



# В гуще событий

Как системному аналитику проектировать метрики

---

# Содержание

**1**

## Метрики

Назначение и виды

**2**

## Архитектура

Агрегация и отправка

**3**

## Лайфхаки

С живыми примерами

**4**

## Ошибки

Запоминаем, не  
повторяем

# Обо мне

- Лид системного анализа в SaluteDevices
- Веду канал и авторский курс [@Analyst\\_Boost](#)
- Проповедаю [@Plant\\_UML](#)
- Сделал [pUML tg-бота](#), веду мастер-классы
- Веду консультации как ментор
- Иногда побеждаю в [хакатонах](#)
- Люблю конференции :)



# 1

## Метрики

Назначение и виды

---

“Измеряй измеримое и делай  
**неизмеримое измеримым**”

— Галилео Галилей

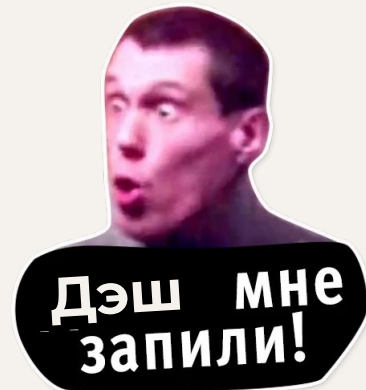
## Какие типы анализа данных использует ваша компания для измерения клиентского опыта (CX) в цифровых продуктах?



Источник: [Harvard Business Review Analytic Services](#), Sponsored by Amplitude, 17.02.2021

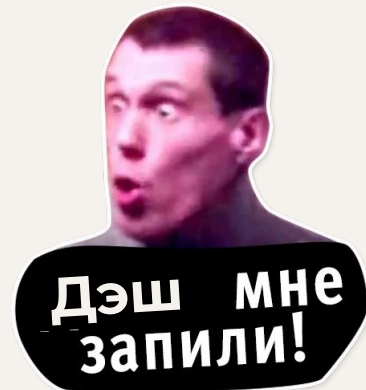
# Кому и зачем нужны метрики

- Продакты и инвесторы: гипотезы и приоритеты



# Кому и зачем нужны метрики

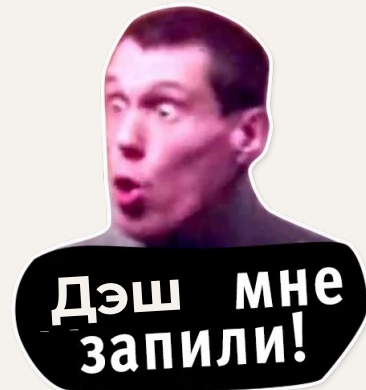
- Продакты и инвесторы: гипотезы и приоритеты
- Дизайнеры и разработчики: удобство и производительность





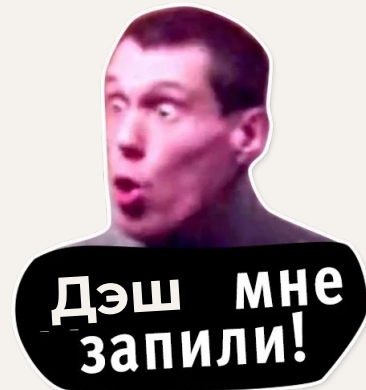
# Кому и зачем нужны метрики

- Продакты и инвесторы: гипотезы и приоритеты
- Дизайнеры и разработчики: удобство и производительность
- Сопровождение и безопасность: аномалии и предсказания



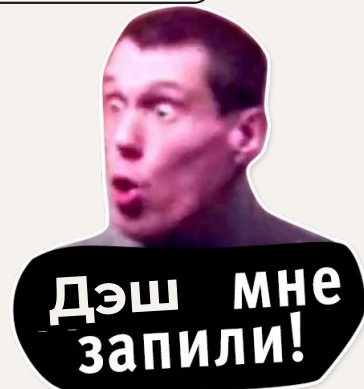
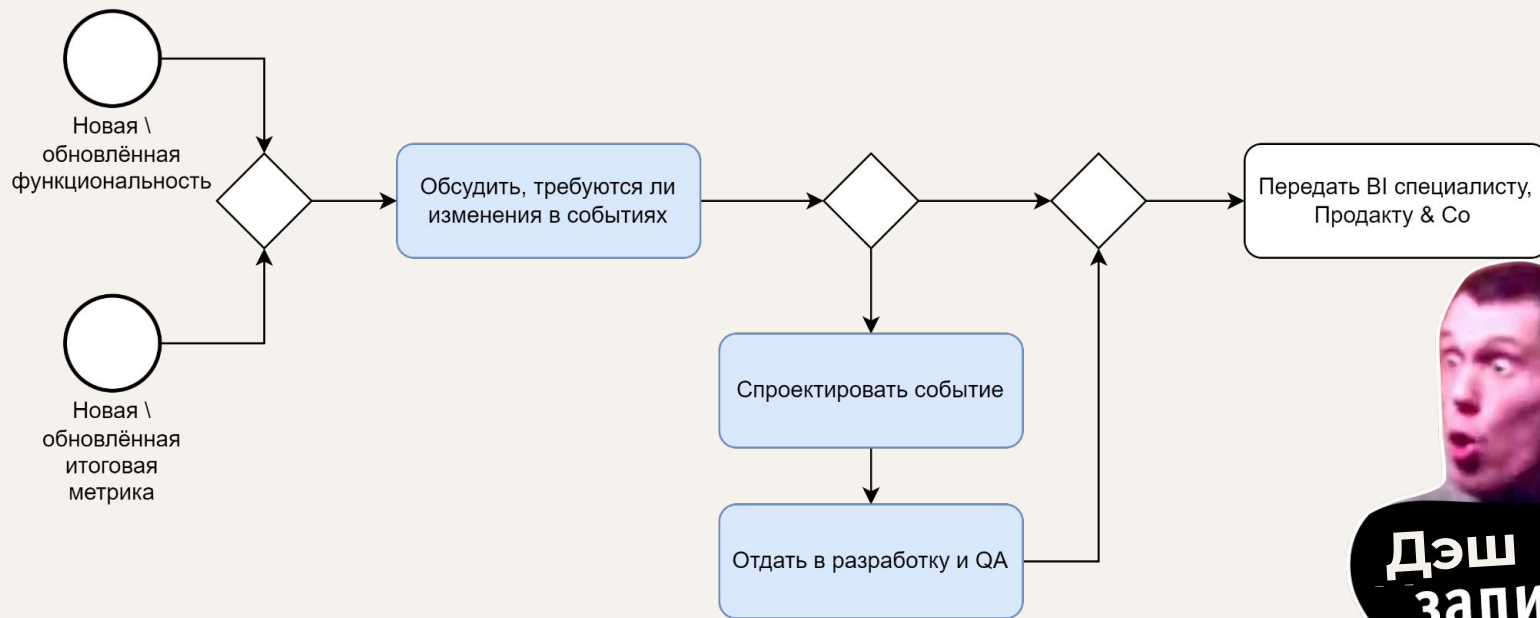
# Кому и зачем нужны метрики

