

MLOPS

Михаил Марюфич



Что делаю?

Разрабатываю и поддерживают инфраструктуру для ML и Big Data

Что сделал?

Различные реко-пайплайны

Поиск дубликатов в Юле

Распознавание текста в ОК

Распознавание документов для Ситимобил

Автоматизация обучения и выкатки моделей

Образование:

МатМех СПбГУ

Computer Science Center по направлению Data Science и Software Engineering

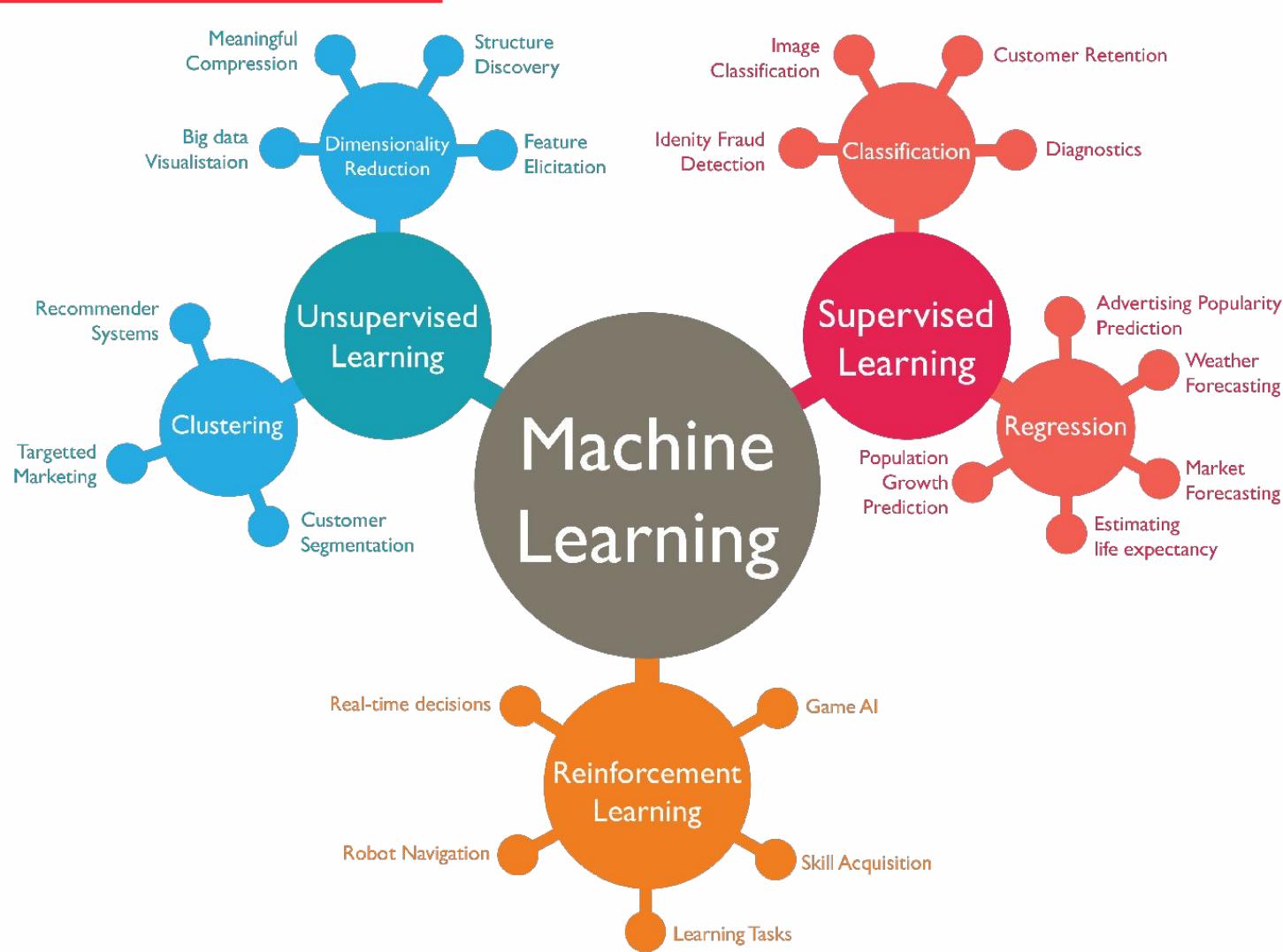
<https://compscicenter.ru/students/362/>



Михаил Марюфич

Что такое ML?

ML - это про науку



Для чего компании внедряют ML?



Модель должна быть в проде!



Infra - ЭТО ВАЖНО

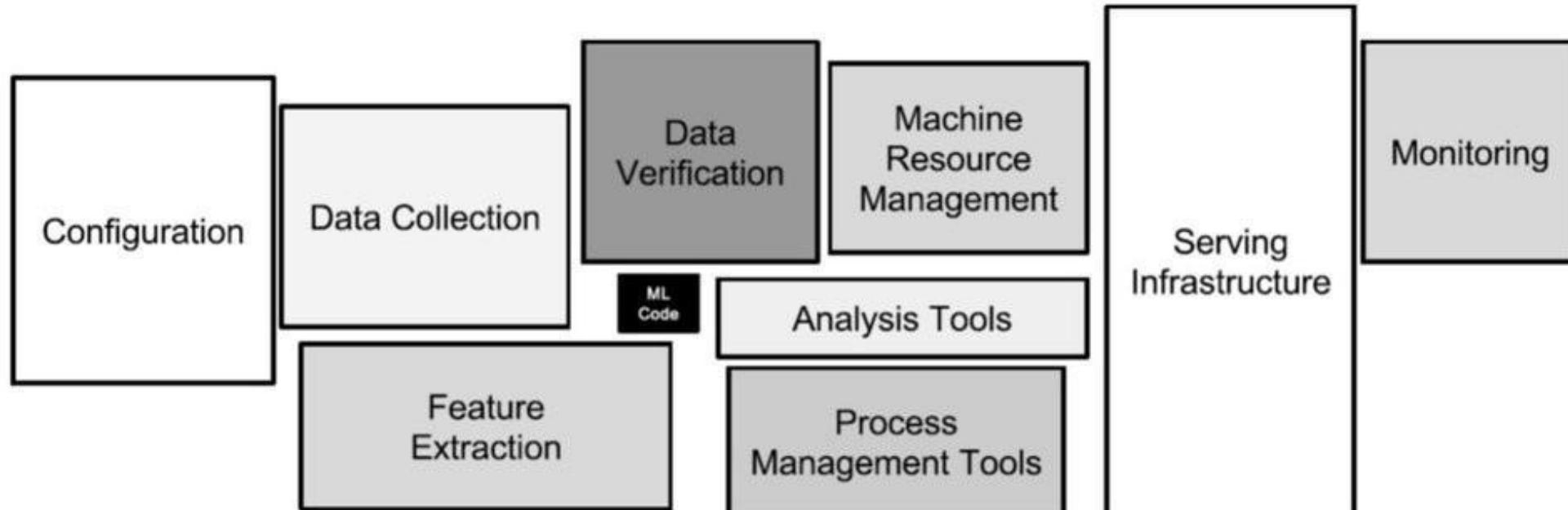


Figure 1: Only a small fraction of real-world ML systems is composed of the ML code, as shown by the small black box in the middle. The required surrounding infrastructure is vast and complex.

VALOHAI NEWSLETTER
JANUARY 2021

HAPPY NEW YEAR 😊

2021 – The Year of MLOps

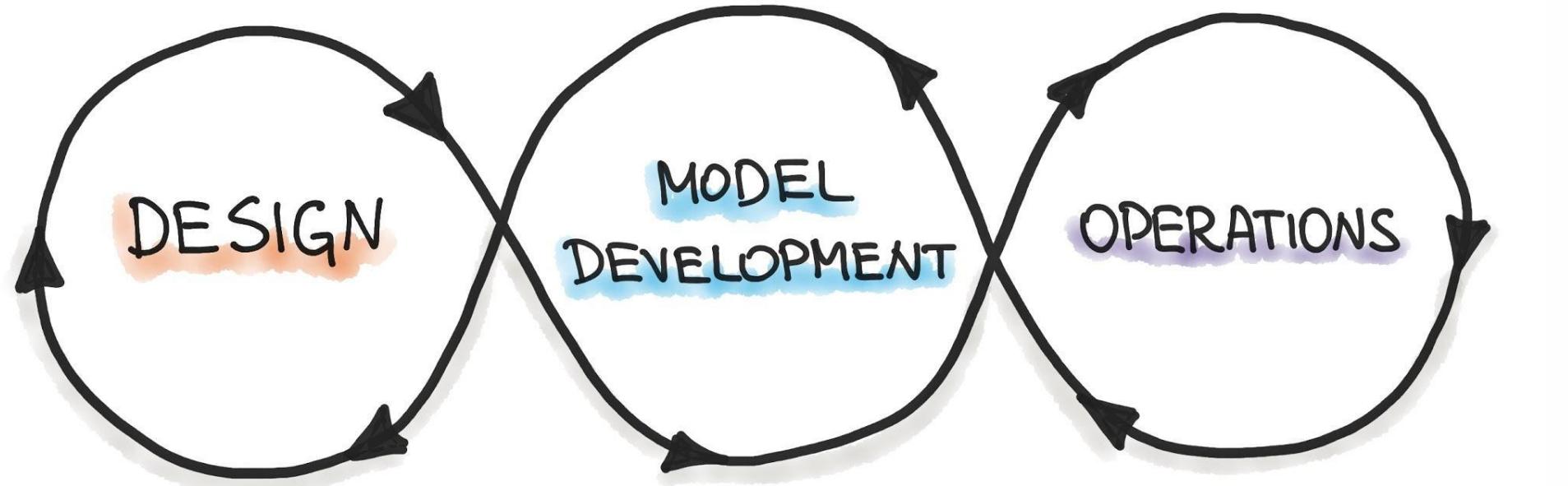


2020



2021

MLOps



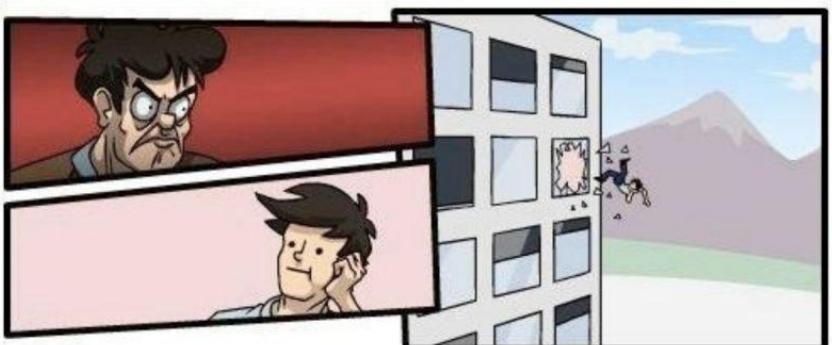
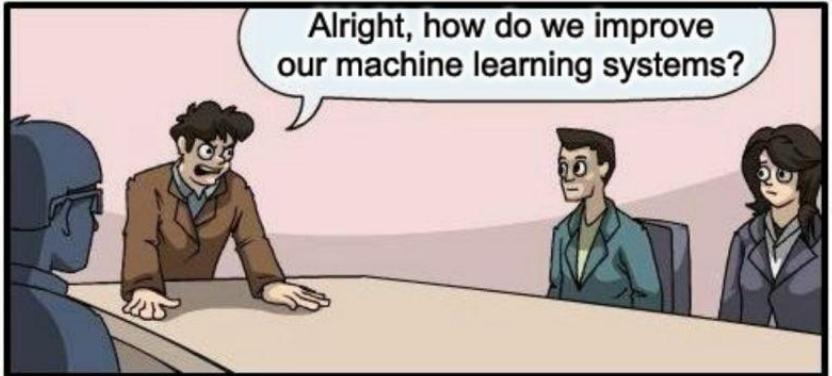
- Requirements Engineering
- ML Use-Cases Priorization
- Data Availability Check

- Data Engineering
- ML Model Engineering
- Model Testing & Validation

- ML Model Deployment
- CI/CD Pipelines
- Monitoring & Triggering

Давайте знакомиться!

Заполняем опрос!



<https://forms.gle/rhTn7YLCBeq7zucf9>



О курсе

Проходит каждый вторник в 19-00
По формату лекции + демки

На курсе будет 5 домашних работ. За все это дело можно будет набрать 100 баллов, критерии будут при выдаче (ближайшая на следующем занятии)

На 5 - 85 баллов

На 4 - 65 баллов

На 3 - 45 баллов

+ могут быть необязательные активности, которые также будут оцениваться(все гибко).

Приступаем к делу =)

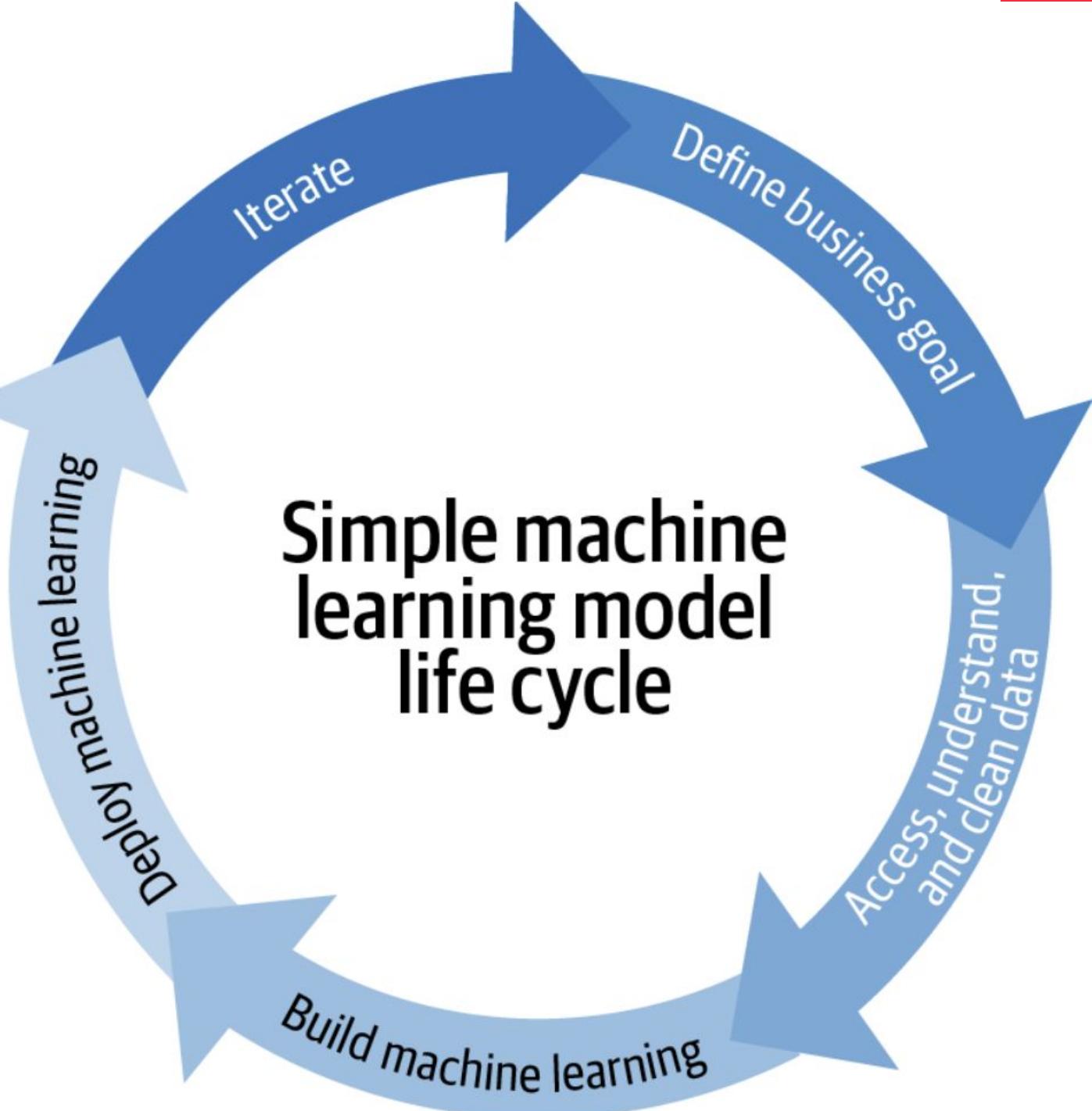
```
background-color: #F5F5F5;
text-shadow: 0px -1px 0px #EAEAEA;
filter: dropshadow(color:#EAEAEA);
color:#777;

}
header #main-navigation ul li span:hover,
box-shadow: 0px 0px 1px #EAEAEA;
-webkit-box-shadow: 0px 0px 2px #EAEAEA;
moz-box-shadow: 0px 0px 1px #EAEAEA;
background-color:#F9F9F9;
active span,
li span.dashboard,
..../img/dashboard.png
ul li span.dashboard,
finaldashboard.html
finaldashboard.html
```



План занятия

- 1) Обсуждаем различные этапы жизненного цикла ML
 - а) Постановка целей
 - б) Работа с данными
 - в) Тренировка моделей
 - г) Деплой
 - д) Мониторинг
 - е) Итерация
- 2) Роли в ДС
- 3) GIT



Основные этапы
разработки модели

Определяем цели



Примеры “целей”

- 1) Сделать нейронную сеть, которая будет детектировать спам.
- 2) Внедрить в продакшен 100 моделей
- 3) Разобраться, как работает BERT



Примеры целей

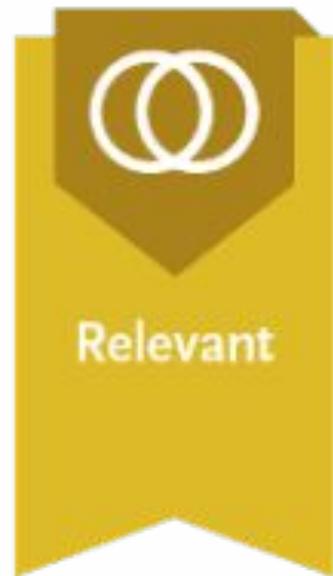
- 1) Снизить количество людей, которые видят спам до 1%

- 1) Снизить количество людей, которые не возвращают большие кредиты(более 1 млн рублей до 0.05% от всех, кому вы выдаем такие.

