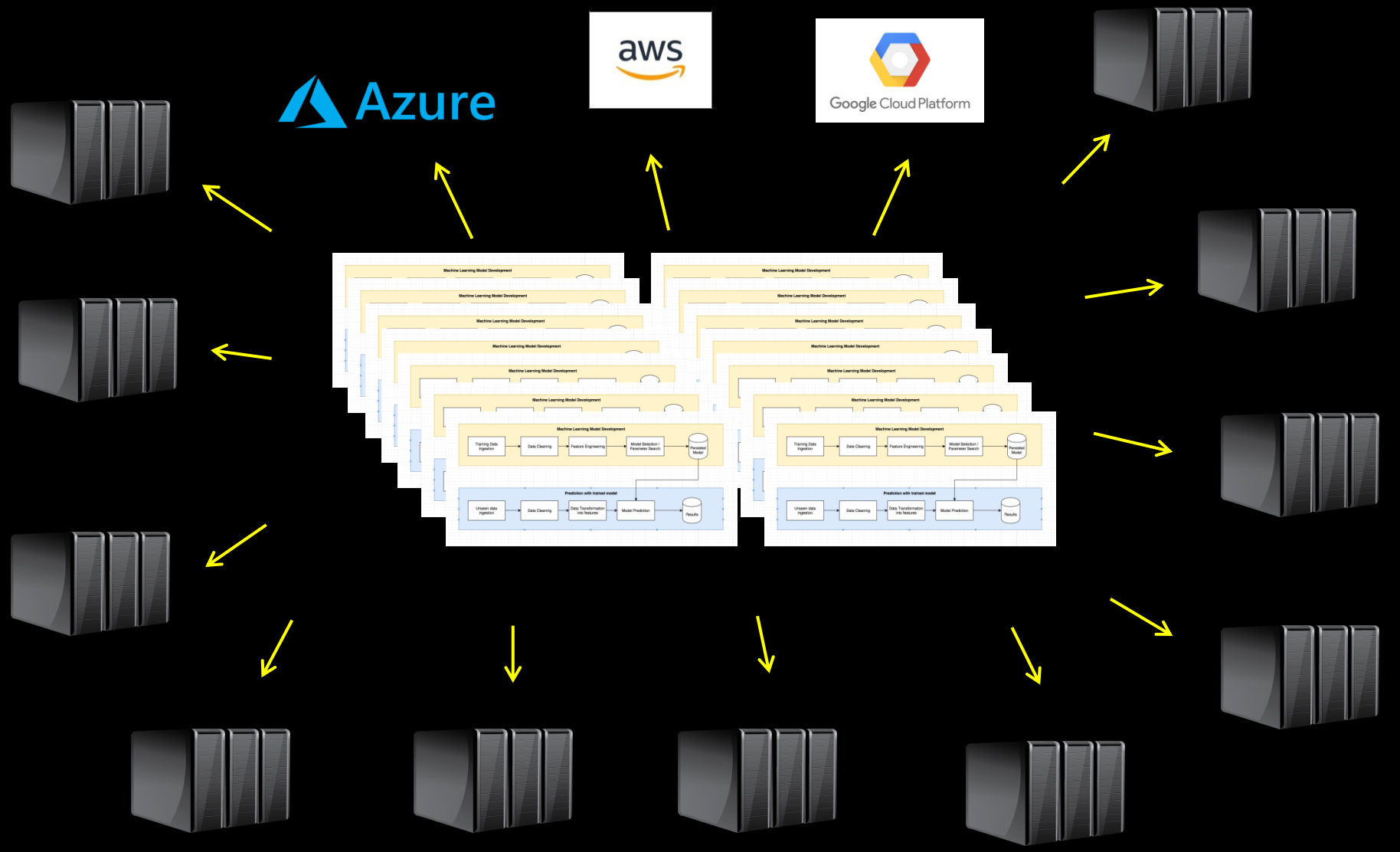
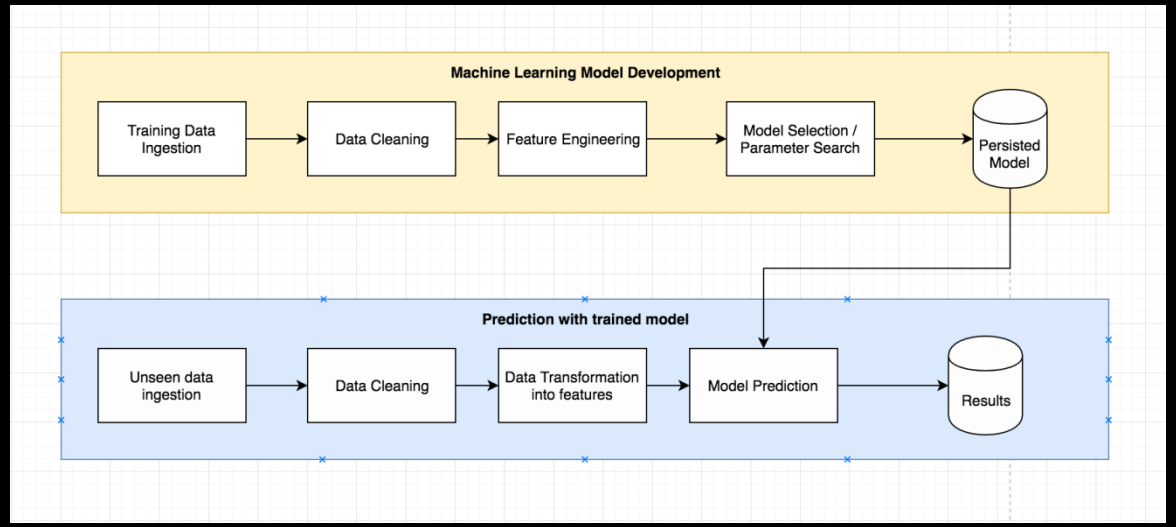


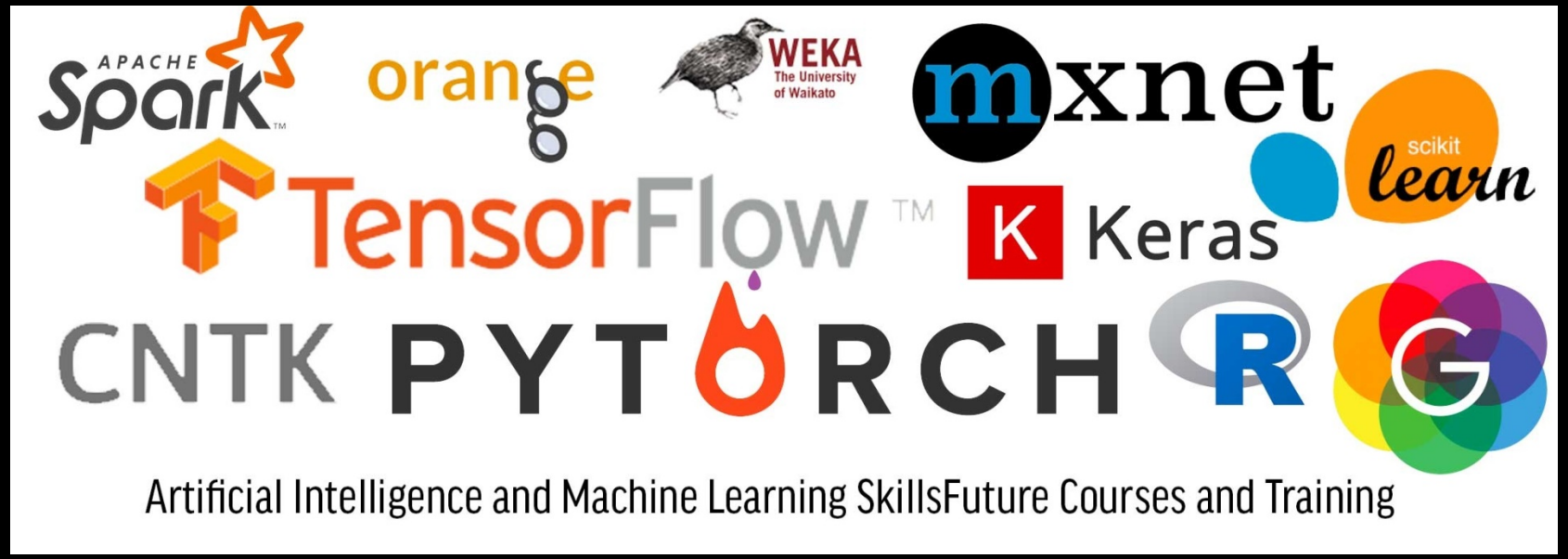
Image : https://axsauze.github.io/scalable-data-science

데이터 플로우의 복잡성 증가

- 많은 데이터 워크플로우의 수
- 표준화된 추적없이 데이터가 수정됨
- 규모가 커질수록 플로우와 스케줄링의 복잡성을 관리하는 것은 불가능해짐

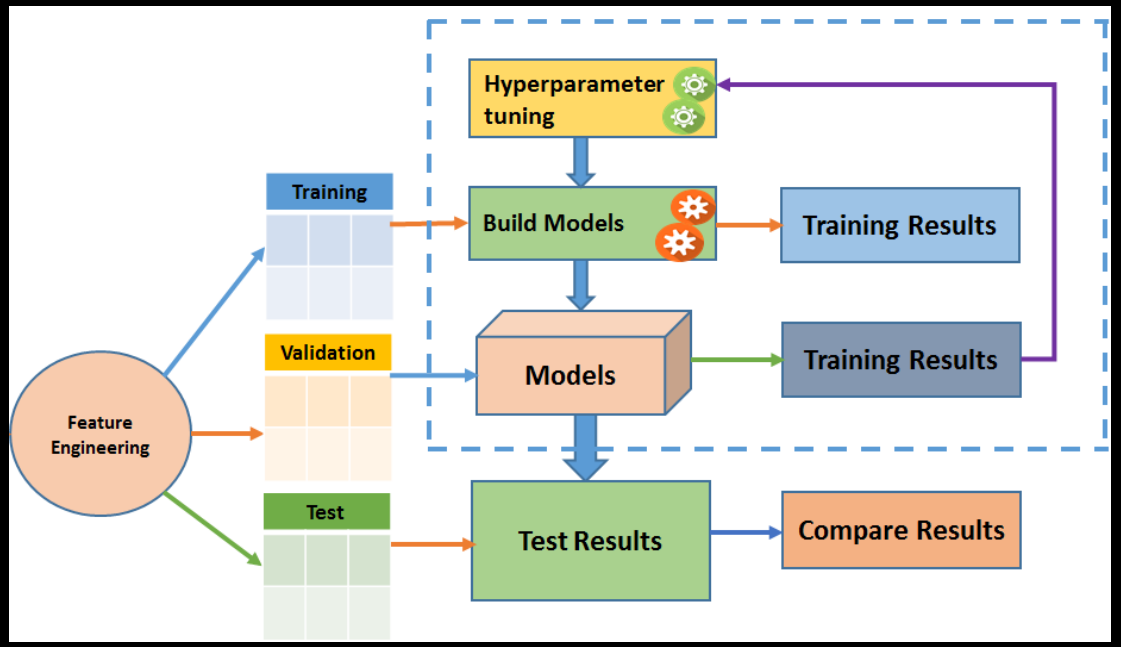


각자 선호하는 툴은 다 다릅니다.



