



# Staging и Differential testing для регрессионного тестирования

Andrey Kuleshov

Deutsche Bank Technology Centre. Moscow, Russia



# Проблематика регрессионного тестирования



- Важная часть тестирования
- Что требуется?



Время



Ресурсы



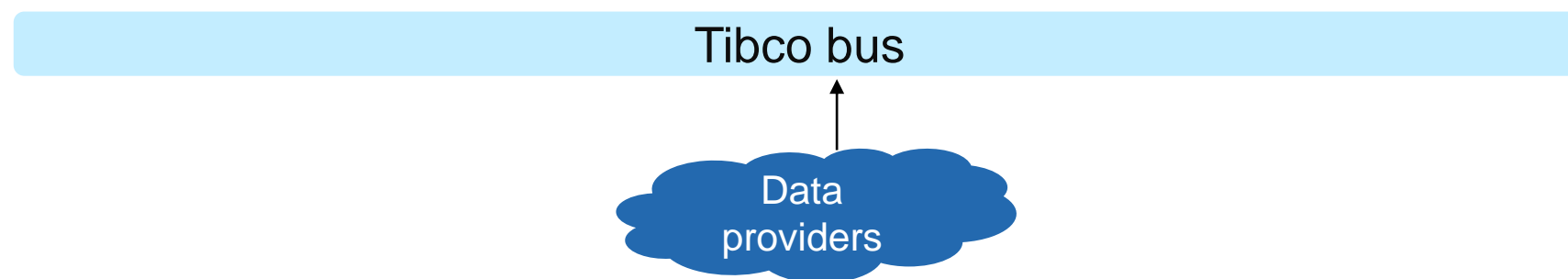
Объем функционала



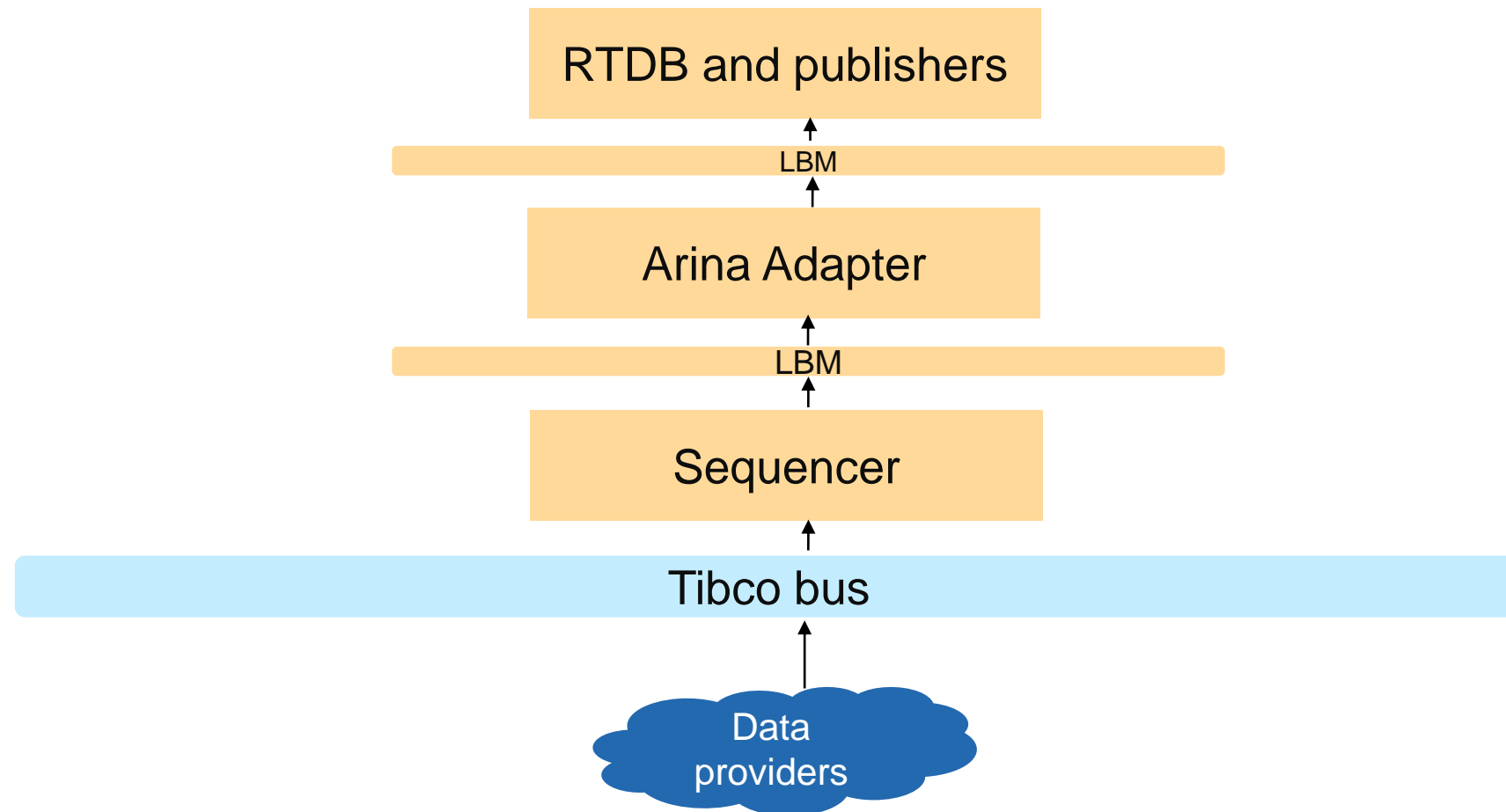
Объем знаний тестировщиков

- Поиск регрессий влияет на скорость разработки

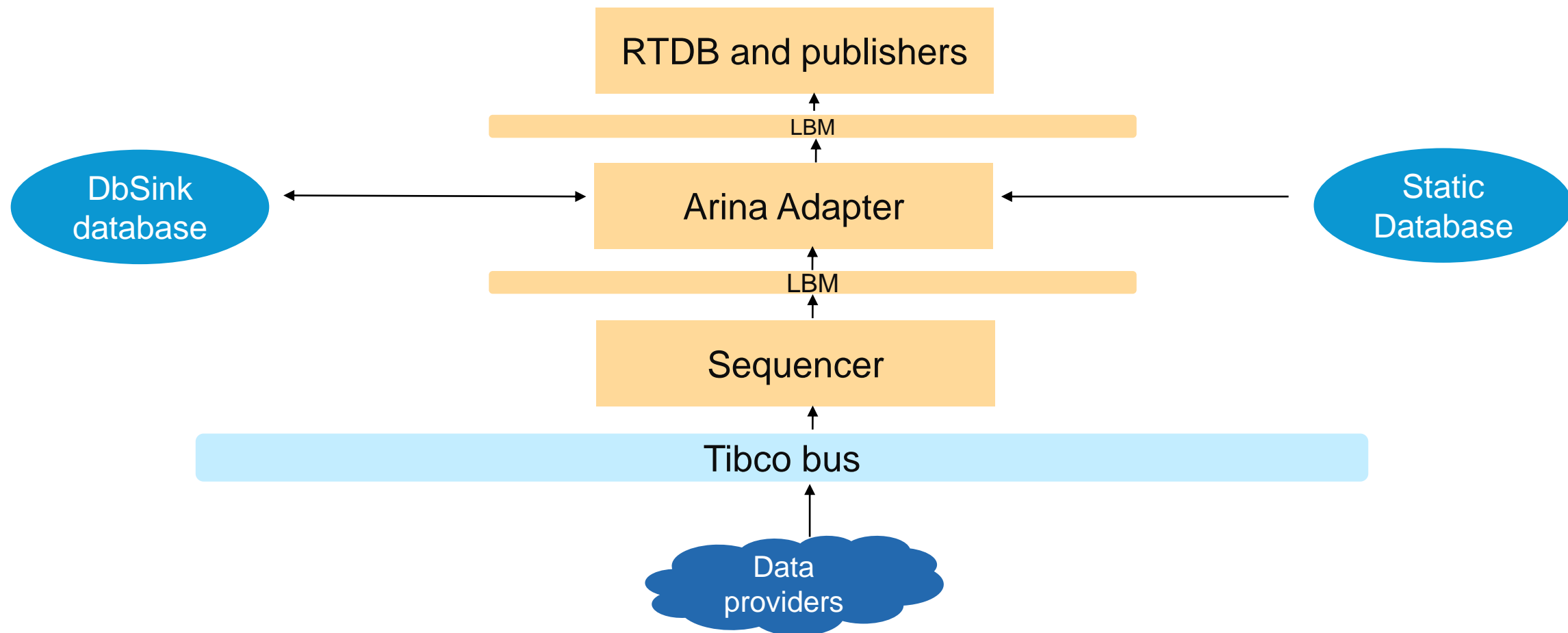
# Обзор приложения



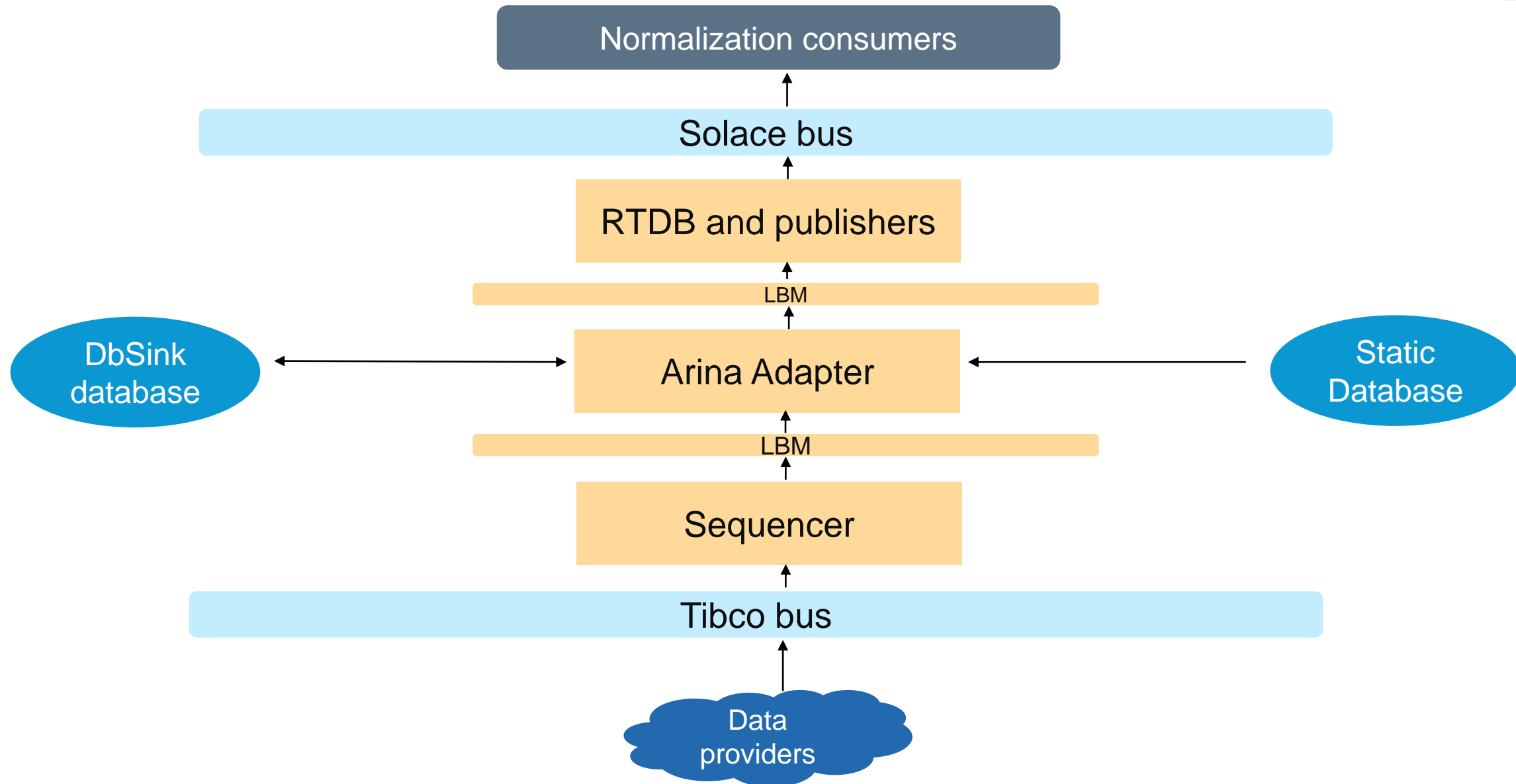
# Обзор приложения



# Обзор приложения



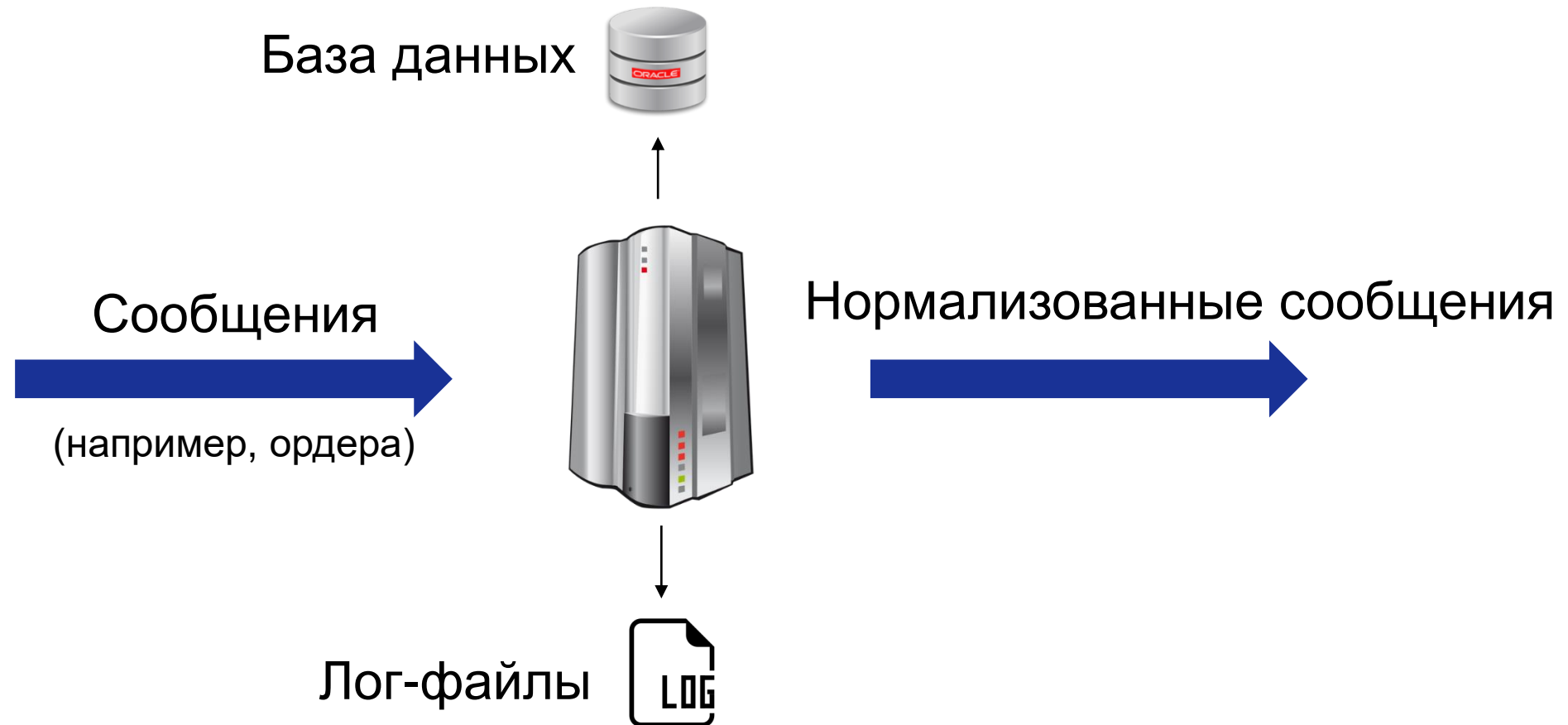