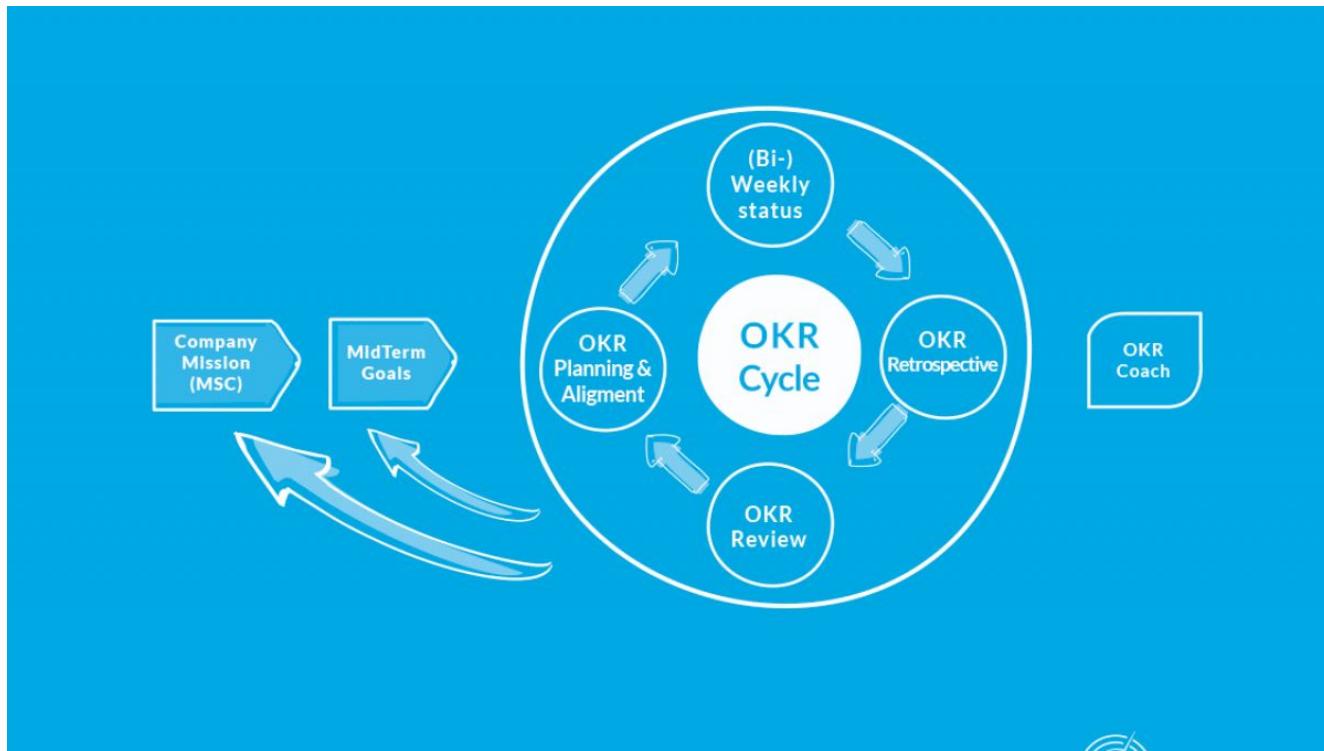
- 2) Повысить среднее время пользования нашим сервисом до 100 минут в день.

Цели (SMART)

S M A R T



OKR (objectives and key result)



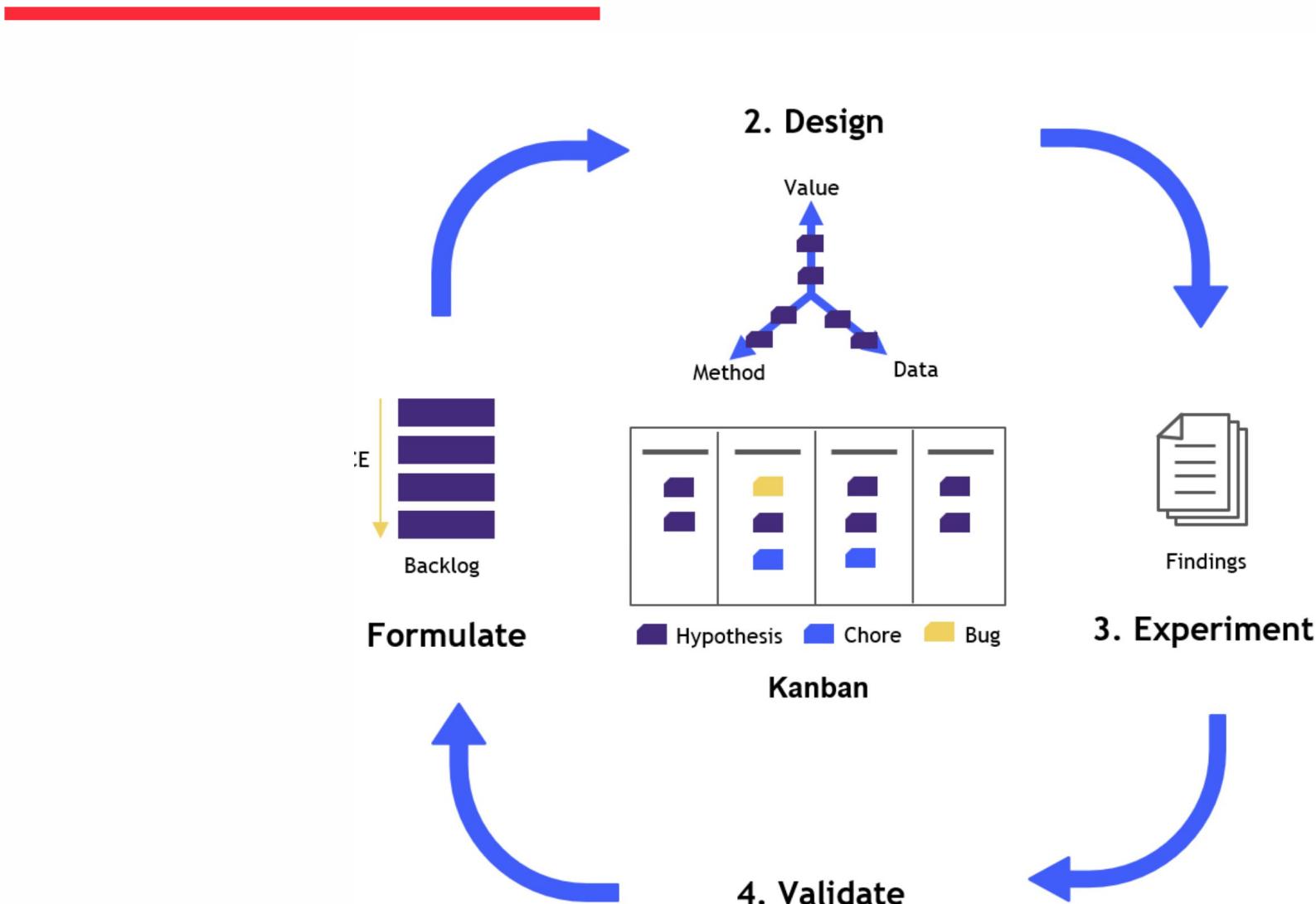
Иерархическая структура целей, от главных ключевых целей для организации до целей конкретных сотрудников.

Все, что мы делаем — ведет к достижению ключевых целей

Доклады по OKR/проверке гипотез



Lean Machine Learning





ИТОГИ

1. Не делаем то, что делать не надо
2. Используем SMART для формулировки целей
3. Думаем об измерении бизнес метрик

Работа с данными



Зачем нам данные?

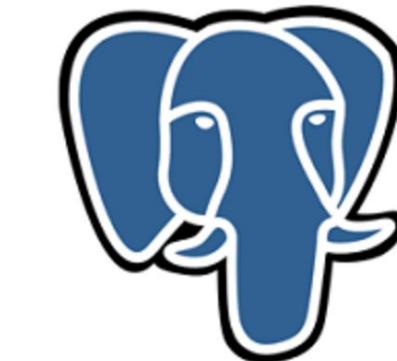
Учим на них модели

Ищем в них инсайты

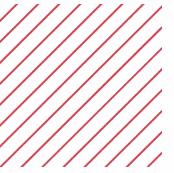
Делаем по ним предсказания

Оцениваем по ним результат

Data Store - данные должны где-то храниться



Postgre^{SQL}



Вопросы к данным

Какие релевантные датасеты доступны?

Достаточно ли точны и надежны данные?

Как получить доступ к этим данным?

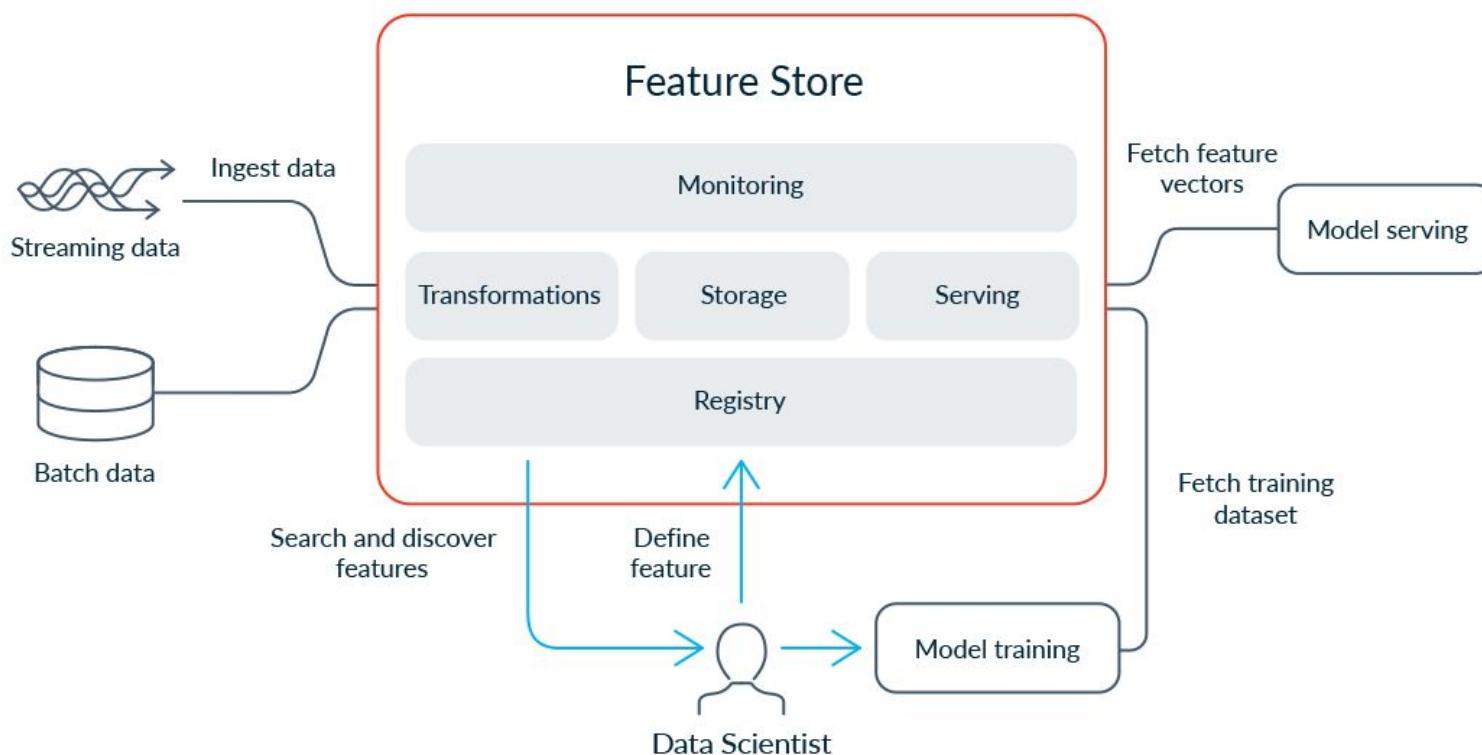
Какие фичи можно получить при джойне?

Часто ли обновляются данные?

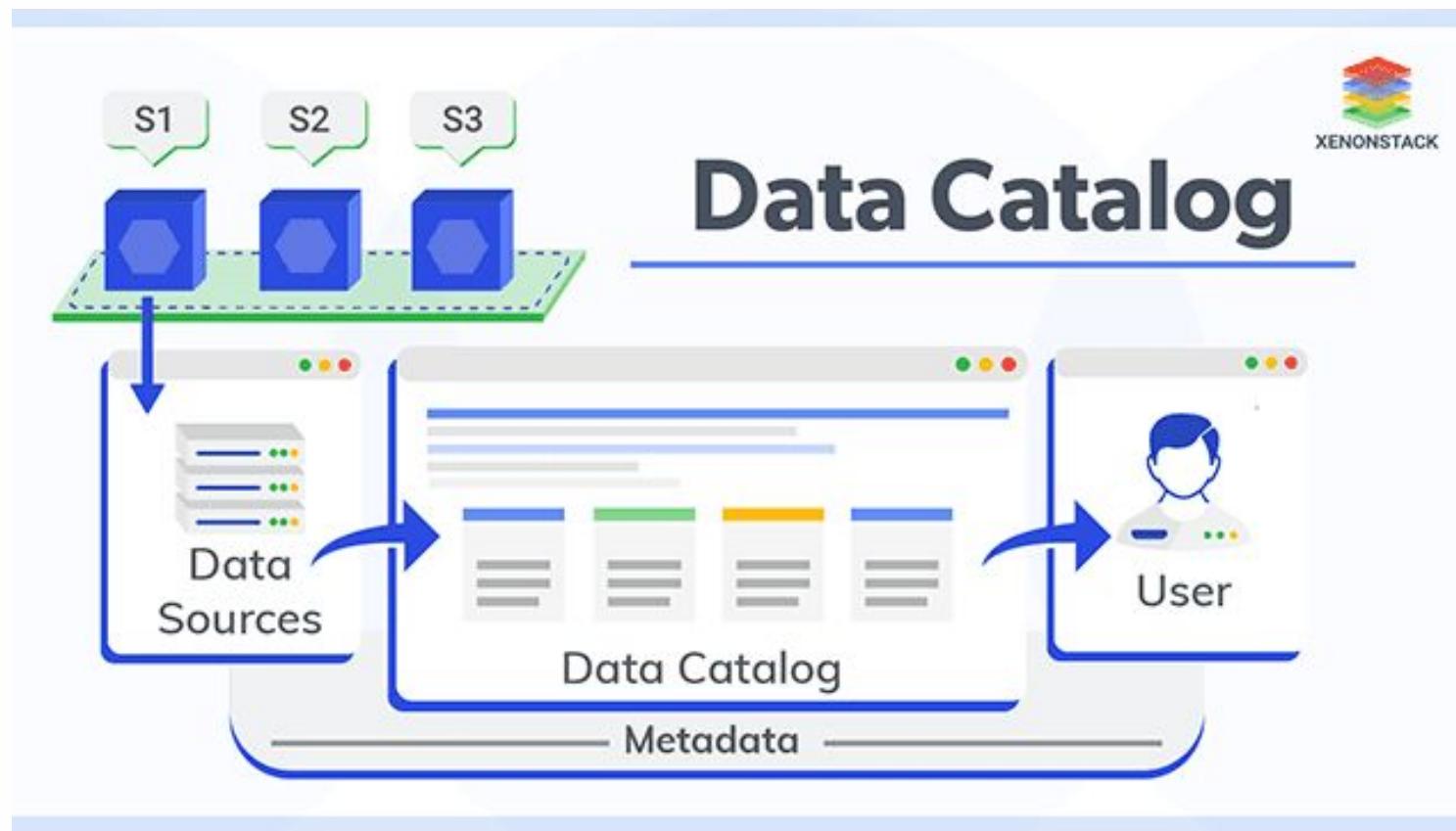
Можно ли будет получать эти фичи в realtime?

