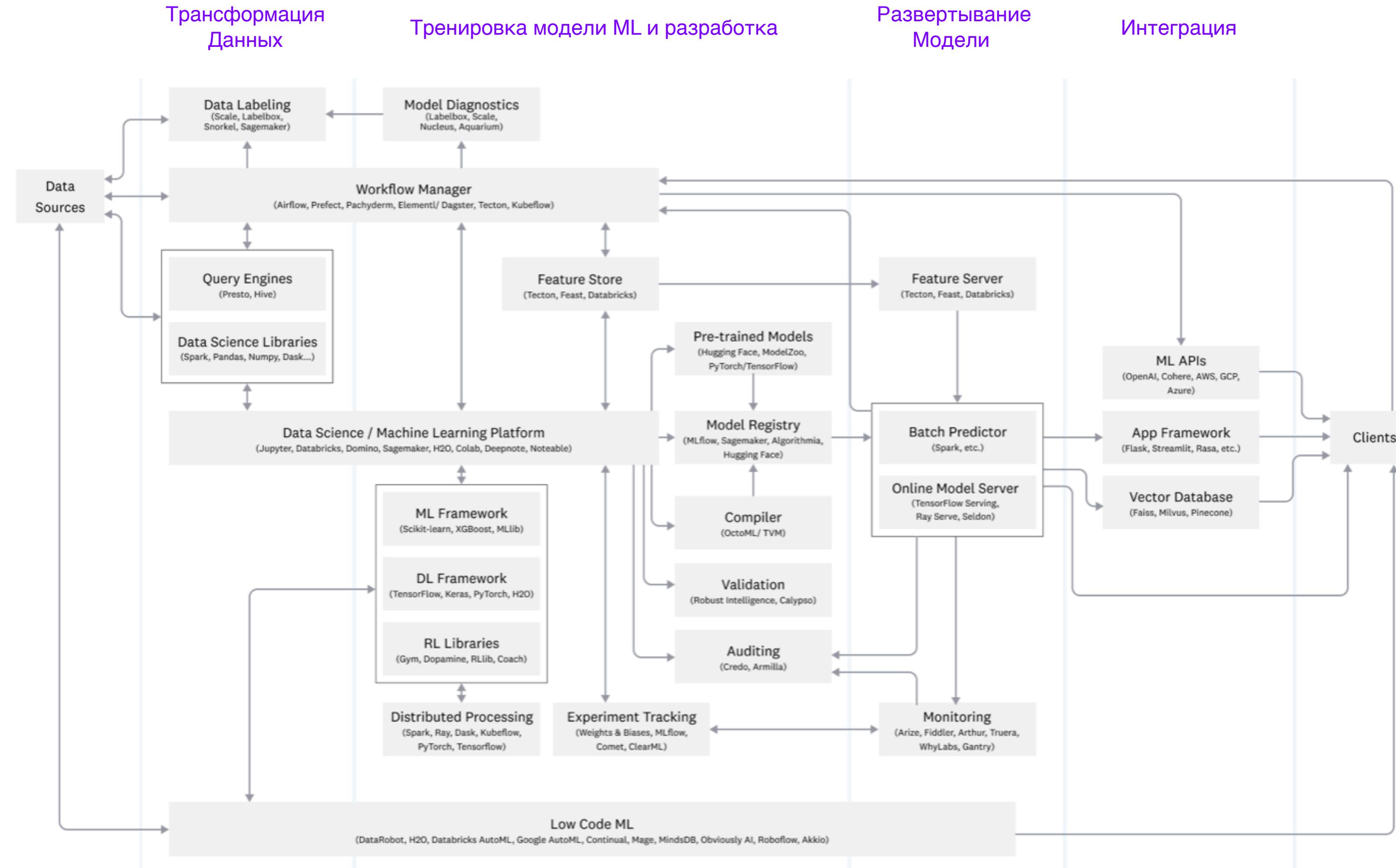


Разбор инструментов для создания уведомлений об отклонении (Алерting)