까다로워지는 모델 서빙

- 각자 다른 버전의 모델이 각자 다른 환경에서 실행
- 점점 더 모델 배포와 버전 되돌리는 것이 복잡해짐



따라서 까다로워지는 track back

우리같은 엔지니어는
굳이 논문같은거 안보고도
모델이 잘못됐다는 걸
알 수 있습니다.

VS

데이터 파이프라인에
벌래같은 거 키우니까
데이터가 잘못 들어와서
모델이 뻑나는 거잖아

DATA engineer

그래서 전 뭐하는데요?

ML engineer

DATA scientist

어....그러니까 음....

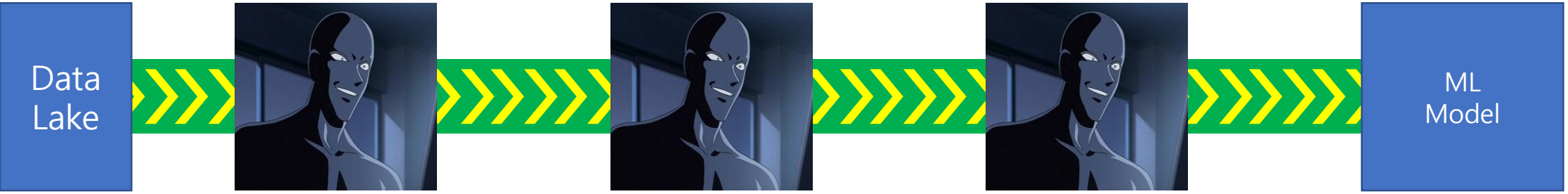
전부다?!


굳이... 할 필요가

ML engineer 는
데이터 파이프라인, 모델의 개발
그리고 서빙까지 신경써야 합니다.

ML-OPS

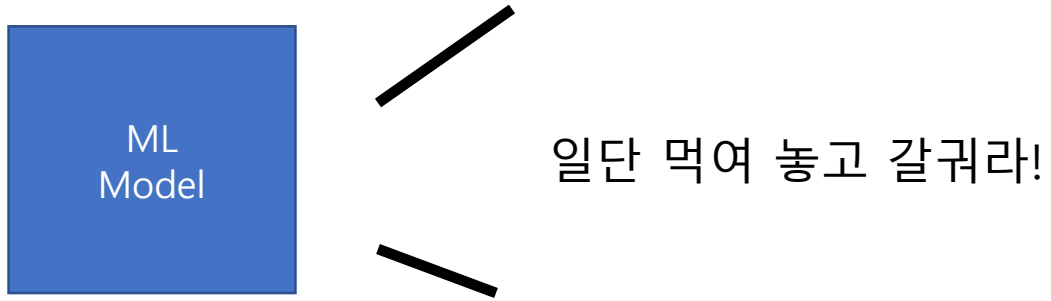
Learning workflow



Service workflow

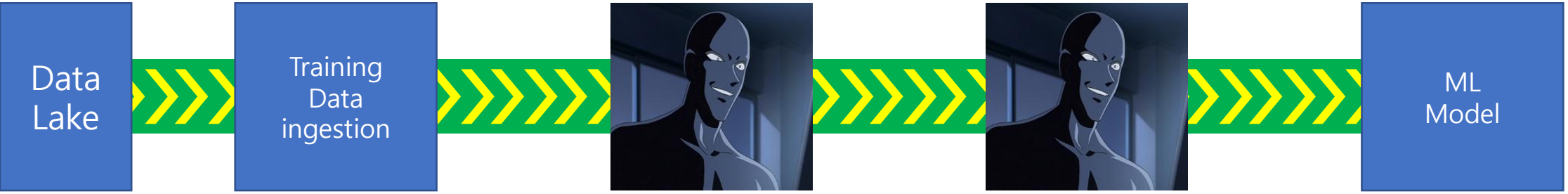
학습하려면 필요한건?

DATA



일단 먹여 놓고 갈궈라!

Learning workflow



Service workflow

근데 어떤 데이터?

~~~쁜 DATA

Yee~~쁜 데이터를 얻기위한

preprocessing
~~~

Yee~~쁜 데이터를 얻기위한

preprocessing

Data Cleaning

쓸데없는 부분 제거
노이즈 제거
이상한 부분 수정

그렇다고 프로토스 행성정화 마냥 싹 태우면 안됨

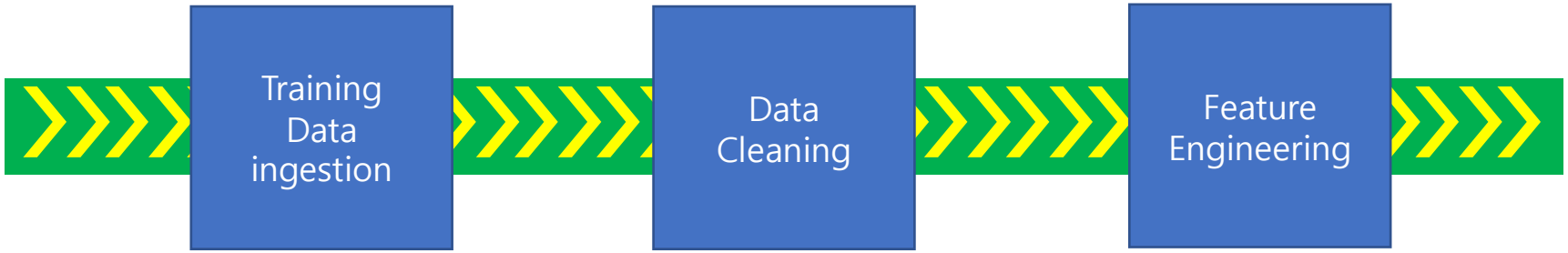
Feature engineering

모델이 좋아하는 형태로 변환 : Dataset
특징 선택(feature selection)
차원 축소(dimension reduction)

그 피처말고 임뫄

Learning workflow

Service workflow



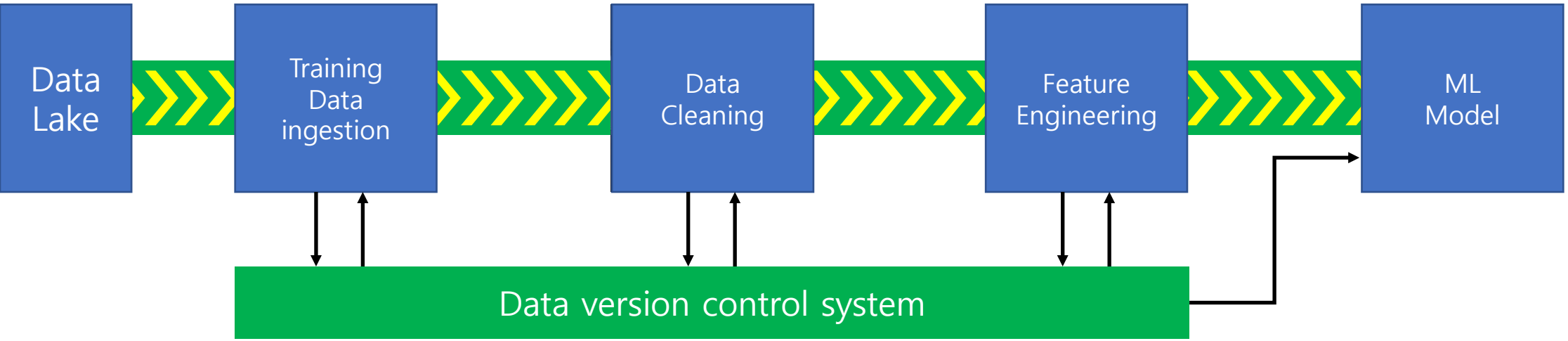
Data는 처리될 때마다 계속 바뀝니다.
모델에 따라 처리되는 형태가 바뀌기도 합니다.

이럴 때 필요한 것은 무엇인가

Versioning

데이터도 CI/  가 필요하다~ 이  이야

Learning workflow



Service workflow

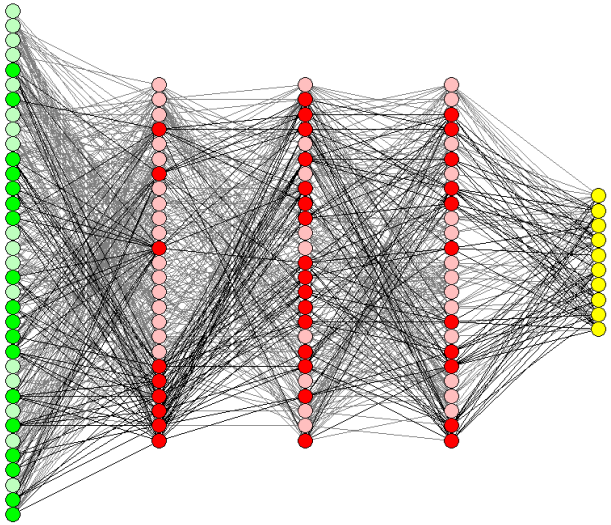
우리

얘 이야기 좀 더 해봅시다

Modeler

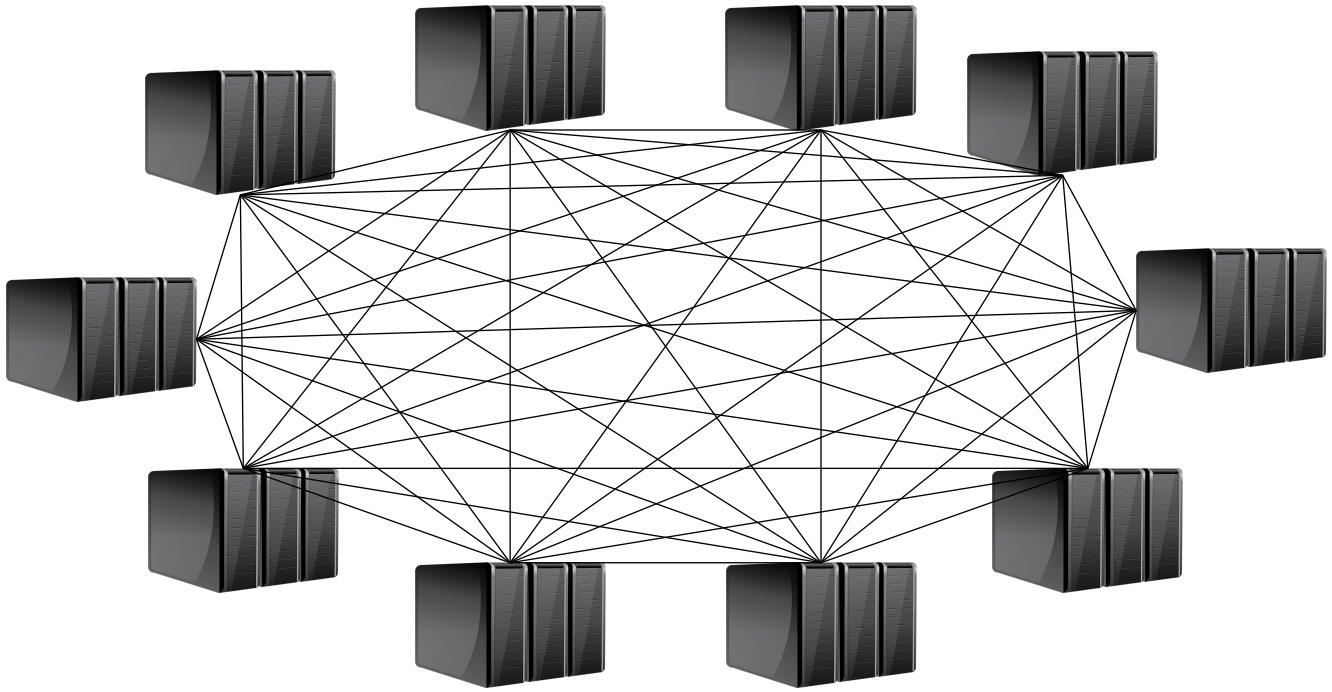
쟤는 (Researcher) 가 만듭니다.