Feature Store

Доступ к фичам в реал-тайме



Data Catalog





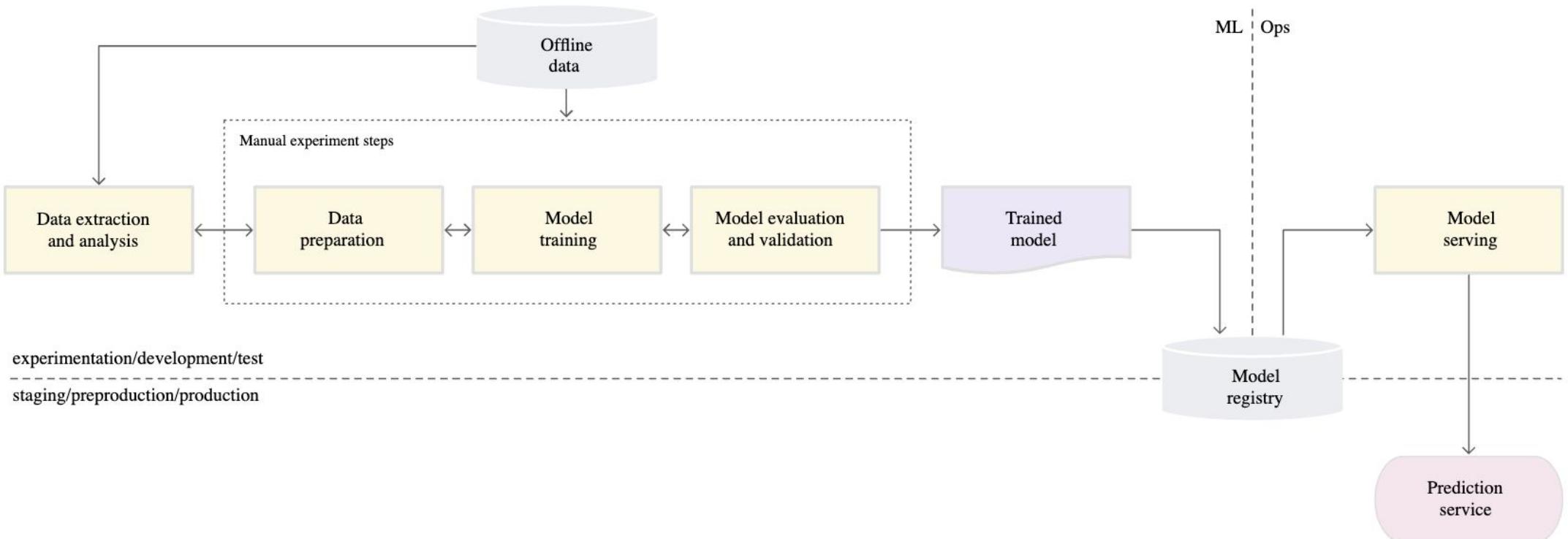
ИТОГИ

1. Данные храним и структурируем=)
2. Стремимся уметь отвечать на вышеизложенные вопросы

Тренировка и оценка качества
моделей.

MLOPS Level 0

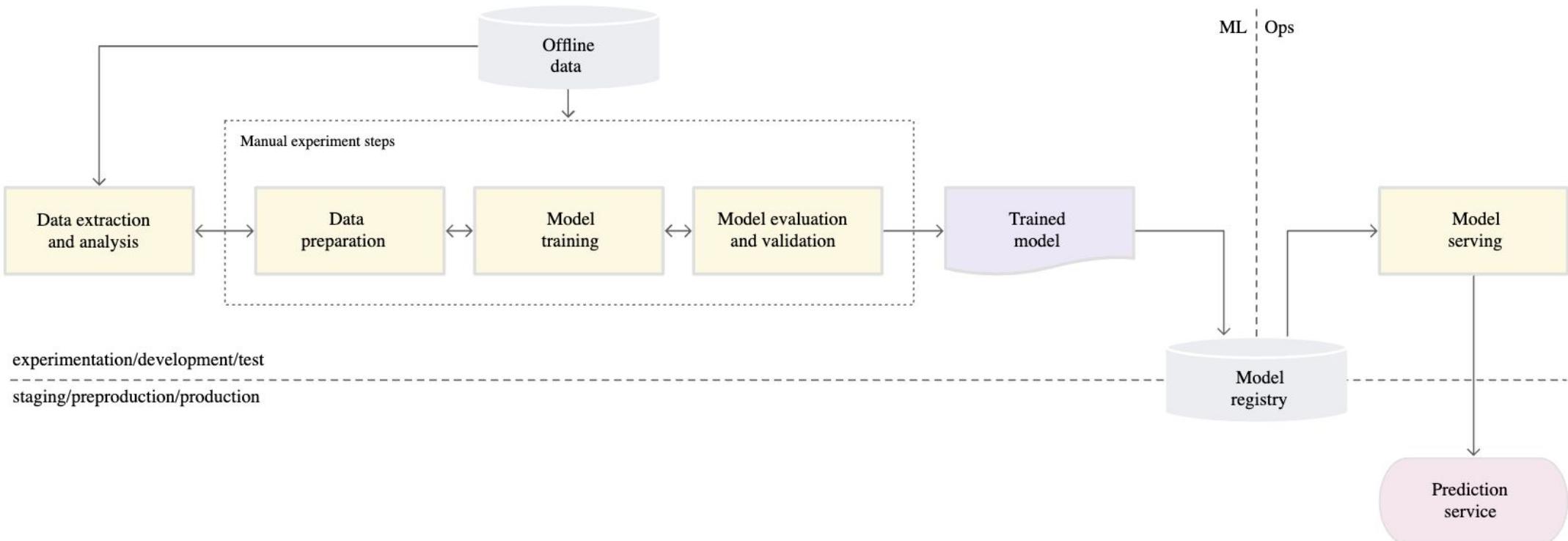
DS делает модели



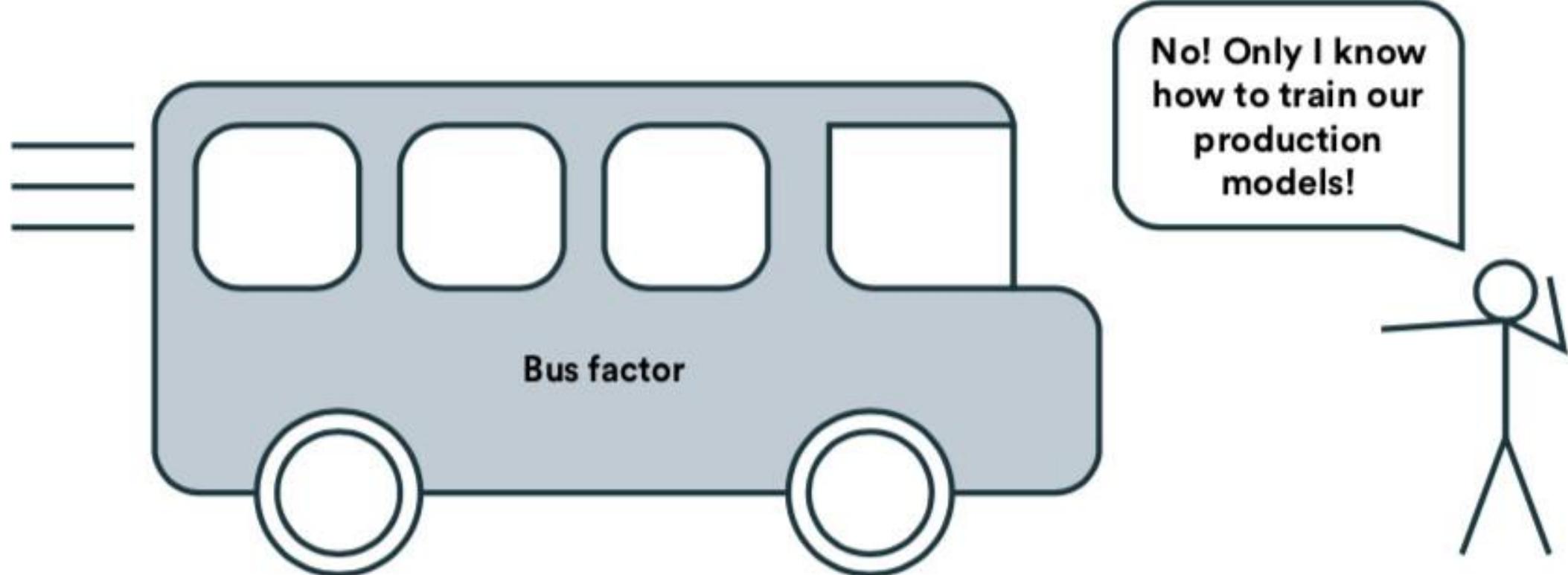
MLOPS Level 0



DS делает модели



BUS FACTOR





Проблемы

- Потери данных
- Потери кода
- Потери знаний о том, как обучать и выкапывать

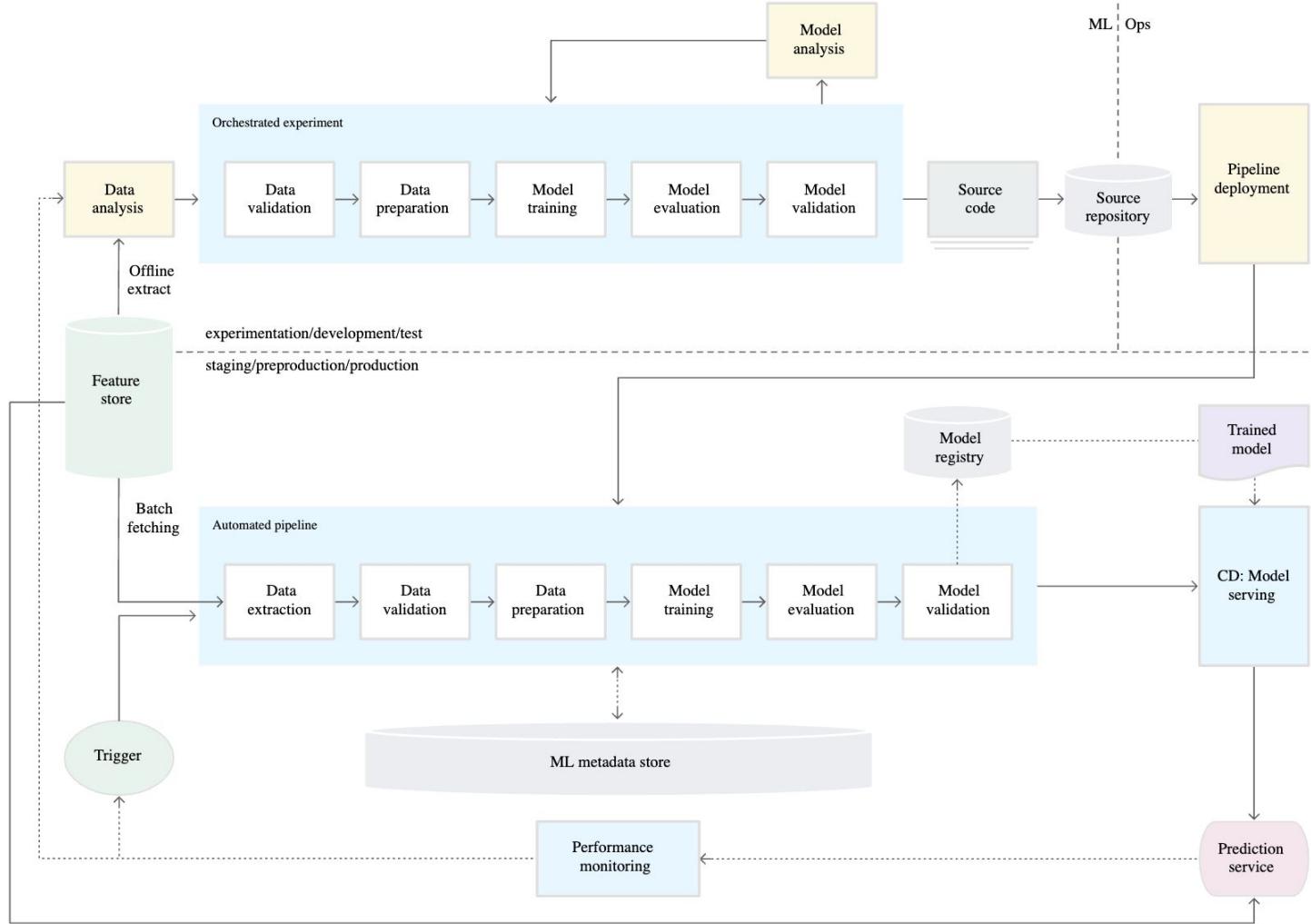


Training-Serving Skew

- Наши данные не такие, как на проде
- Фичи получаются не тем же способом, что на проде
- DS не думает о том, что на проде
- Теряется способ воспроизвести модель

MLOPS Level 1

DS делает пайплины,
которые делают модели



Что поможет реализовать?



Версионирование
кода



Шедулинг и
оркестрация



Версионирование
данных



Трекинг экспериментов



ИТОГИ

1. Нужно писать переиспользуемый, версионируемый, тестируемый код
2. Нужно осознание того, что модель — это не цель

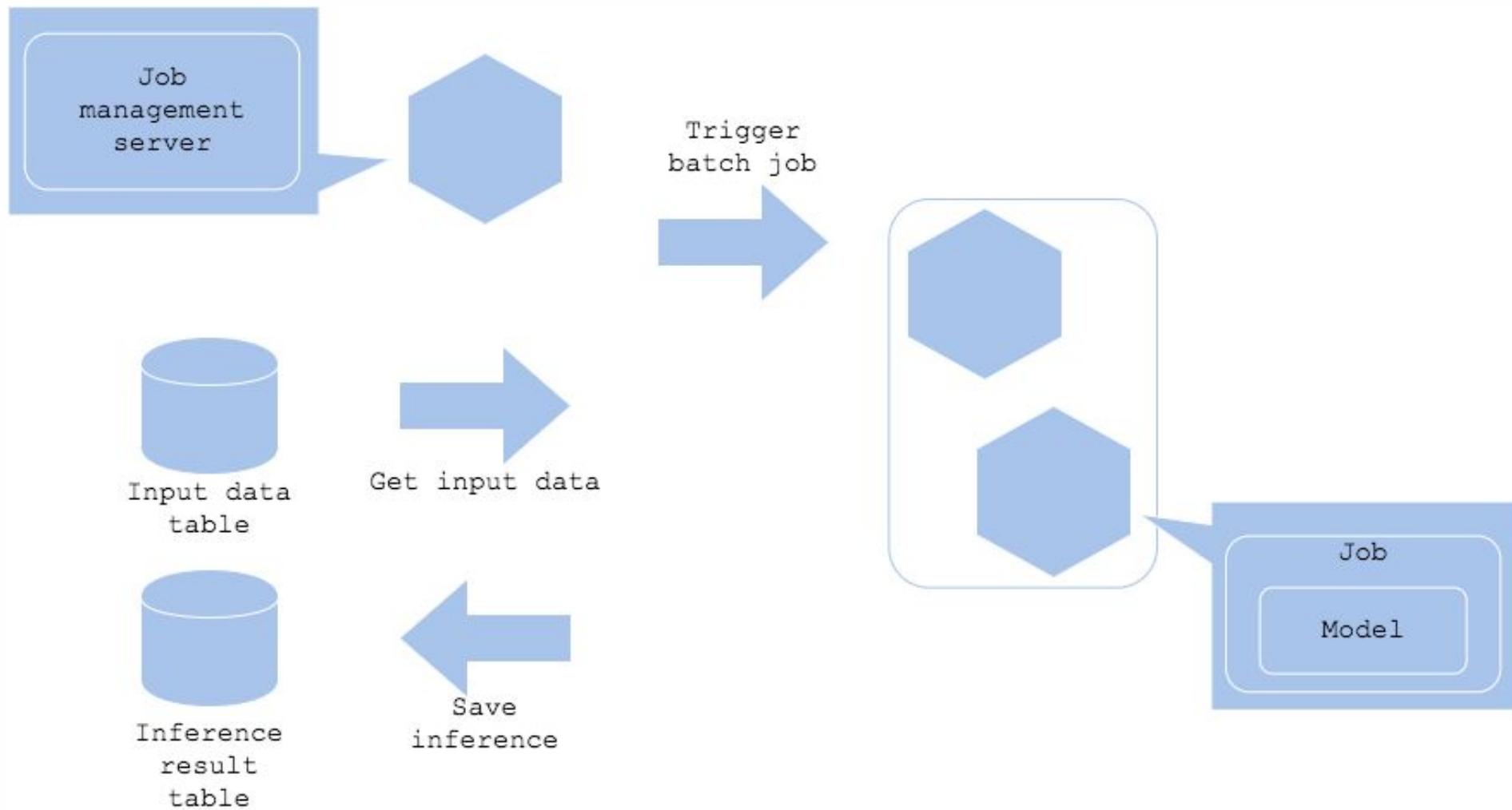
Использование моделей

Классический банковский ML

Index	ПОЛ	ВОЗРАСТ	ЗП	PREDICT
0	М	23	300 к/сек	1
1	М	33	45 000 р/мес	0.5
2	М	34	15 000 р/мес	0.1
3	Ж	55	55 000 р/мес	0.7

AI →

Пакетный паттерн





Когда нужно использовать?

- Если вам **НЕ** нужно получать результат прогноза в реальном времени или почти в реальном времени.
- Для массивной обработки данных
- Когда для корректной работы продакшена достаточно запускать PREDICTION по расписанию(раз в сутки, в час, в 10 минут)



Что поможет реализовать?