# Обзор приложения



# Axiom Arina Adapter (в действительности)



## Адаптер (транслятор) с детерминированной обработкой



# Что же мы делаем?



Front Office

Trading Desk



Trading Platform

Middle Office

## Middle Office системы со своим API

Фин. отчетность

Учет заявок

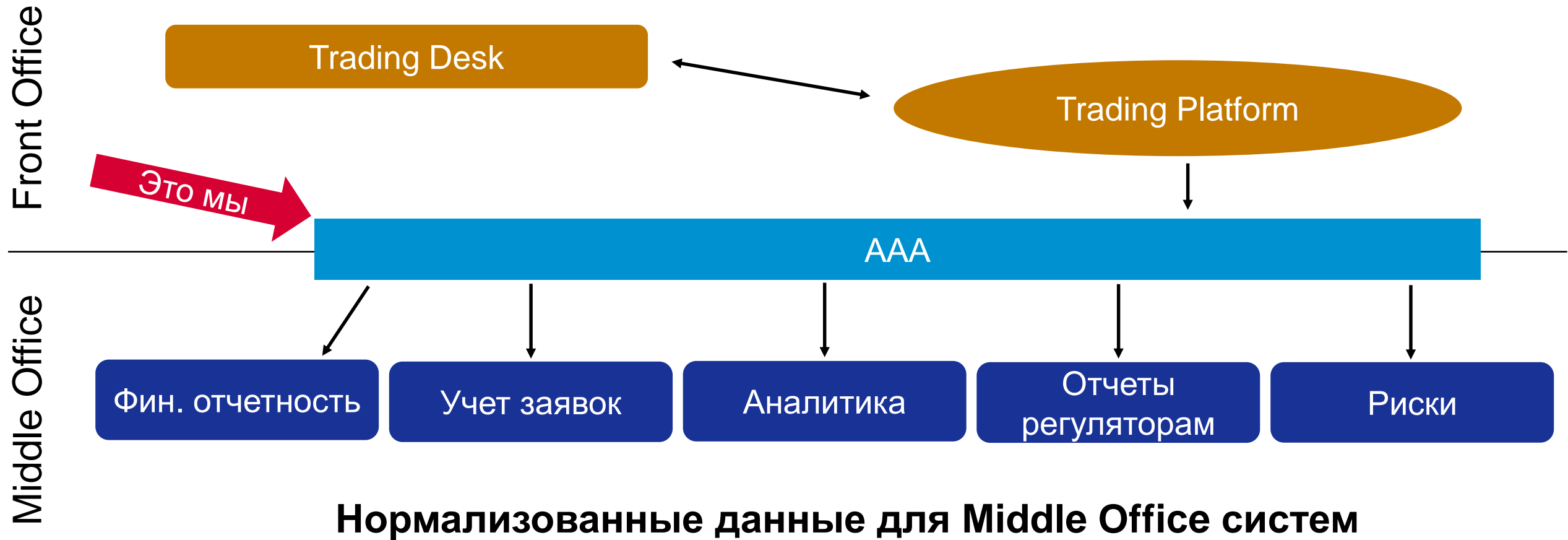
Аналитика

Отчеты  
регуляторам

Риски



# Что же мы делаем?



# Сложность тестирования



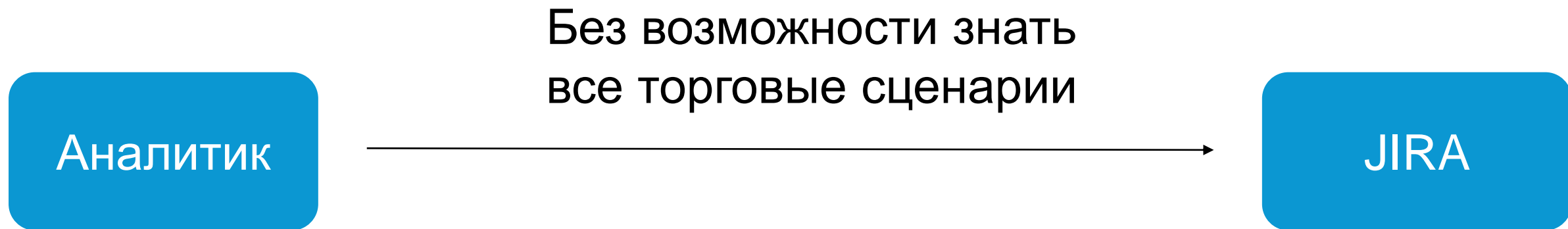
- Большое количество сценариев  
*(около 20 различных типов сообщений, сотня полей)*
- Десяток систем поставляющих/получающих данные
- Большой объем данных  
*(~ 3 миллиона сообщений в день)*