- **Продакты и инвесторы: гипотезы и приоритеты**
- Дизайнеры и разработчики: удобство и производительность
- Сопровождение и безопасность: аномалии и предсказания
- И другие, кому интересна судьба продукта



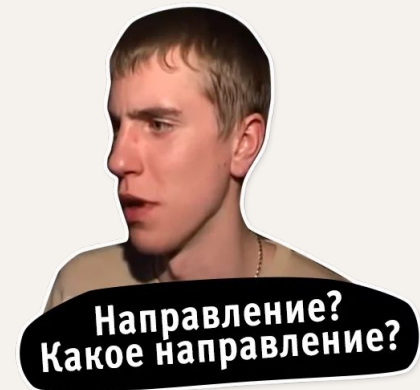
# Проектирование метрик

Система сбора событий и построения метрик существует



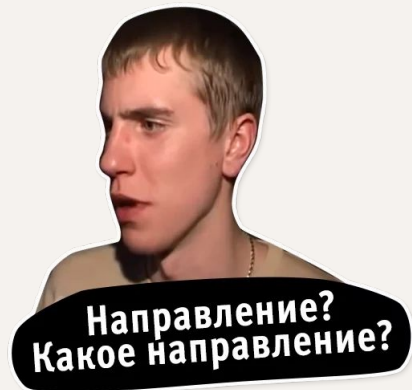
# Основные направления

- Количественно измеримые показатели, собираемые из атомарных событий



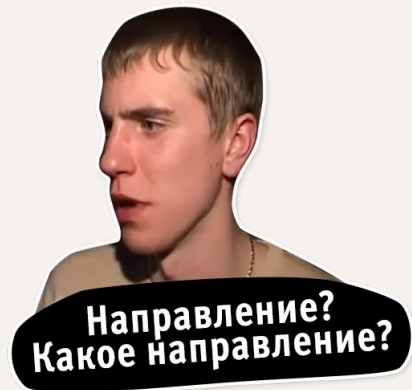
# Основные направления

- Количественно измеримые показатели, собираемые из атомарных событий
- Популярные виды метрик:
- Привлечения (Acquisition): Cost Per Install, Average Revenue Per User ...



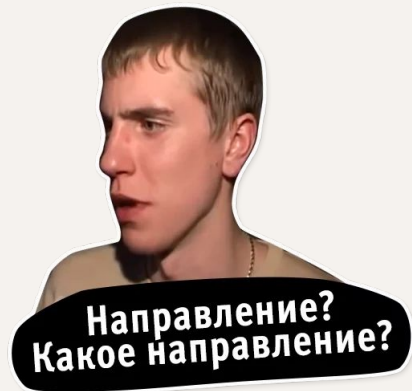
# Основные направления

- Количественно измеримые показатели, собираемые из атомарных событий
- Популярные виды метрик:
- Привлечения (Acquisition): Cost Per Install, Average Revenue Per User ...
- Производительности (Performance): App Load Time, App Crashes ...