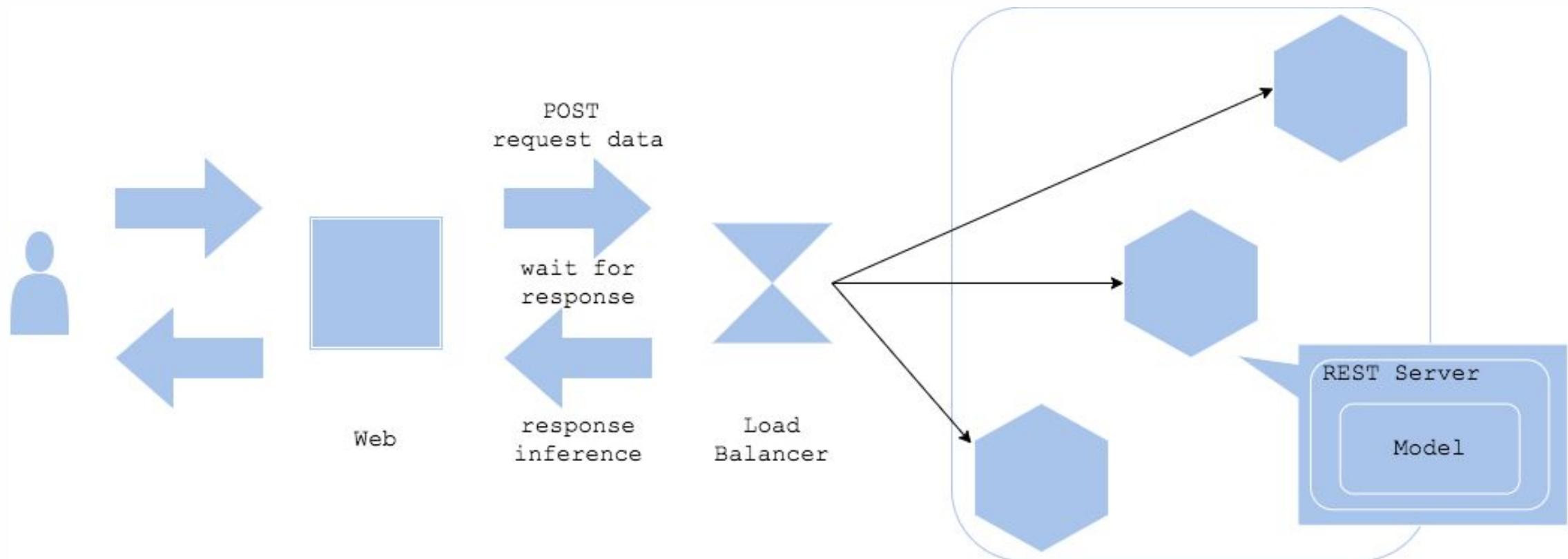


Apache
Airflow

Загружаешь фотку - получаешь ответ



Синхронный паттерн



https://github.com/mercari/ml-system-design-pattern/blob/master/Serving-patterns/Synchronous-pattern/design_en.md



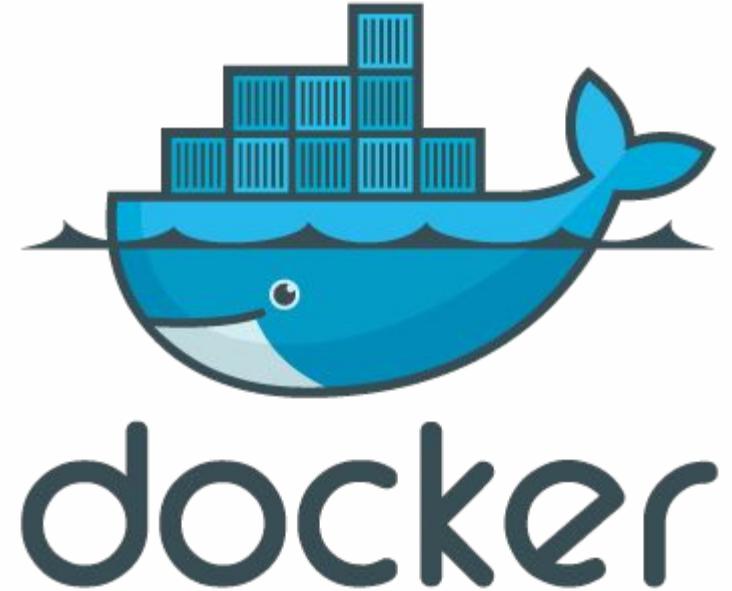
Синхронный паттерн

- когда вам нужен результат предсказания для того, чтобы сделать следующий шаг

ПЛЮСЫ	МИНУСЫ
Просто в реализации	Скорость предсказания будет бутылочным горлышком производительности вашей системы
Как правило, низкое latency	Если предсказание слишком долгое, то нужно сделать так, чтобы пользователь этого не замечал

https://github.com/mercari/ml-system-design-pattern/blob/master/Serving-patterns/Synchronous-pattern/design_en.md

Что поможет реализовать?



Удаление спам сообщений в Instagram



irina_almazova_proff Классные вы ❤️
_777777755 Шоколад и мармелад ❤️
anikeeva600 Вы классные!!! Любите
друг друга.... И будьте счастливы!!!
iiodas ❤️
sweetypresentbouquet Очень
красивая пара 😊 ❤️
nastyawol wol ❤️❤️❤️⭐⭐⭐
marinavictory ❤️
cornershop_777 🔥🔥🔥
expressbeautykiev Привет!
Подпишитесь на нашу страницу и Вы
окунетесь в атмосферу
совершенства, красоты и
привлекательности. 💋😊🦋

master_school_socchi СЧАСТЬЯ ВАМ
РЕБЯТА ❤️❤️❤️



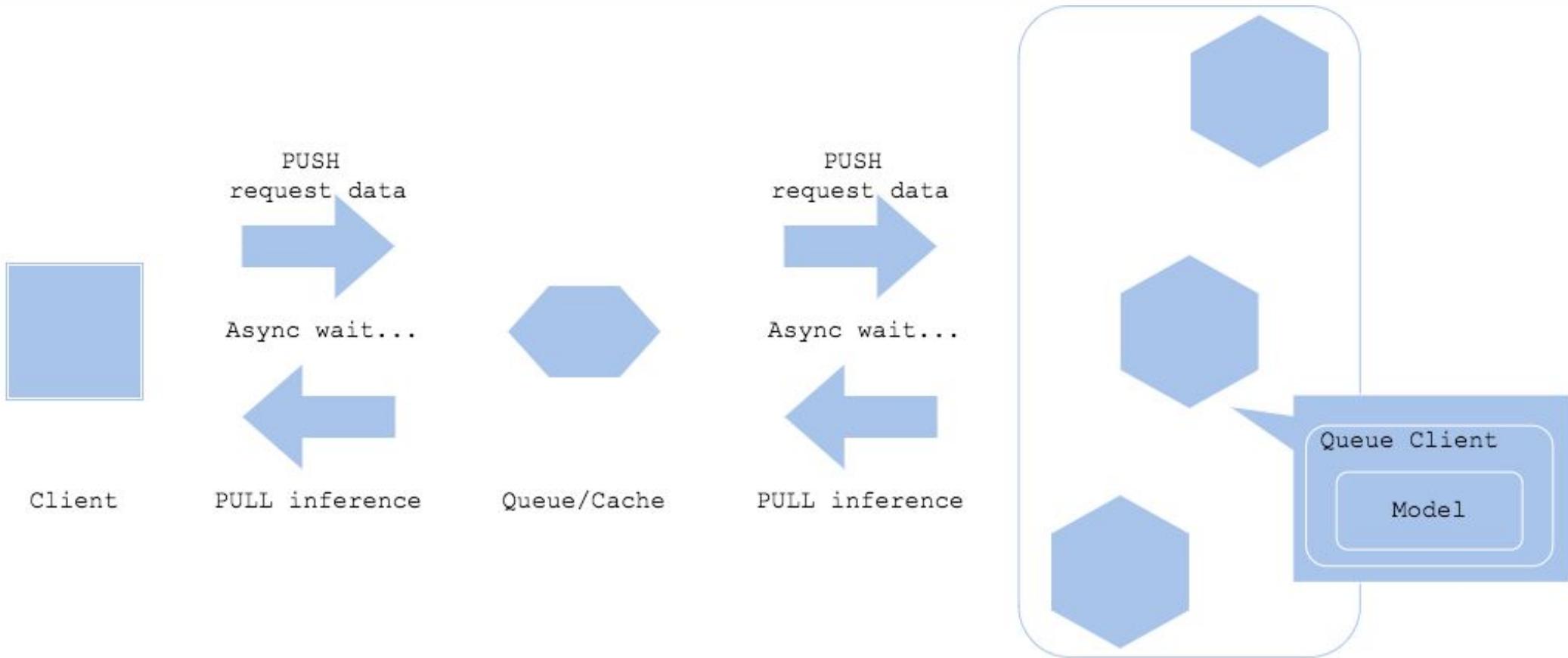
273 933 отметок "Нравится"

17 ЯНВАРЯ

Добавьте комментарий...

...

Асинхронный(near realtime) паттерн



https://github.com/mercari/ml-system-design-pattern/blob/master/Serving-patterns/Asynchronous-pattern/design_en.md



Асинхронный(near realtime) паттерн

- когда вам не нужен результат прямо сейчас

ПЛЮСЫ	МИНУСЫ
Можно отделить бизнес логику и логику предсказания	Как правило, не подходит для использования в реальном времени.
Более высокая(по сравнению с синхронным) пропускная способность	Нужны очереди/кеши, etc
Вы не заблокированы временем предсказания	

https://github.com/mercari/ml-system-design-pattern/blob/master/Serving-patterns/Asynchronous-pattern/design_en.md



Что поможет реализовать?





ИТОГИ

1. Есть несколько сценариев использования ML моделей
2. Нужно уметь выбирать нужный в зависимости от ограничений, целей
3. Реализуются по разному

Деплой

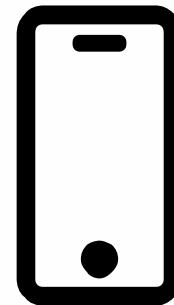
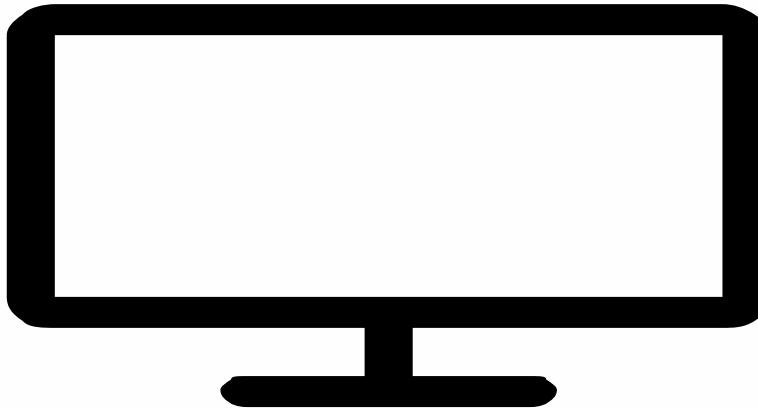
Куда деплоить?



kubernetes



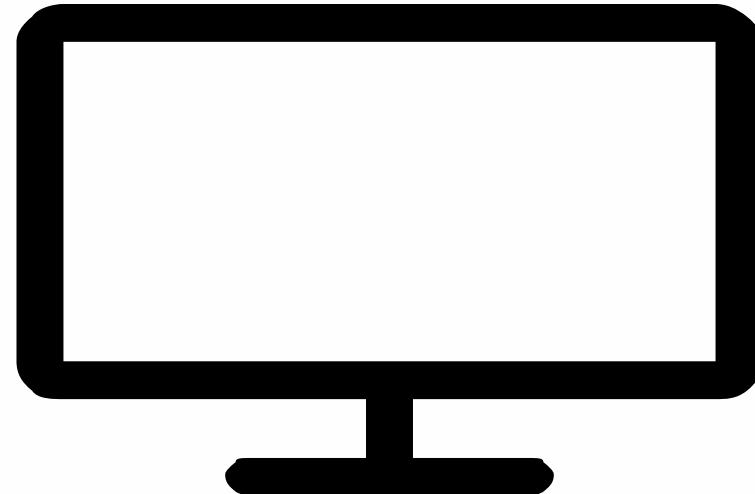
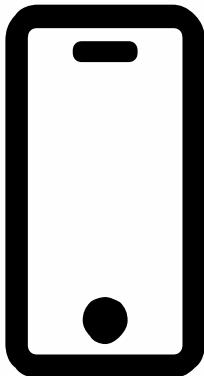
Google Cloud





Embedded

Использование модели встроено в приложение





Embedded: характеристики

Нет сетевой задержки
можно запустить на девайсе
Сложно масштабировать
независимо от приложения