Азат Якупов
az.yakupov@innopolis.ru



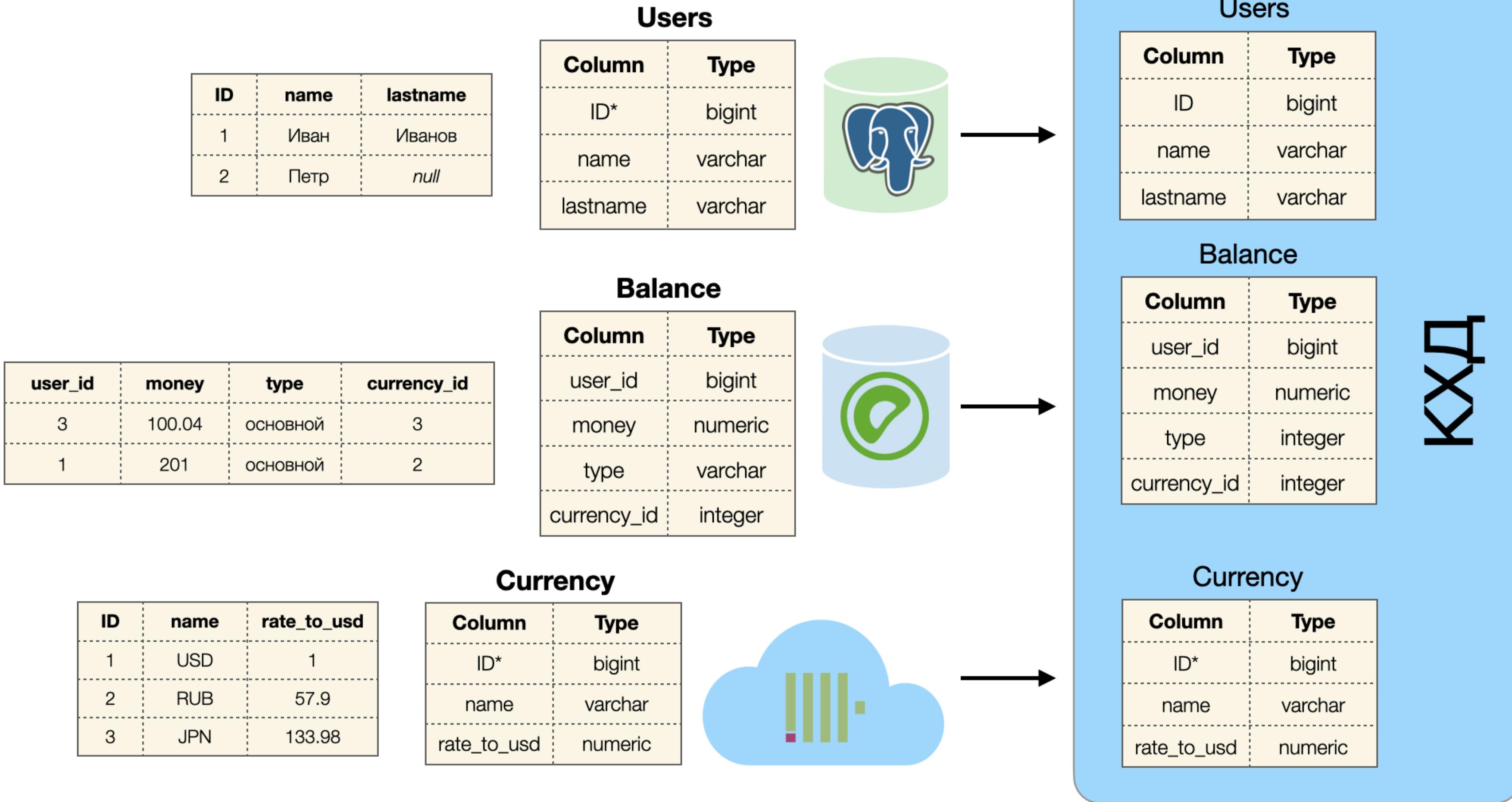


```
(base) azat_yakupov@Azats-MacBook-Pro ➤ ~ ➤ ipython
Python 3.9.13 (main, Aug 25 2022, 18:29:29)
Type 'copyright', 'credits' or 'license' for more information
IPython 7.31.1 -- An enhanced Interactive Python. Type '?' for help.
```