그 사람들은



이런 거 설계하기도 바쁩니다.
논문도 보고 말이죠

그런데 학습모델을 돌리려면



넘나 많은 GPU를

넘나 분산된 환경에서

운영하(고 싶어하)게 됩니다.

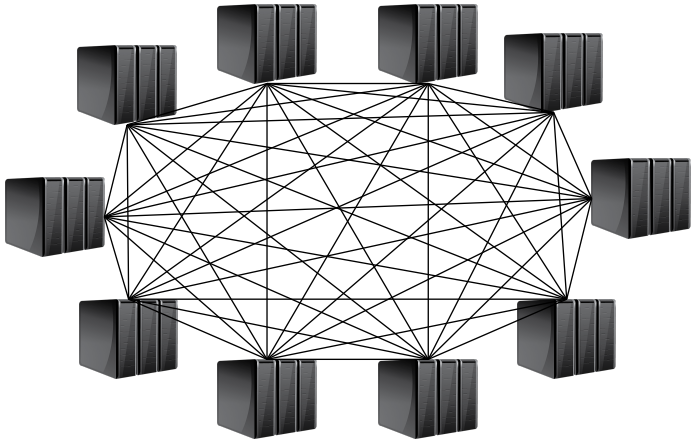
~~행복한 고민~~

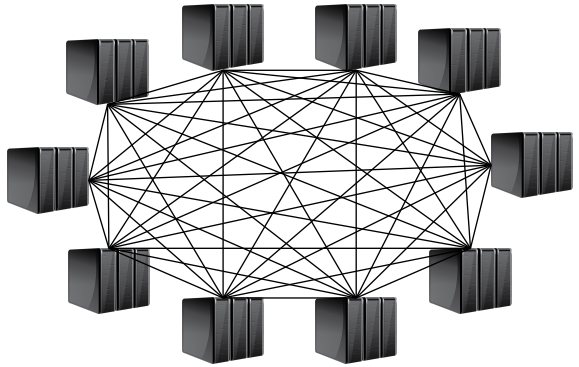
자 모델 알아서 돌려봐

선생님 자비 좀....

나에게 주피터를 달라!

~~그런 건 없어~~





Experiment ENV flatform

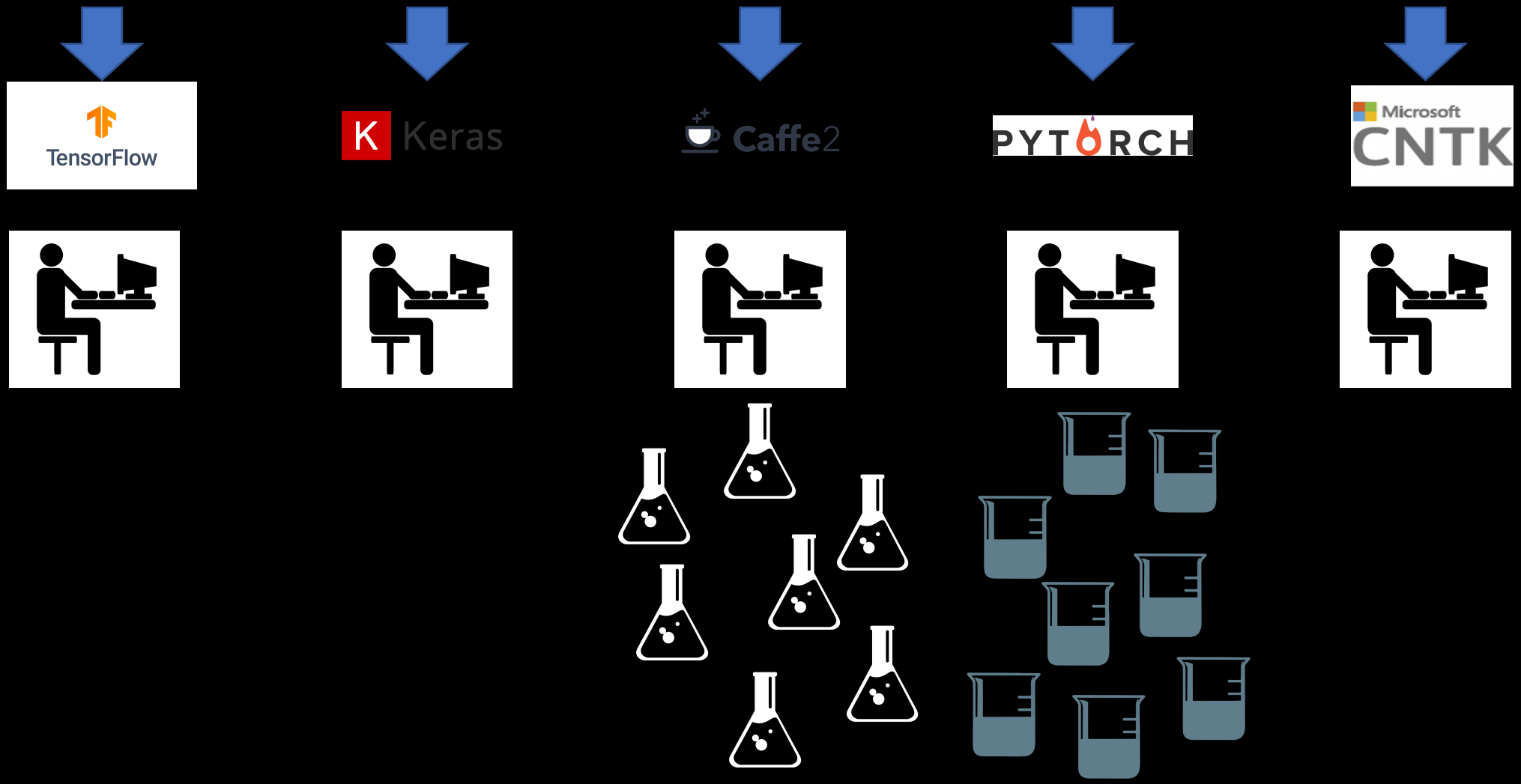
Env

jupyter

~~심심하면 커널 나가버리는 내가 싫어하는 주피터노트북 읍읍~~

하지만 모델러가 늘어나면 어떨까?