Model as a service

Использование модели вынесено в отдельный сервис



Model as a service: характеристики

Сетевой задержка

Можно масштабировать
независимо от приложения



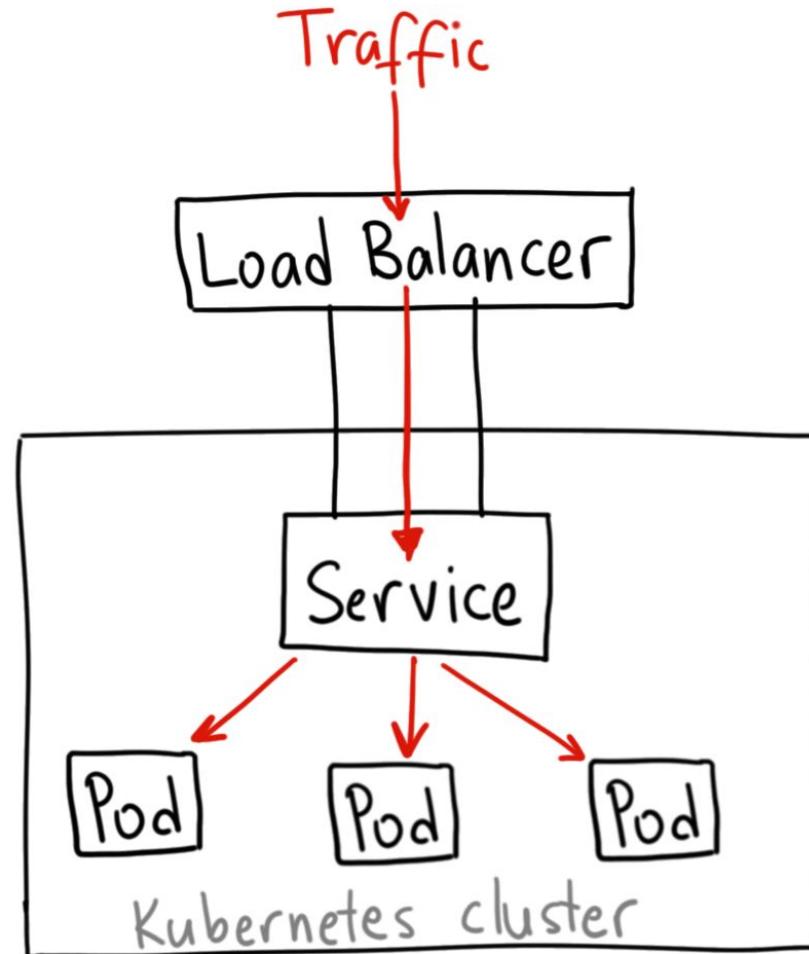
Масштабирование приложения

Сетевая задержка

Независимый релизный цикл

Можно масштабировать
независимо от приложения

Масштабирование

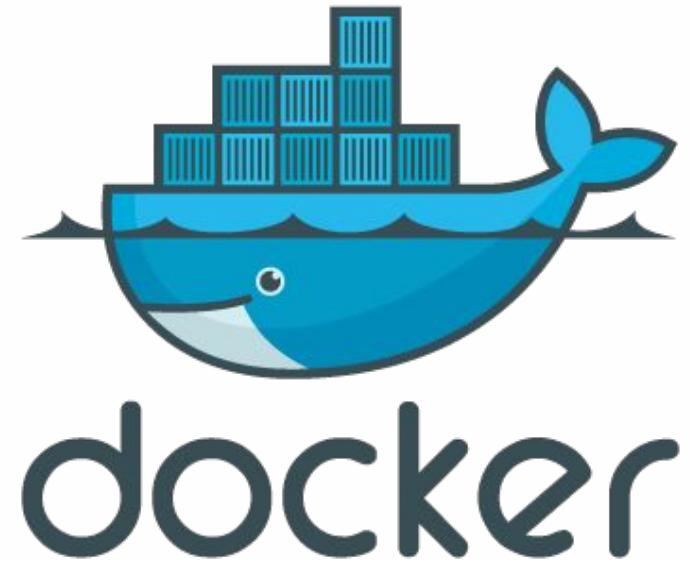


UPDATE модели



V1.1 -> V1.2

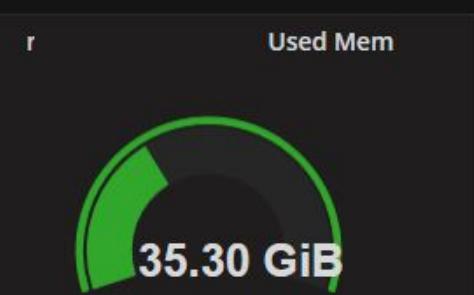
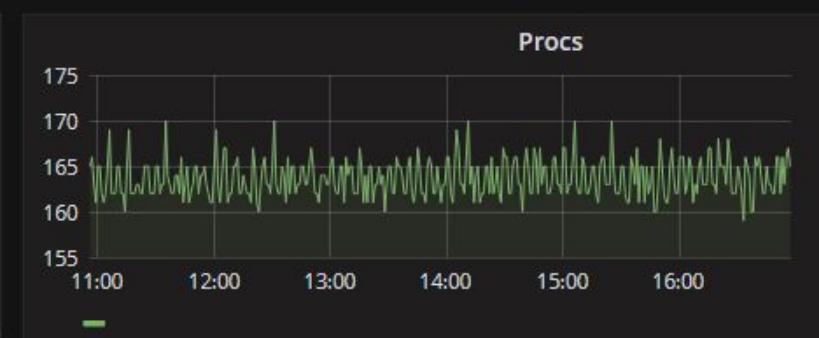
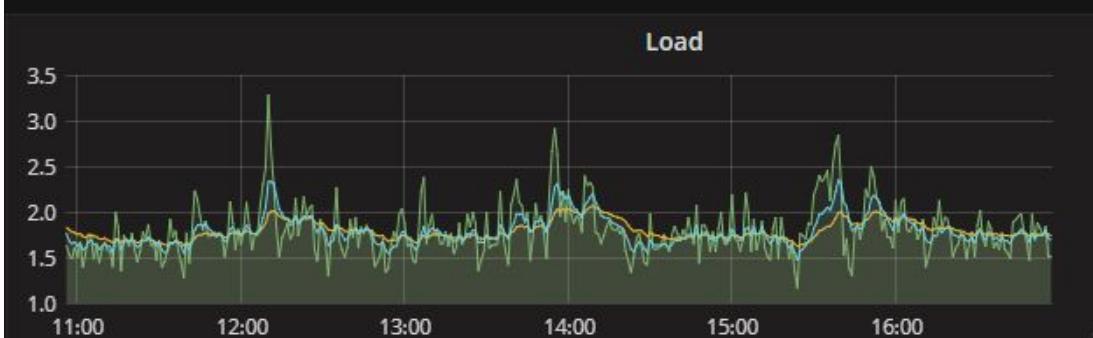
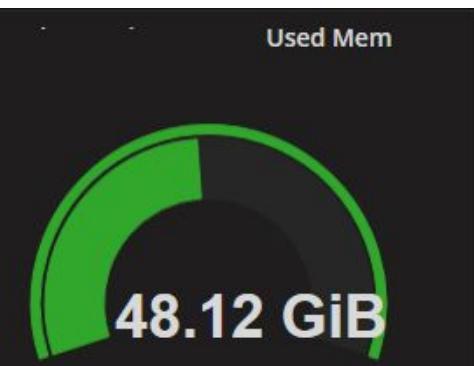
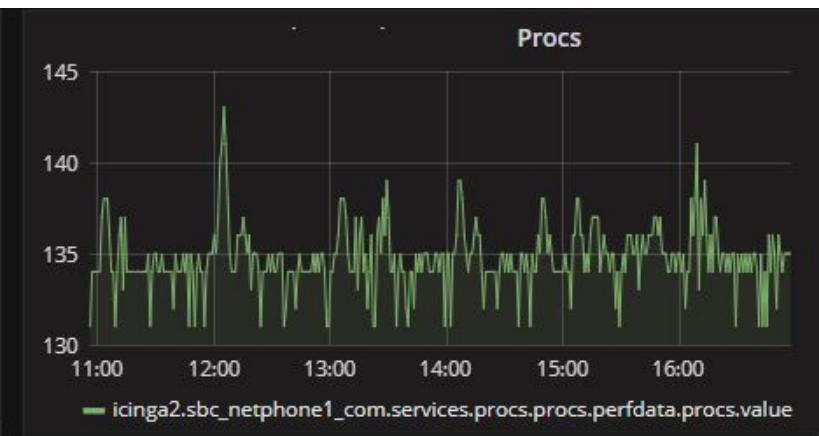
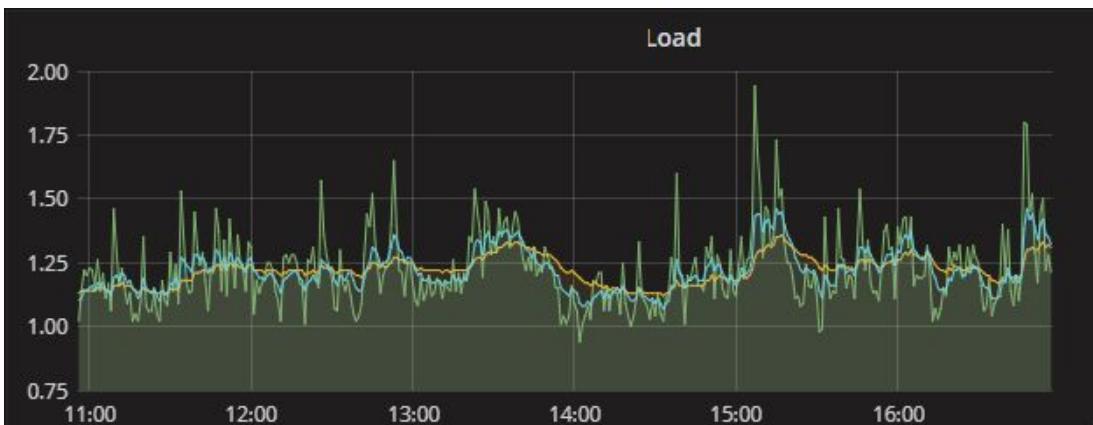
Что поможет реализовать?



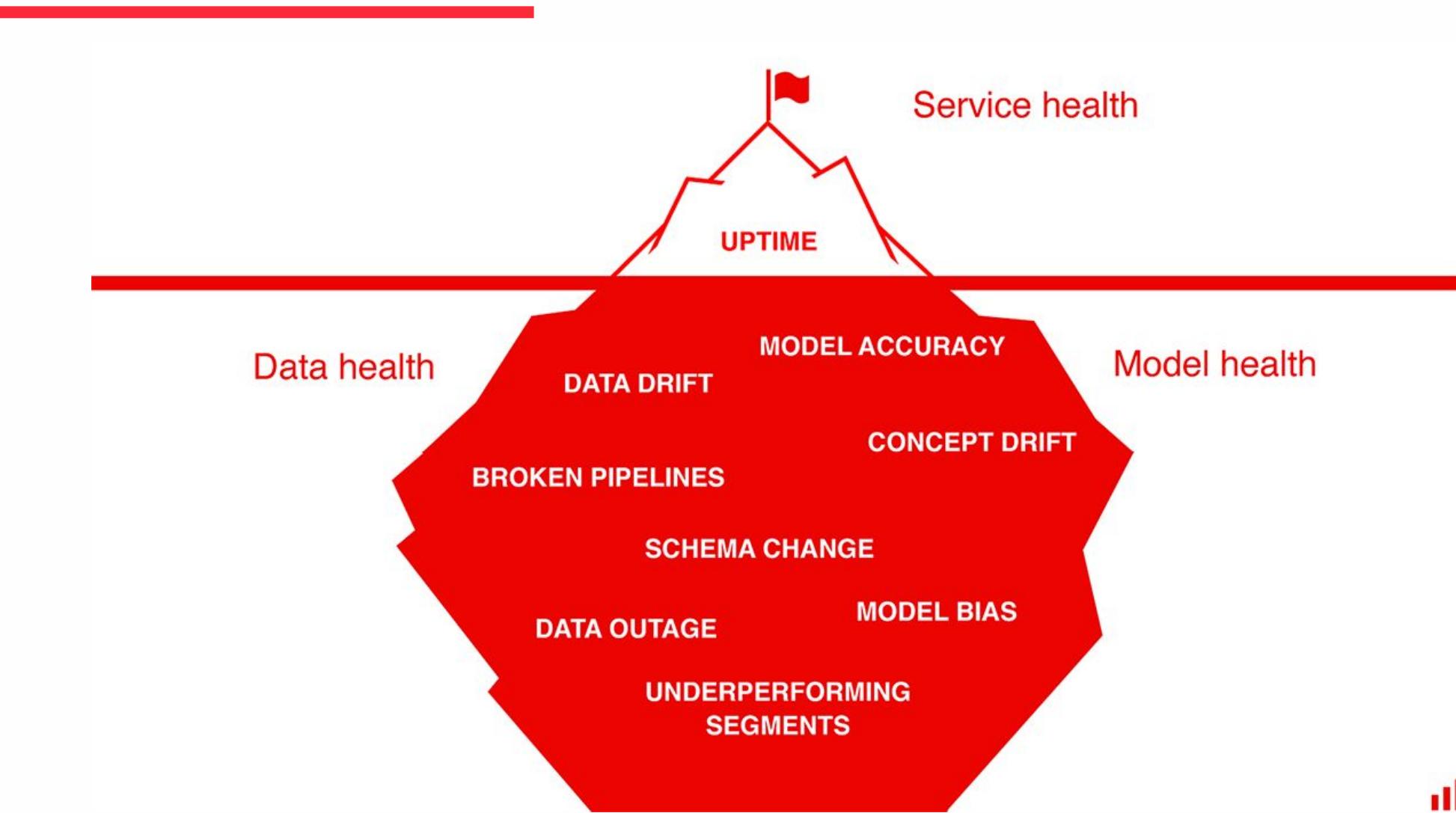
Мониторинг

Технические метрики

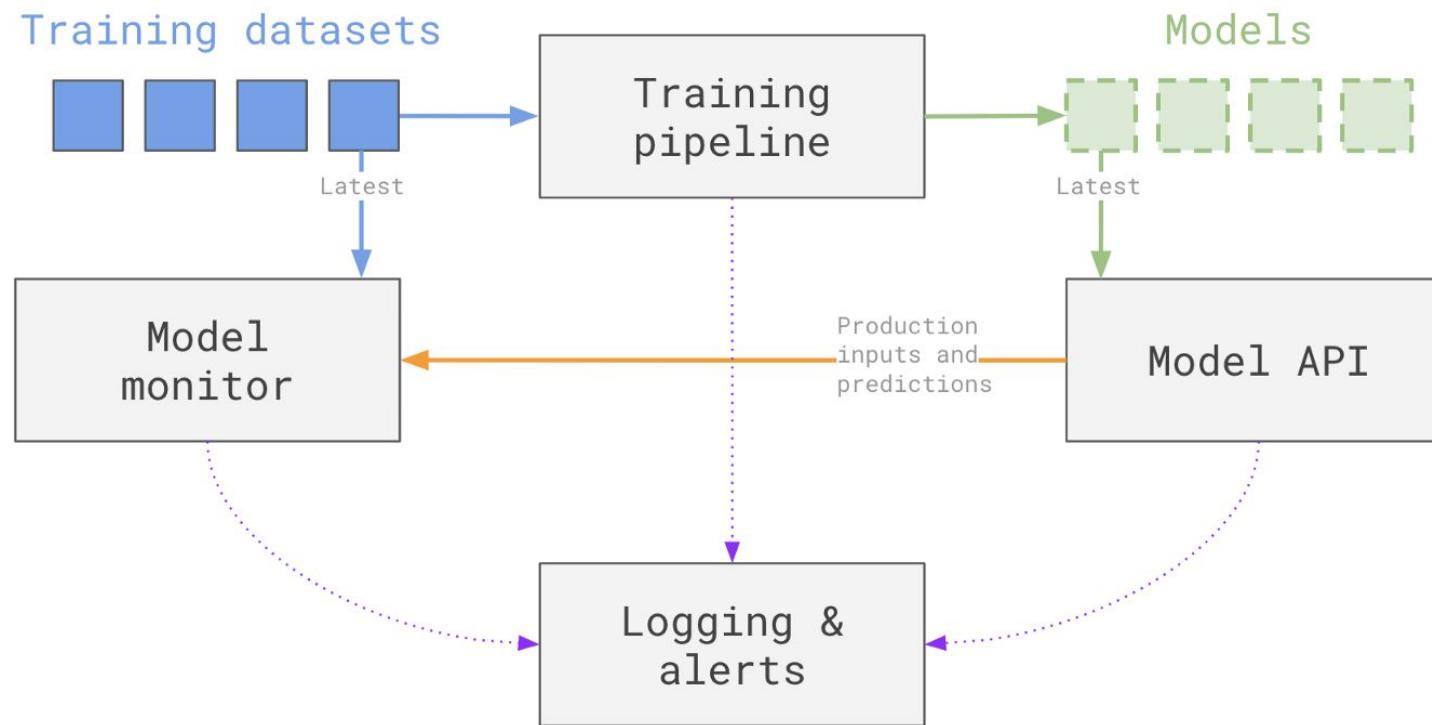
cput, gput, query per seconds, errors



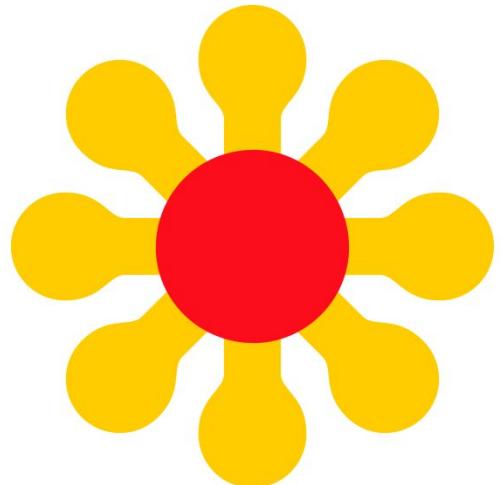
ML specific



Monitoring Scheme

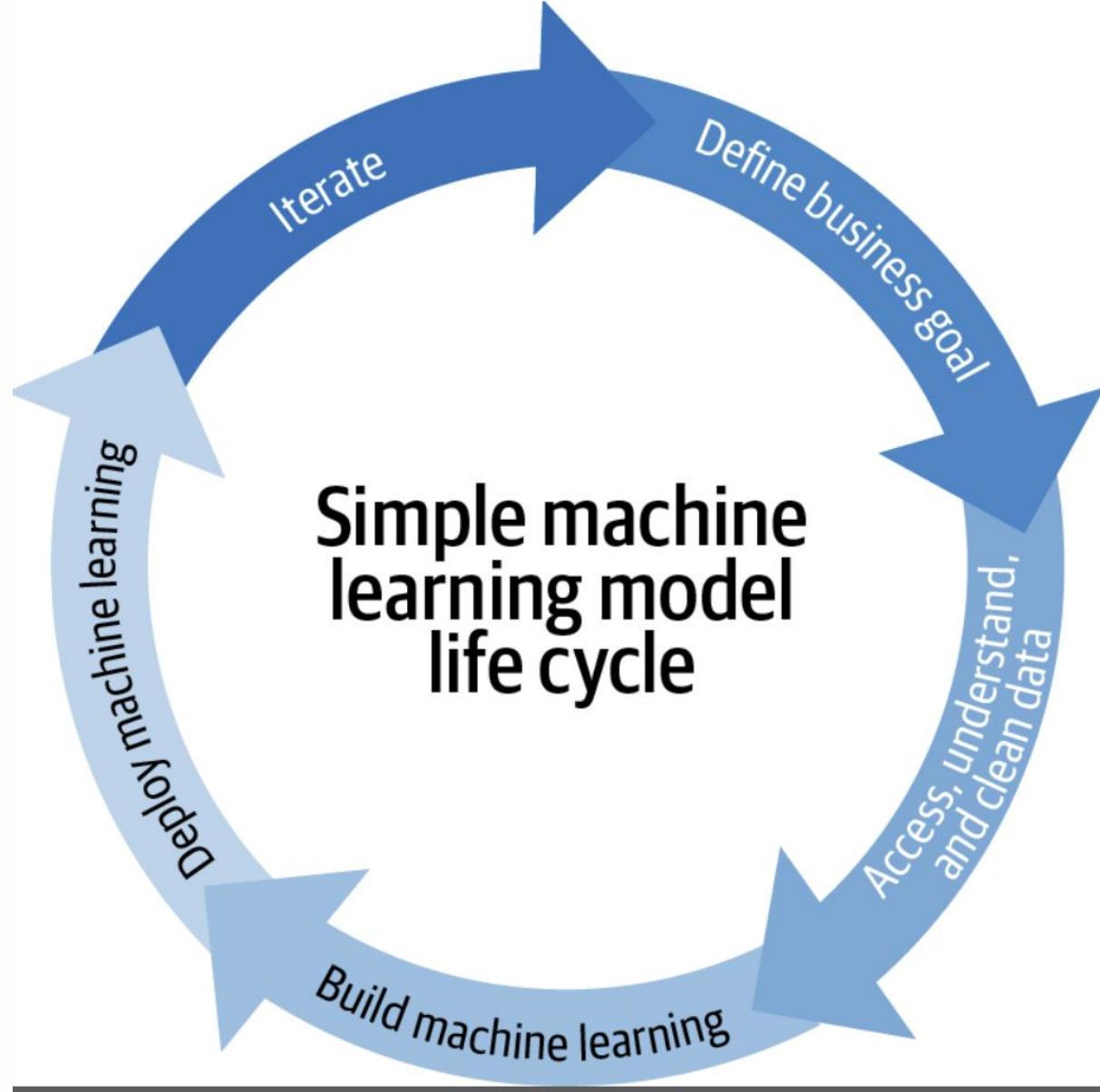


Что поможет реализовать?