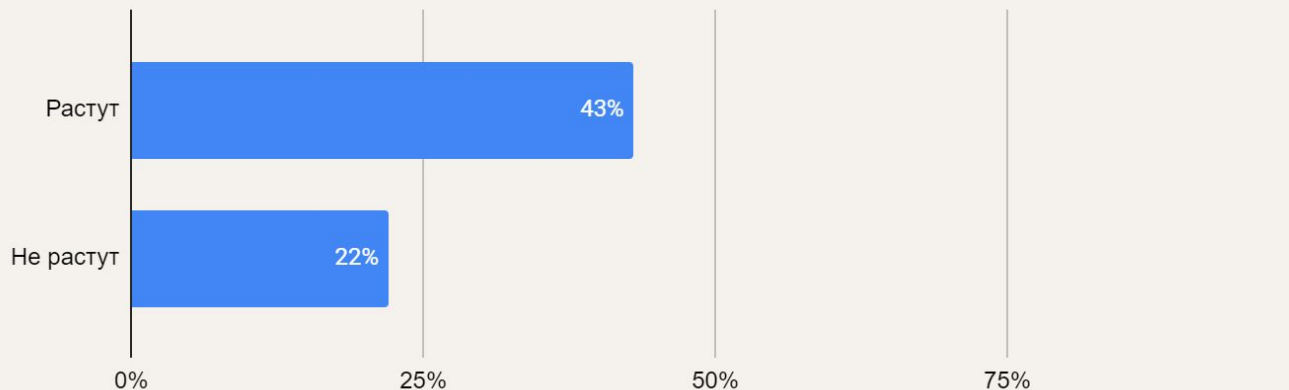


# Основные направления

- Количественно измеримые показатели, собираемые из атомарных событий
- Популярные виды метрик:
- Привлечения (Acquisition): Cost Per Install, Average Revenue Per User ...
- Производительности (Performance): App Load Time, App Crashes ...
- Вовлеченности (Engagement): DAU/WAU/MAU, Retention Rate ...



Сколько продуктовых менеджеров использует аналитику для сбора и анализа данных о СХ, в компаниях, которые:



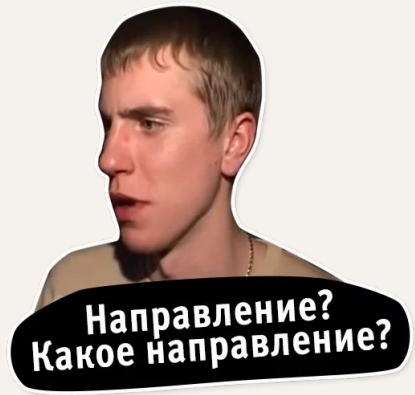
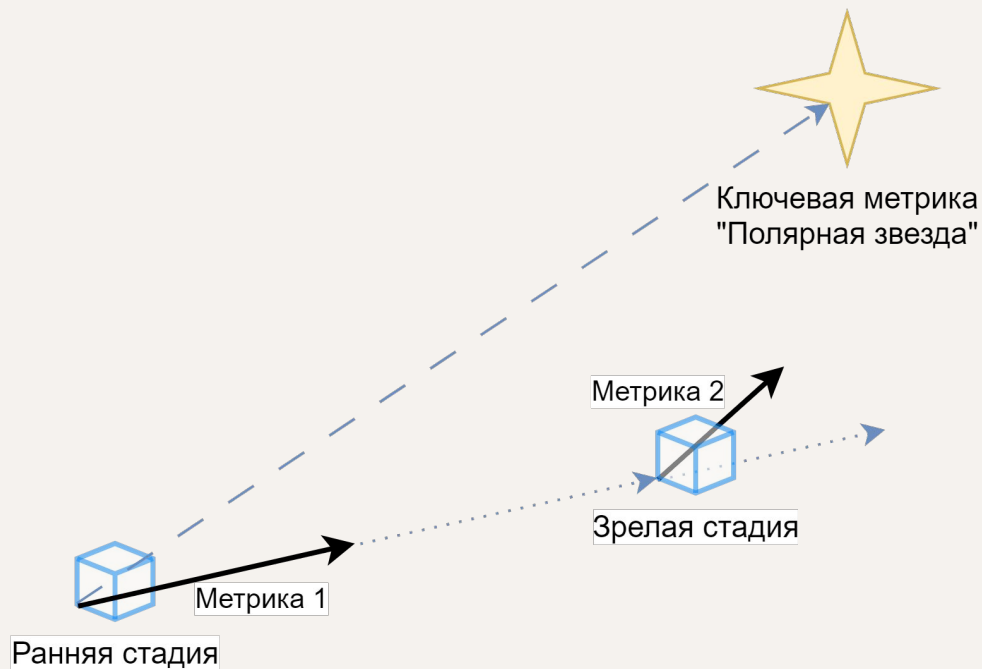
*\*Организация определяется как*

- *Растущая, если она имела положительный рост дохода с 2018 по 2019 год и, как ожидается, будет иметь положительный рост дохода с 2019 по 2020 год*
- *Нерастущая, если она имела неизменный или снижающийся доход с 2018 по 2019 год, с аналогичным прогнозируемым доходом за 2019-2020 года*