# Сложность тестирования

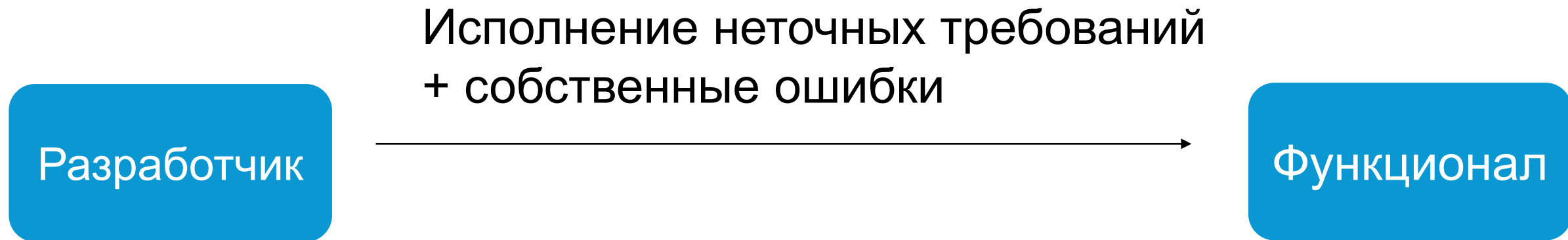


- Сложность воспроизведения тестовых сценариев
- Сложность интеграции в цепочку клиент-биржа
- Главное ничего не сломать!

# Как мы жили раньше



# Как мы жили раньше

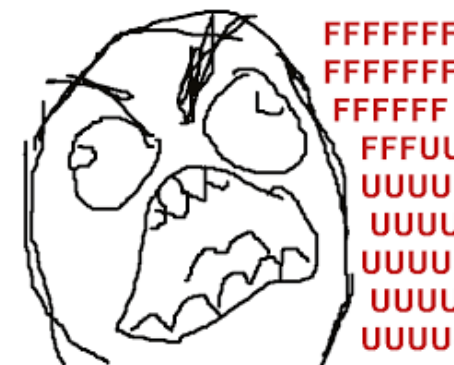


# Как мы жили раньше



Тестировщик

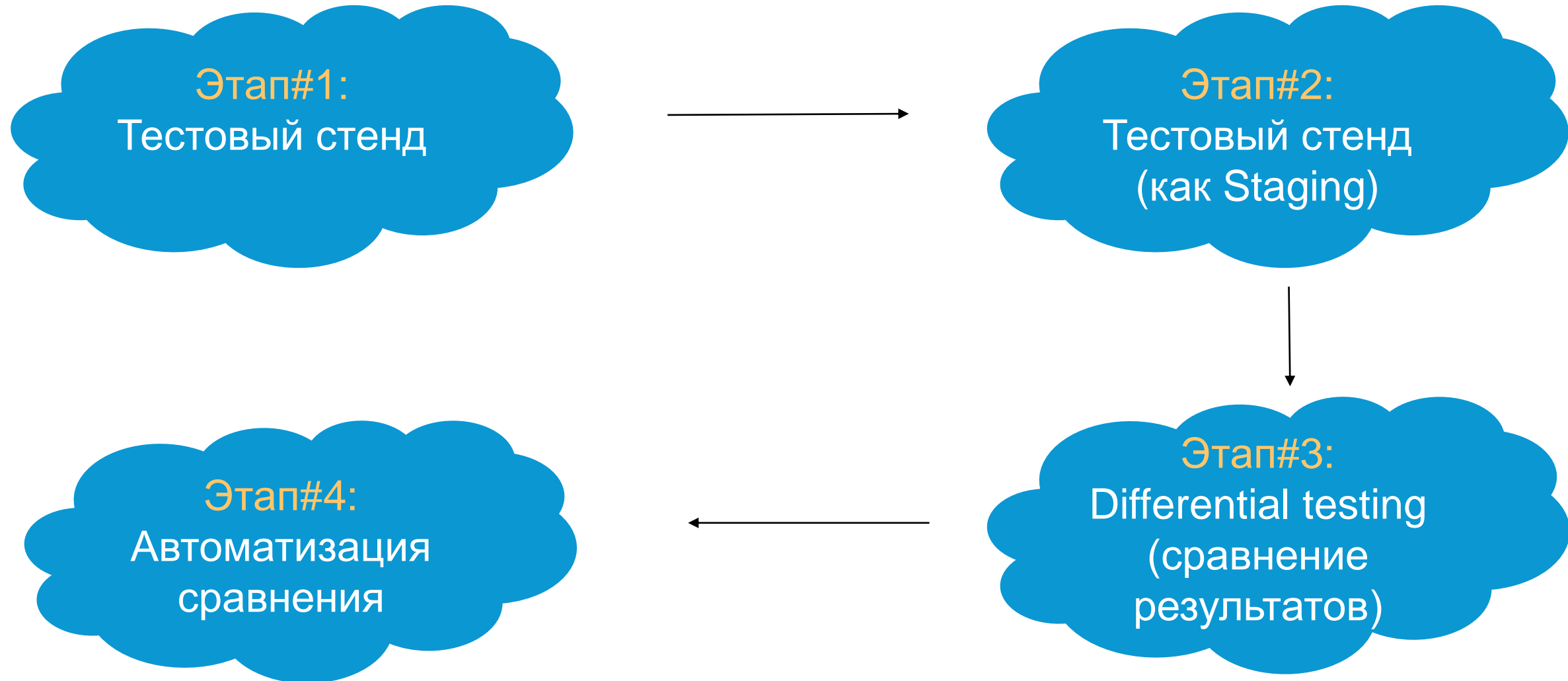
Прогоняет ручные регрессионные тесты,  
проверяет 0.000001% функциональности  
и не находит ни одной ошибки.



В итоге ловили 1-2 ошибки в месяц в PROD

С новым подходом регрессионных ошибок больше нет

# Эволюция подхода в проекте



# Параллельное тестирование



Параллельное тестирование (ProdParallel) – это сочетание:

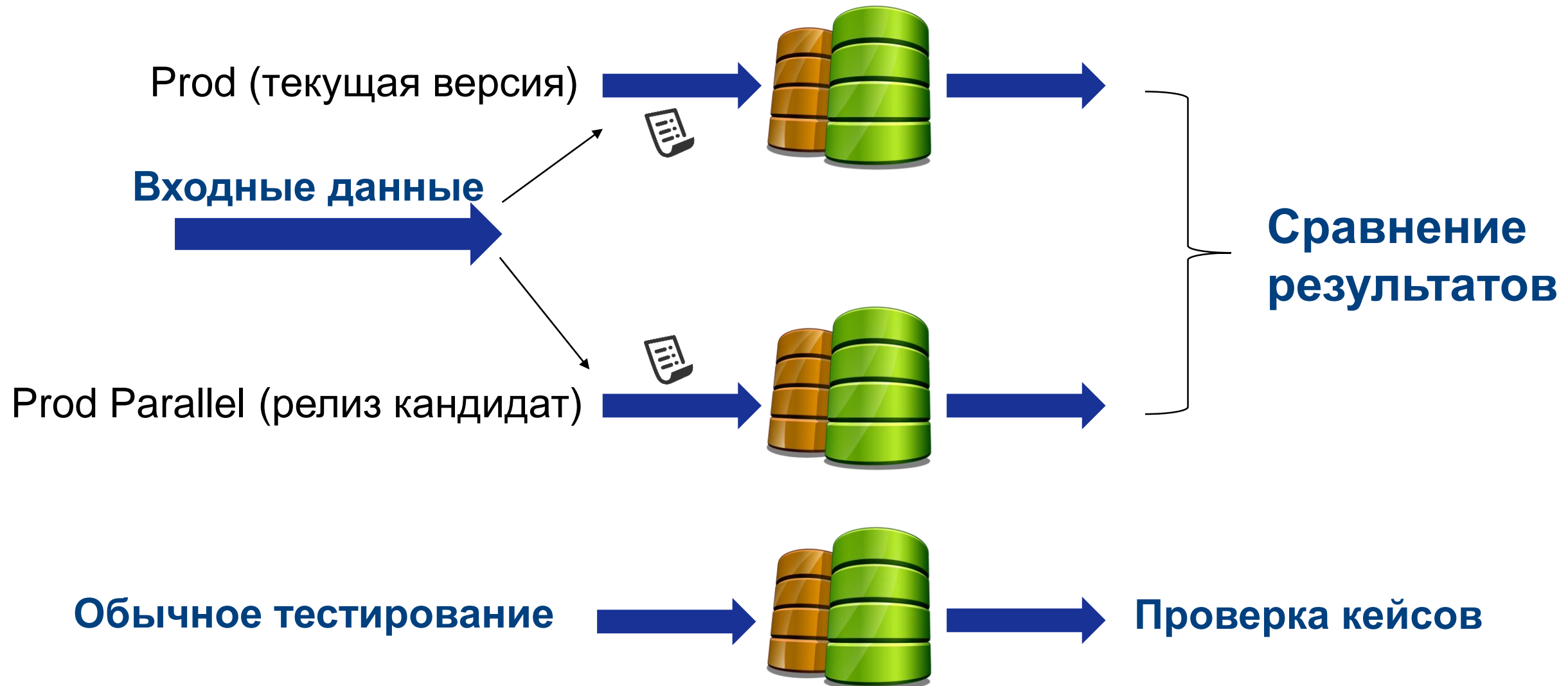
- Staging
- Differential testing
- Набора утилит для мониторинга и сравнения результатов

Что нужно:

- Большой объем разнообразных данных в продуктовой версии
- Различные сценарии работы в прод версии
- Детерминированность
- Энтузиазм и готовность к плюсам и минусам



# Обзор подхода



# Подход к реализации: ProdParallel как Prod



- Конфигурация тестового инстанса точно такая же, как PROD
  - разница только в конфигурации, необходимой для новой функциональности
  - (неприятная часть)

# Подход к реализации: ProdParallel как Prod

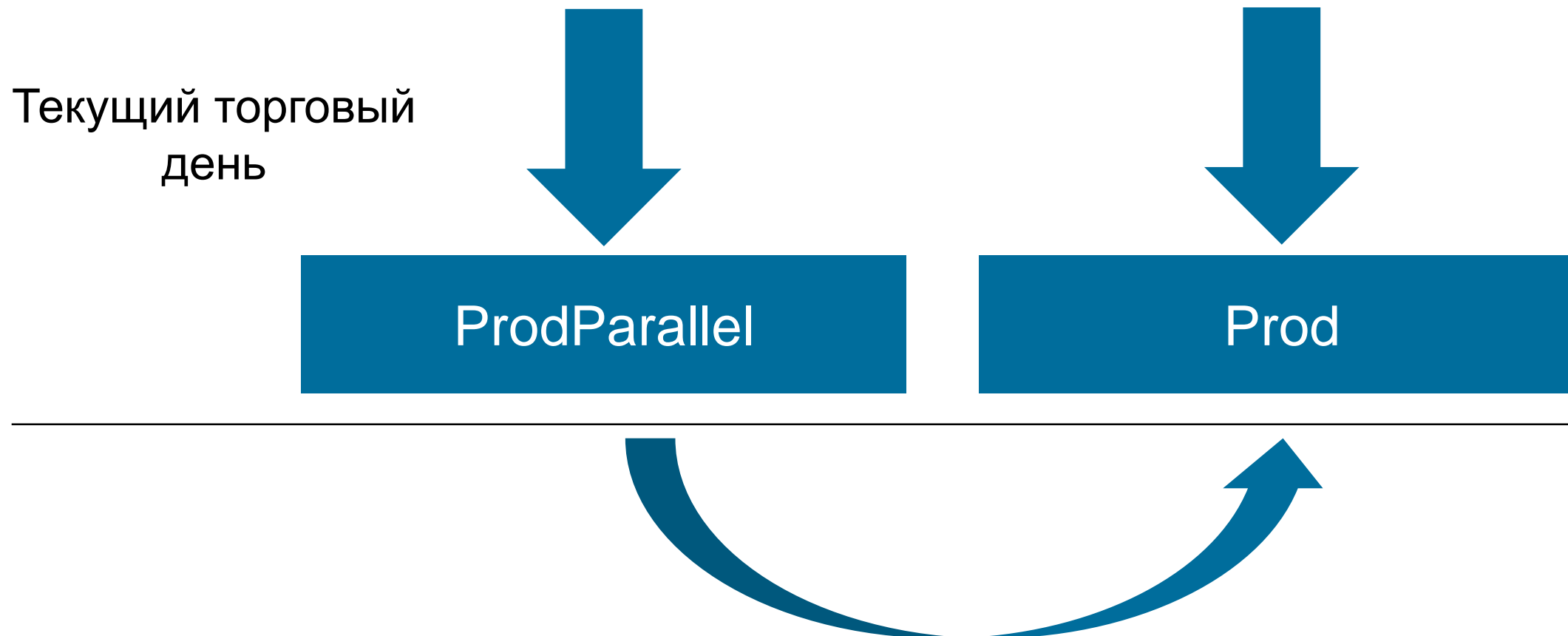


- Конфигурация тестового инстанса точно такая же, как PROD
  - разница только в конфигурации, необходимой для новой функциональности
  - (неприятная часть)
- В тестовой среде установлен релиз кандидат
- Одинаковое железо в PROD/тестовом стенде

# Воспроизведение данных: подход #1



Слушаем PROD **и параллельно** с ним обрабатываем данные



# Воспроизведение данных: подход #1



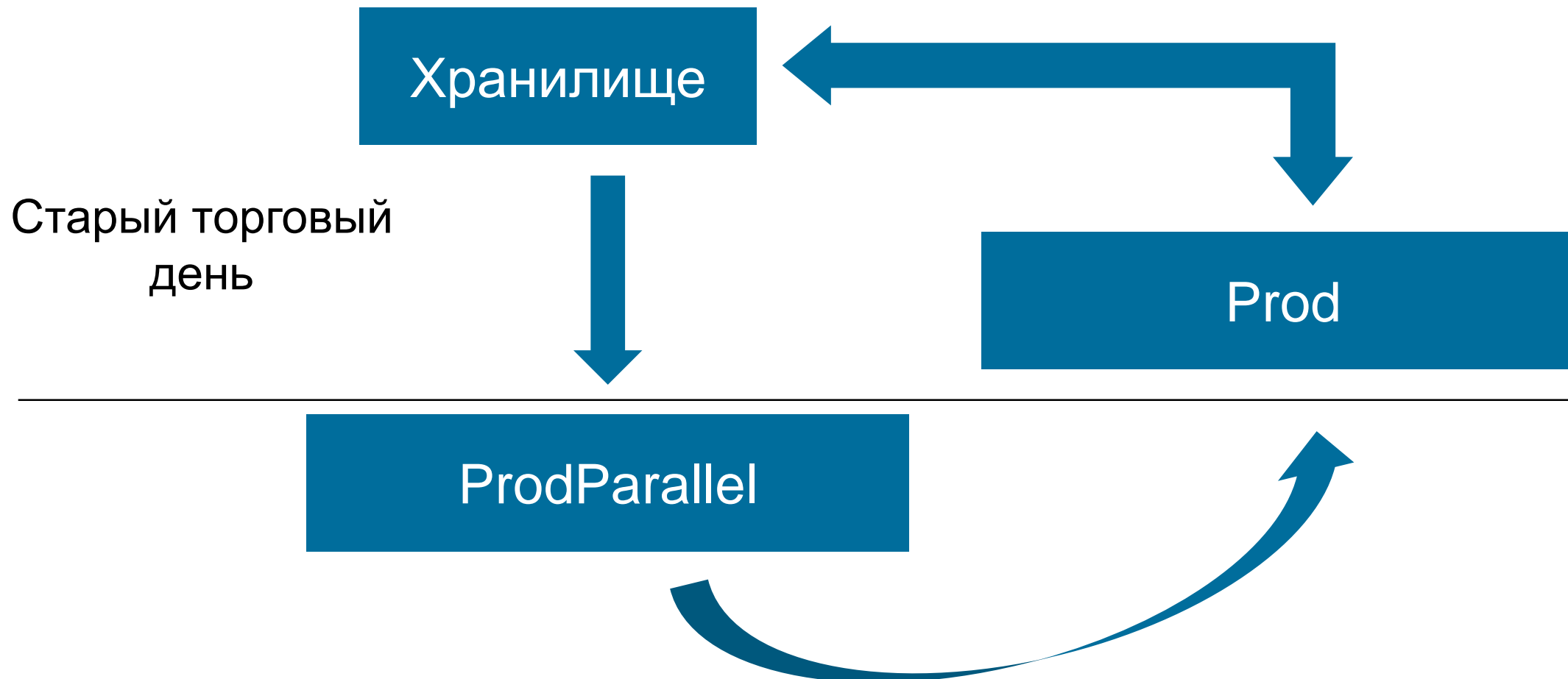
**Плюсы:** - real-time результаты

**Минусы:** - проблемы с it-аудитом  
- потери данных сравнения  
- невозможность перезапустить по несколько раз

## Воспроизведение данных: подход #2



Записываем **старые PROD данные**, проигрываем и сравниваем результат с соответствующим днем в PROD