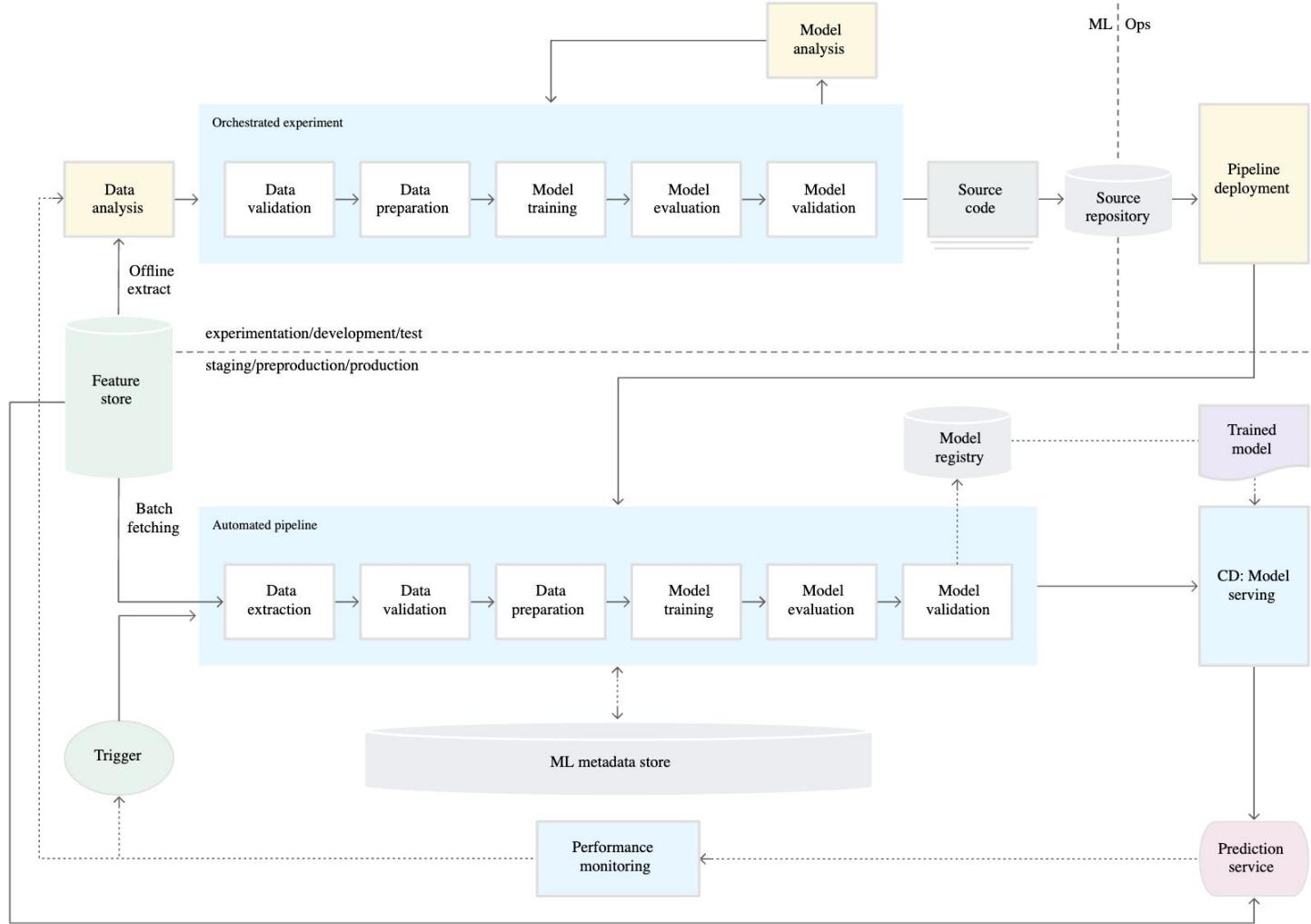
Итерация

Модель разработана не раз и навсегда

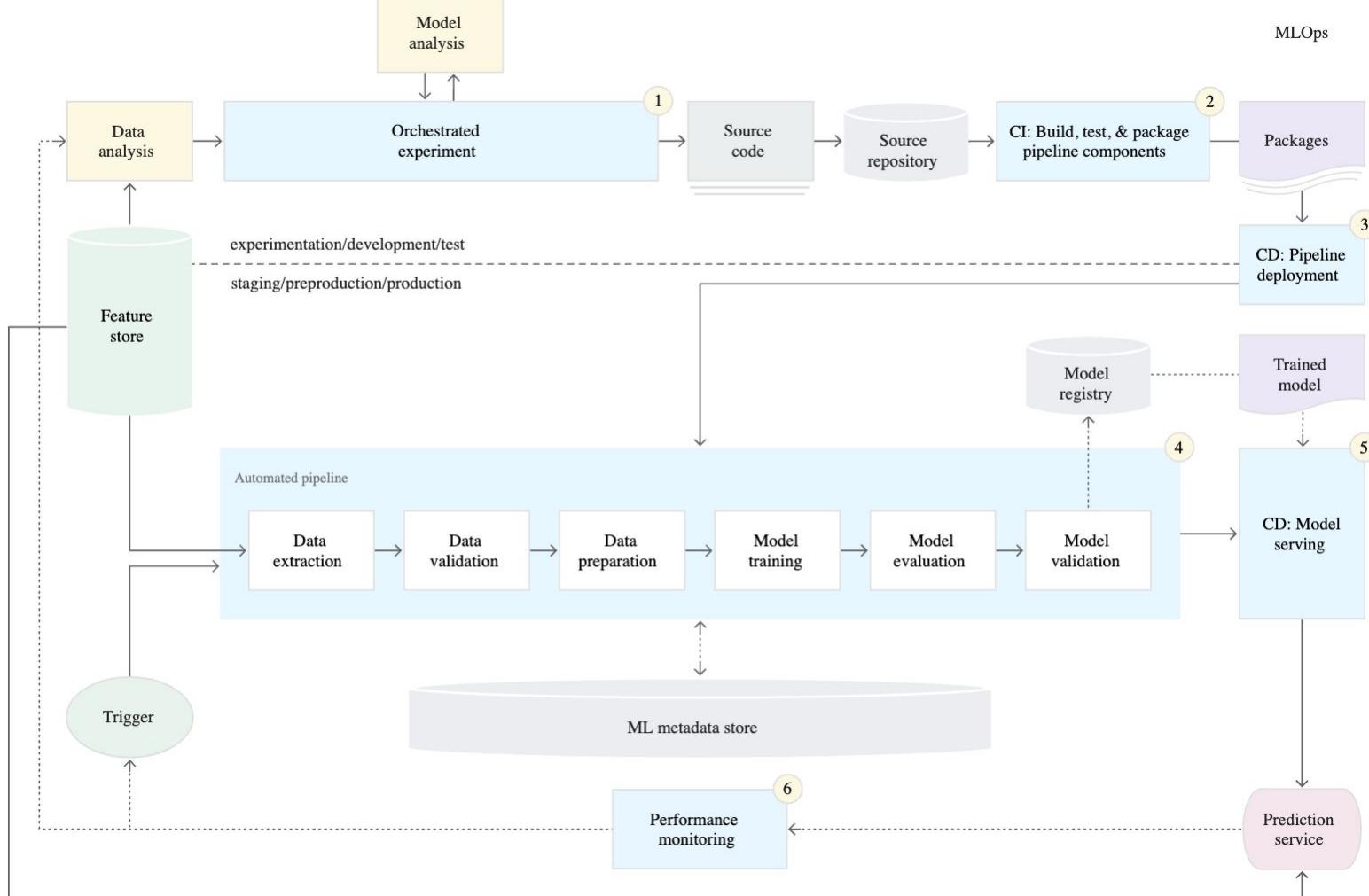


MLOPS Level 1

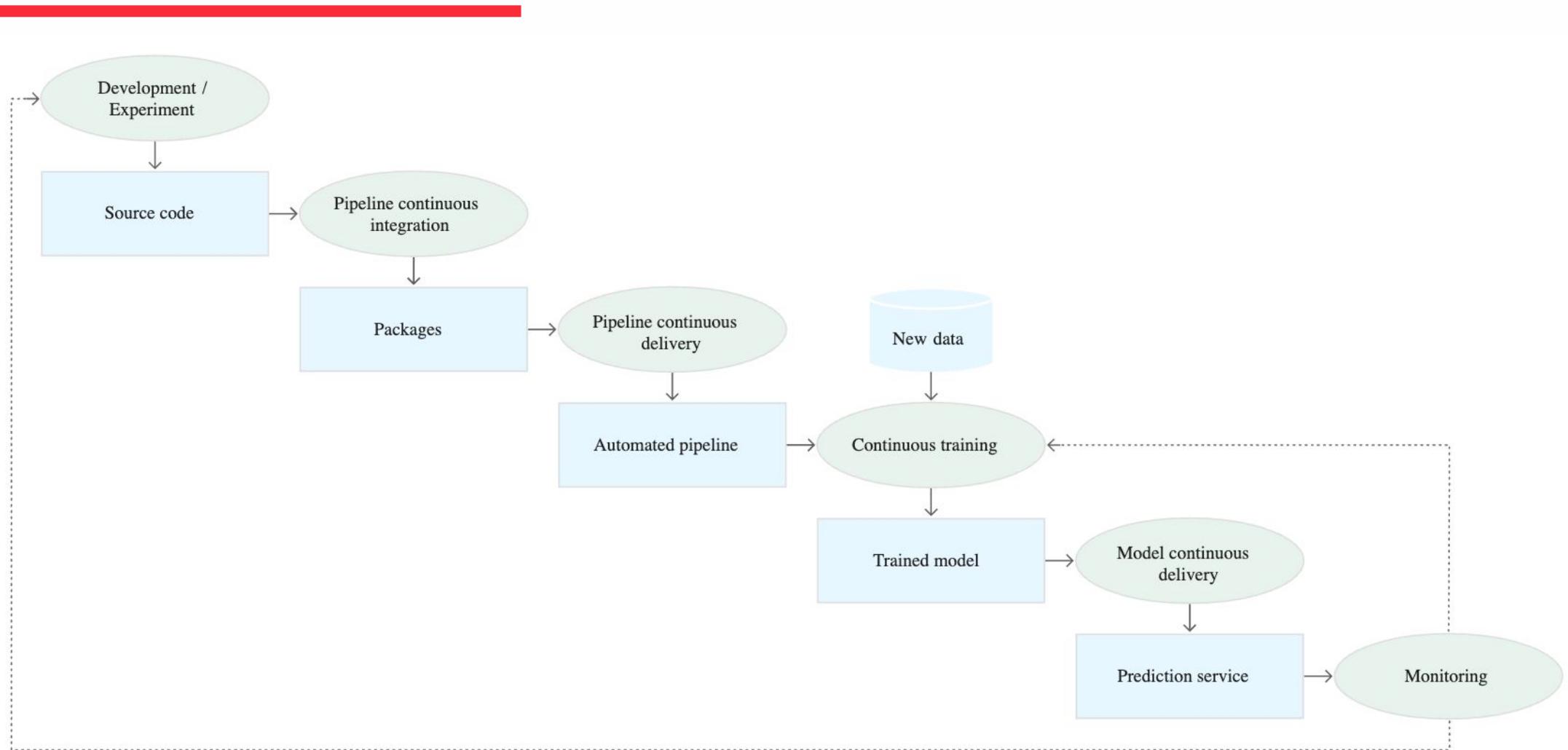
DS делает пайплины,
которые делают модели



MLOPS: level 2



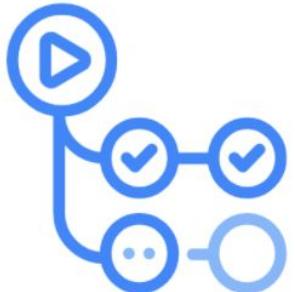
Автоматизация помогает двигаться быстрее



Что поможет реализовать?



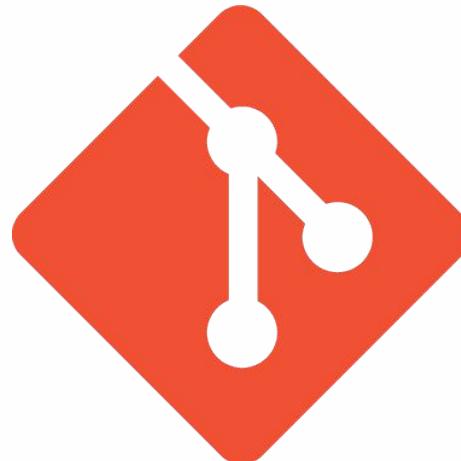
Apache
Airflow



GitHub Actions



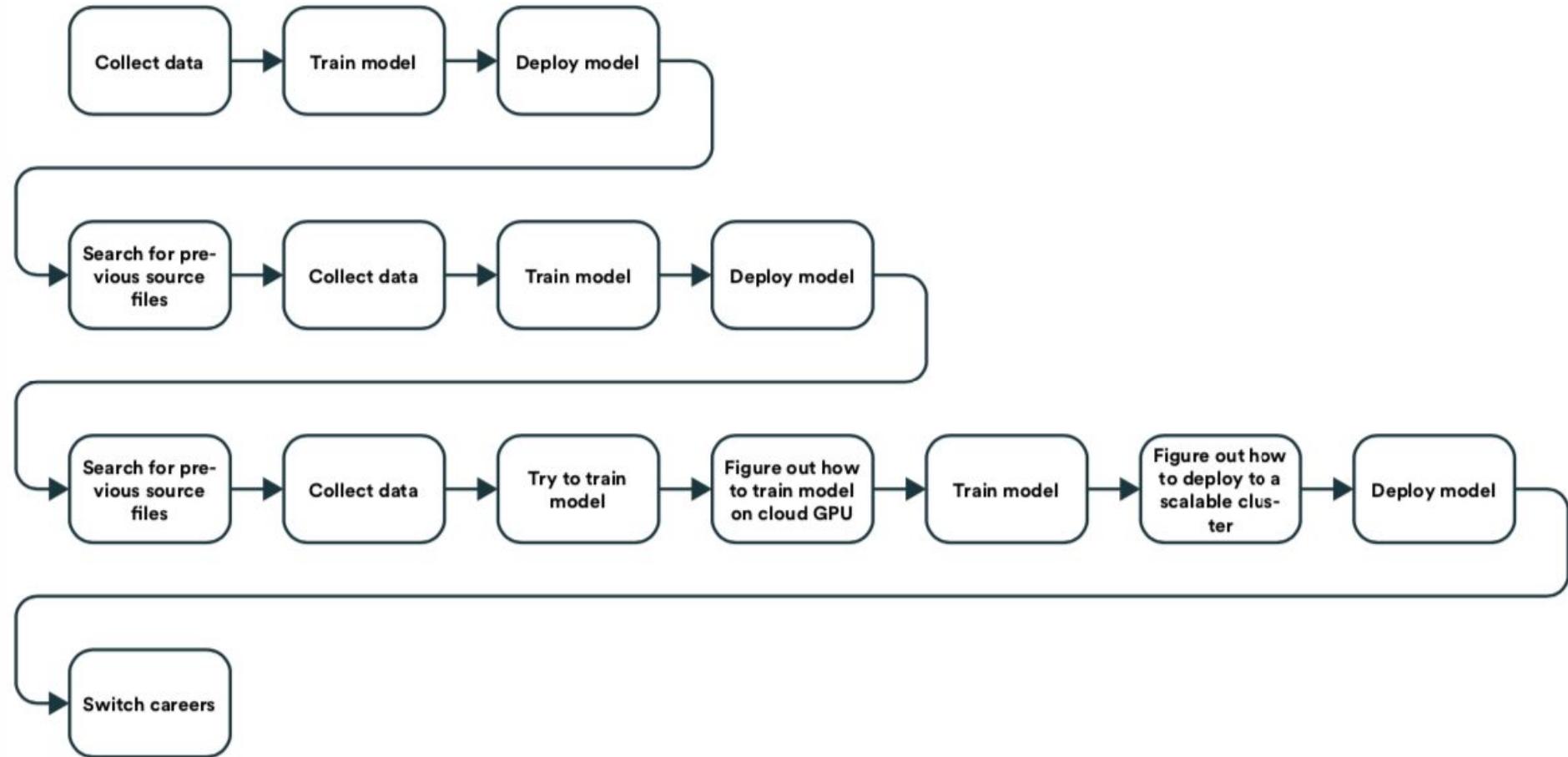
kubernetes



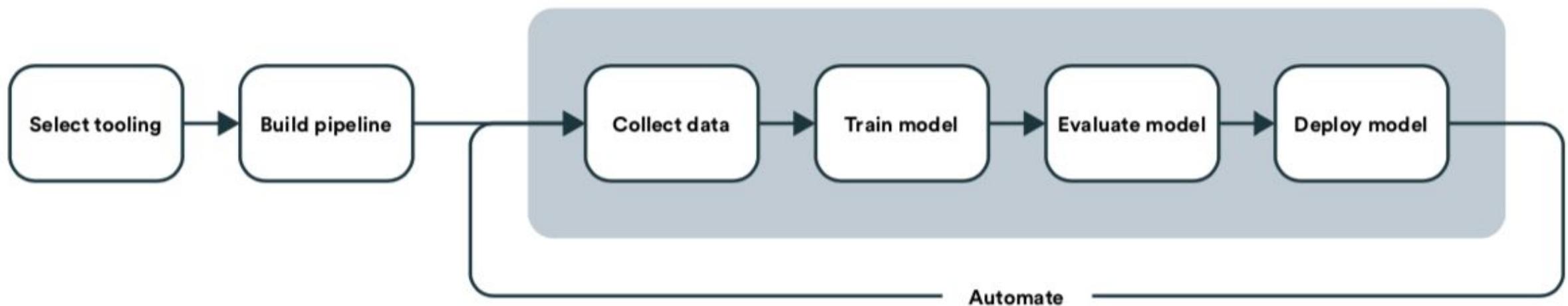
git

История 2-х компаний

Company 1



Company 2



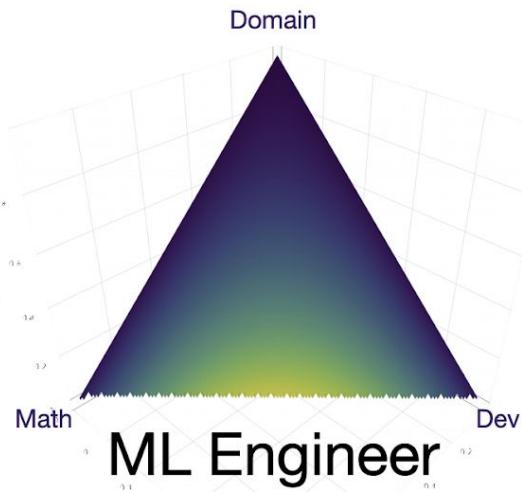


Итоги

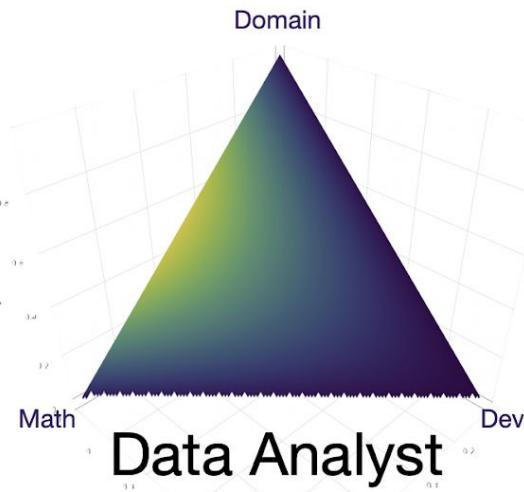
- Двигаться быстро важно
- Но вложиться в инфра/процесс тоже важно