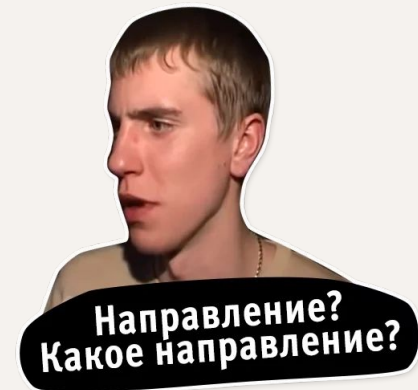
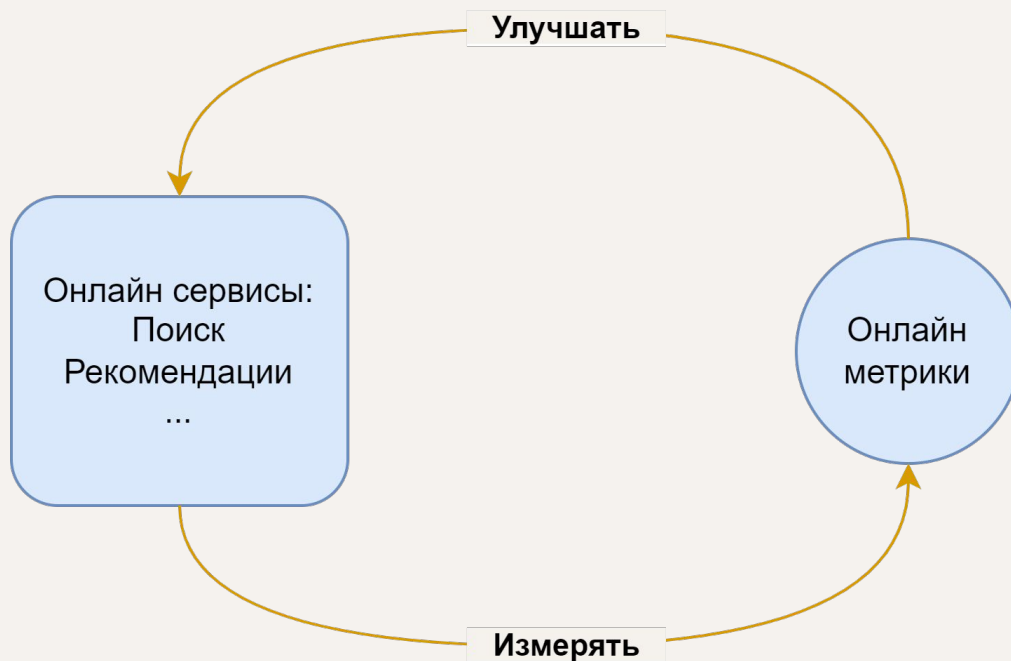
Источник: [Gartner, STAMFORD, Conn. 31.03.2020](#)