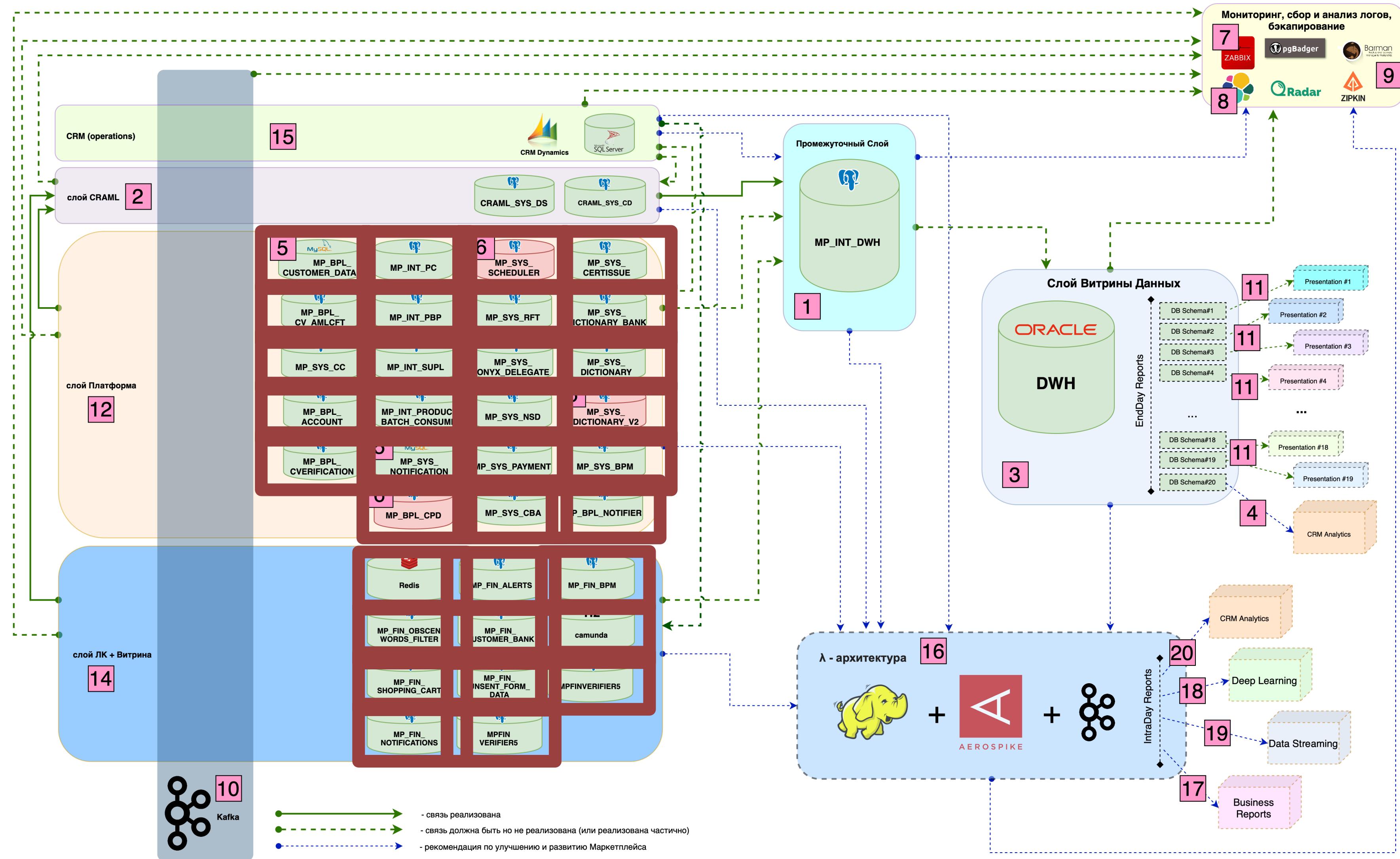
```
In [1]: 0.1 + 0.1 + 0.1
```

```
Out[1]: 0.3000000000000004
```

```
In [2]: █
```