Роли в ML

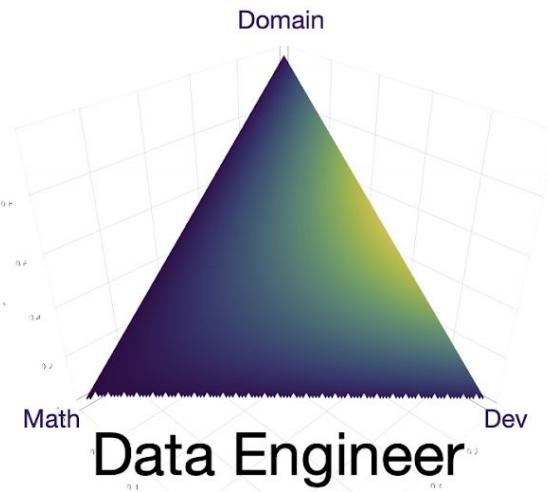
Роли в ML



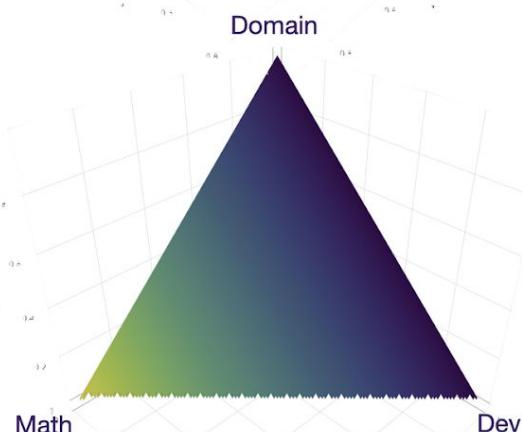
ML Engineer



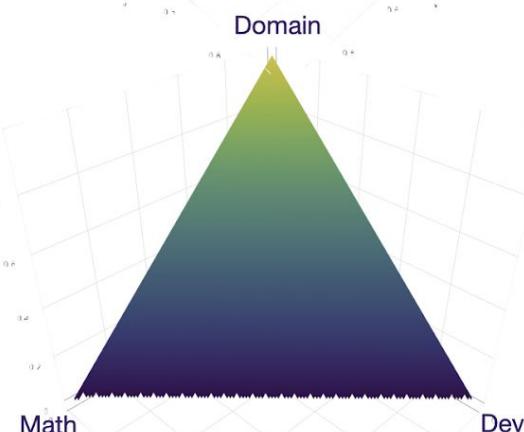
Data Analyst



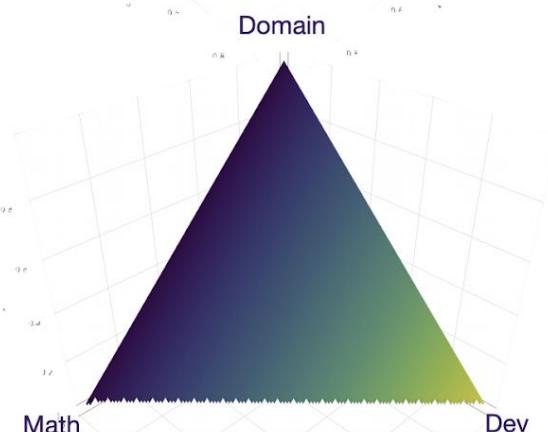
Data Engineer



ML Researcher

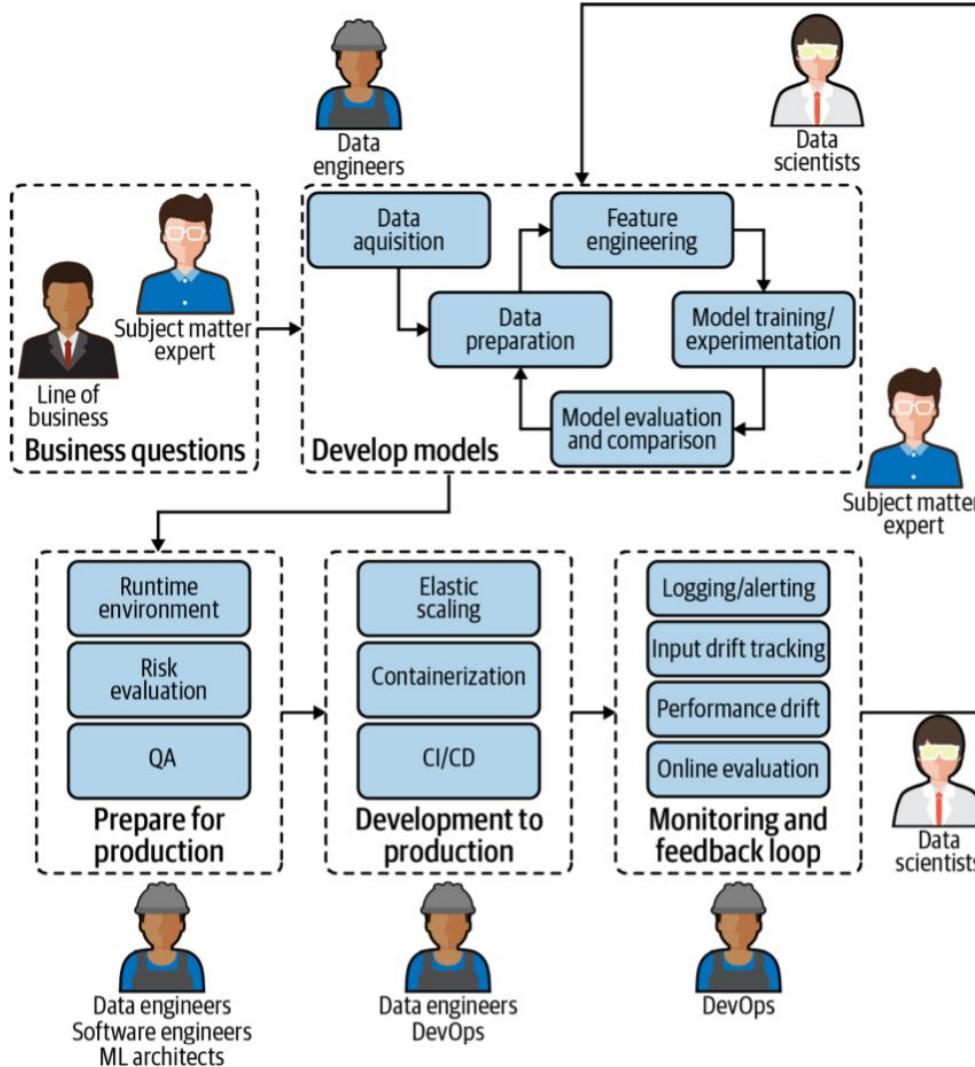


Analyst



Devops

Enterprise ML





Data Scientist/ML Engineer

Разработку модели

За то, что модель готова к эксплуатации

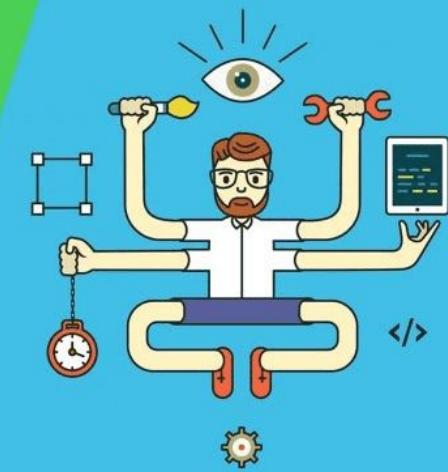
Развертывание модели +-

Оценка качества модели онлайн-оффлайн

Итеративное улучшение моделей

Data Engineer

**BIG
DATA
ENGINEER**



Создание платформы данных

Разработку конкретных источников для
моделей

Оптимизацию производительности data
pipelines

Software Engineer

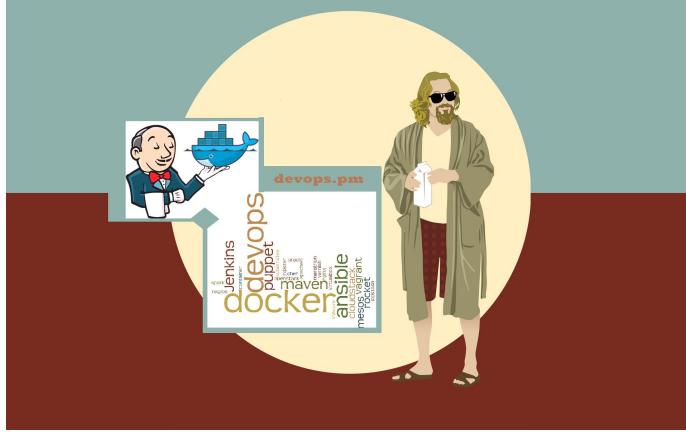


Встраивание моделек в продукт

DEVOPS

Создание платформы для развертывания и мониторинга приложений (в том числе и моделей)

Создание CICD пайплайнов для компонентов ML систем





ИТОГИ

1. Люди всякие нужны, люди всякие важны
2. Data Scientist должен иметь возможность доставлять результаты своего труда САМ
3. Переписывать за DS код тренировки/инференса моделей — антипаттерн

Be T-shaped



Mindset



CI/CD Pipeline



Frontend



Cloud Platform



Backend API



Container



Research



Machine Learning



Math

<https://towardsdatascience.com/t-shaped-skills-builder-guide-in-2020-for-end-to-end-data-scientist-33d2652511b0>

GIT

```
background-color: #333;
text-shadow: 0px -1px 0px #000;
filter: dropshadow(color="#333");
color:#777;

}

header #main-navigation ul li span:hover,
header #main-navigation ul li span:active {
    border: 1px solid #ccc;
    border-bottom: none;
    background-color: #F9F9F9;
    box-shadow: 0px 0px 1px #ccc,
    -webkit-box-shadow: 0px 0px 1px #ccc,
    moz-box-shadow: 0px 0px 1px #ccc;
    padding: 5px 10px;
    font-weight: bold;
}
```



План

- 1) Зачем нужны системы контроля версия
- 2) GIT - основные возможности

Зачем нужны системы контроля версий?



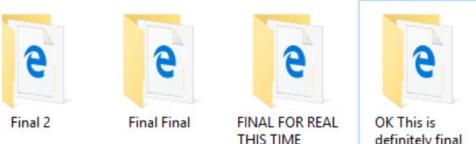
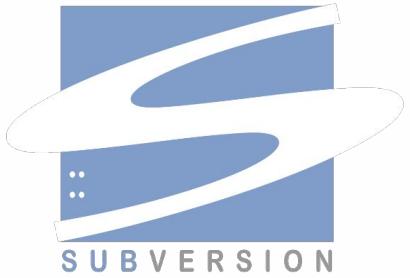
Системы контроля версий

- 1) Возврат к старым версиям (safety net)
- 2) Отслеживание изменений
- 3) Хранение истории
- 4) Совместная работа

Системы контроля версий

 Mikhail-M	Update python-package.yml	✗ f6f7b5f 9 days ago	⌚ 10 commits
	.github/workflows	Update python-package.yml	9 days ago
	configs	commit project	2 months ago
	docs	commit project	2 months ago
	ml_example	Sklearn pipelines (#1)	2 months ago
	models	commit project	2 months ago
	notebooks	commit project	2 months ago
	references	commit project	2 months ago
	reports	commit project	2 months ago
	tests	Sklearn pipelines (#1)	2 months ago
	.gitignore	commit project	2 months ago
	LICENSE	commit project	2 months ago
	README.md	commit project	2 months ago
	requirements.txt	commit project	2 months ago
	setup.py	commit project	2 months ago
	tox.ini	fix flake8	2 months ago

Как можно сделать?

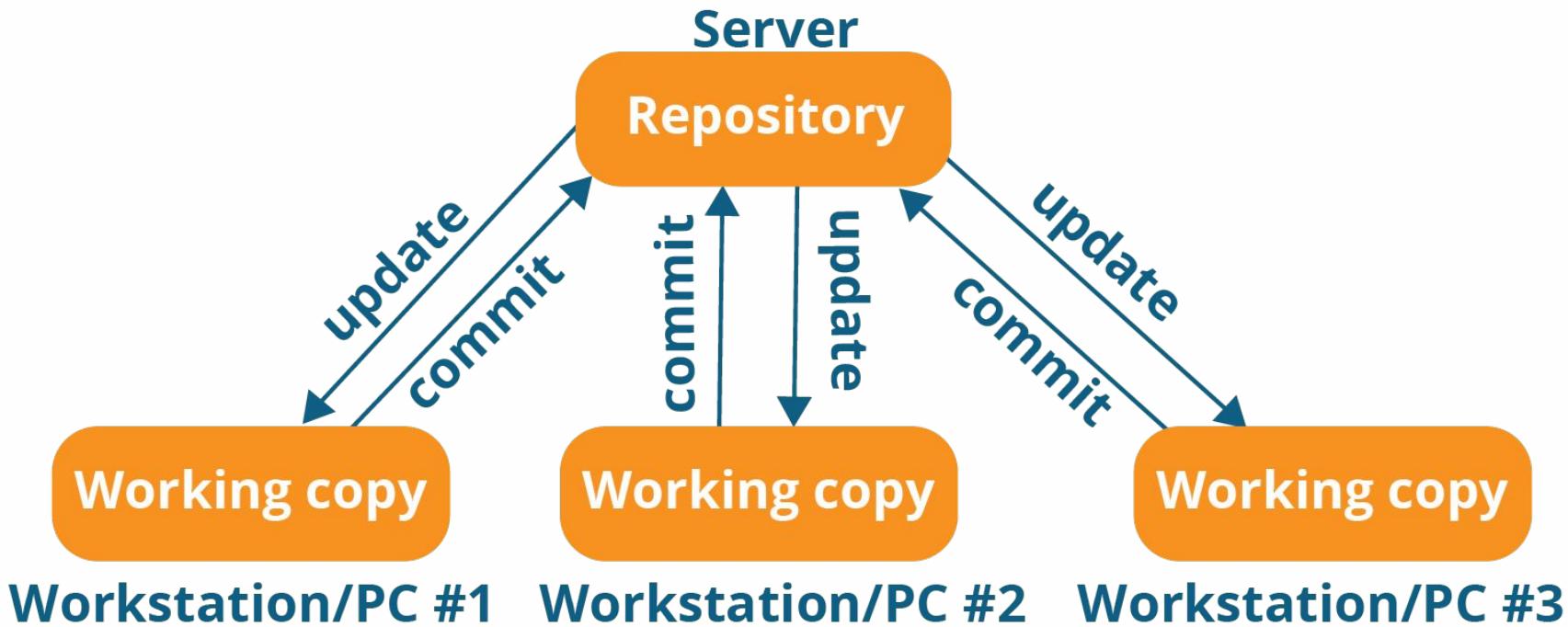


Версионировать файлы

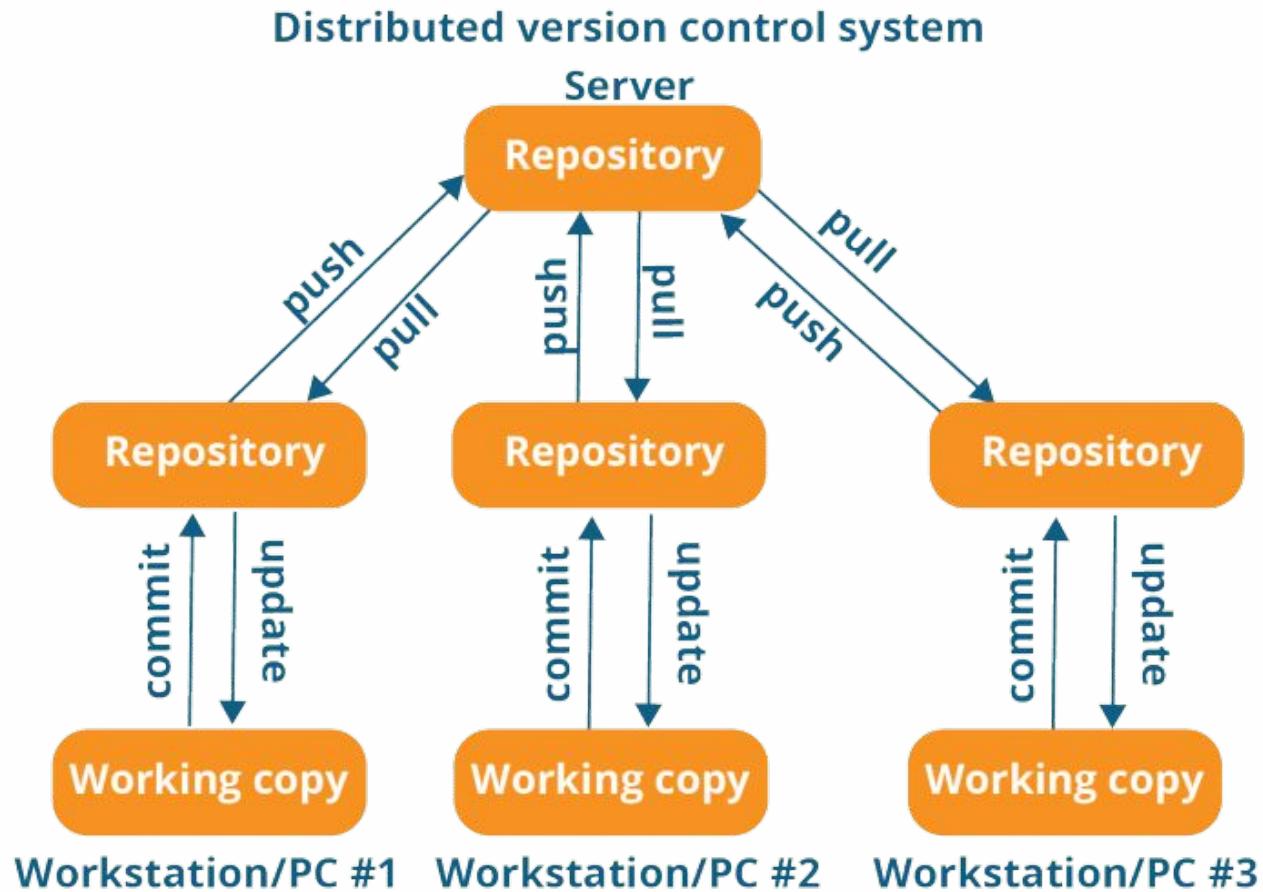
```
9      13
10     - def impute_features(df: pd.DataFrame, strategy: str) -> pd.DataFrame:
11      14 + def get_imputer(strategy: str) -> _BaseImputer:
12      15     imputer = SimpleImputer(missing_values=np.nan, strategy=strategy)
13      16 -     features_transformed = imputer.fit_transform(df)
14      17 -     features_pandas = pd.DataFrame(
15      18 -         features_transformed, columns=df.columns, index=df.index,
16      19 -     )
17      20 -     return features_pandas
18      21
19      22 +     return imputer
20      23
21      24 - def impute_categorical_features(df: pd.DataFrame):
22      25 -     return impute_features(df, strategy="most_frequent")
23      26
24      27 + def get_categorical_imputer() -> _BaseImputer:
25      28 +     return get_imputer(strategy="most_frequent")
26      29
27      30 - def impute_numerical_features(df: pd.DataFrame):
28      31 -     return impute_features(df, strategy="mean")
```