프레임워크 고만 가져와 미친놈들아

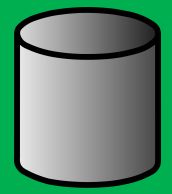

TensorFlow
Keras
Caffe2
PYTORCH
Microsoft
CNTK


이건 미친짓이야
난 여기서 나가야겠어

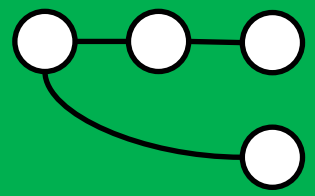

julia
mxnet
K
DEEP
LEARNING
theano
R
Microsoft
CNTK
Caffe2
Chainer

TensorFlow Keras Caffe2 PYTORCH Microsoft CNTK

Model Abstraction Layer



Model DB

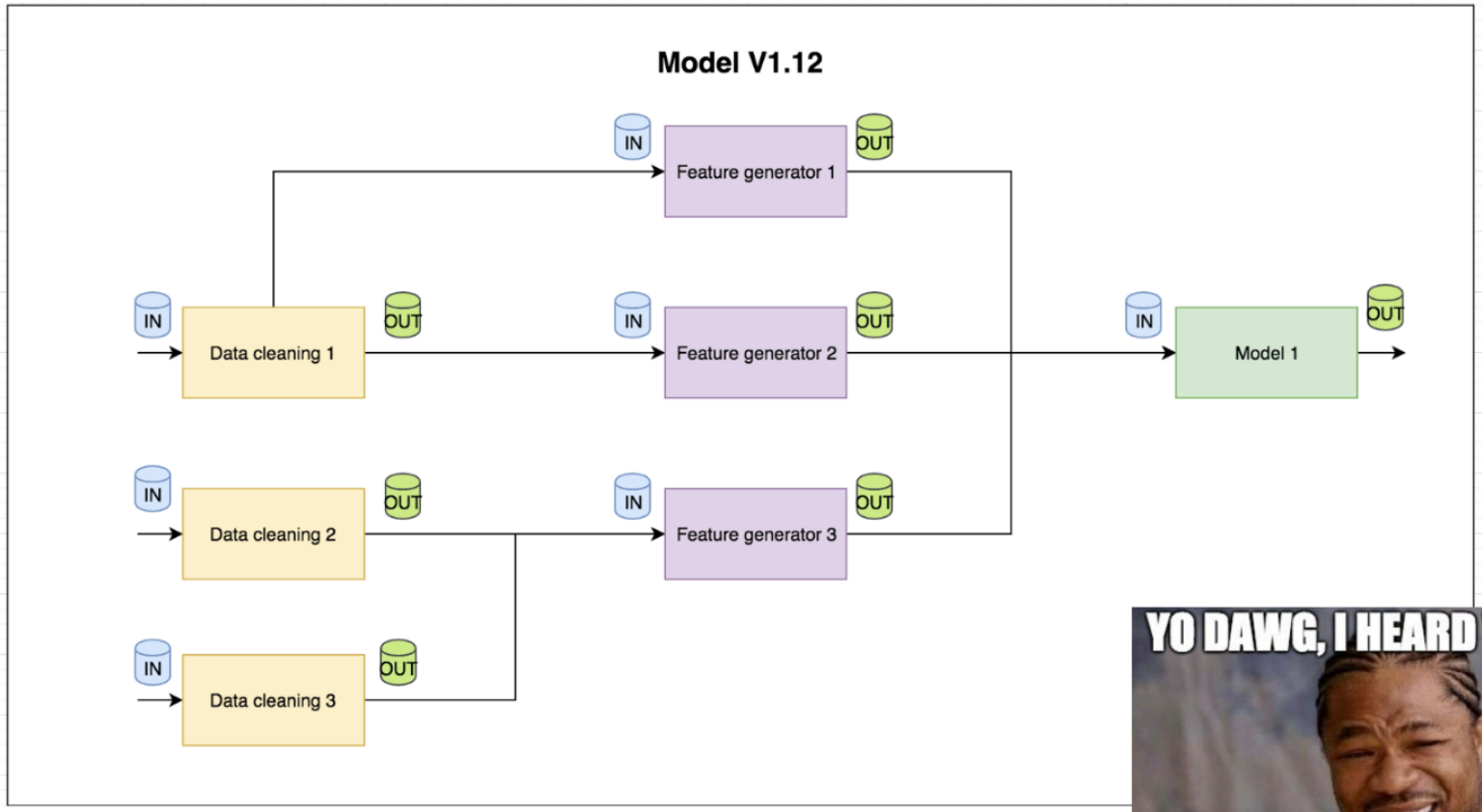


Version control

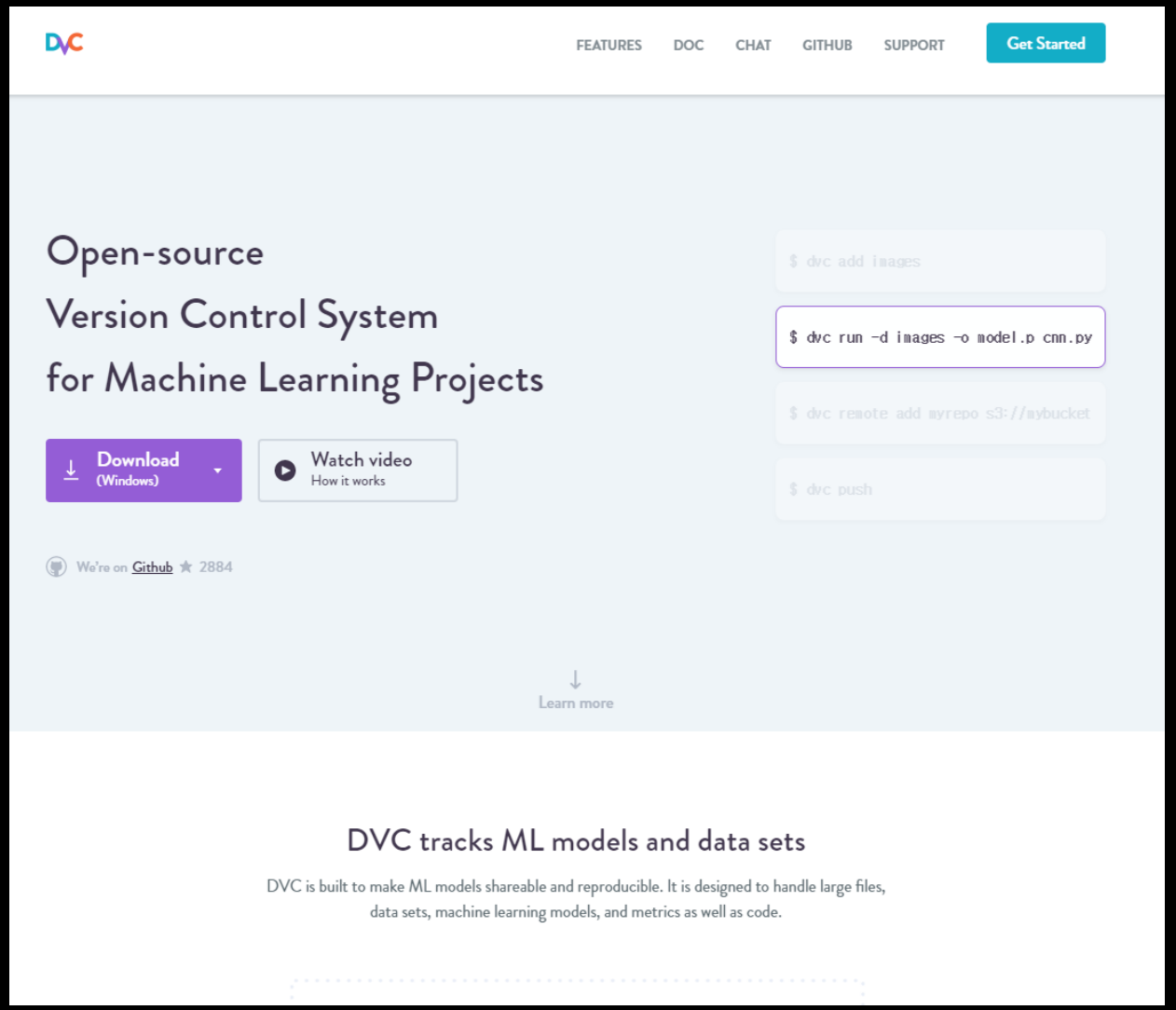
monitoring

Model storage Layer

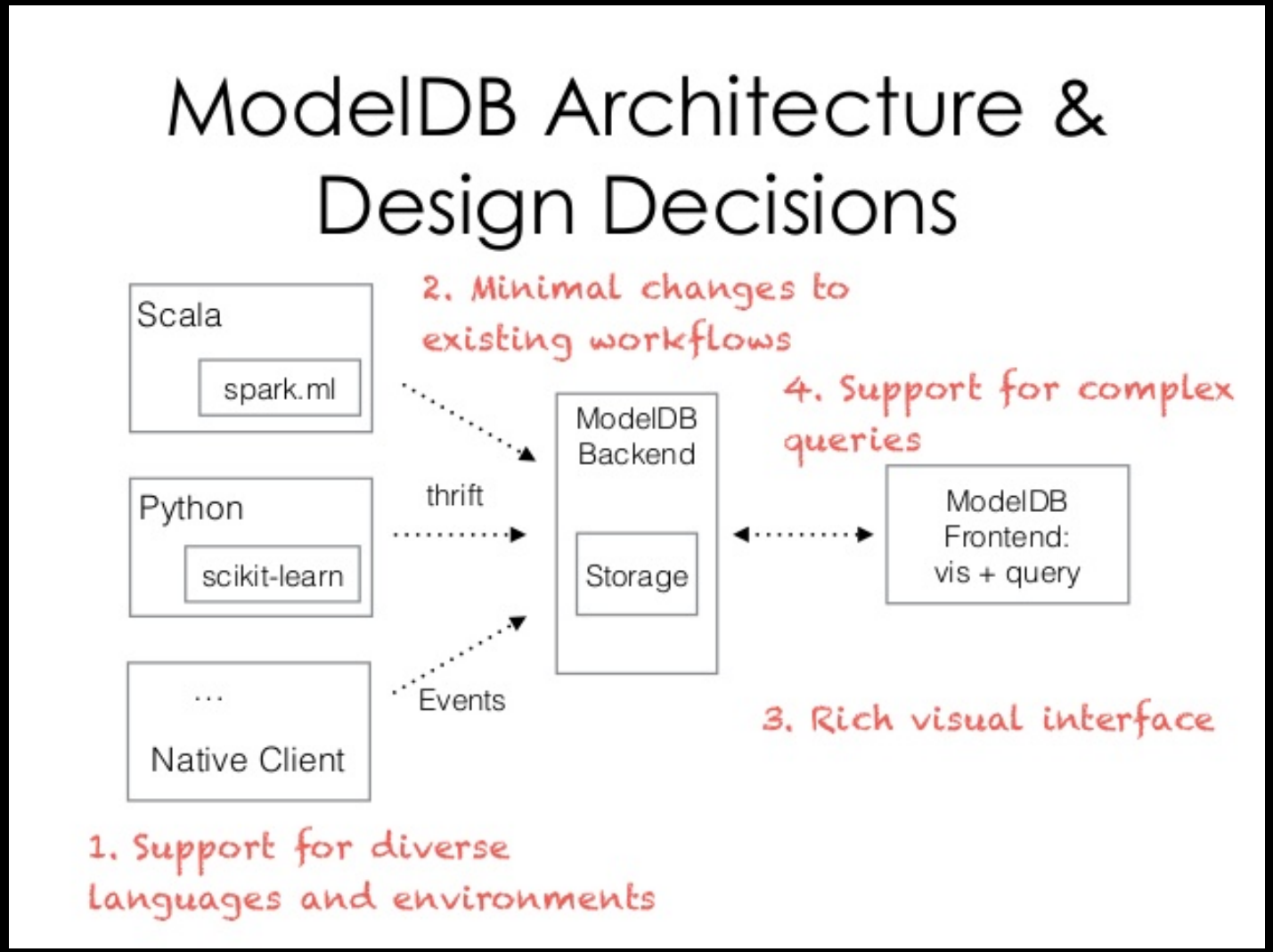
We can abstract our entire pipeline and data flows



어떻게 하면 이 문제를 해결할까?



어떻게 하면 이 문제를 해결할까?



어떻게 하면 이 문제를 해결할까?

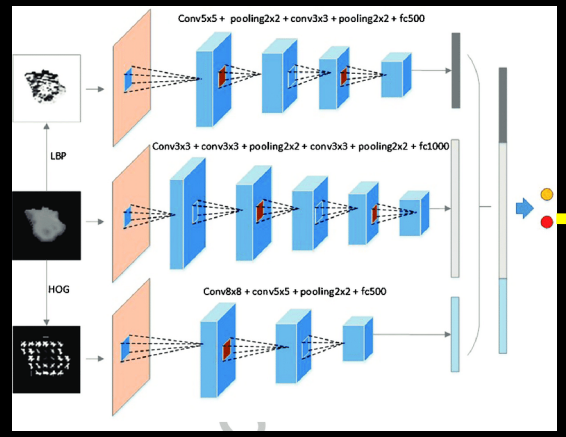
Model Description Script

model_scheme_sample.json 1.94 KB

```
{
  "data" : {
    "batch_size" : 500,
    "image_size" : [30,30,1],
    "label_dim" : 3,
    "data_orientation" : "vertical",
    "label_type" : "classification",
    "normalize" : "min_max",
    "data_path" : "path"
  },
  "model" : {
    "learning_rate": 0.001,
    "iteration": 50,
    "save_interval": 1,
    "optimizer" : "adam_optimizer",
    "network": [
      {
        "name": "conv_layer_set_1",
        "type": "conv2d_set",
        "kernel": [3,3],
        "channel": 64,
        "padding": "same",
        "batch_normalization": "true",
        "activation": "leaky_relu",
        "maxpooling": {
          "pooling_size": [2,2],
          "pooling_stride": [2,2],
          "padding": "same"
        },
        "dropout": 0.7
      },
```

```
      {
        "name": "conv_layer_set_2",
        "type": "conv2d_set",
        "kernel": [
          3,
          3
        ],
        "channel": 128,
        "padding": "same",
        "batch_normalization": "true",
        "activation": "leaky_relu",
        "maxpooling": {
          "pooling_size": [
            2,
            2
          ],
          "pooling_stride": [
            2,
            2
          ],
          "padding": "same"
        },
        "dropout": 0.7
      },
```