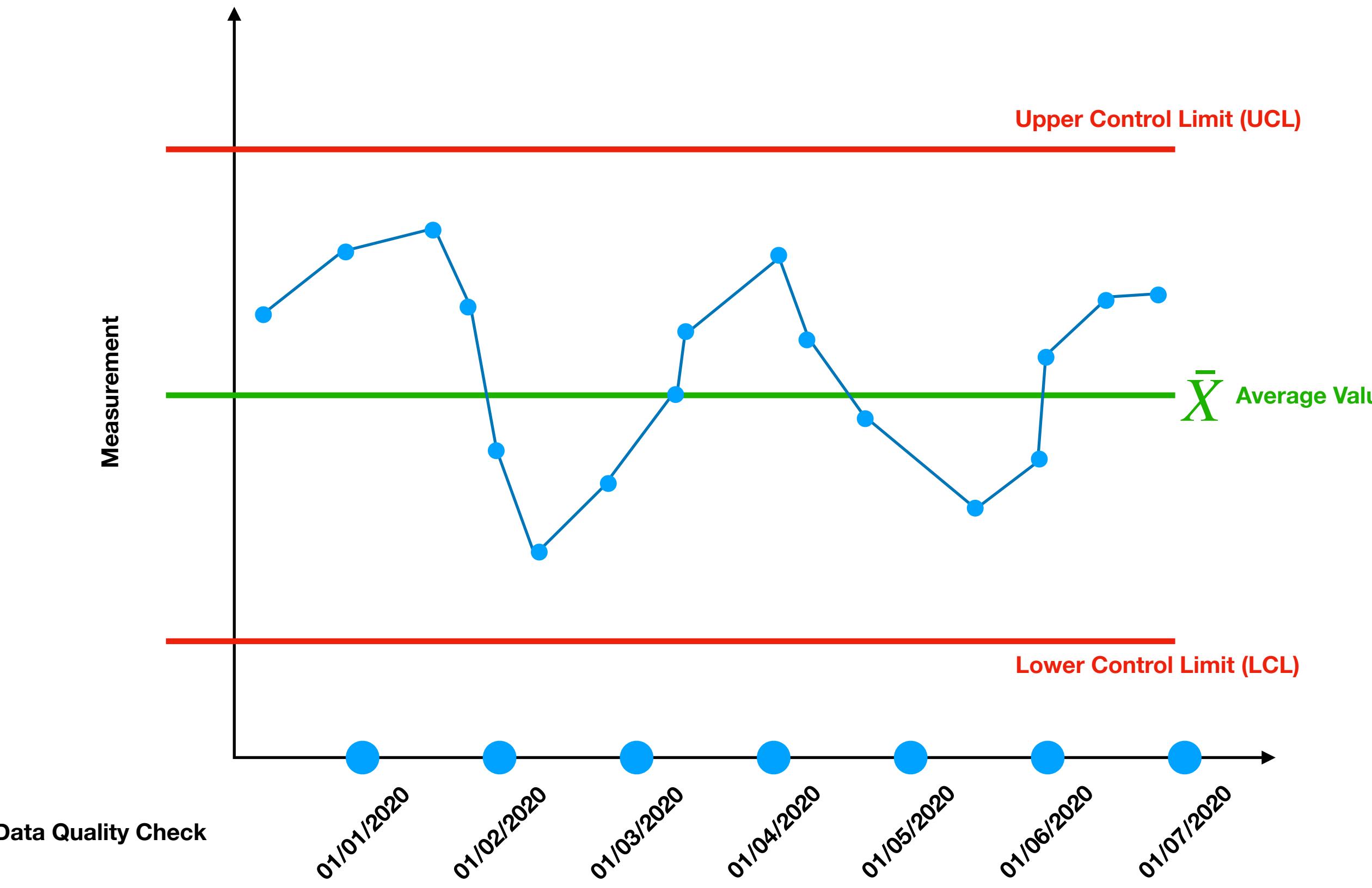
**Почему данные
становятся
неконсистентными?**



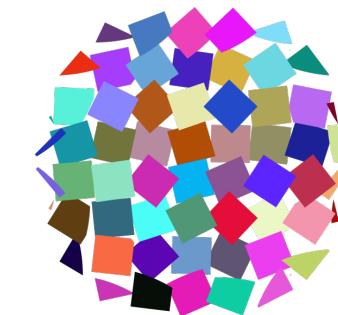
МикроСервисы??!!