# Воспроизведение данных: подход #2



- Решаются все проблемы первого подхода
- Подobie smoke testing
- He ~~Real-time~~ результаты

# Подход к реализации: сравнение



- Определить, что же нужно сравнивать:  
**логи/базы данных/xml-файлы/excel-файлы** и т.п.
- Организовать сравнение и анализ результатов:  
**вручную/автоматически**
- Организовать мониторинг ошибок и статуса тестового стенда



# ProdParallel как Prod



- Два набора конфигураций:
  - 1) ведутся параллельно для Prod/ProdParallel
  - 2) разработчики отвечают за изменения
- **Важно:** основная однотипная часть конфигураций максимально внесена в общий дистрибутив

# ProdParallel как Prod



- В тестовой среде и в ProdParallel установлена **самая свежая сборка** (для этого требуется continuous integration):
  - 1) коммит в git (Stash)
  - 2) авто-билд в TeamCity после каждого коммита
  - 3) авто-импорт в Udeploy
  - 4) установка на сервер
  - 5) воспроизведение данных
  - 6) работа по сравнению результатов

# TeamCity + Udeploy integration



		Results	Artifacts	Changes
refs/heads/master	#2.17.0-20171017.123159-47	✓ Tests passed: 80   ▾	View   ▾	Changes (8)   ▾
refs/heads/master	#2.17.0-20171017.100305-46	✓ Tests passed: 80   ▾	View   ▾	Changes (4)   ▾



aaa-apac-replay		Моментальная копия: Нет
Компонент	Версия	
<input type="text"/>	<input type="text"/>	
aaa-thirdparty-tibco	8.4.4-2 (Показать сведения)	
aaa-thirdparty-instantclient	11.2.0.4.0-1 (Показать сведения)	
aaa-thirdparty-lbm	6.8-3 (Показать сведения)	
AAA-distro-uat	2.16.0-20170928.125955-5 (Показать сведения)	
aaa-thirdparty-java	1.8.0_131-15 (Показать сведения)	
AAA-configs-apac-replay	master-20171002.182536-16 (Показать сведения)	

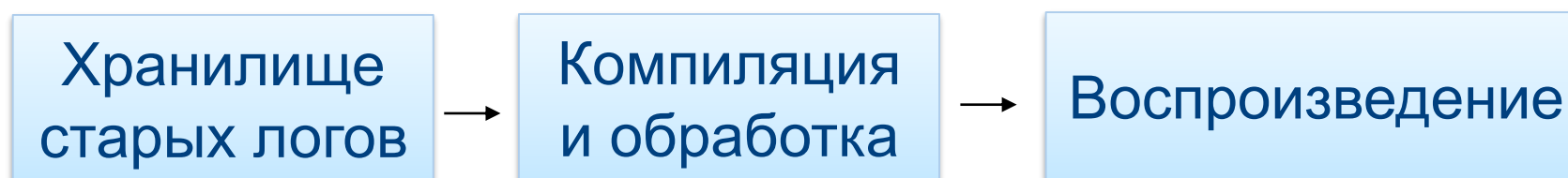
# Воспроизведение данных



Ежедневное **авто**воспроизведение старых данных

Проигрывание в любой момент времени

Требуется собирать, хранить и обрабатывать данные перед воспроизведением





- Почтовые уведомления о состоянии тестового инстанса в течение дня
- Ежедневный дайджест, содержащий список ошибок  
(в противовес real-time системе мониторинга в PROD: **ITRS GENEOS**)
- Ошибки сгруппированы по паттернам, что позволяет оценить их критичность
- *Удаляются уникальные идентификаторы, выбирается общая подстрока в ошибке*

# Мониторинг



Version: distro-2.18.0-20171112.172610-13  
[PP is using Latest Version]

проверка последней версии

=====AAA=====

встретилось 15 ошибок с подобным патерном

текст ошибки

**Error [15 matches]:** (enrichers.impl.AbstractClientFieldsEnricher) - Arina provided tibco.quorumClientID different from resolved

**Error [2 matches]:** (enrichers.impl.SettlDateEnricher) - GbaSettlDate (val) Thu Nov :: GMT () differs from ArinaSettlDate (val) Wed

=====SEQ=====

/export/data/seq\_standalone.out.2017-11-14

для группировки ошибок по подстроке,  
уникальные идентификаторы удалены

=====RTDB=====

/export/data/rtdb\_sinkwriter\_standalone.out.2017-11-14

**Error [1 matches]:** (transports.solace.SEMPProvision) - Retry of on semp request timeout...

не забываем про warnings

=====AAA=====

**Warning [3 matches]:** (vas.wcc.ComponentVersionReporter) - WCC Reporting successfully competed

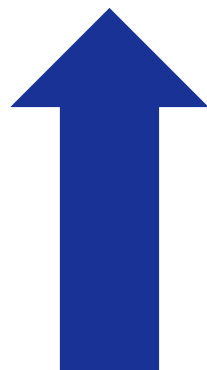


- Объекты для сравнения – **таблицы в базах данных и записи о транзакциях**
- Java утилита для сравнения Oracle баз – TableDiff:
  - 1) позволяет сравнивать запросы из любых таблиц по уникальному ключу
  - 2) позволяет делать сравнения результатов join по нескольким таблицам
  - 3) прост в использовании и настройке (достаточно первичного ключа)

# Сравнение и анализ (TableDiff)

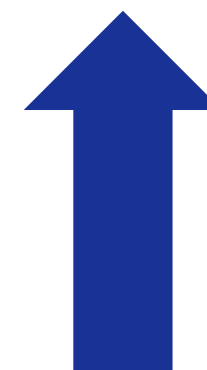


Prod (база данных №1)



Первичный ключ  
↔  
(определяется  
пользователем)

Replay (база данных №2)





# Сравнение и анализ (TableDiff)



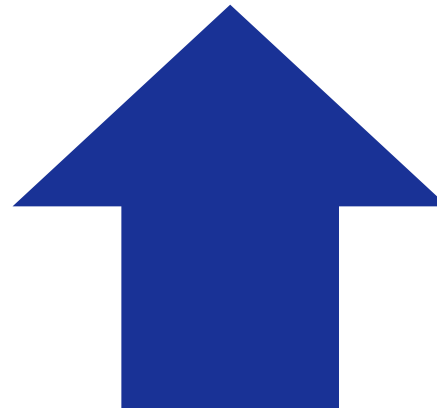
Prod (база данных №1)



Первичный ключ  
(Col#1)



Replay (база данных №2)



# Сравнение и анализ (TableDiff)



Prod (база данных №1)

Col#1	Col#2
123	A

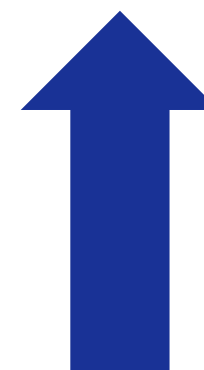
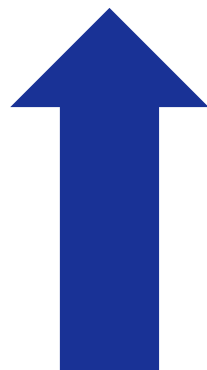
Replay (база данных №2)