JSON.NET



Model Meta

Model Description

Model Parser

Training

Serving

Data Lake

Training Data ingestion

Data Cleaning

Feature Engineering

ML Model

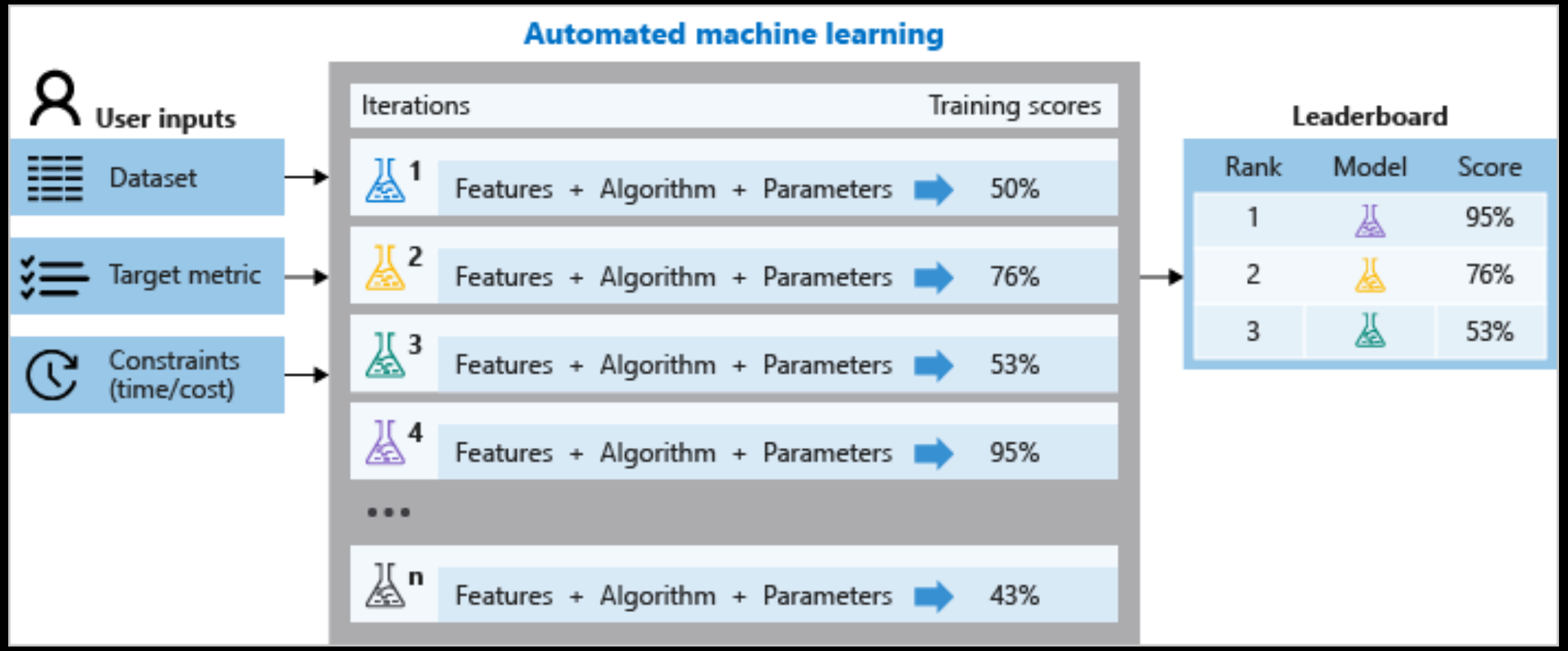
Data version control system

Model manager

Request

Response

그런데 요즘은 자동으로 합니다.



Auto ML

모델러 : AI 앞잡이가 되면 고용보장은 될 줄 알았는데 흐흑

(~~제프 딘을 원망해라~~)

Data Lake

Training Data ingestion

Data Cleaning

Feature Engineering

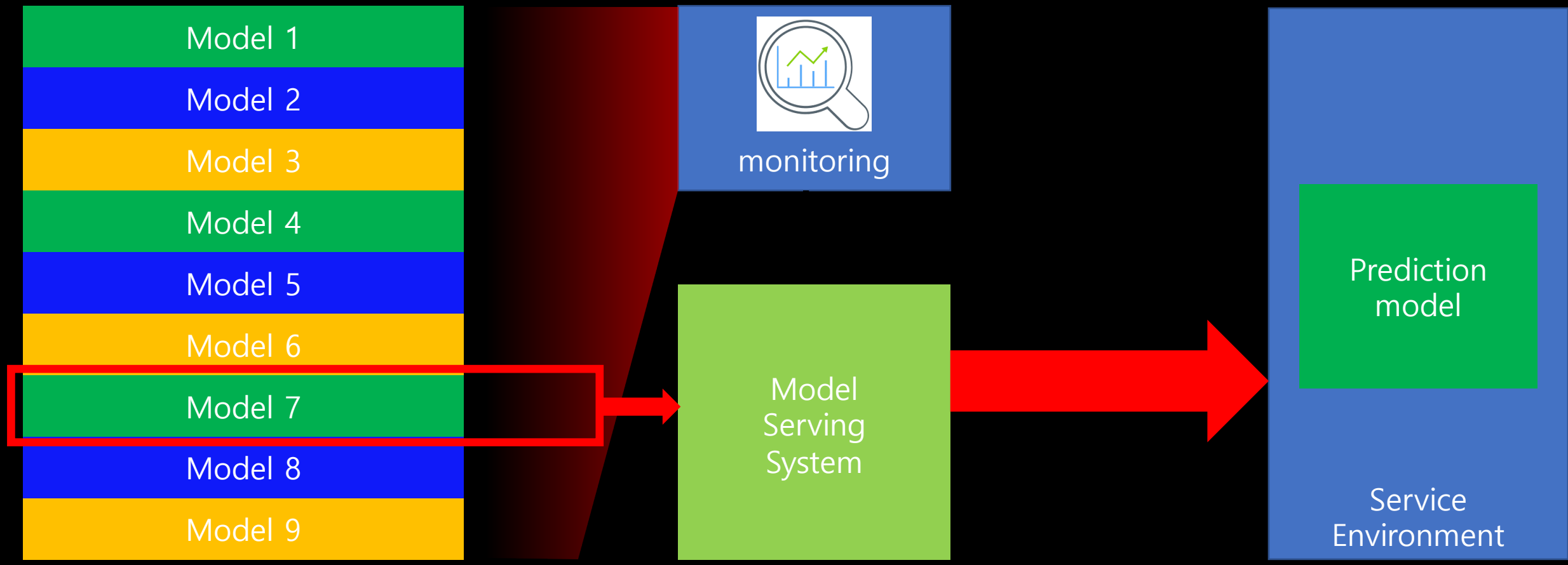
ML Model

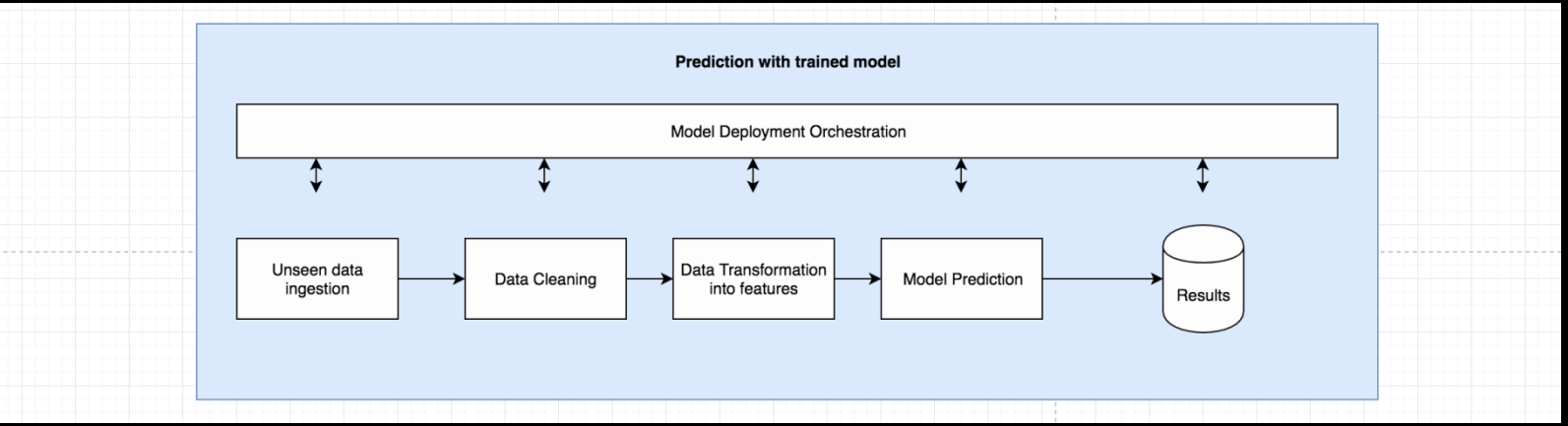
Data version control system

Model manager

Request

Response


Model 1
Model 2
Model 3
Model 4
Model 5
Model 6
Model 7
Model 8
Model 9
monitoring
Model Serving System
Prediction model
Service Environment



일반 웹 CI/CD/모니터링
하고 비슷하다고 생각할 수 있는데

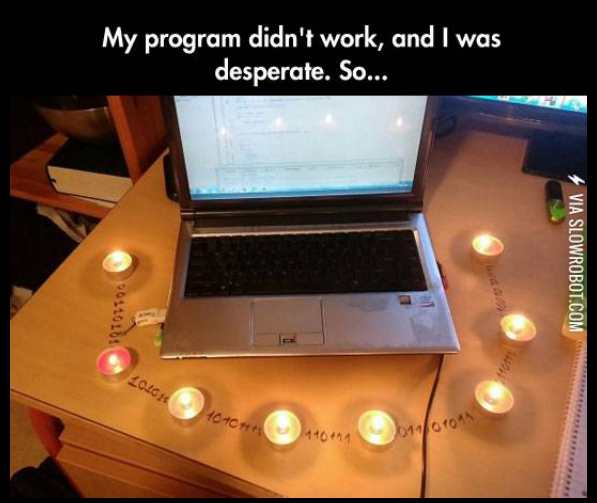
좀 다릅니다.