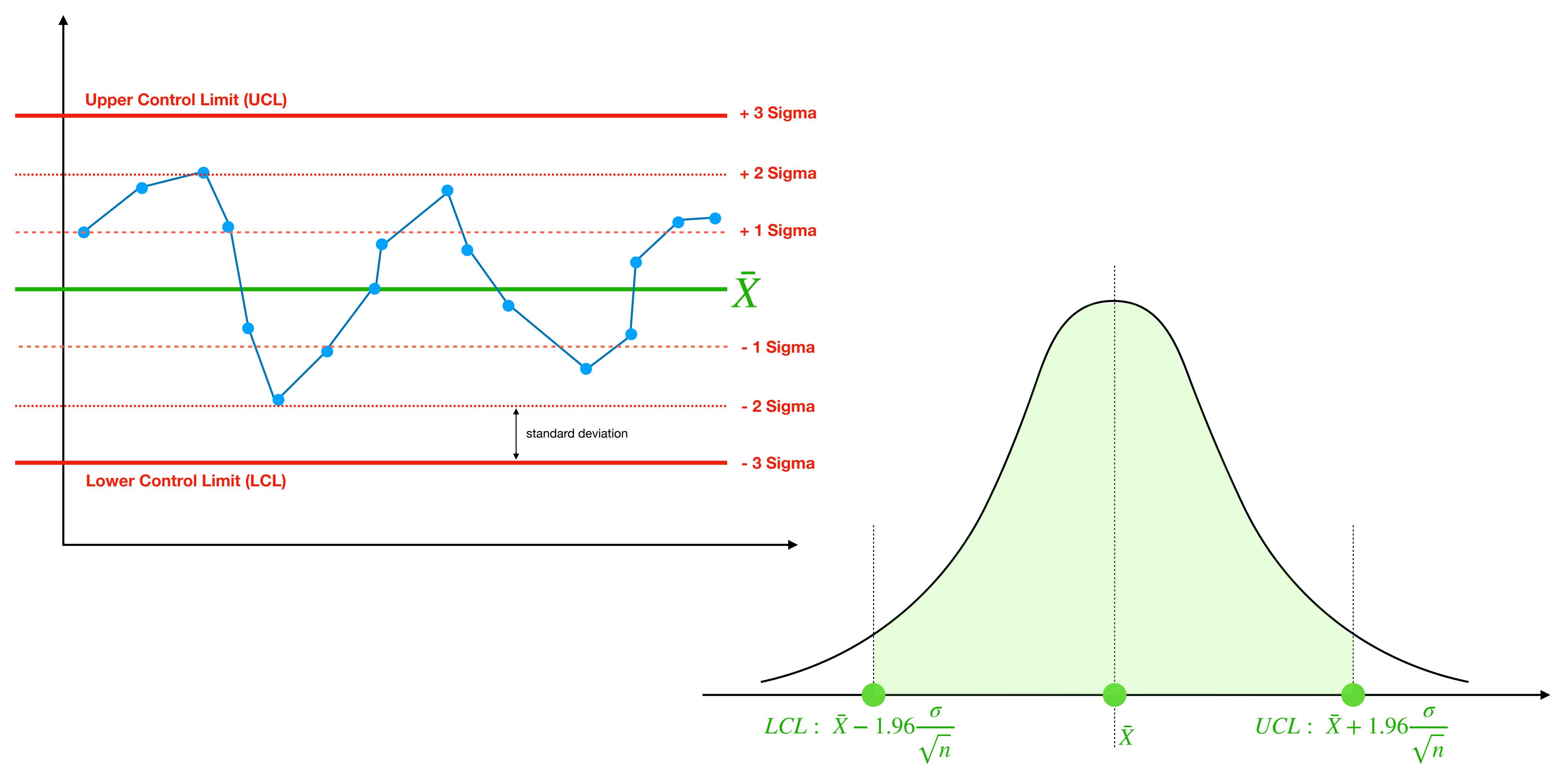
Как делать
данные
консистентными?

```
CREATE TABLE Employee
(
    ID NUMBER,
    SALARY DECIMAL(9,2)
        CONSTRAINT CH_SAL
        CHECK (SALARY>=100000),
    DNAME VARCHAR(10)
        CONSTRAINT CH_DNAME
        CHECK (DNAME IN ('HR', 'IT')),
    BONUS DECIMAL(9,2) DEFAULT 0
);
```