# Вектор и прирост



# Вектор и прирост

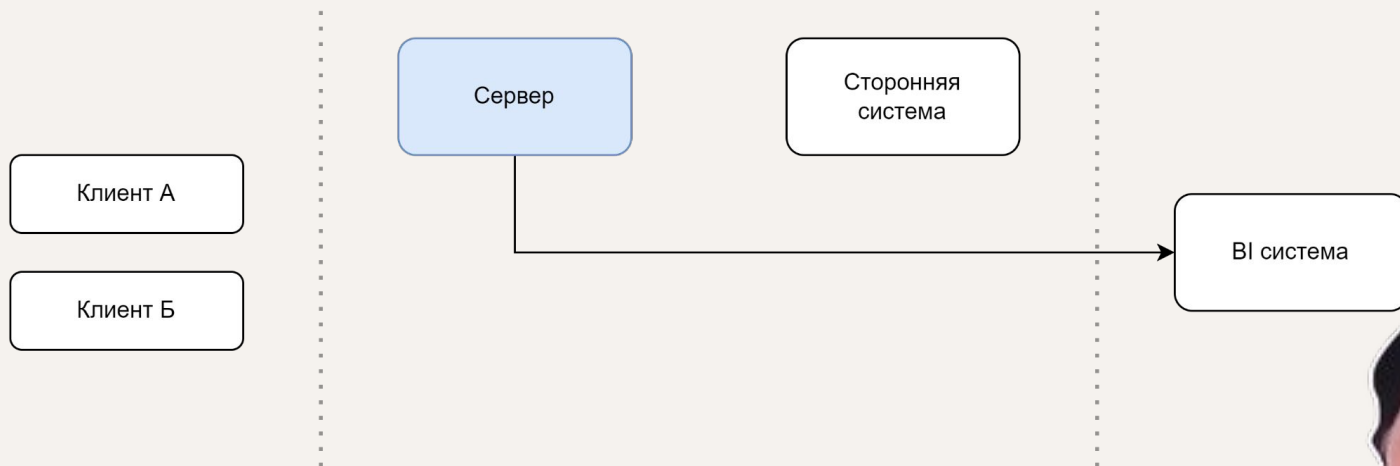


# 2

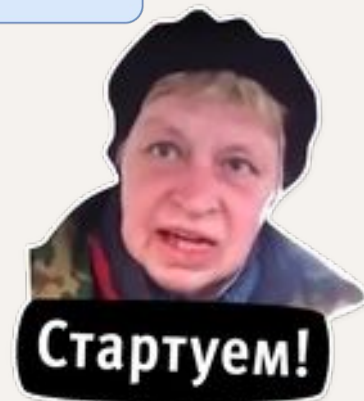
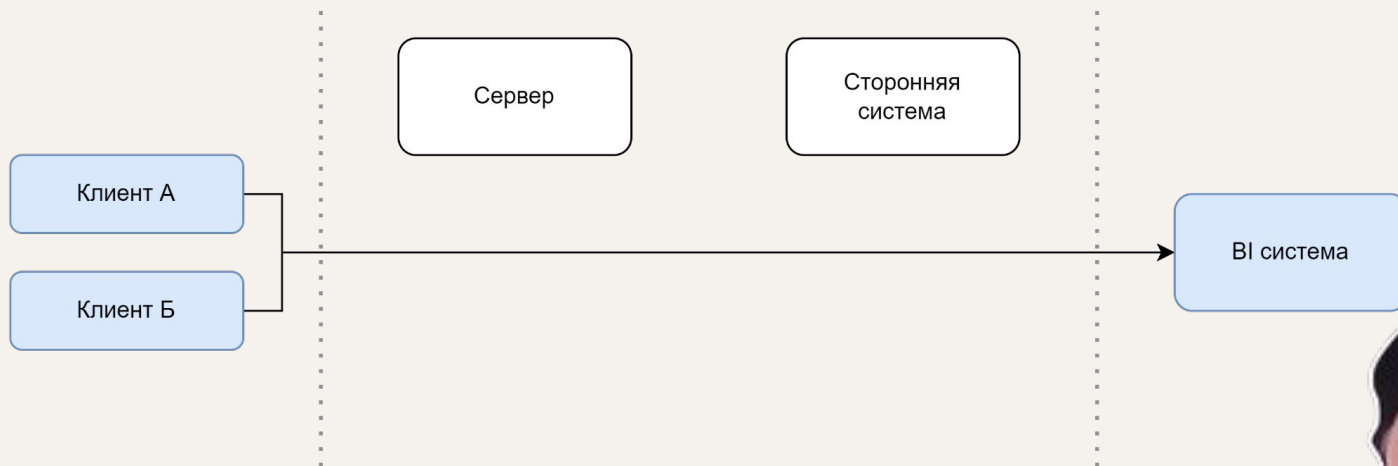
## Архитектура