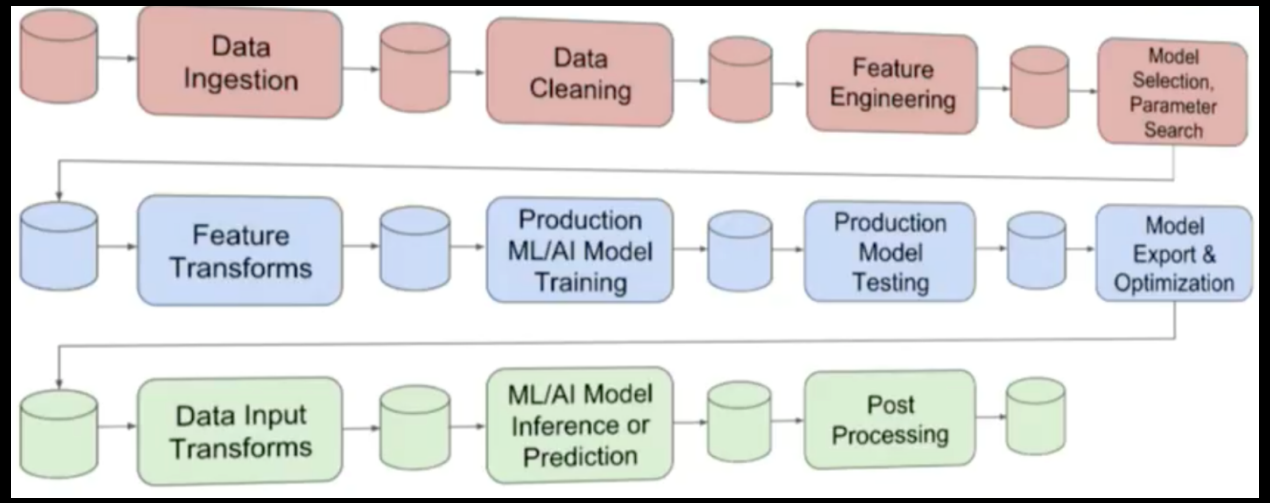
모니터링

소프트웨어 모니터링 외에

- 모델 퀄리티
- 데이터의 분포
- 데이터의 형태

WE CAN STOP PRAYING TO THE DEMO GODS





결국 기도메타

Data Lake

Training Data ingestion

Data Cleaning

Feature Engineering

ML Model for training

Data version control system

Model manager

Model serving system

Request

ML Model for prediction

Response

실제 서비스에서 모델을 사용할때도

필요한건? DATA

나는 유저가 준 데이터를 원한다

Data Lake

Training Data ingestion

Data Cleaning

Feature Engineering

ML Model for training

Data version control system

Model manager

Model serving system

Request

Real (by user) Data ingestion

ML Model for prediction

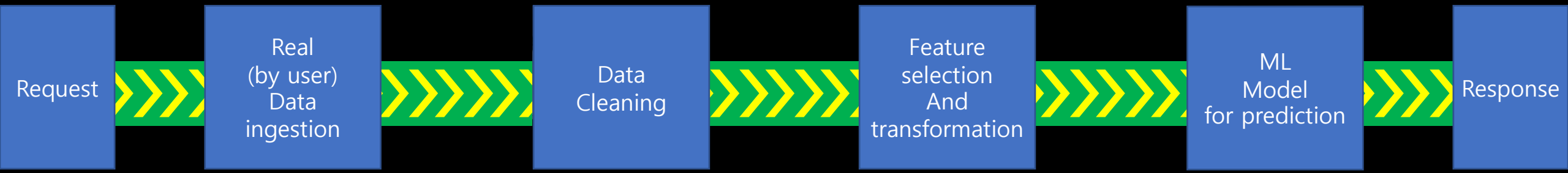
Response

~~~쁜 DATA

학습 과정에서 본 동일한 성능을

재현성 을 위한

학습 과정에서 한 동일한 과정으로

preprocessing
~~~

Data version control system

Model manager

Model serving system



Data pipelining을 하려면

Orchestration 도 중요합니다.

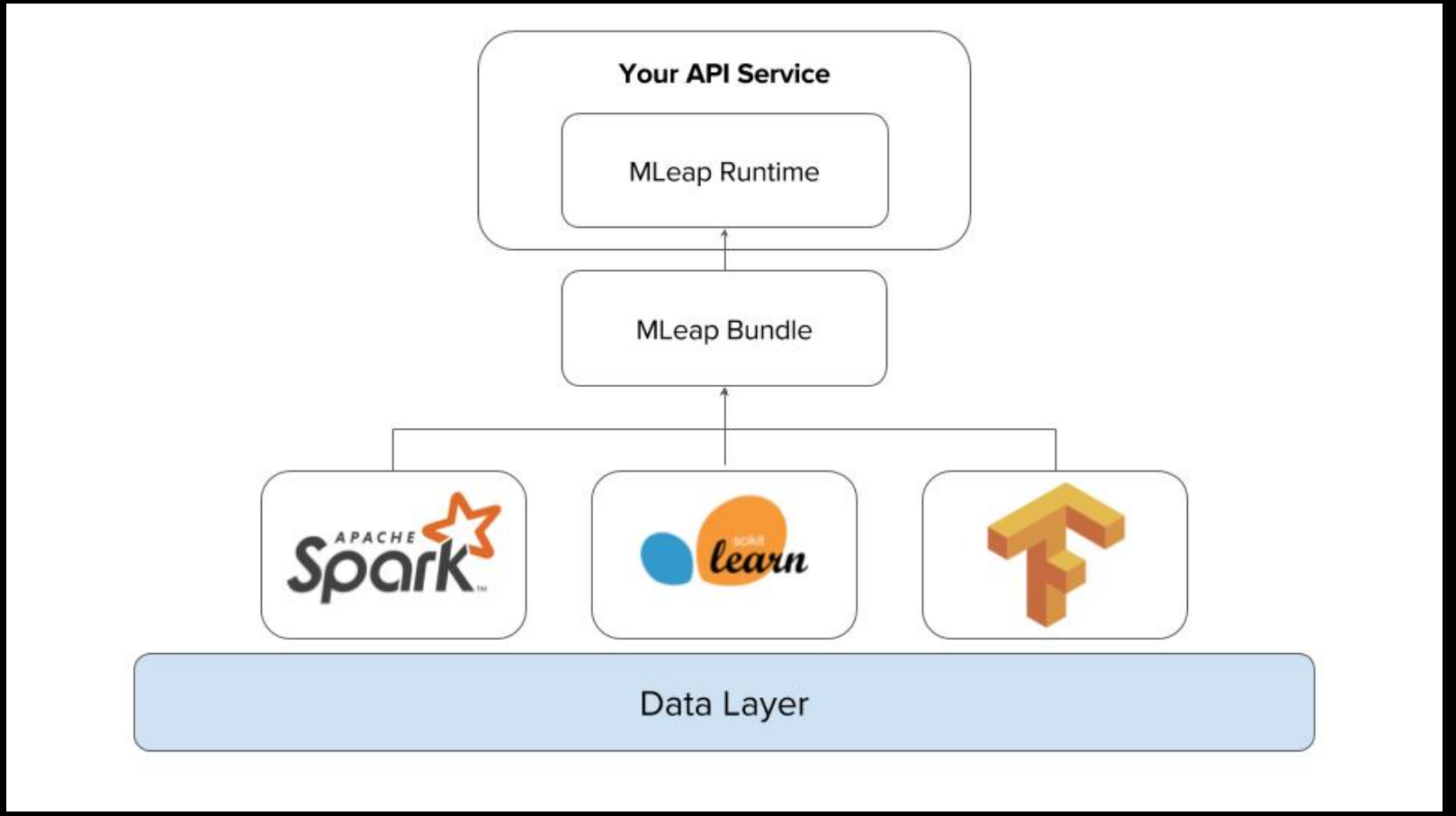
COMPUTATIONAL RESOURCE ALLOCATION

THIS IS A HARD PROBLEM

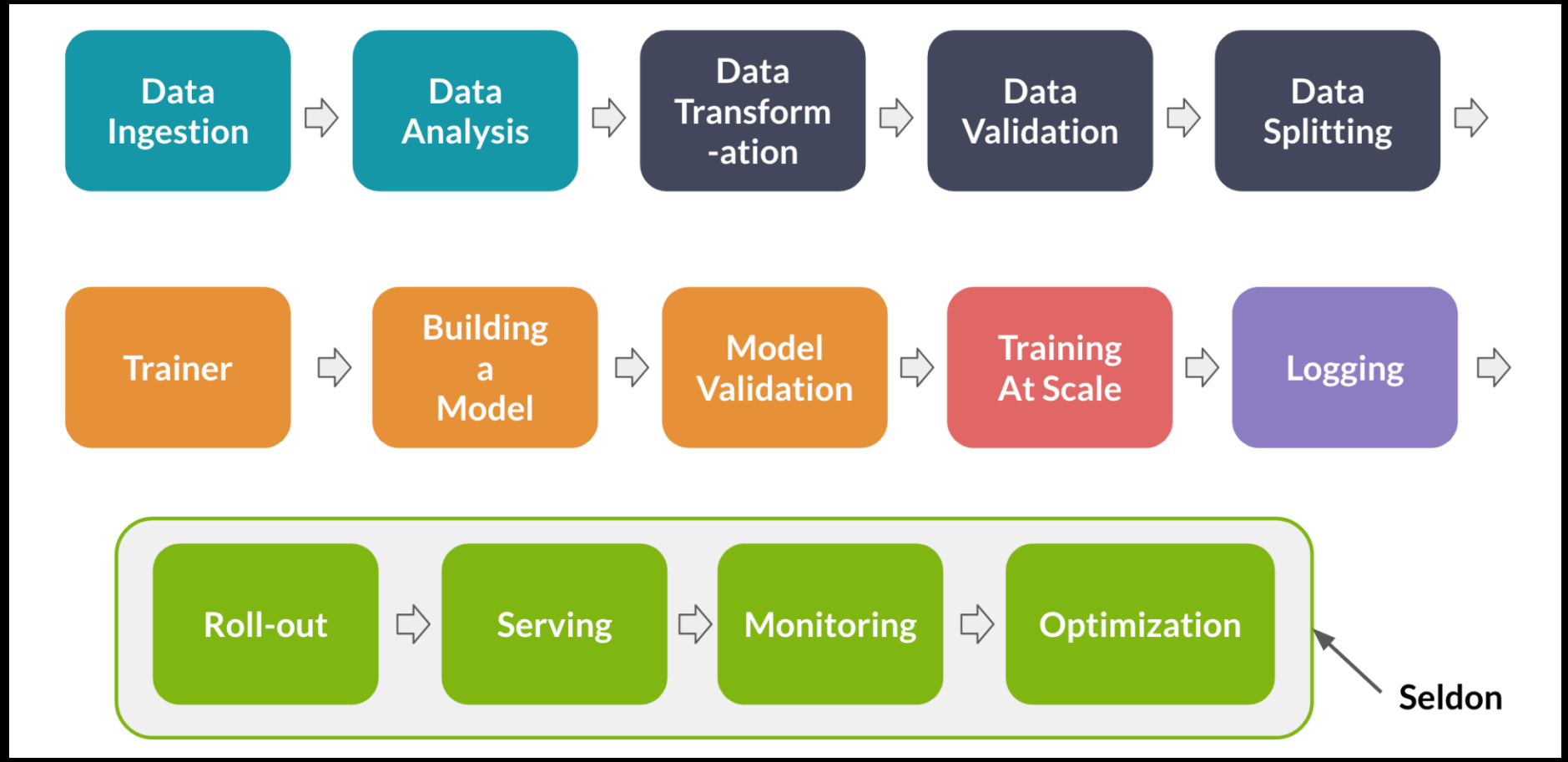
오케스트레이션 레벨에서 리소스 추상화

- An ETL framework
- A HDFS-based service
- A Kubernetes cluster
- Any distibuted framework!