Почему мы измеряем метрику
качества вероятностью





Правило 6-сигма

Техническая задача

Дано: A, B - стандартные (**Heap**) реляционные таблицы РСУБД

$|A|_r = |B|_r = m$, r - строка реляционной таблицы

$|A|_C = |B|_C = n$, C - столбец реляционной таблицы

Показать: $A \neq B, P(A \neq B) = 1 \Rightarrow \exists \forall i, j = \overline{1, m}, A_{r_i} \neq B_{r_j}$

A

	C_1	C_2	C_3	...	C_n
r_1	1	0	1	...	1
r_2	0	0	1	...	0
...	1	1	1	...	1
r_m	0	0	1	...	1

Большая
таблица



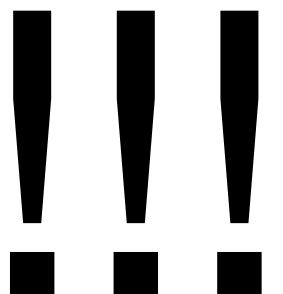
B

	C_1	C_2	C_3	...	C_n
r_1	1	0	1	...	1
r_2	0	0	0	...	0
...	1	1	1	...	1
r_m	0	0	1	...	1

Большая
таблица

Технические метрики качества данных

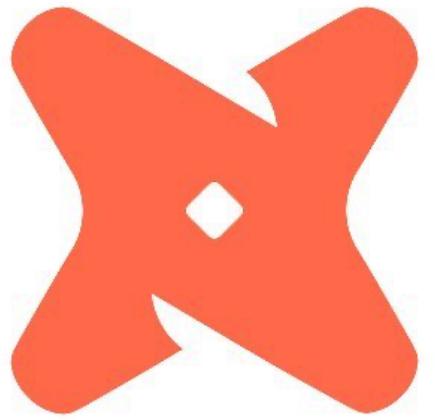
	A	B	C	D	E	F	G
1	Идентификатор	Позиция прибора	Описание	Тип значения	Единица измерения	max	max
2	1	FIRCA-001	Регулятор расхода сырья дебутанизатора в С-1	PV	кг/ч	14750,0	19750,0
3	2	FIRC-003	Регулятор расхода резервного сырья в С-1	PV	кг/ч	14750,0	19750,0
4	3	PZSA-111	Давление шлемовой трубы С-1	PV	КПа	500,0	600,0
5	4	PIRCA-001	Регулятор давления сверху С-1	PV	КПа	500,0	600,0
6	5	PIR-107A	Давление сверху колонны	PV	КПа	500,0	600,0
7	6	PIR-107B	Давление на тарелке питания	PV	КПа	500,0	680,0
8	7	PIR-107C	Давление в кубе колонны	PV	КПа	500,0	680,0
9	8	PDIRA-107C	Перепад между верхом и кубом	PV	КПа	0,0	33,0
10	9	PDIRA-107A	Перепад между верхом и тарелкой питания	PV	КПа	0,0	70,0
11	10	PDIRA-107B	Перепад между тарелкой питания и кубом	PV	КПа	0,0	33,0
12	11	TIRCA-002	Регулятор температуры сверху колонны	PV	град. С	46,0	53,0



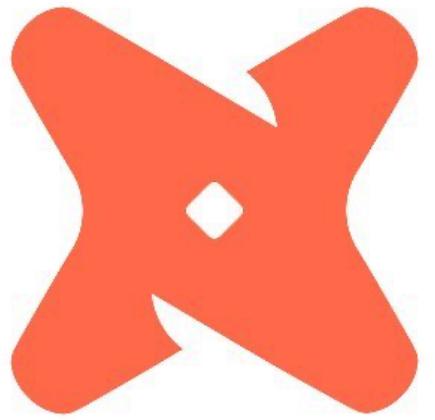
Бизнес метрики качества данных



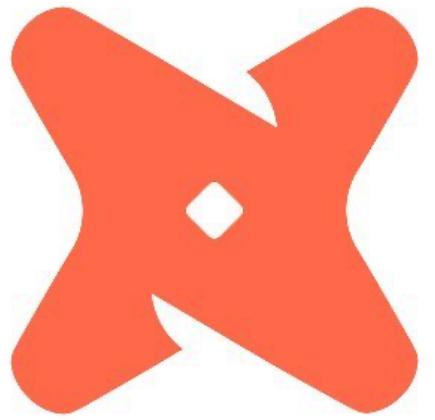
это сервис для разработки, планирования и мониторинга рабочих процессов, в частности сценариев выгрузки, преобразования и загрузки данных (ETL).