Col#1	Col#2
123	B

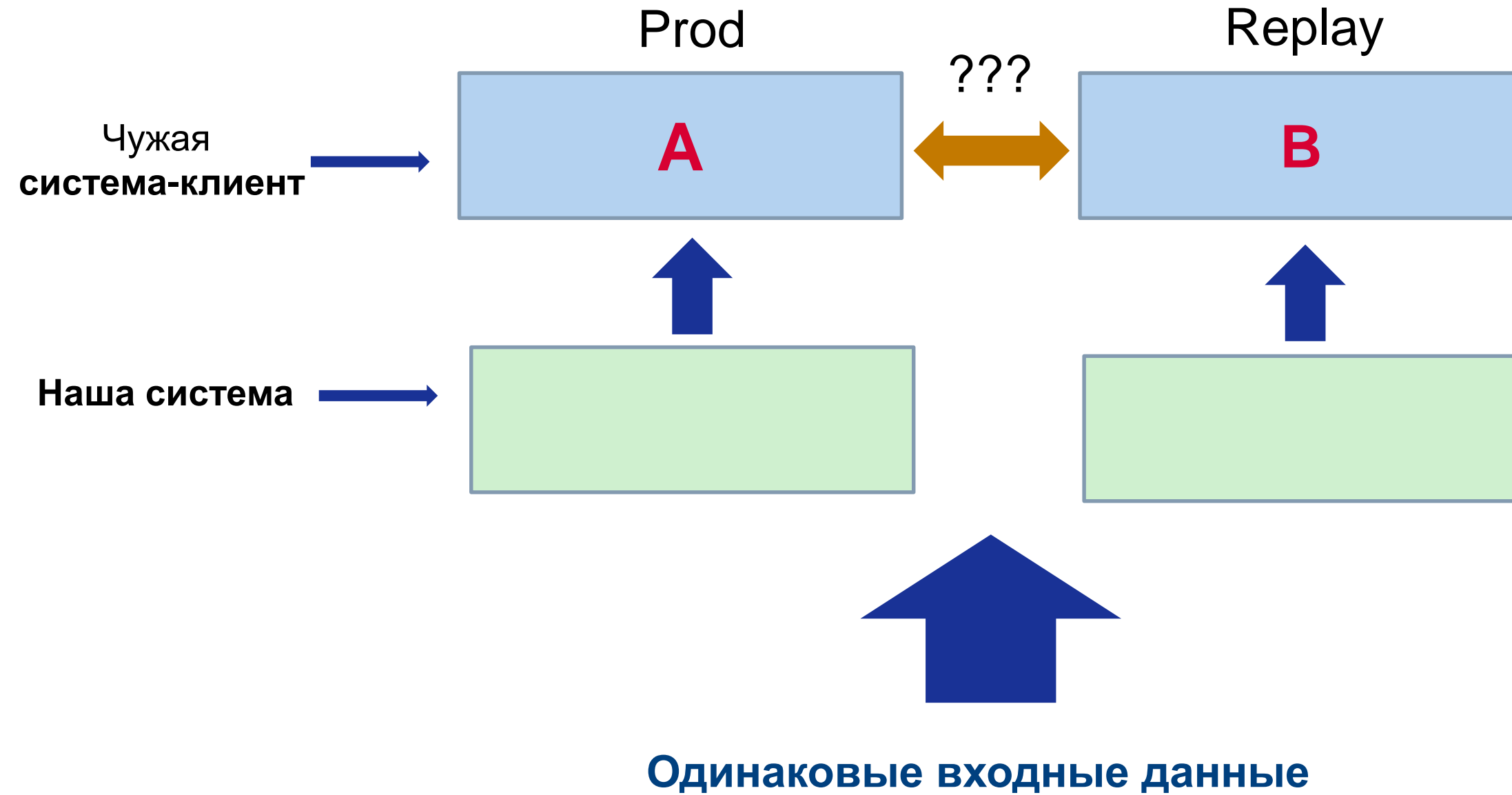
Первичный ключ  
(Col#1)



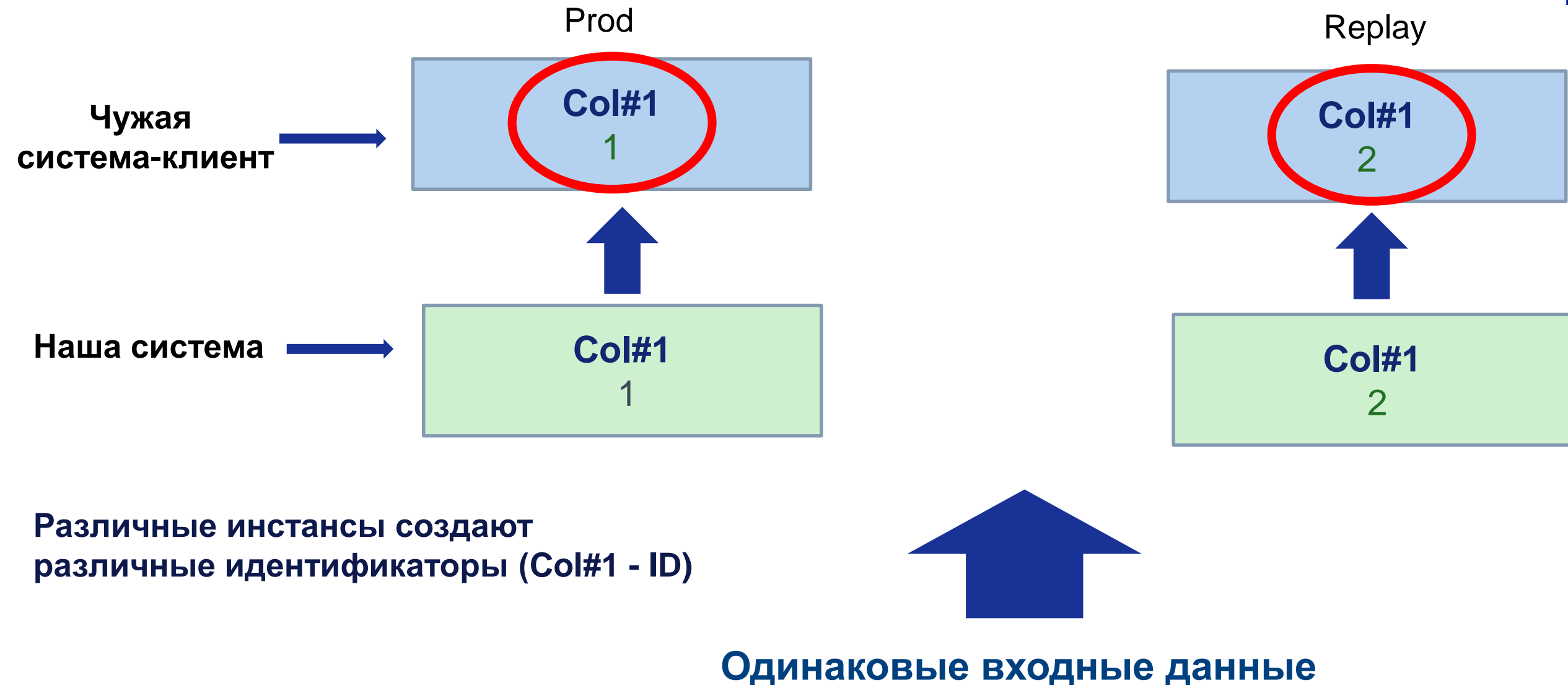
Diff: A vs B в Col#2



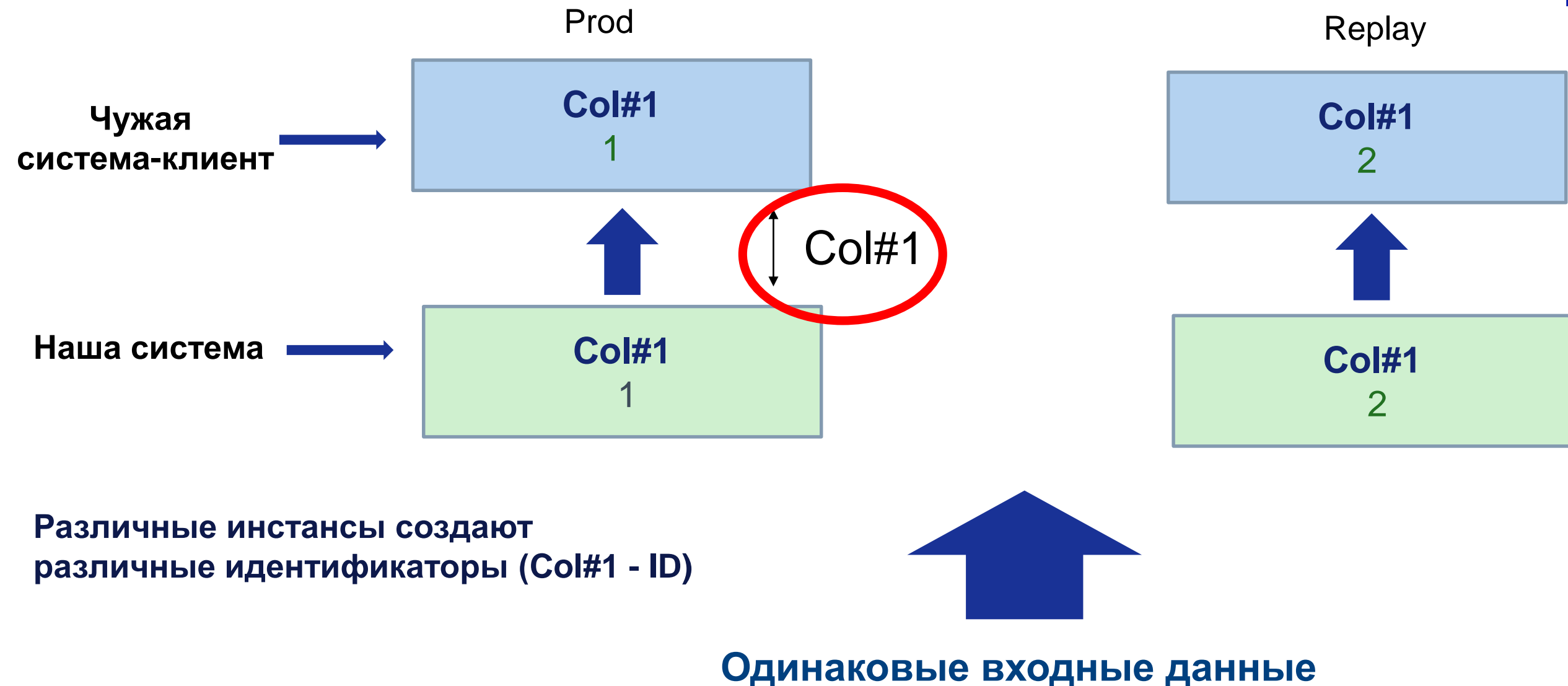
# Сравнение и анализ (TableDiff Join)