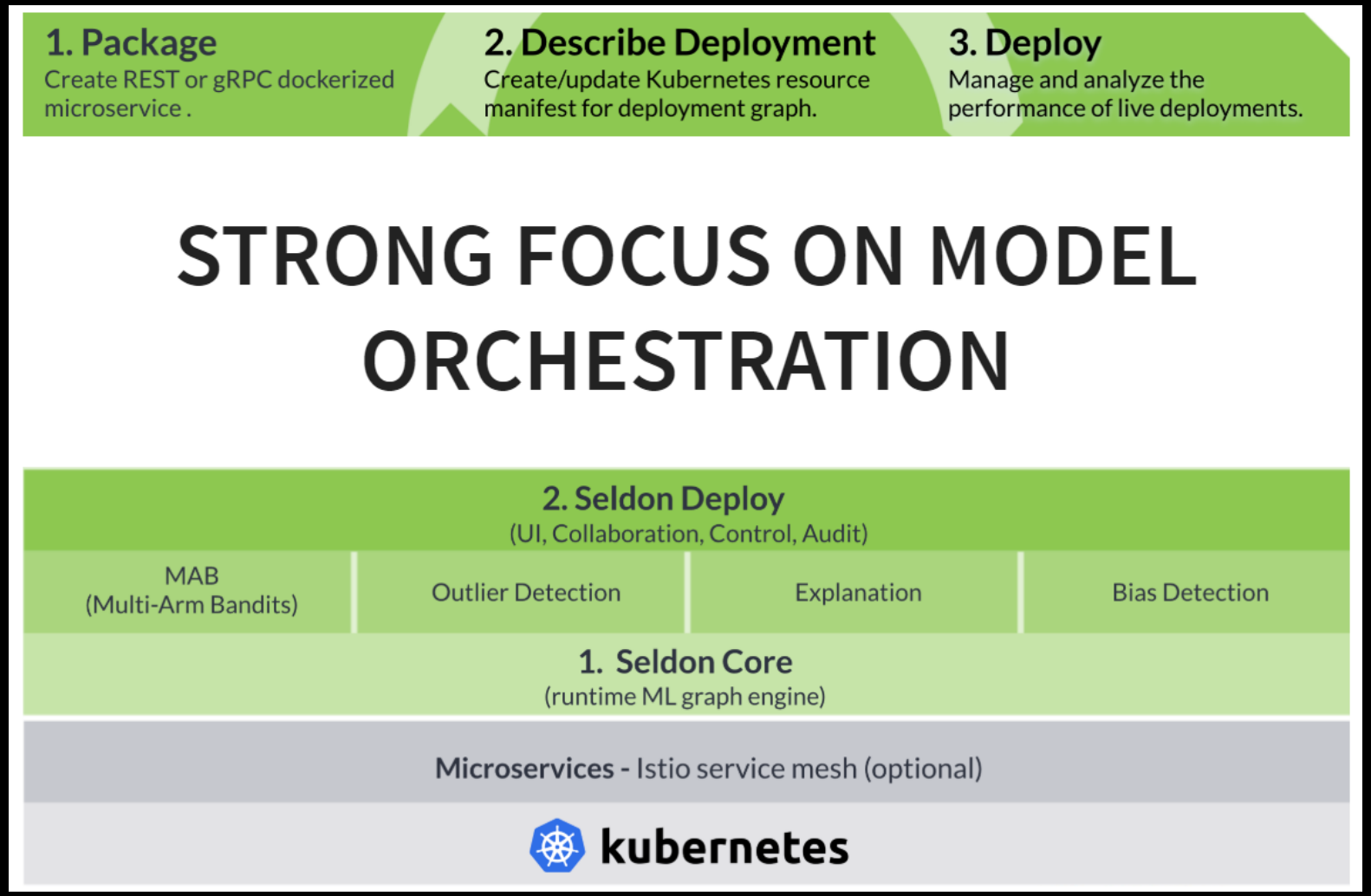
MLEAP


Your API Service
MLeap Runtime
MLeap Bundle
APACHE
Spark
scikit
learn
Data Layer

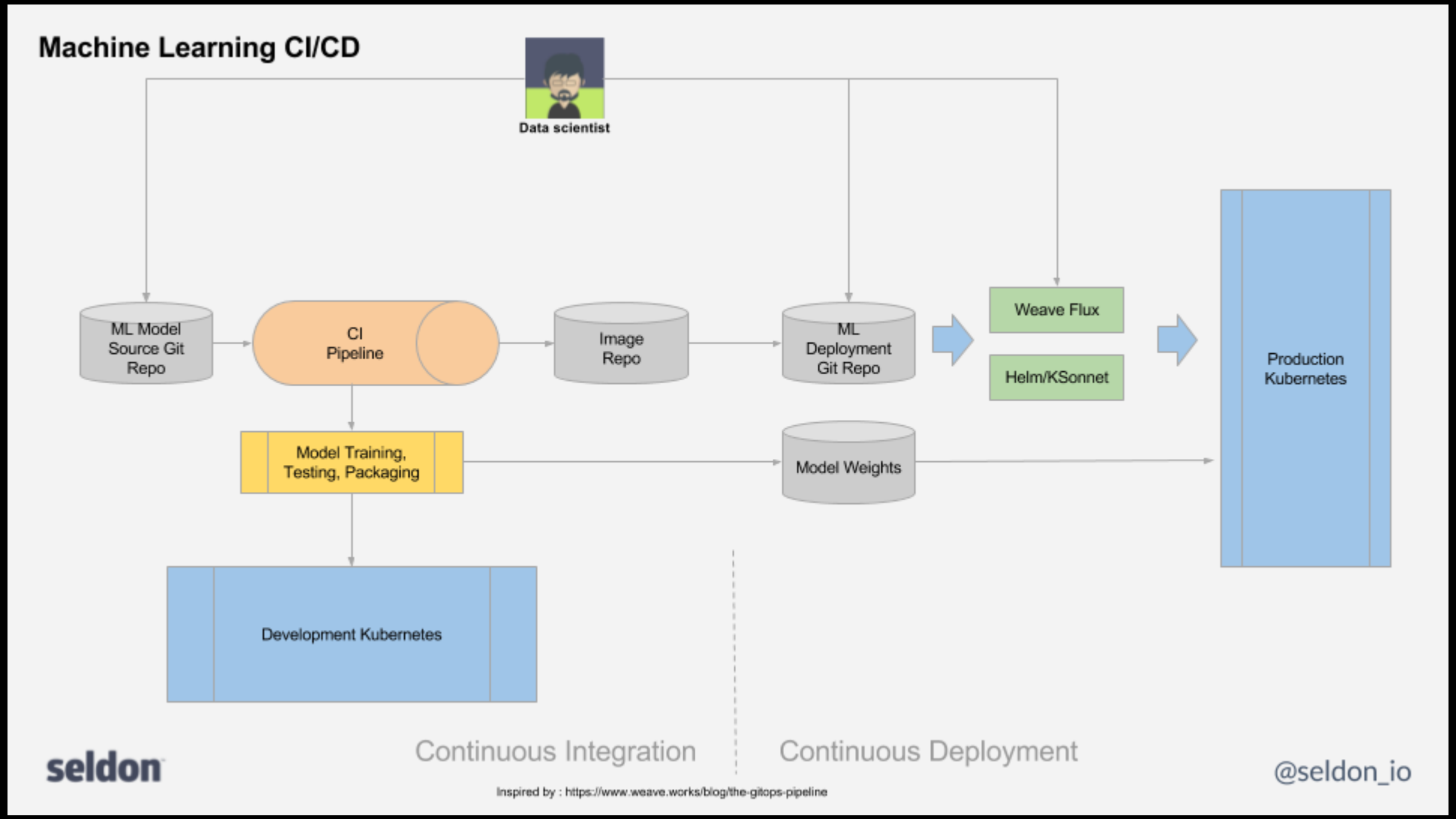
SELDON-CORE

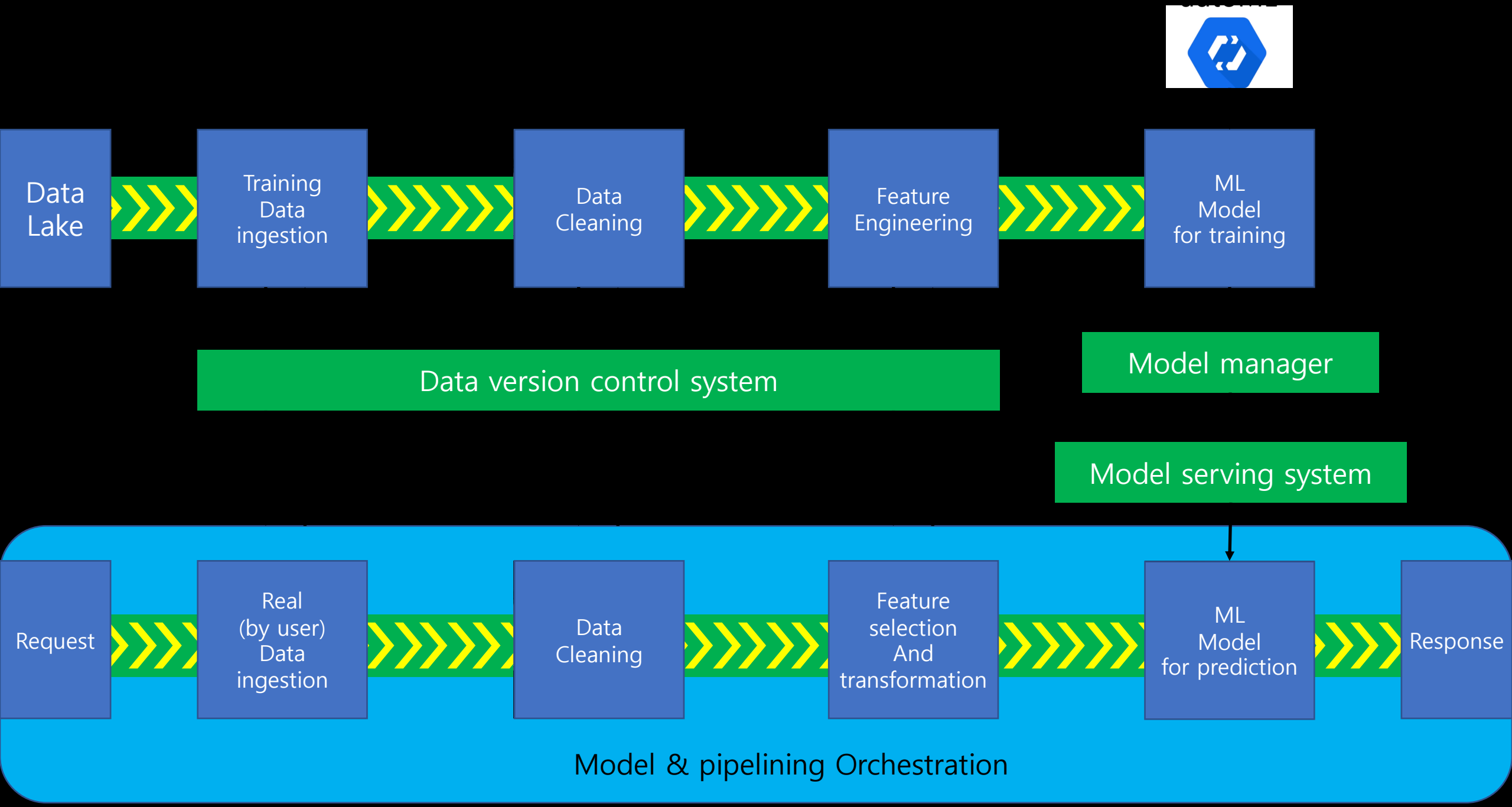

Data Ingestion
Data Analysis
Data Transform -ation
Data Validation
Data Splitting
Trainer
Building a Model
Model Validation
Training At Scale
Logging
Roll-out
Serving
Monitoring
Optimization
Seldon

SELDON-CORE


1. Package
Create REST or gRPC dockerized microservice .
2. Describe Deployment
Create/update Kubernetes resource manifest for deployment graph.
3. Deploy
Manage and analyze the performance of live deployments.
STRONG FOCUS ON MODEL ORCHESTRATION
2. Seldon Deploy
(UI, Collaboration, Control, Audit)
MAB
(Multi-Arm Bandits)
Outlier Detection
Explanation
Bias Detection
1. Seldon Core
(runtime ML graph engine)
Microservices - Istio service mesh (optional)
kubernetes

TACKLING THE CI/CD CHALLENGE




Data Lake
Training Data ingestion
Data Cleaning
Feature Engineering
ML Model for training
Data version control system
Model manager
Model serving system
Request
Real (by user) Data ingestion