это современный инструмент для трансформации данных в хранилищах данных. Вместо того чтобы выполнять ETL (Extract, Transform, Load), с помощью **ДВТ** можно работать по принципу **ELT** (Extract, Load, Transform), когда сначала данные загружаются в хранилище данных, а затем преобразуются уже внутри хранилища.



представляет собой средство **командной строки**, предназначенное для трансформации данных в хранилище данных.



my_project/

|-- dbt_project.yml

|-- analysis/

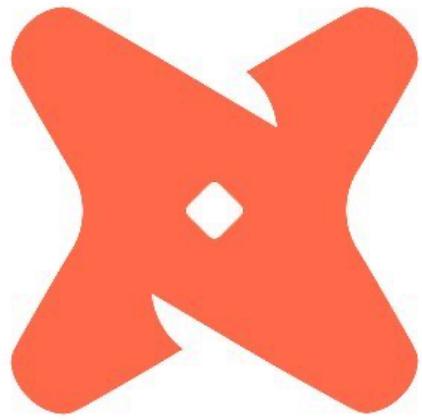
|-- models/

|-- tests/

|-- macros/

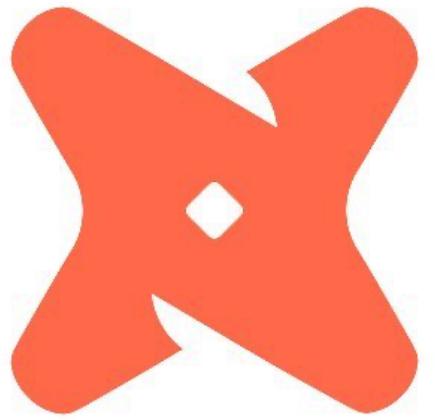
|-- snapshots/

|-- ... (и другие каталоги и файлы)



my_project/models/
|-- users.sql
|-- orders_summary.sql

```
SELECT * FROM raw.users;  
  
WITH orders_aggregated AS (  
    SELECT  
        user_id,  
        COUNT(order_id) AS number_of_orders,  
        SUM(amount) AS total_amount  
    FROM raw.orders  
    GROUP BY user_id  
)  
  
SELECT  
    u.user_id,  
    u.user_name,  
    u.email,  
    o.number_of_orders,  
    o.total_amount  
FROM {{ ref('users') }} u  
LEFT JOIN orders_aggregated o ON u.user_id =  
o.user_id;
```



Чтобы запустить модели
dbt run

Чтобы протестировать модели
dbt test

Автоматически генерировать документацию
dbt docs generate
dbt docs serve



pip3 install dbt-postgres
pip3 install dbt-core

```
from datetime import datetime, timedelta
from airflow import DAG
from airflow.providers.dbt.operators.dbt import DbtRunOperator, DbtTestOperator

default_args = {
    'owner': 'airflow',
    'depends_on_past': False,
    'email_on_failure': False,
    'email_on_retry': False,
    'retries': 1,
    'retry_delay': timedelta(minutes=5)
}

dag = DAG(
    'dbt_dag',
    default_args=default_args,
    description='A simple dbt DAG',
    schedule_interval=timedelta(days=1),
    start_date=datetime(2023, 1, 1),
    catchup=False,
)
```



```
dbt_run = DbtRunOperator(  
    task_id='dbt_run',  
    profiles_dir='/path/to/your/dbt/profiles/',  
    dir='/path/to/your/dbt/project/',  
    dag=dag,  
)  
  
dbt_test = DbtTestOperator(  
    task_id='dbt_test',  
    profiles_dir='/path/to/your/dbt/profiles/',  
    dir='/path/to/your/dbt/project/',  
    dag=dag,  
)  
  
dbt_run >> dbt_test
```

демонстрация

Спасибо
за внимание



Азат Якупов
az.yakupov@innopolis.ru