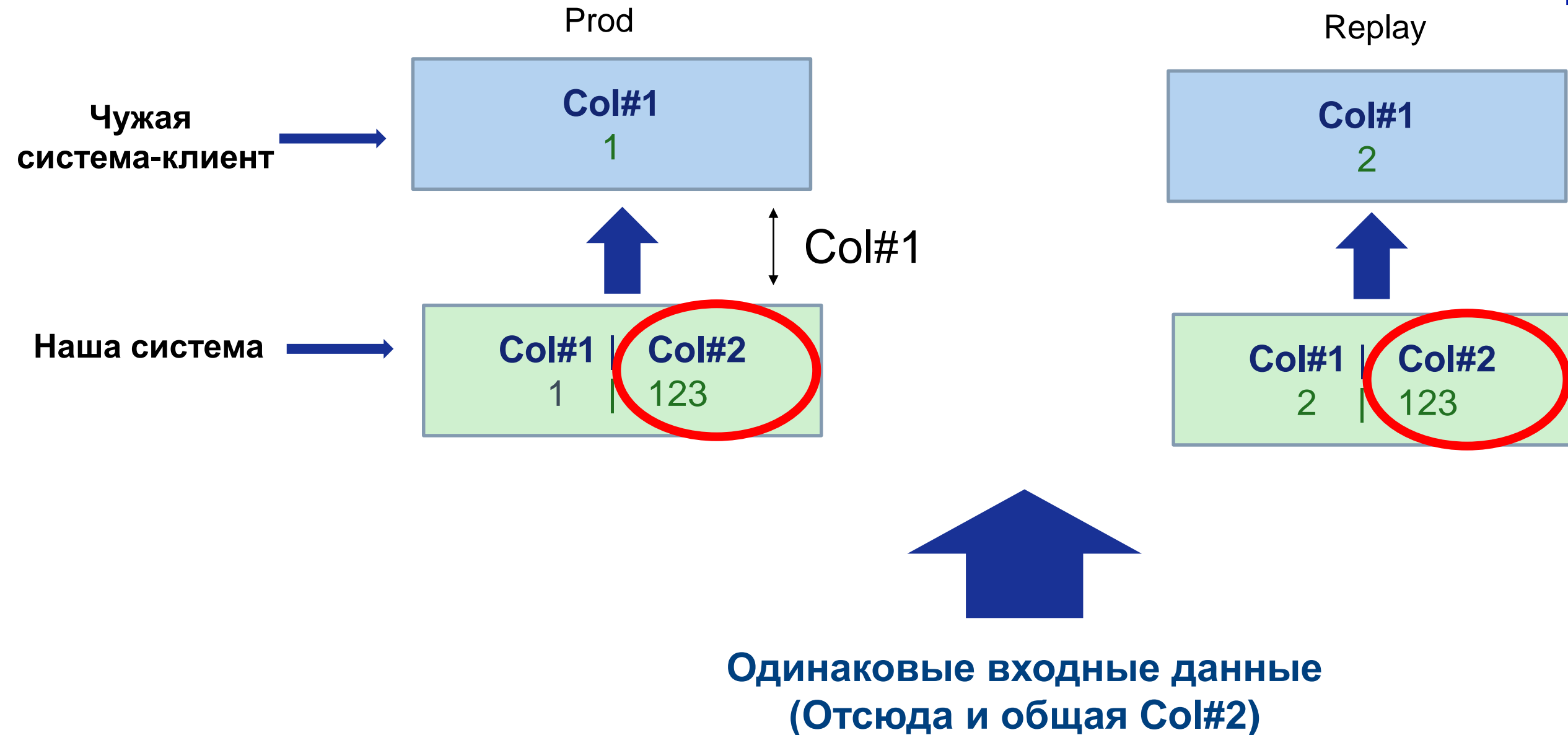
# Сравнение и анализ (TableDiff Join)



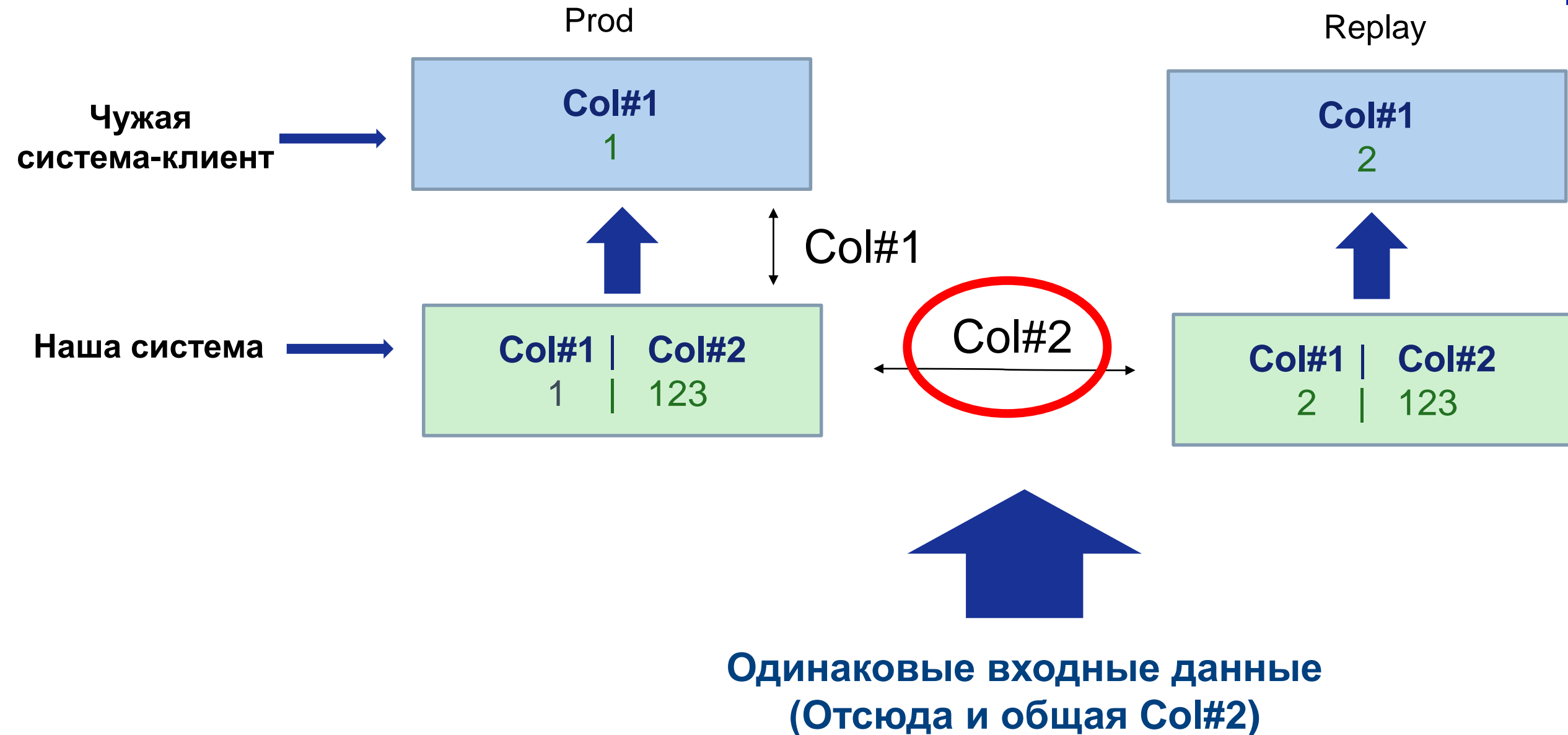
# Сравнение и анализ (TableDiff Join)



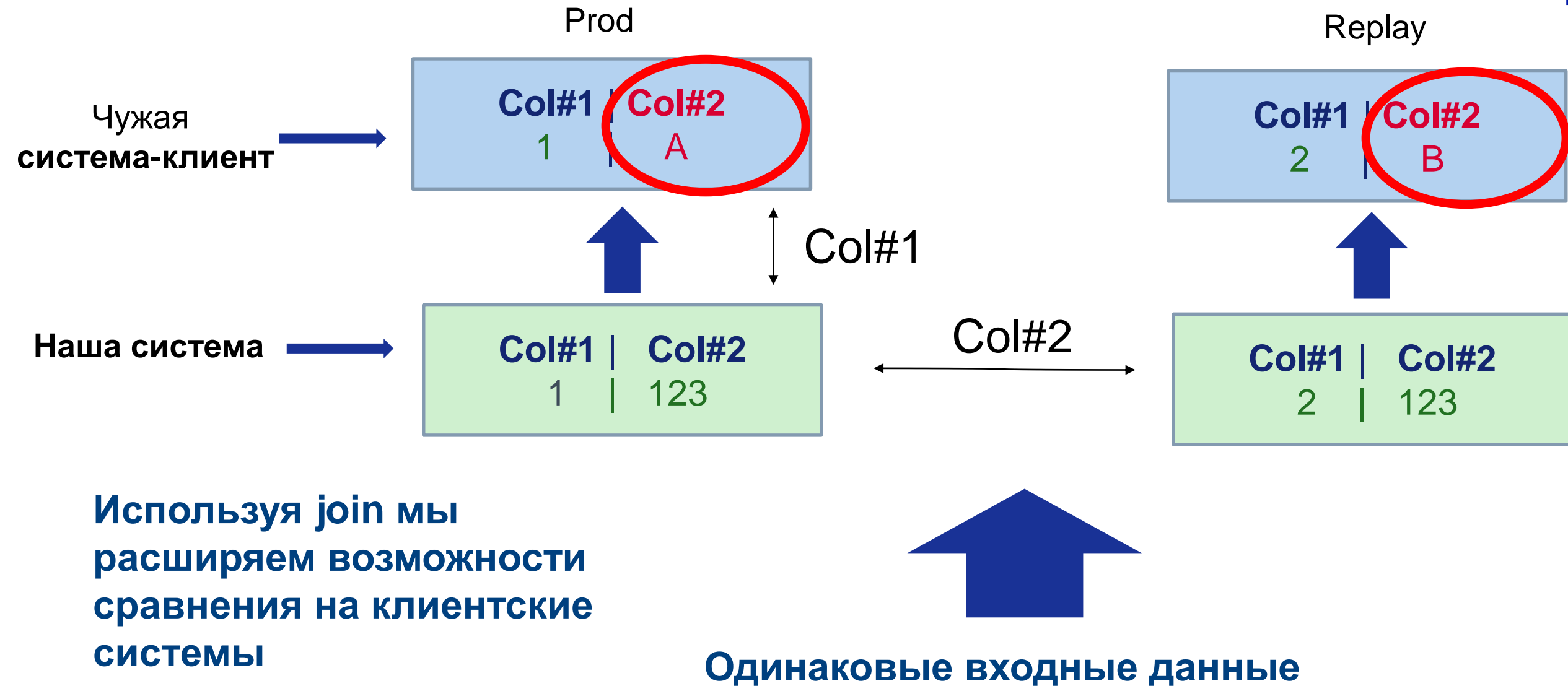
# Сравнение и анализ (TableDiff Join)