Центральная система контроля версия

Centralized version control system



Распределенная система контроля версий



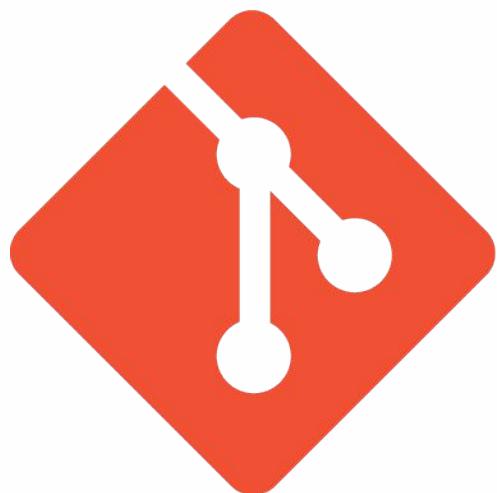
Какие бывают?



TOP VERSION CONTROL SYSTEMS

Git

- Стандарт де-факто
- Распределенный
- github.com, gitlab.com, <https://bitbucket.com>



git



Github

- Стандарт де-факто
- Распределенный
- github.com, gitlab.com, [https://bitbucket](https://bitbucket.org)



git add

Добавляем файлик под контроль версий

```
git add file.txt
```

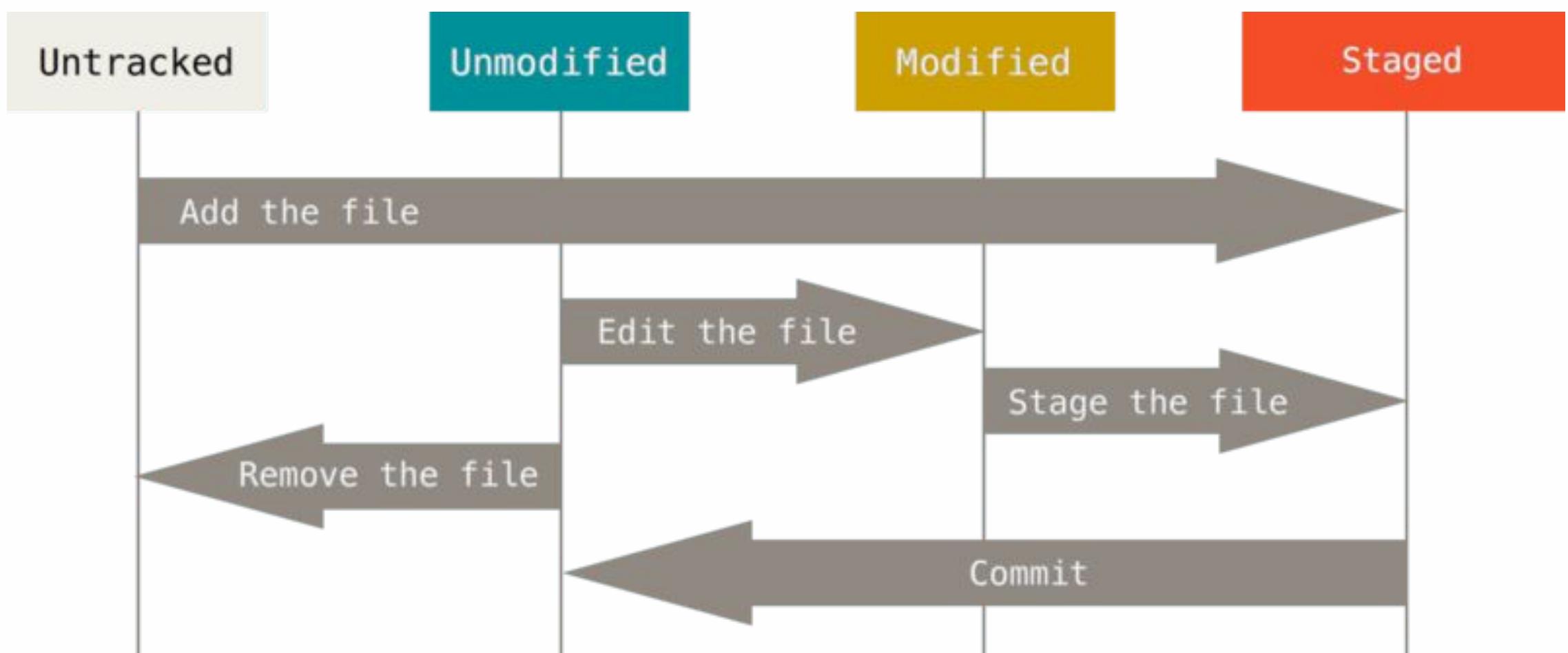


git commit

Фиксируем изменение

git commit -m "my commit"

Состояния файлов



git status

```
@:~/week-4-game <master>$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

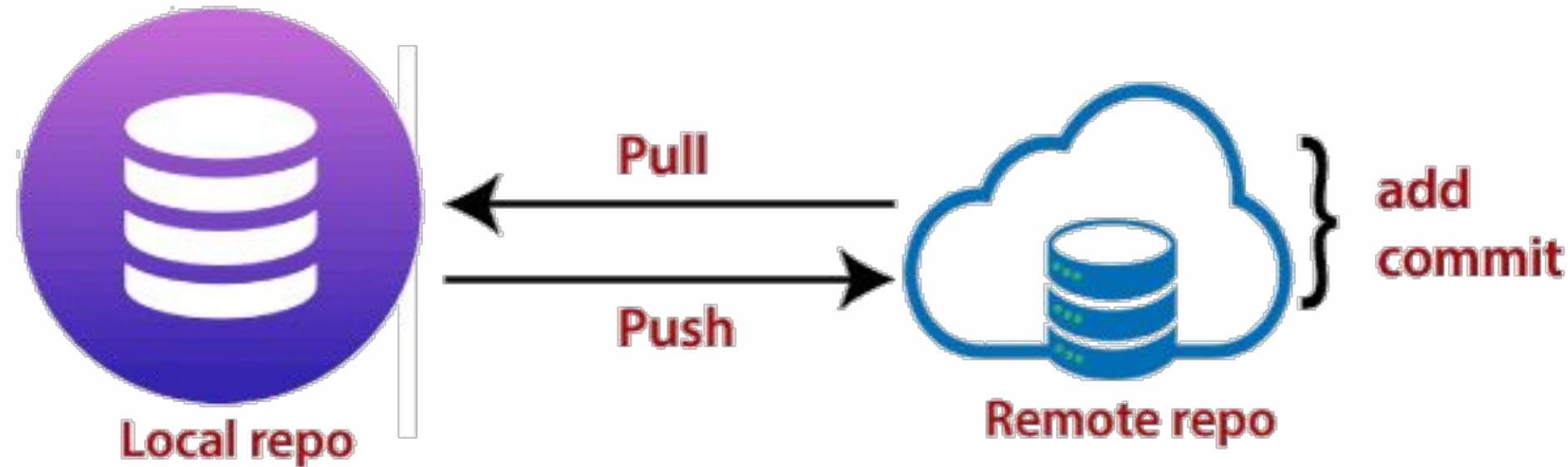
    modified:   assets/css/style.css
    modified:   index.html

Unmerged paths:
  (use "git reset HEAD <file>..." to unstage)
  (use "git add <file>..." to mark resolution)

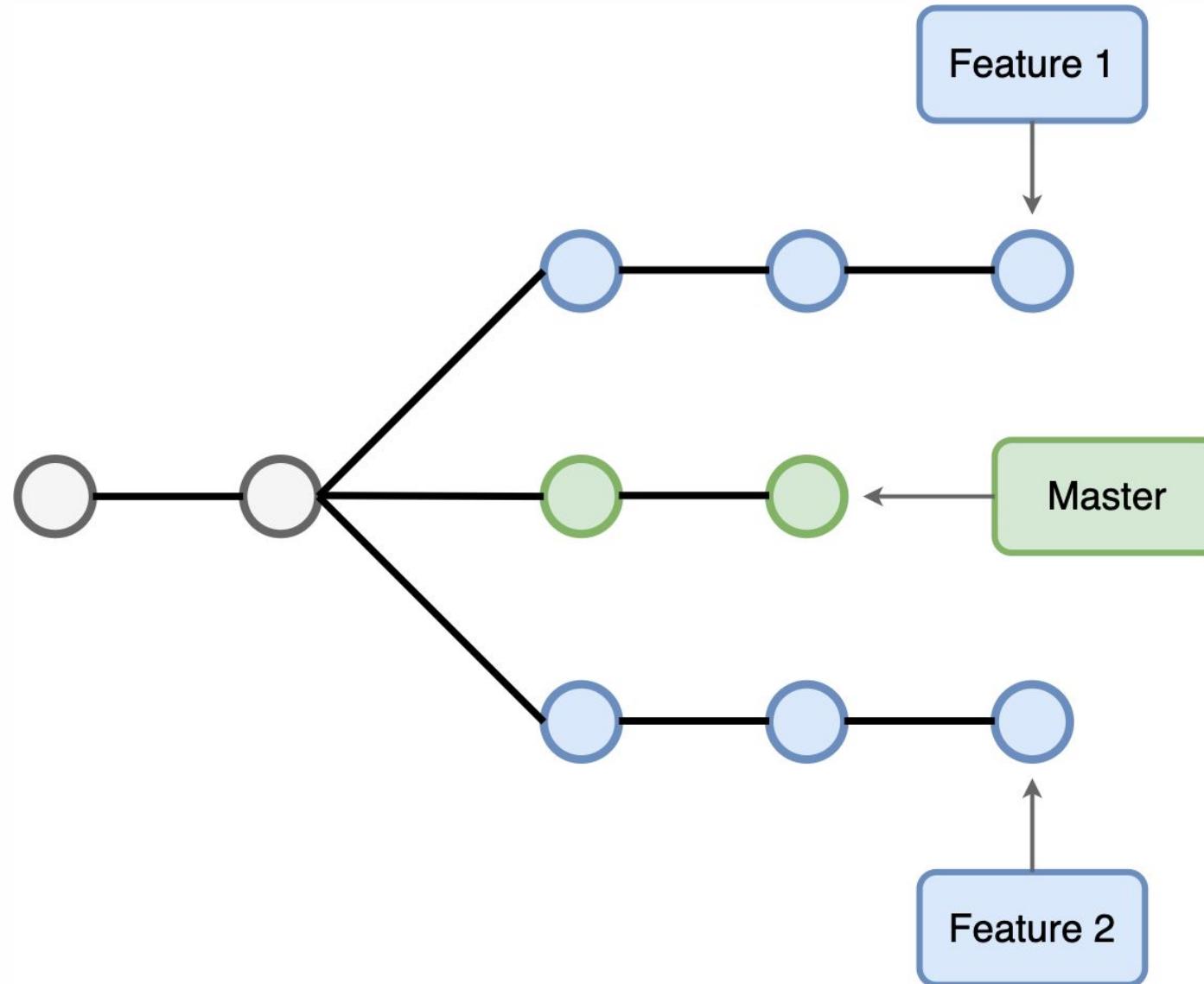
    both modified:  assets/javascript/game.js

@:~/week-4-game <master>$ █
```

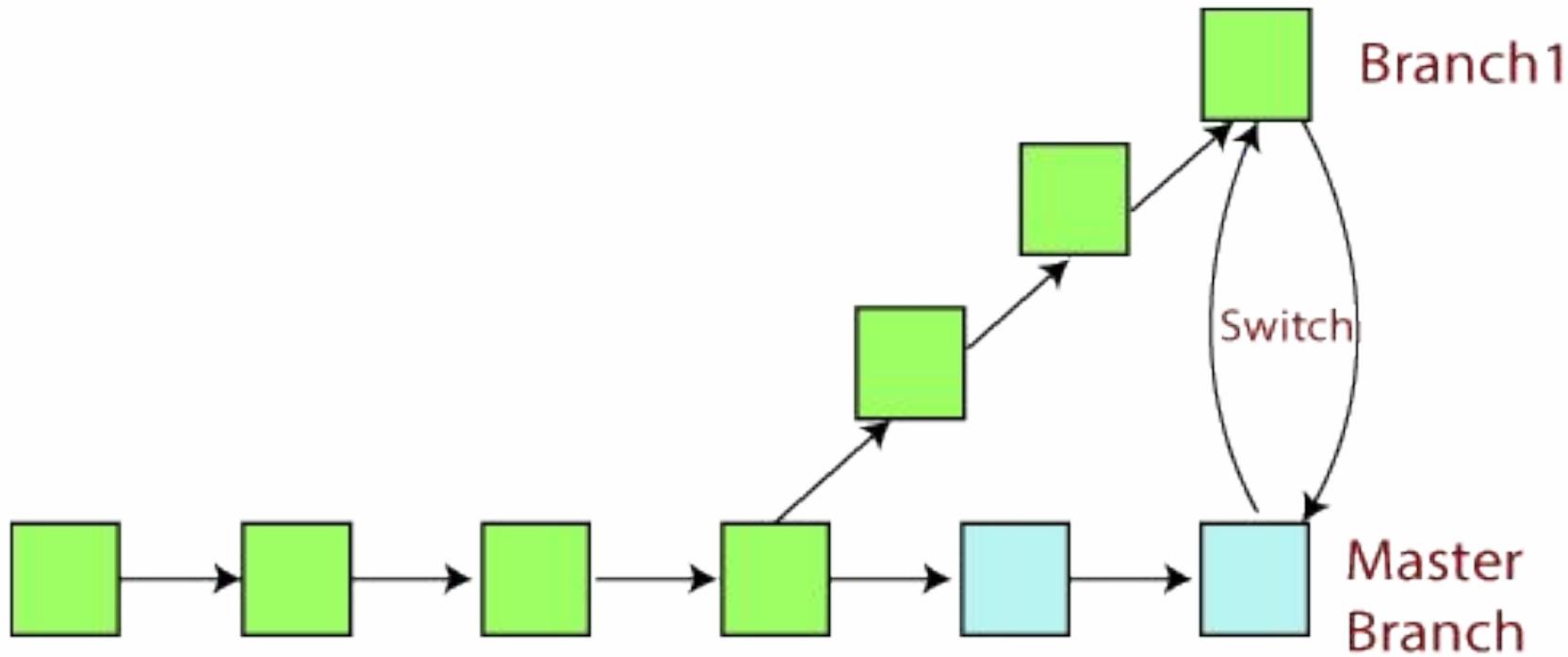
git pull/push



git branch



git checkout



Git Checkout



Играемся с гитом

<https://git-school.github.io/visualizing-git/>



Практика

- 1) Создаем репозиторий
- 2) Создаем ветку
- 3) Создаем файлы, делаем git add/commit/push
- 4) Создаем пулл реквест в мастер
- 5) Добиваемся конфликтной ситуации и разрешаем ее
- 6) git reset/revert



Структура хранения в GIT

- 1) <https://habr.com/ru/company/badoo/blog/163853/>
- 2) <https://git-scm.com/book/ru/v2/Git-%D0%B8%D0%B7%D0%BD%D1%83%D1%82%D1%80%D0%B8-%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D1%8B-Git>



Итог

Гита на таком уровне должно
быть достаточно=)

MLOPS

Михаил Марюфич