Data Cleaning
Feature selection And transformation
ML Model for prediction
Response
Model & pipelining Orchestration

이러면 워크플로우는 끝

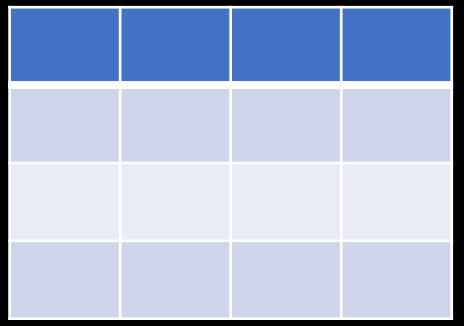
~~이러면 워크플로우는 끝~~

이라고 생각하면 망합니다

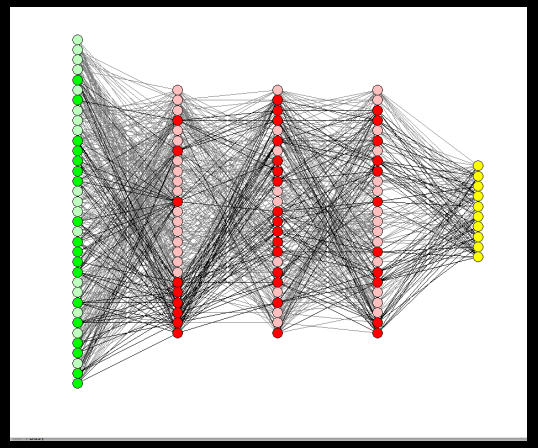
ML은 어떻게 Test를 할까?

- 과연 이 코드의 모델 그래프는 내가 의도한 대로 나온 것인가?
- 팀에서 서비스 가능하다고 판된되는 정확도가 나오는가?
- 테스트 결과에서 나온 정확도의 신뢰도는 어떻게 평가하는가?
- 정확도가 높아졌는데 나중에 알고보니 구성된 신경망 모델이 내가 의도한 모델이 아니었다면?
- Test Code는 어떻게 짜나?

Match Down



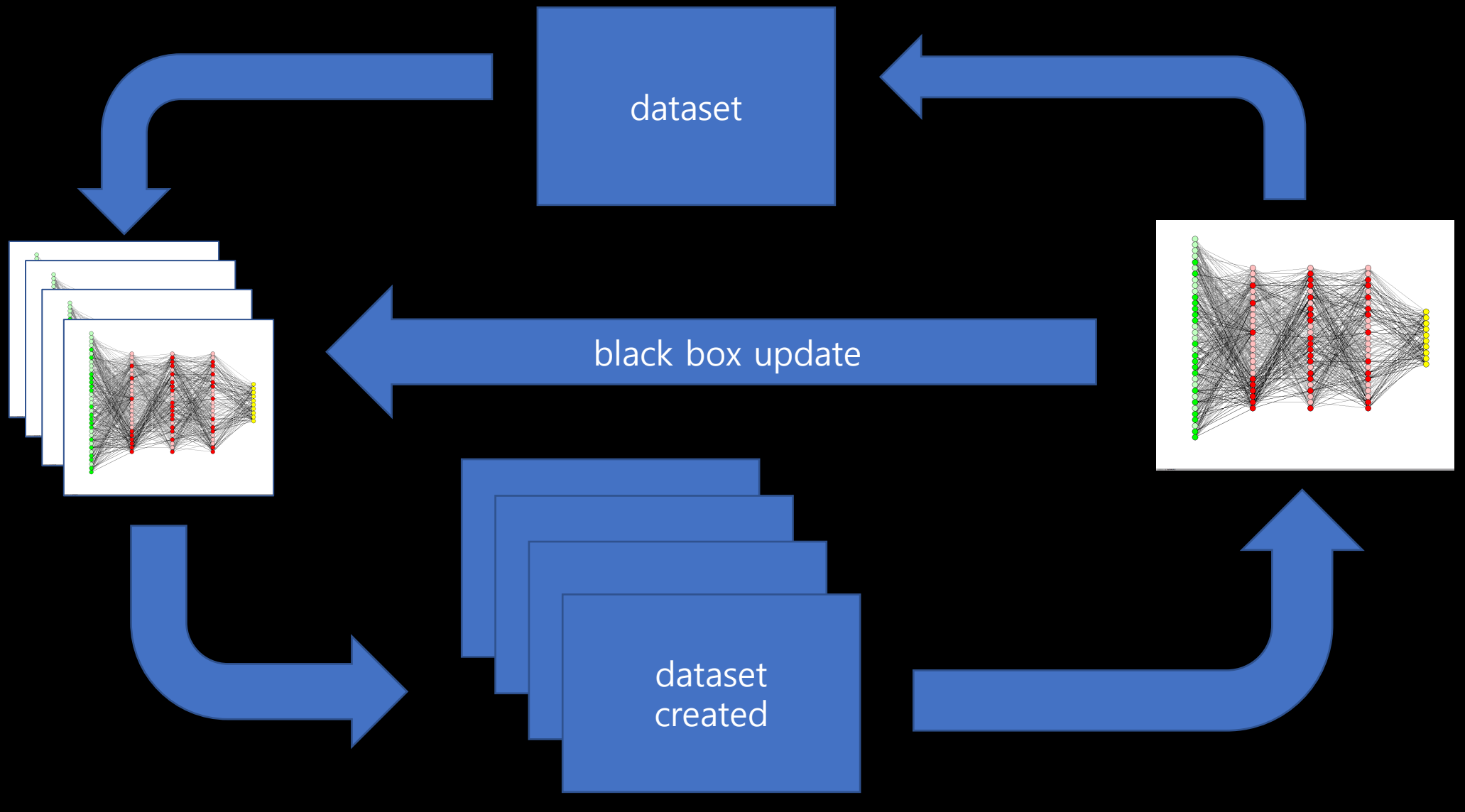
model table



ONNX

Match Down
Engine

black label(정확도의 정확도 측정)



그리고 가장 중요한 업무 트레킹과 평가

엔지니어는
하루에 수십 수백 줄의
코드를 작성 수정하지만

연구자는
실험의 목적에 맞게
하루에 딱 한 줄 추가할 때도 있다

연구팀한테 개발팀이 할 수도 있는 말
(제가 직접 들었던 말)

쟤들은 도대체 뭘 하는지 모르겠다

그리고 가장 중요한 업무 트레킹과 평가

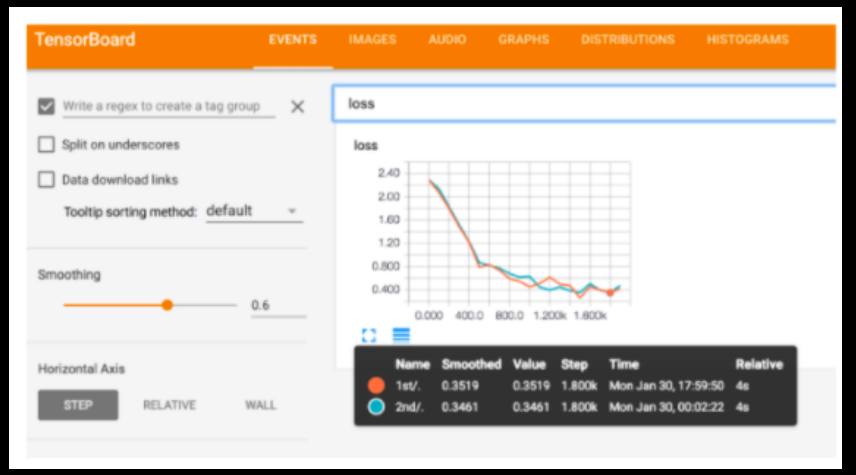
엔지니어는
하루에 수십 수백 줄의
코드를 작성 수정하지만

연구자는
실험의 목적에 맞게
하루에 딱 한 줄 추가할 때도 있다

뭔가 많이 했는데
논 것이 되어버림
그래서 쓰는게 랩노트인데.....
절망편 : 문서만 쓰면 일하는 것이 되어버림

실험 리포팅 툴이 필요해서
만들고 있는 중입니다

JSON.NET



tensorboard log

reporting engine

Report

결론
1. 협업은 어렵다
2. 가능한 서로의 업무방식을 유지하면서
   협업할 수 있는 방법이 필요하다
3. 그래서 여러 도구가 필요하다

ML-Ops Lab을 개설할 예정입니다.
많은 분들과 함께 MLOps tool (open source)
개발하고 싶습니다.