Агрегация и отправка

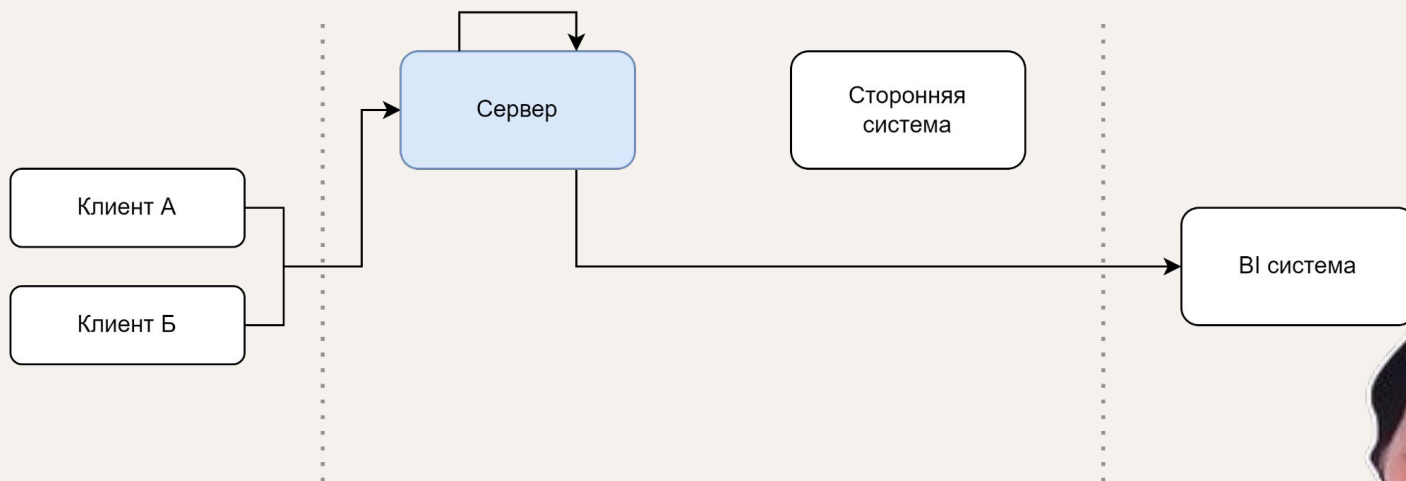
# Варианты реализации



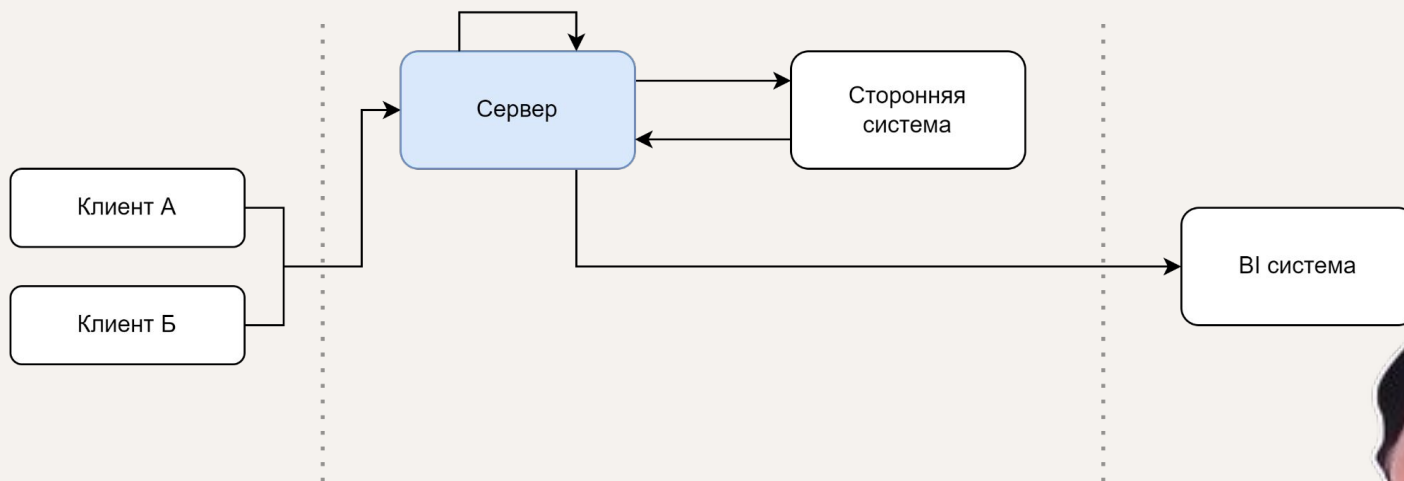
# Варианты реализации



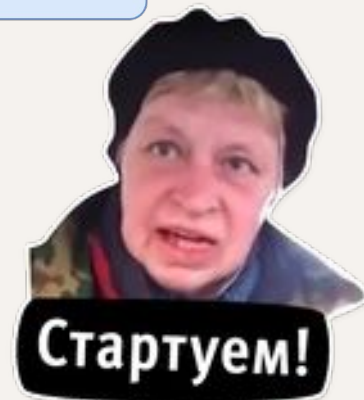
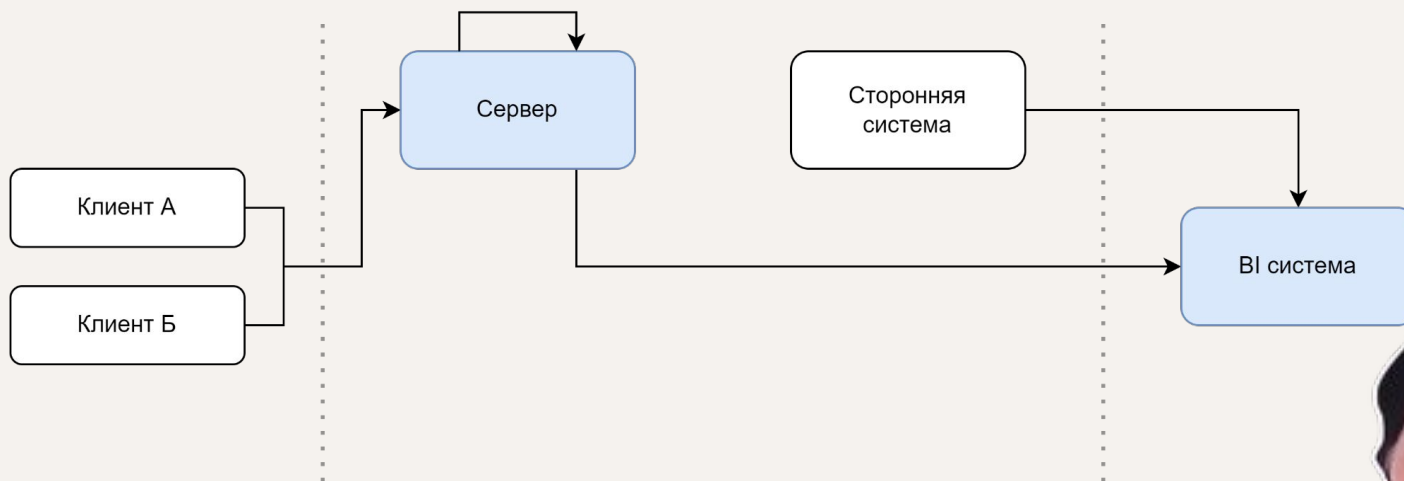
# Варианты реализации



# Варианты реализации

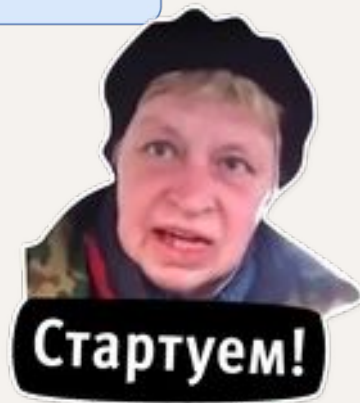
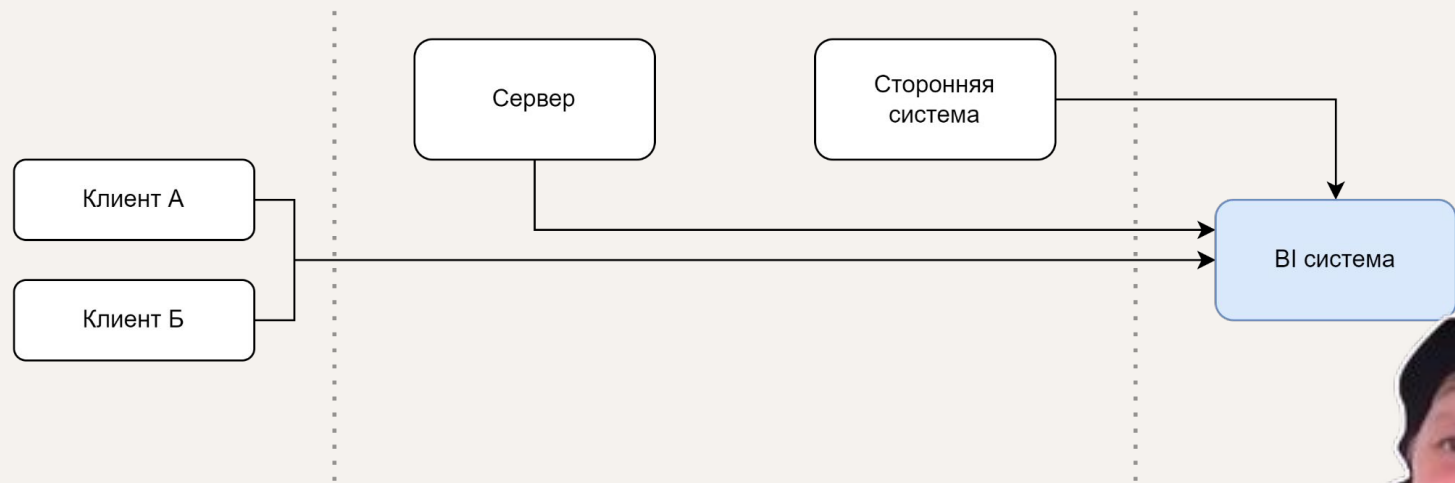