# Сравнение и анализ (TableDiff Join)



# Сравнение и анализ (TableDiff Join)





# Сравнение, интеграция с JIRA и анализ результатов



- TableDiff позволяет сравнивать большое количество результатов
- Мы можем сравнивать и другие клиентские системы работающие в режиме ProdParallel
- Но где хранить и как анализировать результаты?

# Сравнение, интеграция с JIRA и анализ результатов



- Python утилита для анализа результатов
- Набор колонок для сравнения и первичный ключ создается аналитиками
- Исключаются “динамические” данные: времена/генерируемые ID
- Хранение репортов в JIRA (ради GUI), используя xmlrpc и Jira API

# Анализ результатов (JIRA как удобный GUI)



Sessiondate: 2017-10-06

Запрос для анализа по БД

<div>TX_TXN_ID</div> <div>Имя колонки</div>	<div>PROD=13752163 vs PP=4991483</div> <div>Разница</div>	<div>SELECT TX_TXN_ID FROM TEXECUTIONS_TX WHERE SESSIONDATE=TO_DATE ('2017-10-06', 'YYYY-MM-DD') AND ORIGINAL_ORDER_ID='43546457323114'</div>
---	---	---

# Анализ результатов (JIRA как удобный GUI)



 **technical: 26550-11** added a comment - 09/Oct/17 7:07 AM

===AUTOGENERATED REPORT===

Sessiondate: 2017-10-06

Запрос для анализа по БД

<b>TX_TXN_ID</b>	PROD=13752163 vs PP=4991483	SELECT <b>TX_TXN_ID</b> FROM TEXCUTIONS_TX WHERE SESSIONDATE=TO_DATE ('2017-10-06', 'YYYY-MM-DD') AND ORIGINAL_ORDER_ID='43546457323114'
<b>EXEC_BROKER</b>	PROD=null vs PP=2699950	SELECT <b>EXEC_BROKER</b> FROM TEXCUTIONS_TX WHERE SESSIONDATE=TO_DATE ('2017-10-06', 'YYYY-MM-DD') AND ORIGINAL_ORDER_ID='43546457323114';

Имя колонки

Разница

в значениях

# Сравнение, интеграция с JIRA и анализ результатов



- QA команда с помощью аналитиков исследует результаты
- Ежедневный анализ разницы – часть процесса
- Такой подход позволяет иногда **тестировать новую функциональность**

# Sanity checks



- Если есть возможность еще и **верифицировать данные** – надо это использовать!
- Предпосылки: необходимость контролировать и сравнивать логику работы приложения как в PROD, так и в PP
- **Ежедневное исполнение тестовых sql-запросов** для проверки бизнес логики как Prod, так и ProdParallel

# Sanity checks



- Простейшее решение при наличии у приложения записи в базу
- SQL запросы пишутся QA командой для проверки реализованной функциональности
- Некоторая часть функционального тестирования
- **Sanity** – проверка на “**вменяемость/здравомыслие**” данных

# Sanity checks (реализация)





- Python script для исполнения запросов (cx\_Oracle) и создания e-mail репортов
- Интеграция с Confluence используя xmlrpclib и Confluence API
- Confluence – как простой и удобный GUI для хранения запросов и истории изменений










# Интеграция с Confluence



Простой и удобный способ хранения, изменения запросов с историей (GUI)

 Confluence Spaces ▾ People Questions Calendars **Create**



Pages / Home / ArinaRTDBAdapter (aka ArinaAxiomAdapter)

## [AAA DBsink] Data Sanity Checks

Created by [REDACTED] last modified by Andrew Kuleshov just a moment ago

**#1.Rejected or Filled order with leaves qty**

```
select count (*) from torders_tx where sessiondate >= trunc(sysdate) and is_last=1 and leaves_qty > 0 and ord_status in ('8','2') and creation_time>= trunc(sysdate) ;
```

**#2.cum\_qty+leaves\_qty!=order\_qty**

```
select count (*) from torders_tx where sessiondate >= trunc(sysdate) and leaves_qty+cum_qty!=order_qty and leaves_qty>0;
```

**#3a.short sale reason for not-short sell sides**

```
select count (*) from torders_tx where sessiondate >= trunc(sysdate) and is_last=1 and side not in (5, 6,1) and short_sale_reason is not null;
```

**#3b.No SS/SSE/SellPlus/BuyMinus in EU**

```
[EU]select count(*) from torders_tx where sessiondate>=trunc(sysdate) and side in ('3','4','5', '6');
```

# Sanity checks (отчеты)



Письма – лучшие уведомления об ошибках 😊

===Sanity check failed===

Please see **FAILED** sanity checks below

=====

#02/10/2017 16:28:37 GMT #Example of comment

— информация о запросе **negative test**

-----

SQL QUERY: select count(\*) from **EXAMPLE\_TABLE** where **EXAMPLE\_CONDITION**

**EXAMPLE:** select \* from **EXAMPLE\_TABLE** where drop copy order id=**EXAMPLE\_ID**

**FAILED** (32)

=====

пример для исследования проблемы

# Как это работает



## Не забывайте!

- Никто не отменяет стандартные методы тестирования
- **Differential testing** – дополнение, которое облегчает работу
- Каждая новая функциональность также покрывается юнит тестами и функциональными тестами
- Ручное тестирование полезно и иногда незаменимо в рамках интеграции

# Плюсы



- ✓ Широкая применимость и возможность **быстро** создать подобную инфраструктуру
- ✓ Поиск и **анализ тонких мест** в требованиях
- ✓ Поиск сайдэффектов в работе приложения

# Плюсы



- ✓ Широкая применимость и возможность **быстро** создать подобную инфраструктуру
- ✓ Поиск и **анализ тонких мест** в требованиях
- ✓ Поиск сайдэффектов в работе приложения
- ✓ Быстрое получение результата
- ✓ Нахождение регрессионных ошибок
- ✓ Тестирование в условиях, **максимально приближенных к реальности**

# Минусы



- **Нельзя рассчитывать на 100% покрытие функциональности**
- Требуется ресурсы для поддержки инстанса и инфраструктуры
- Возможна разница в работе, связанная с железом и конфигурациями
- Необходимы ресурсы для анализа и исследования разницы

# Минусы



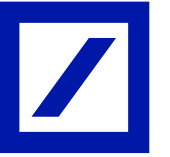
- **Нельзя рассчитывать на 100% покрытие функциональности**
- Требуются ресурсы для поддержки инстанса и инфраструктуры
- Возможна разница в работе, связанная с железом и конфигурациями
- Необходимы ресурсы для анализа и исследования разницы
- **Ограничения**, которые накладываются на систему:
  - разнообразие сценариев/входных данных в PROD
  - необходимость в хранении результатов
  - детерминированная логика работы приложения

# Личные ощущения



- У нас не было **ни одной регрессионной ошибки** в PROD за время использования этого подхода
- В **разы** облегчили работу тестировщиков и разработчиков (на регрессию больше не уходит несколько недель)
- Ловим баги, которые не смогли поймать во время ручного тестирования (сайдэффекты)
- Соседние команды адаптируют наш подход
- Советуем к использованию





# Спасибо за внимание!

## Вопросы





Данный материал не является предложением или предоставлением какой-либо услуги. Данный материал предназначен исключительно для информационных и иллюстративных целей и не предназначен для распространения в рекламных целях. Любой анализ третьих сторон не предполагает какого-либо одобрения или рекомендации. Мнения, выраженные в данном материале, являются актуальными на текущий момент, появляются только в этом материале и могут быть изменены без предварительного уведомления. Эта информация предоставляется с пониманием того, что в отношении материала, предоставленного здесь, вы будете принимать самостоятельное решение в отношении любых действий в связи с настоящим материалом, и это решение является основанным на вашем собственном суждении, и что вы способны понять и оценить последствия этих действий. ООО "Дойче Банк Техцентр" не несет никакой ответственности за любые убытки любого рода, относящихся к этому материалу.