# Варианты реализации





# Варианты реализации



# 3

## Лайфхаки

С живыми примерами

# Заголовки

- Trace ID (для отслеживания на уровне инфраструктуры)



# Заголовки

- Trace ID (для отслеживания на уровне инфраструктуры)
- Session id (Например, unixtimestamp первого события сессии)



# Заголовки

- Trace ID (для отслеживания на уровне инфраструктуры)
- Session id (Например, unixtimestamp первого события сессии)
- Авторизация, аутентификация
- Версия



# Иерархия тела. Контракт

- Client properties (device id, ip + geo, provider, OS + version, brand ...)



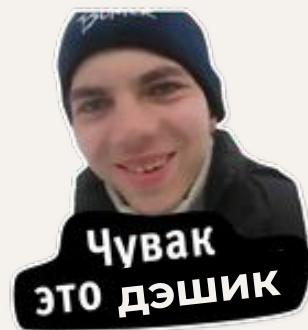
# Иерархия тела. Контракт

- Client properties (device id, ip + geo, provider, OS + version, brand ...)
- User properties (user id, role, organization, ...)



# Иерархия тела. Контракт

- Client properties (device id, ip + geo, provider, OS + version, brand ...)
- User properties (user id, role, organization, ...)
- Event (event id, name, date, category)





# Иерархия тела. Контракт

- Client properties (device id, ip + geo, provider, OS + version, brand ...)
- User properties (user id, role, organization, ...)
- Event (event id, name, date, category)
- Event properties (custom object)



# Описание параметра

- Название параметра (e.g. provider\_name)
- Тип данных (e.g. string)



# Описание параметра

- Название параметра (e.g. provider\_name)
- Тип данных (e.g. string)
- Обязательность (e.g. 0) и условие заполнения (e.g. при подключении по моб. сети)
- Описание (e.g. название оператора интернет-услуг)



# Описание параметра

- Название параметра (e.g. provider\_name)
- Тип данных (e.g. string)
- Обязательность (e.g. 0) и условие заполнения (e.g. при подключении по моб. сети)
- Описание (e.g. название оператора интернет-услуг)
- Пример (e.g. T-mobile)
- Бизнес-вопрос (e.g. кто провайдер наших клиентов?)



# Описание события

- Полное описание параметров
- Триггер и момент замера (e.g. пользователь вызвал открытие контента, на клиенте)



# Описание события

- Полное описание параметров
- Триггер и момент замера (e.g. пользователь вызвал открытие контента, на клиенте)
- Бизнес-вопрос (e.g. Какой тип контента потребляют пользователи больше всего?)
- Или маппинг на показатель отчётности / дэшборд (многие-ко-многим)



# Описание события

- Полное описание параметров
- Триггер и момент замера (e.g. пользователь вызвал открытие контента, на клиенте)
- Бизнес-вопрос (e.g. Какой тип контента потребляют пользователи больше всего?)
- Или маппинг на показатель отчётности / дэшборд (многие-ко-многим)
- Путь к экрану (e.g. главный экран -> настройки)
- Или в формате Source screen + Current Screen



# Мета информация события

- Статус разработки события (e.g. PROD)
- Версия события (e.g. 1)
- Ответственный (e.g. Команда А)





# Статус разработки события

- DRAFT – Event в процессе проектирования
- PLANNED – Event спроектирован и запланирован
- DEV – Event в разработке (доступен на dev-стендах), но ещё не в PROD
- DONE – Event в PROD'e
- CANCELLED – Event упразднён



# Версионирование события

№	IOS	Android	Web	Desktop	Комментарий
2	Заплани- ровано  <i>(можно связать с task tracker)</i>	Заплани- ровано	Заплани- ровано	Заплани- ровано	Атрибут <b>добавлен</b> \ <b>изменён</b> \ <b>удалён</b> <ul style="list-style-type: none"><li>• <b>название атрибута соответствующим цветом</b><ul style="list-style-type: none"><li>○ пояснения тем же цветом</li></ul></li></ul>
1	Номер_ версии	Номер_ версии	Номер_ версии	Номер_ версии	Исходная версия



# Schema & Example

===== Допустим, JSON =====

```
{  
  "param_name": {  
    "type": "string"  
  },  
  "is_param": {  
    "type": "boolean"  
  },  
  "param_size": {  
    "type": "number"  
  }  
}
```

===== Конкретный пример =====

```
{  
  "param_name": "hello",  
  "is_param": true,  
  "param_size": 15  
}
```

# Возможный нейминг

- Любой, главное, единообразии и осведомлённости. Например:
- `lower_snake_case` в формате <глагол\_сущность\_обстоятельства>



# Возможный нейминг

- Любой, главное, единообразии и осведомлённость. Например:
- `lower_snake_case` в формате <глагол\_сущность\_обстоятельства>
- <Сделать что-то>\_<над чем совершаем действие>
- Примеры: `add_movie_to_list` | `tap_create_button` | `delete_post`



# Ведение в Confluence

- Или в любой другой системе. 1 Event – 1 страничка wiki-системы
- Если по одной теме 2 Event'а – то это будет две странички
- Название странички: [Product\_name] your\_event\_name



# Ведение в Confluence

- Или в любой другой системе. 1 Event – 1 страничка wiki-системы
- Если по одной теме 2 Event'a – то это будет две странички
- Название странички: [Product\_name] your\_event\_name
- Создать шаблон события
- Создать единую для страниц преамбулу, включить её в шаблон



# 4

## Ошибки

Запоминаем, не повторяем

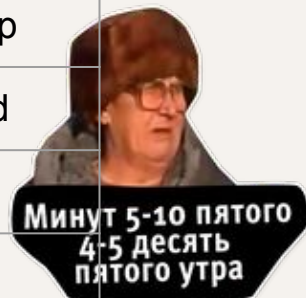


# Найди ошибку

**Сценарий:** пользователь добавил четырёх разных участников в команду, с разных клиентов, двоих добавил в роли администратора.

На дэшборде продакт видит только одного участника, и тот не админ.  
Ниже отфильтрованы поля, по которым строится дэш.

user_id	event_id	event_name	event_properties	platform
AB-01	1234	add_user	{ "member_id": "11", "is_admin": "true" }	Desktop
AB-01	5678	add_user	{ "member_id": 22, "is_admin": "false" }	Android
AB-01	9101	add_user	{ "member_id": 33, "is_admin": true }	iOS
AB-01	1121	add_user	{ "member_id": "44", "is_admin": false }	Web



# Качество данных

- Накапливать события и отправлять пачкой, подумать о кэше
- Определить максимальный размер события или пачки



# Качество данных

- Накапливать события и отправлять пачкой, подумать о кэше
- Определить максимальный размер события или пачки
- Не передавать файлы целиком, только id / ссылки
- Подумать о переотправке и отсутствии сети



# Качество данных

- Накапливать события и отправлять пачкой, подумать о кэше
- Определить максимальный размер события или пачки
- Не передавать файлы целиком, только id / ссылки
- Подумать о переотправке и отсутствии сети
- Помнить про персональные и чувствительные данные
- Помнить про шифрование и хэширование



# Сценарии для тестирования

- Опишите ключевые сценарии вместе с РО & Со
- Оставшиеся сценарии для полного перебора – вместе с QA



# Сценарии для тестирования

- Опишите ключевые сценарии вместе с PO & Co
- Оставшиеся сценарии для полного перебора – вместе с QA
- Подумайте над матрицей всех комбинаций для событий
- На каждый релиз – свой артефакт сценариев



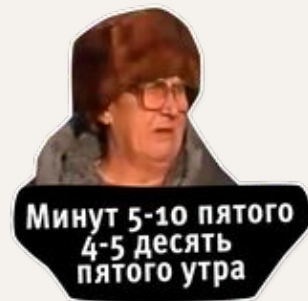
# Бонус: расчёт нагрузки

- Выписать все Event'ы
- Выбрать эталон: или самый частотный, или ключевой, или наглядный
- Для каждого Event'а прикинуть долю относительно эталона



# Выводы

- Помните про метрики – это отдельная задача при работе над функциональностью





# Выводы

- Помните про метрики – это отдельная задача при работе над функциональностью
- Вдумчиво выберите подходящую архитектуру



# Выводы

- Помните про метрики – это отдельная задача при работе над функциональностью
- Вдумчиво выберите подходящую архитектуру
- Учитывайте возможные ошибки при проектировании событий



# Выводы

- Помните про метрики – это отдельная задача при работе над функциональностью
- Вдумчиво выберите подходящую архитектуру
- Учитывайте возможные ошибки при проектировании событий
- Schema однажды вас спасёт



# Контакты

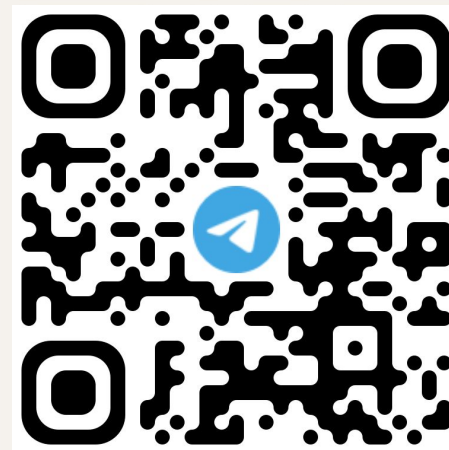
Группа **@plant\_uml**



@Plant\_UML\_Analyst\_Boost\_bot

Mail: nekharichkin@yandex.ru

@Analyst\_Boost



# Спасибо!

# Вопросы

**CREDITS:** This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik**