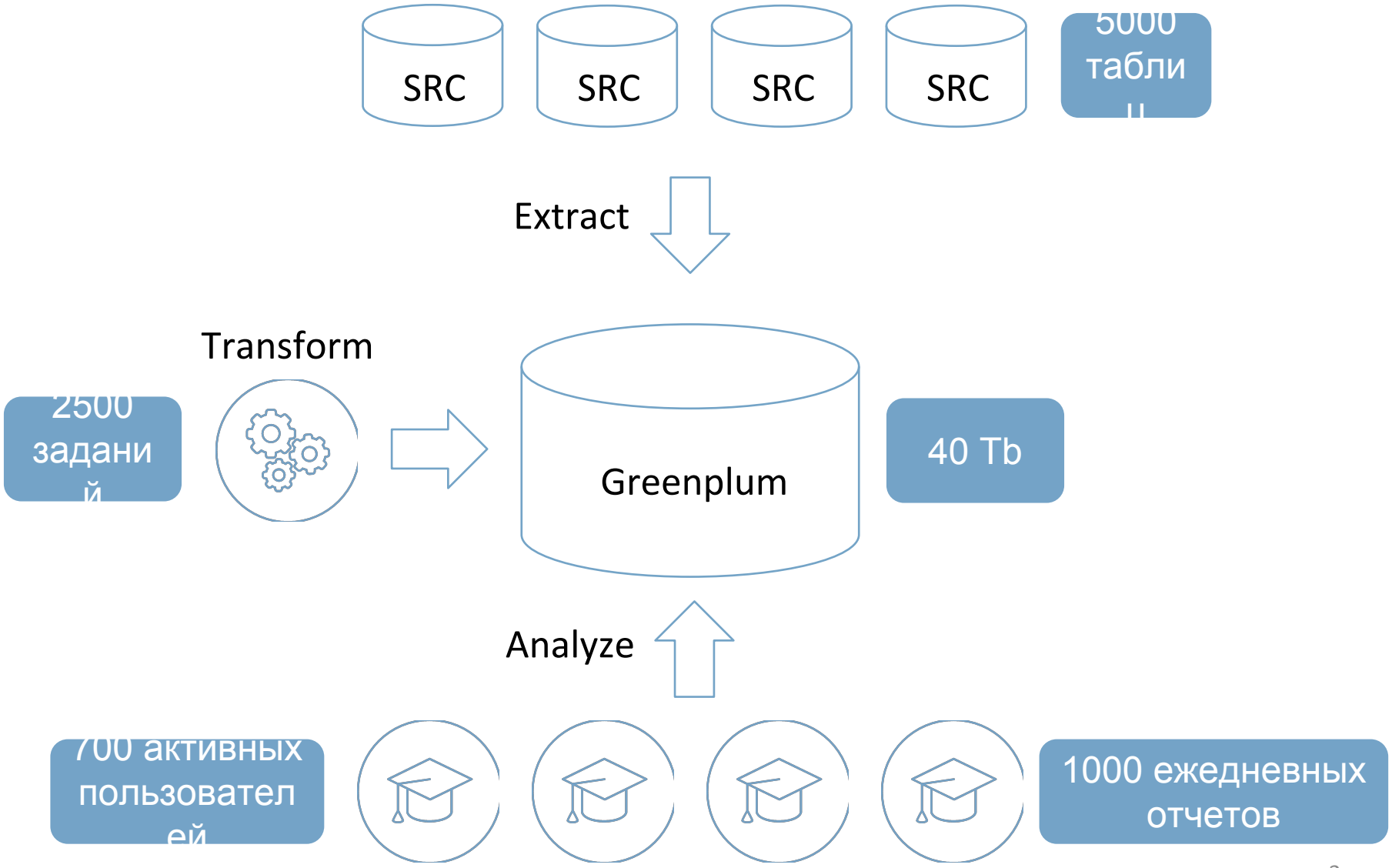
Greenplum
Опыт использования

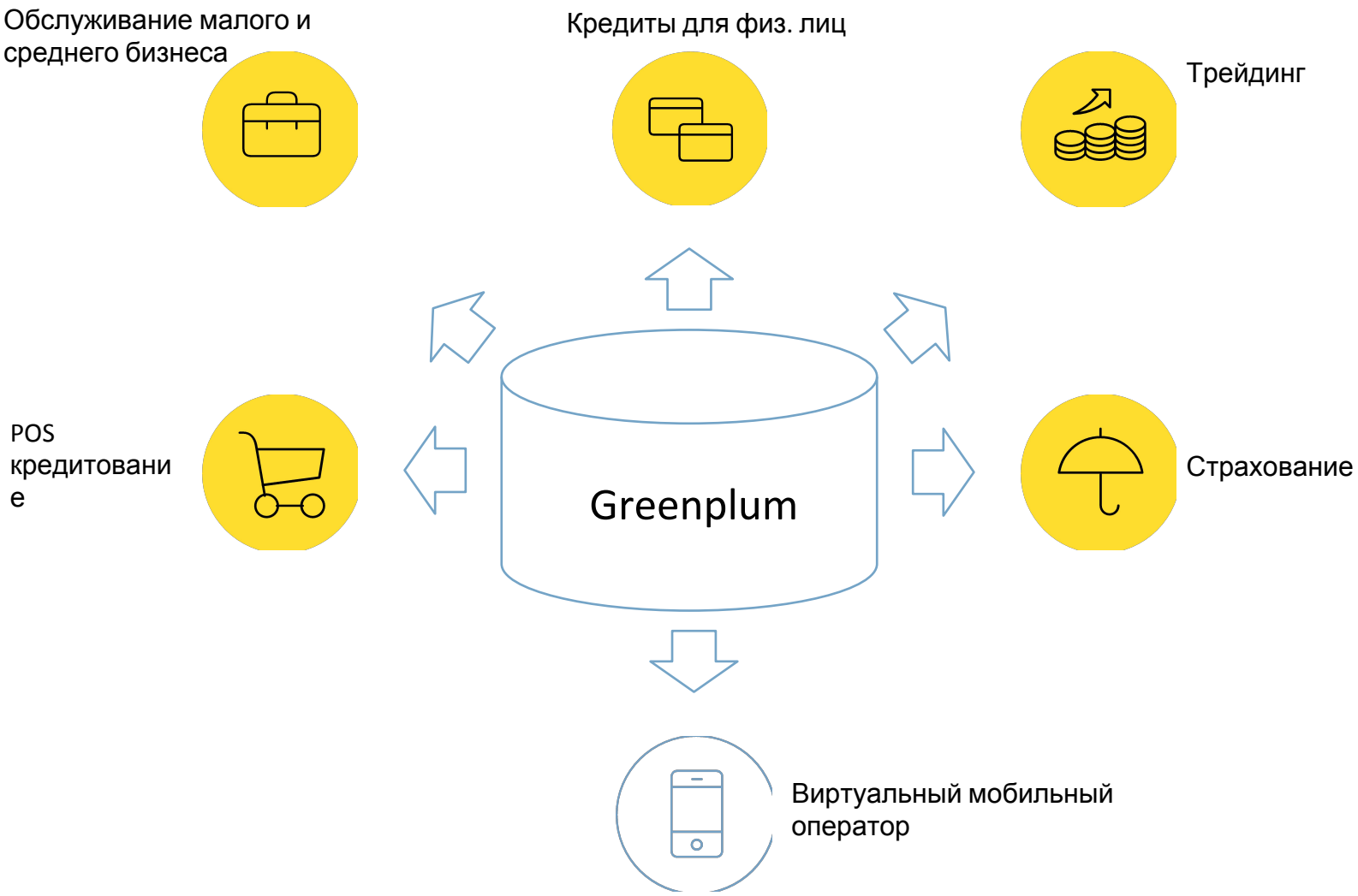
Tinkoff.ru

Опыт использования

Хранилище Данных



Единое Хранилище данных



Давным-давно

2011 год

Давным-давно

Загрузка, обработка (ETL), хранение и аналитика (BI) работает целиком на SAS платформе

Данные хранятся в SAS-файлах на NAS

Объем данных 2 Tb

2011 год

Требования к новой DB

- Оптимальна для аналитической нагрузки.
  - Задачи хранилища данных должны решаться максимально эффективно.
- Линейная масштабируемость
  - При необходимости дозакупаем новые сервера, а не меняем целиком один большой сервер на еще больший сервер
- Интеграция с SAS ETL
- Интеграция с SAS BI
- Soft based (не DCA)
  - Сами контролируем закупку железа
  - Оптимизируем расходы $$$

Выбор

TERADATA

Greenplum

- Soft based
- Интеграция с SAS
- Эффективна на наших запросах

- Сжатие на FPGA
- DCA

NETEZZA

ORACLE®

Данные в Greenplum

Готовим данные правильно!

Данные в Greenplum

- Используем партицирование
  - Позволяет применять разные типы хранения и сжатия для разных частей одной и той же таблицы

```
CREATE TABLE prod_dds.bank_card
(
    card_rk INTEGER NOT NULL,
    ...
    valid_from_dttm timestamp without time zone,
    valid_to_dttm timestamp without time zone NOT NULL,
)
TABLESPACE ssd
DISTRIBUTED BY (card_rk)
PARTITION BY RANGE (valid_to_dttm)
(
    PARTITION p_inact
    START ('2007-03-06 00:00:00'::timestamp without time zone) END ('2050-01-01 00:00:00'::timestamp without time zone) EVERY ('45 years'::interval)
    WITH (orientation=column, appendonly=true),
    PARTITION p_act
    START ('5998-12-31 23:59:59'::timestamp without time zone) END ('5999-01-01 00:00:00'::timestamp without time zone) INCLUSIVE EVERY ('3 years'::interval)
    WITH (appendonly=false));
```

- Используем поколоночное хранение
  - Позволяет читать только необходимые поля
  - Ускоряет запросы, использующие небольшое кол-во полей
  - Сжатие работает более эффективно

Данные в Greenplum

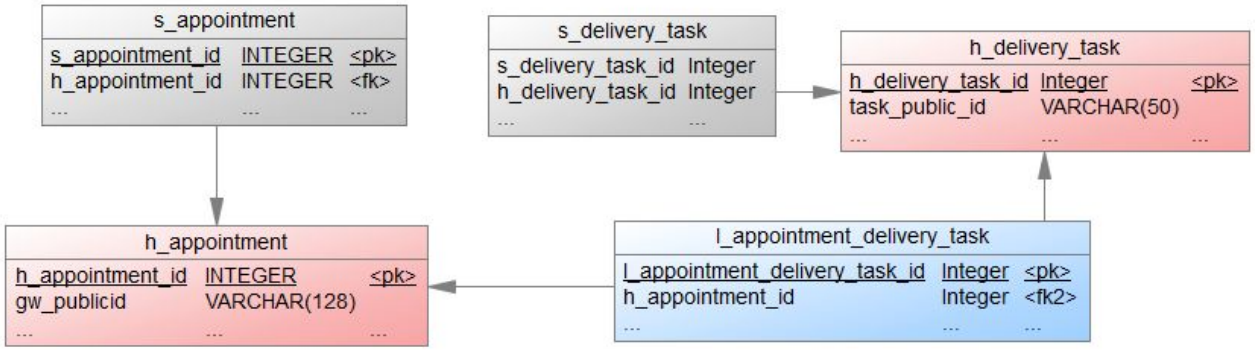
- Используем сжатие
  - RLE – хорошо сжимает повторяющие блоки значений, оптимально использовать совместно с сортировкой
  - Quicklz – быстрое сжатие с низкой нагрузкой на CPU, сокращает объем таблиц на 30-50%
- Правильно выбираем поля распределения
  - Распределять необходимо по наиболее вероятному полю join'а, чтобы избежать Redistribute Motion

Данные в Greenplum

- Стараемся не создавать слишком много объектов в DB
  - При большом количестве объектов в БД, узким местом могут стать системные каталоги – простое установление соединение с базой будет занимать десятки секунд
- Регулярно делаем Vacuum
  - Не забываем про системные каталоги (pg_catalog), особенно если часто создаются/удаляются объекты
- Регулярно делаем Analyze
- Каждое утро стартует процесс, который бежит по списку таблиц в базе и анализирует/вакуумит их по приоритету:
  1. не вакуумились/анализировались вообще
  2. вакуумились/анализировались давно

Данные в Greenplum – data vault

- Презентационный слой строим в Data Vault
  - Hub’ы, Link’и, Satellite’ы



  - Позволяет дробить ETL на небольшие задачи

Данные в Greenplum

- Большое количество Update’ов для SCD2 справочников
  - Избежать update’ов не удалось

account_rk	cash_atm_trxn_fee_amt	valid_from_dttm	valid_to_dttm
1	290.00000	2014-02-27 03:49:42.000000	2014-03-01 02:23:30.000000
1	20.00000	2014-03-01 02:23:31.000000	2014-03-18 23:28:38.000000
1	290.00000	2014-03-18 23:28:39.000000	2014-07-24 22:25:30.000000
1	20.00000	2014-07-24 22:25:31.000000	2014-07-25 22:15:45.000000
1	290.00000	2014-07-25 22:15:46.000000	5999-01-01 00:00:00.000000

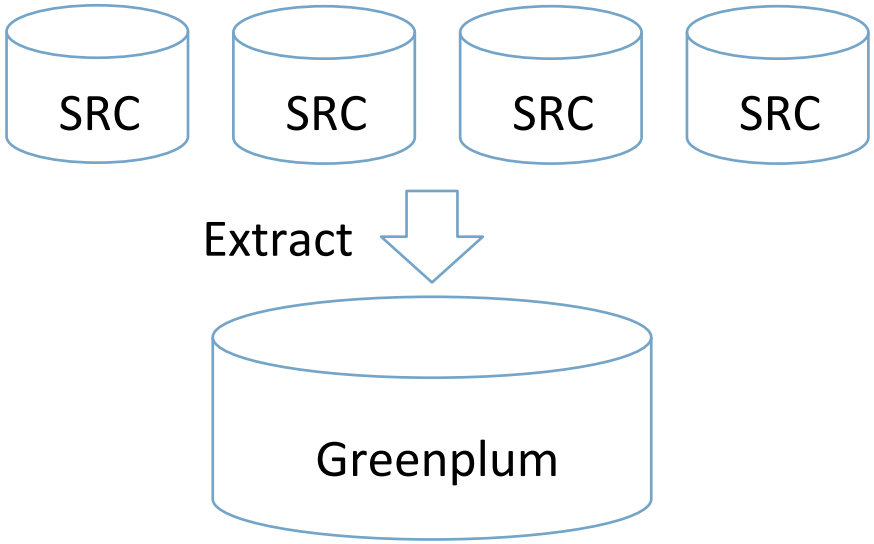
Признак актуальной версии

account_rk	cash_atm_trxn_fee_amt	valid_from_dttm
1	290.00000	2014-02-27 03:49:42.000000
1	20.00000	2014-03-01 02:23:31.000000
1	290.00000	2014-03-18 23:28:39.000000
1	20.00000	2014-07-24 22:25:31.000000
1	290.00000	2014-07-25 22:15:46.000000

Нужно вызывать оконные функции при каждом обращении к таблице чтобы найти актуальную версию!

- Greenplum хорошо справляется с таким объемом update’

Загрузка данных в Greenplum

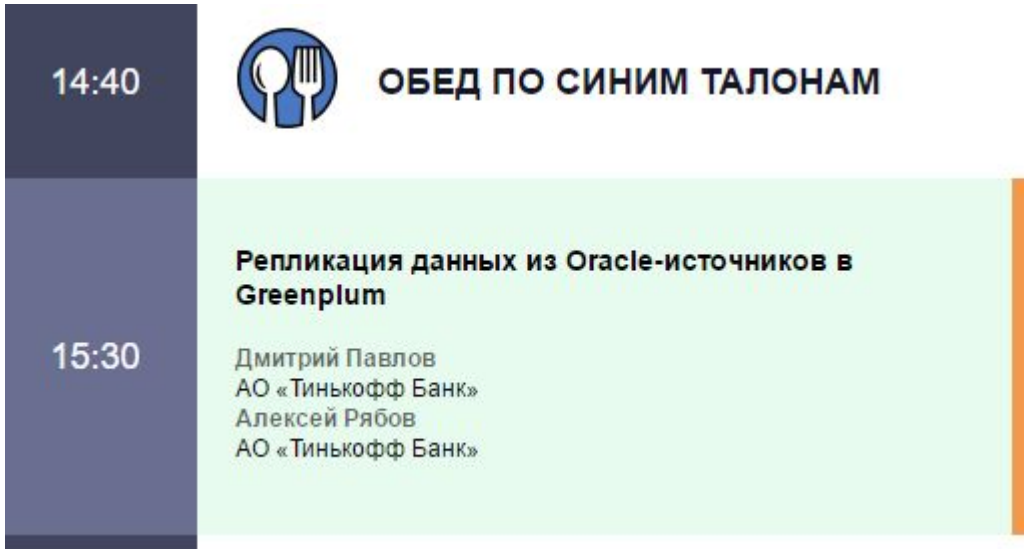


- Большой объем данных в источниках – сотни Tb
- Большое количество изменений в сутки – 1,5 миллиарда

- Загружаем только нужные данные (таблицы)
- Загружаем только изменившиеся данные
- Используем gpfdist – позволяет загружать с очень большой скоростью
- Загружаем новые изменения каждый час

Загрузка данных в Greenplum

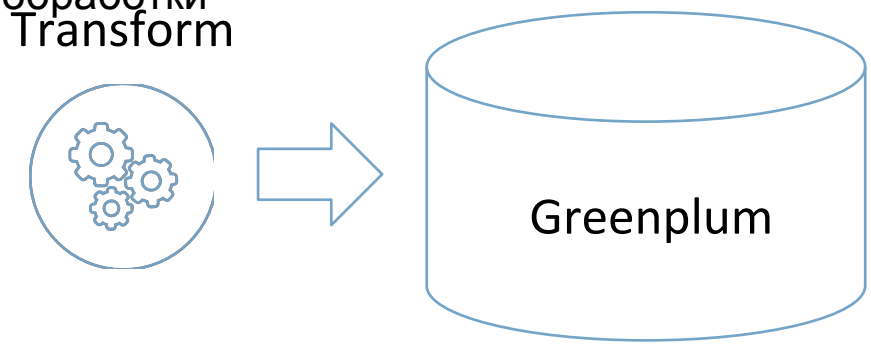
Подробнее завтра расскажут Алексей Рябов и Дмитрий Павлов



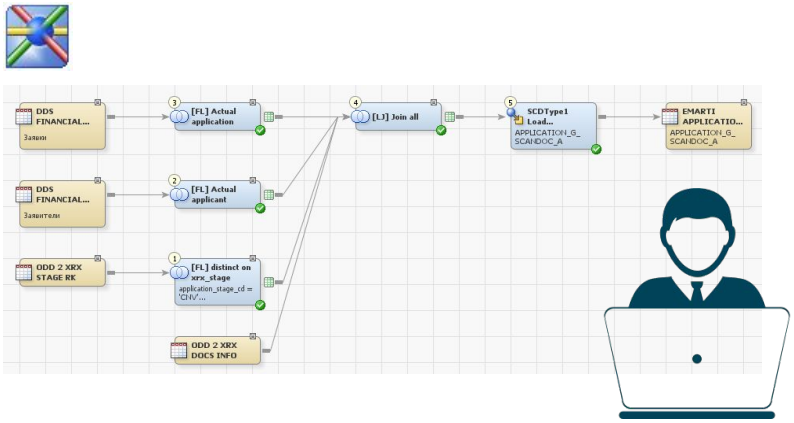
ETL в Greenplum

- Все исходные данные уже в Greenplum
- Большие объемы обрабатываемых данных
- Большое количество разработчиков
- Большое количество процессов
- Не всегда оптимальные планы запросов в GP
- Сложная логика обработки

- Все расчеты нужно делать внутри Greenplum
- Нужна ETL платформа
- Нужна ETL платформа с возможностью кастомизаци



ETL из коробки и Greenplum



Автогенерация

SQL,
SQL,
SQL,
…

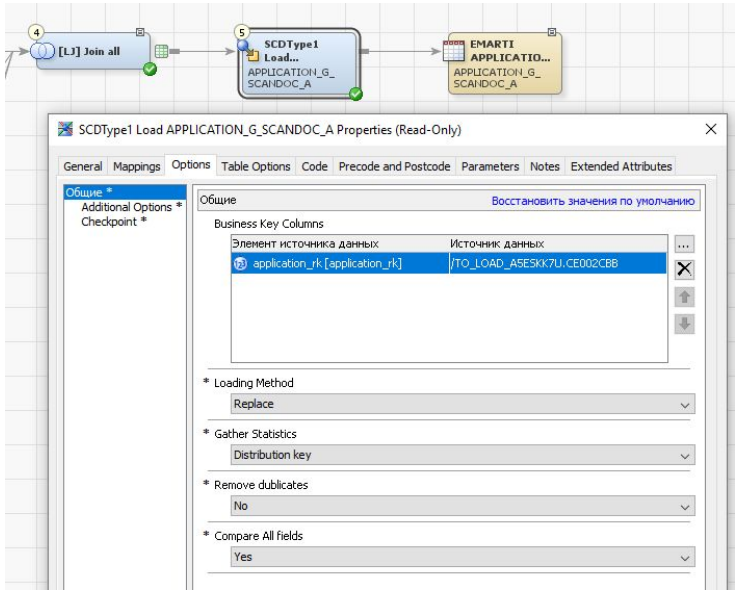
- Насколько эффективен сгенеренный SQL?
- Не будут ли мешать ETL-процессы друг другу?

Настраиваем ETL под Greenplum

- Для каждой ETL сессии создаем новую work схему в Greenplum для временных таблиц

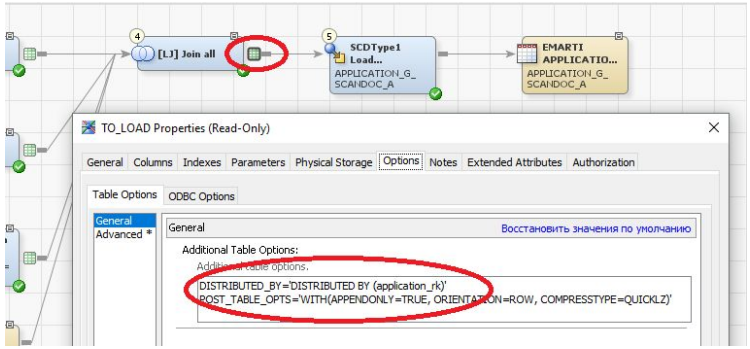
```
create schema work_F1A00000D7F1_dwhsys_srvdsetl01p_55281
```

- Создаем набор стандартных переиспользуемых компонент оптимизированных для Greenplum
  - Построение SCD таблиц
  - Инкрементальная загрузка
  - Генерация ключей
  - …



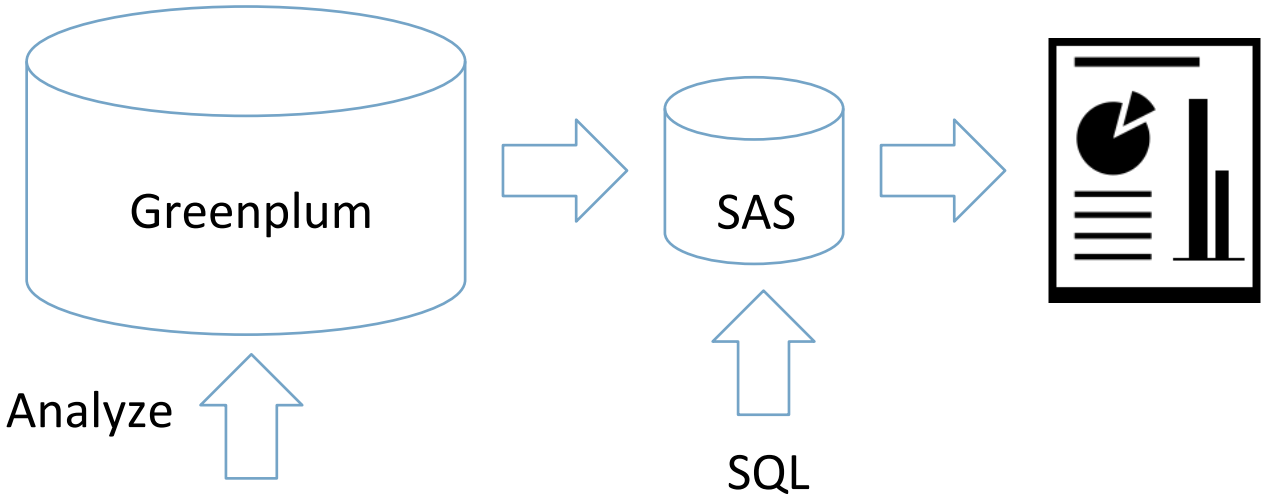
Разработка ETL в Greenplum

- Не забываем про правильную дистрибуцию промежуточных таблиц



- ORCA используем с осторожностью. Он пока оптимален лишь в некоторых случаях. Главный плюс использования - более продвинутая работа с партициями
- Оптимизаторы Greenplum не всегда строят оптимальный план. Нужно разбивать сложную логику (SQL) на небольшие части – так проще и оптимизатору и разработчику.

BI в Greenplum - External

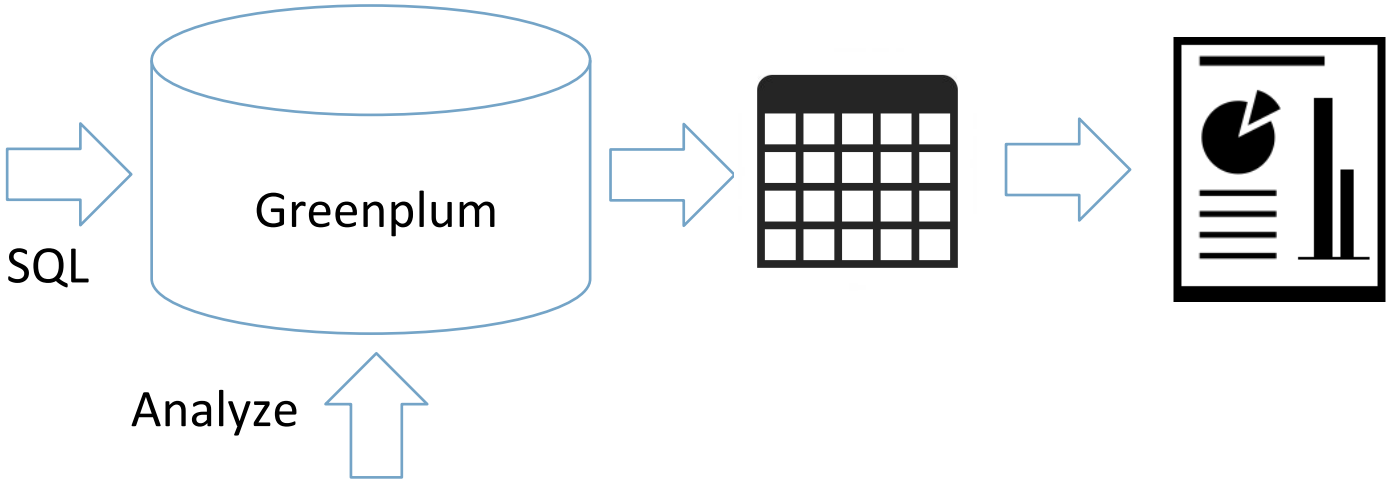


Два варианта аналитики

1

1. Выгрузить из Greenplum нужные данные во внешнюю систему
2. Обработать их во внешней системе
3. Визуализировать

BI в Greenplum – Push Down

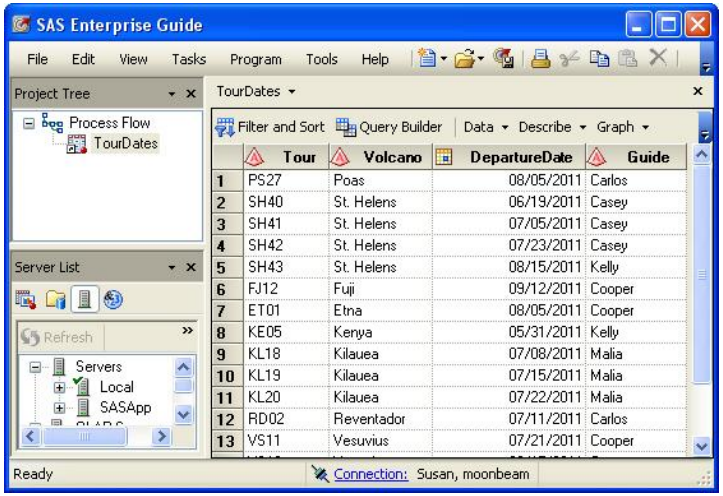


Два варианта аналитики

2

1. Сгенерить или написать правильный SQL для Greenplum
2. Выполнить SQL внутри Greenplum
3. Выгрузить получившийся small результат во внешнюю систему
4. Визуализировать

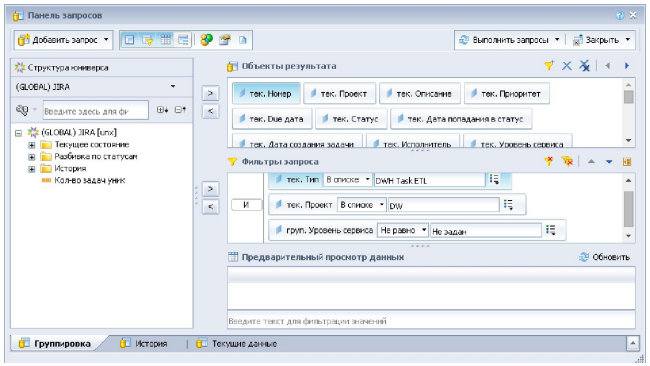
SAS Enterprize Guide



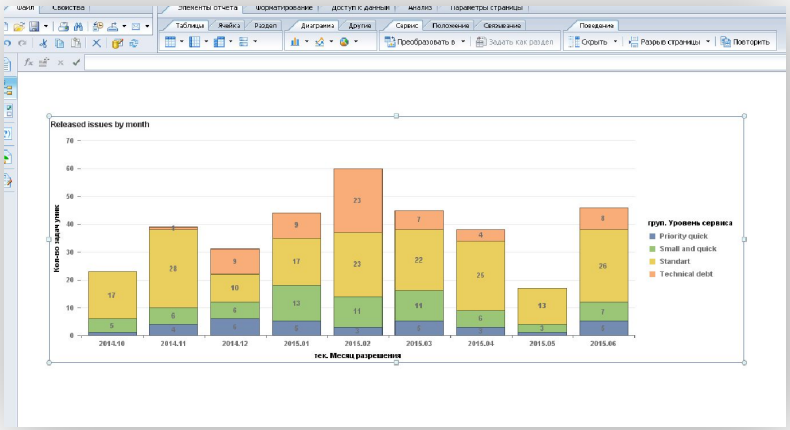
- Собственный драйвер для работы с Greenplum
- Может как External так и Push Down, но больше заточен под External
- Загружает в себя данные очень медленно, 50 Mb/сек
- В 80% случаев генерит не эффективный Push Down код
- Эффективный Push Down код – только написанный руками и/или с использованием специальных макросов

BI в Greenplum

SAP Business Objects

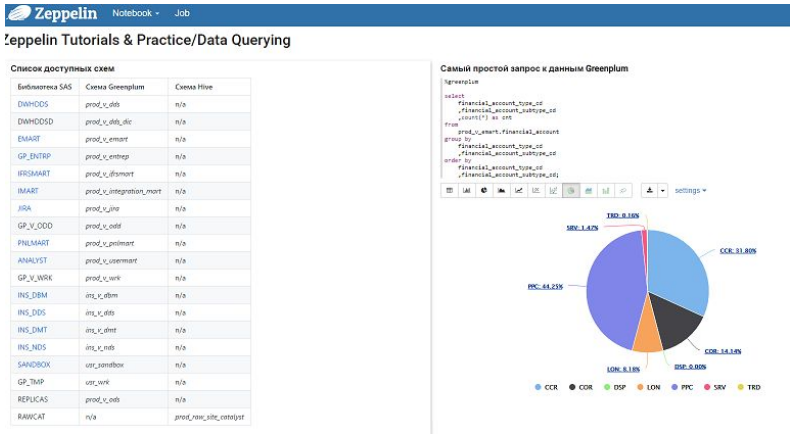


- Только Push Down режим
- Обычный JDBC, но это не проблема
- Автогенерация достойного эффективного кода
- Богатые возможности кастомизации



BI в Greenplum

Apache Zeppelin

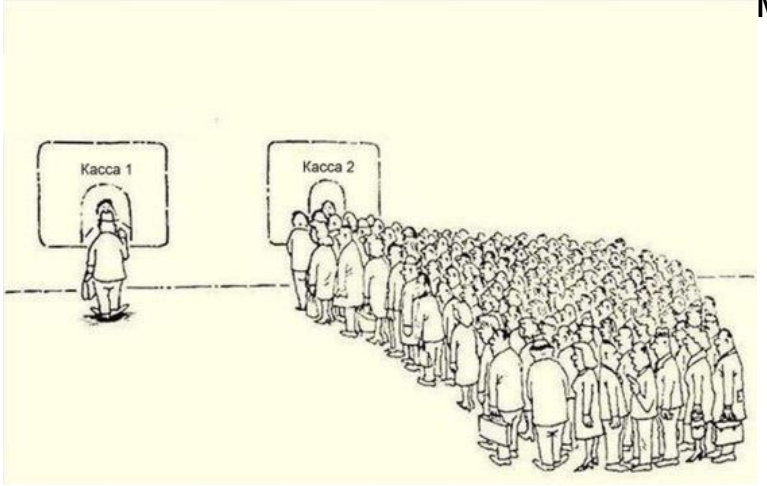


- Логика соединения с Greenplum зависит от интерпретатора
- Можно External и Push Down
- External требует дополнительной настройки интерпретаторов
- Требуется кастомизация для Enterprize-фич

BI в Greenplum – особенности работы в Greenplum

- Два подключения к Greenplum – один с встроенным оптимизатором, второй с ORCA
  - Эффективны на разных типах запросов
  - Более эффективный план выбираем эксперементально

- Настраиваем ресурсные очереди даже внутри одного инструмента
  - например, аналитическая и операционная отчетность имеют разные требования по доступности и разный уровень нагрузки на систему



BI в Greenplum – особенности работы в Greenplum

- Принудительно фильтруем пустые записи в ключах соединений, чтобы избежать перекоса при перераспределении данных
  - один из ключевых способов соединения таблиц - перераспределение по ключу соединения, все пустые значения при этом попадают на один сегмент

Счета	
account_id	**application_type_id**
1	null
2	1
3	2
4	null
…	…

Типы заявок	
application_type_id	**application_type_desc**
1	Интернет заявка
2	Мобильная заявка

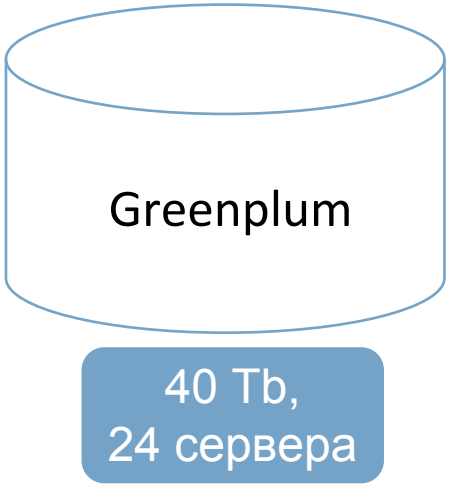
BI в Greenplum – особенности работы в Greenplum

- Денормализуем детальные данные с целью эффективного партиционирования
  - например, сущность «счет" может содержать атрибут "тип заявки", чтобы можно было по нему обратиться в нужную партицию с данными

Счета

account_id	application_type_id	application_type_desc
1	null	
2	1	Интернет заявка
3	2	Мобильная заявка
4	null	
…	…	

Greenplum и Hadoop



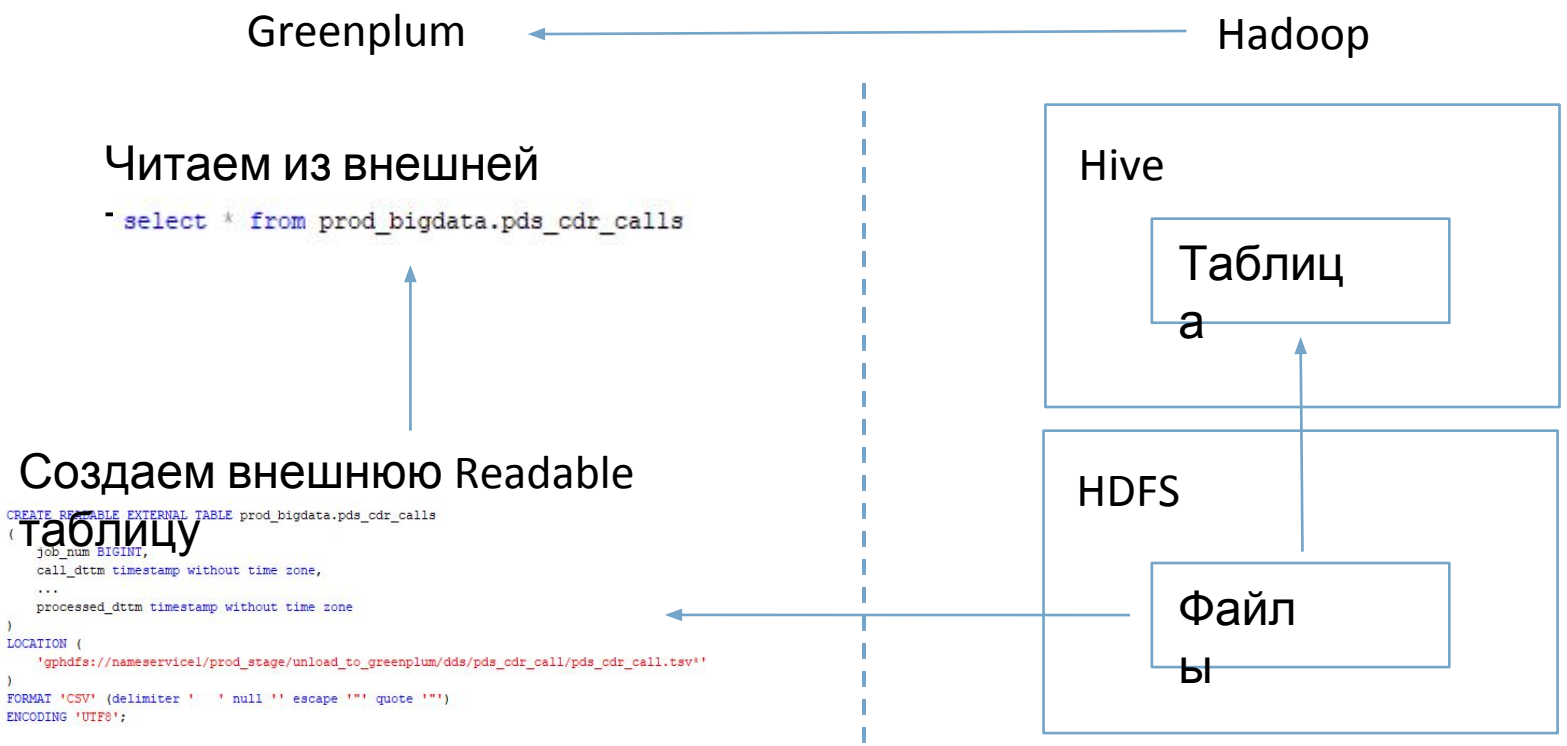
30
серверов

- В Hadoop храним сырые логи, немного их очищаем и стуктуризируем (BigData ETL)
- В Hadoop позволяем бизнес-пользователям анализировать данные
- Сырые логи - тоже источник для DWH

- Нужно уметь передавать данные из Hadoop в Greenplum
- И наоборот тоже надо уметь

Из Hadoop в Greenplum

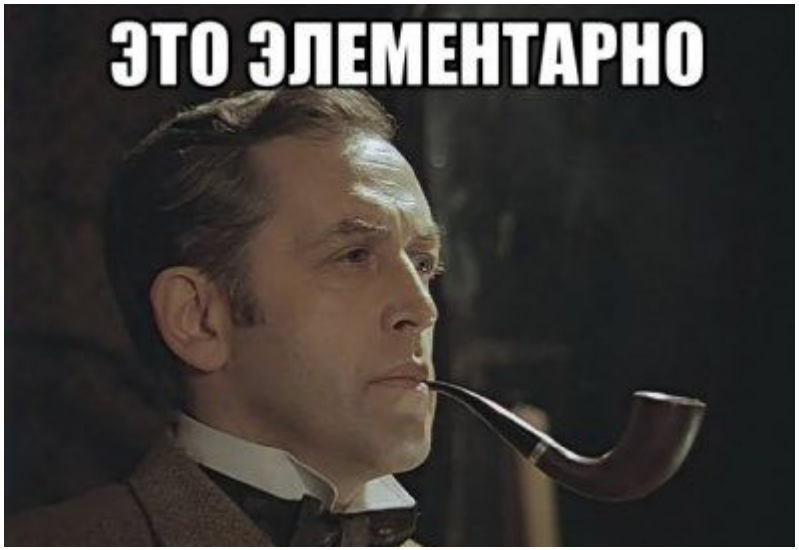
Используем gphdfs



Из Greenplum в Hadoop

Все то же самое, только создаем внешнюю **Writable** таблицу

И не читаем, а пишем



Практика использования gphdfs

- Простая и очевидная интеграция для разработчи
- Легко встроить в любой ETL инструмент
- Достаточная скорость передачи данных. 500 Mb в секунду
- Позволяет читать Avro, CSV, Text, Parquet

- Нет интеграции с Hive
  - Не позволяет читать структуру партиций Hive
- Не позволяет читать ORC

Greenplum – проблемы и решения

- Нет DR из коробки

  Делаем свой DR

- Недостаточный и неудобный стандартный мониторинг (command cente

  Делаем свой мониторинг Linux, Bash, Graphana, Graphite

- Часто выходят диски из строя

  Анализируем ситуацию. Снижаем нагрузку на диски.

- Проблема устаревания поколений

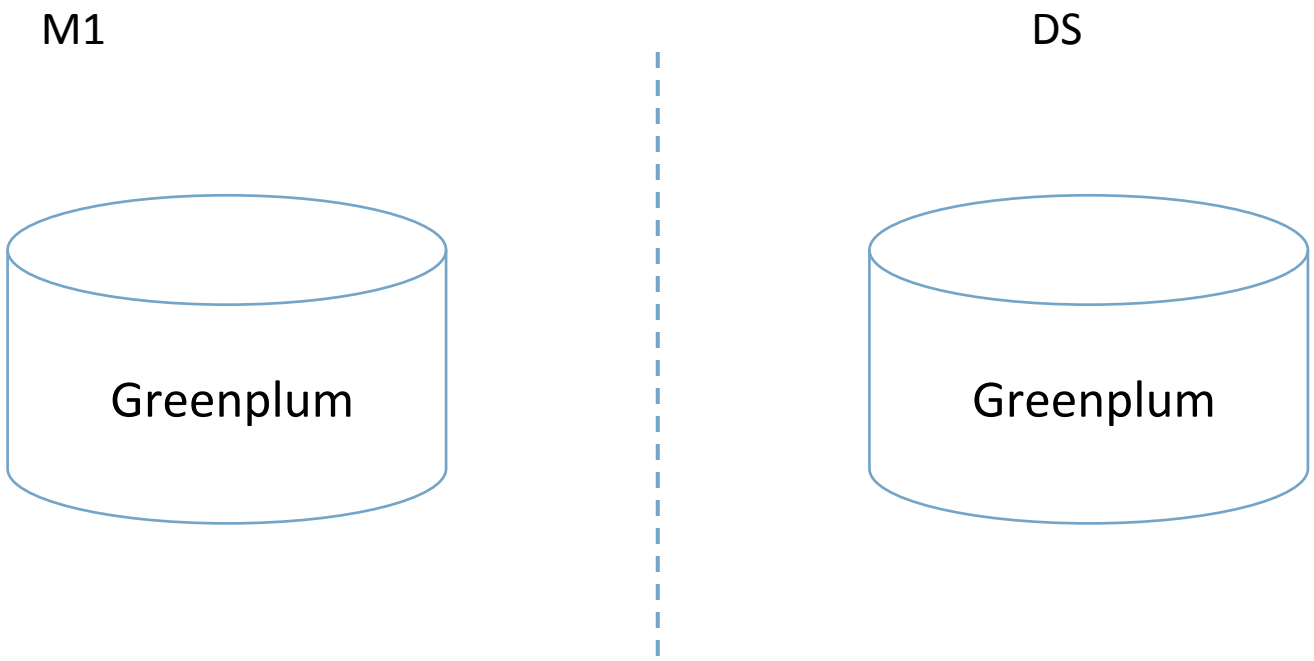
12 серверов.
Поколение 1.
Куплены в 2015.

6 серверов.
Поколение 2.
Куплены в 2017.

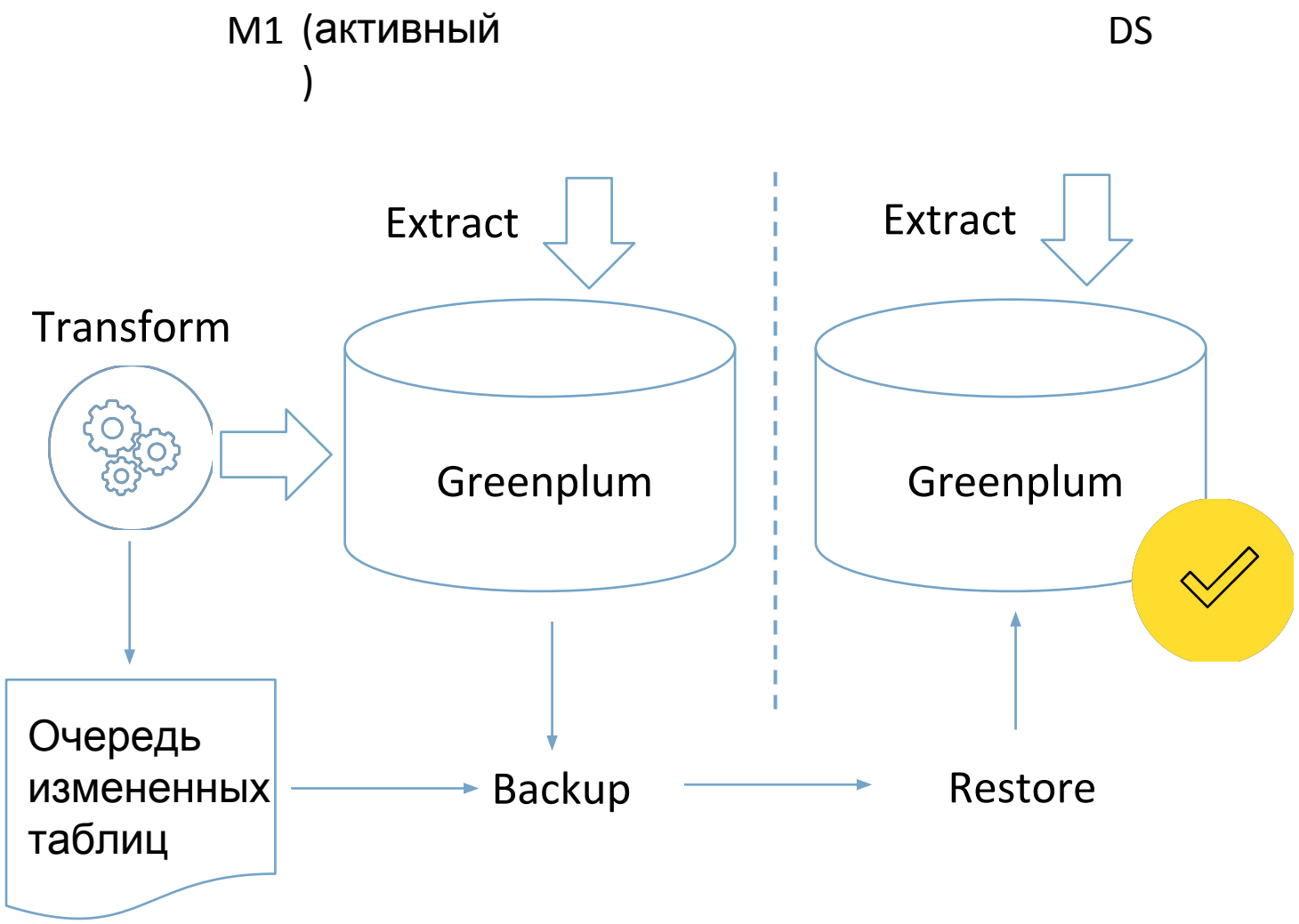
У нас все еще сервера поколения 1

Собственный DR

- Два Дата Центра
- Два одинаковых Greenplum
- В случае выхода из строя всего ДЦ (или всего кластера Greenplum) мы должны подняться во втором ДЦ
- На восстановление 1 час

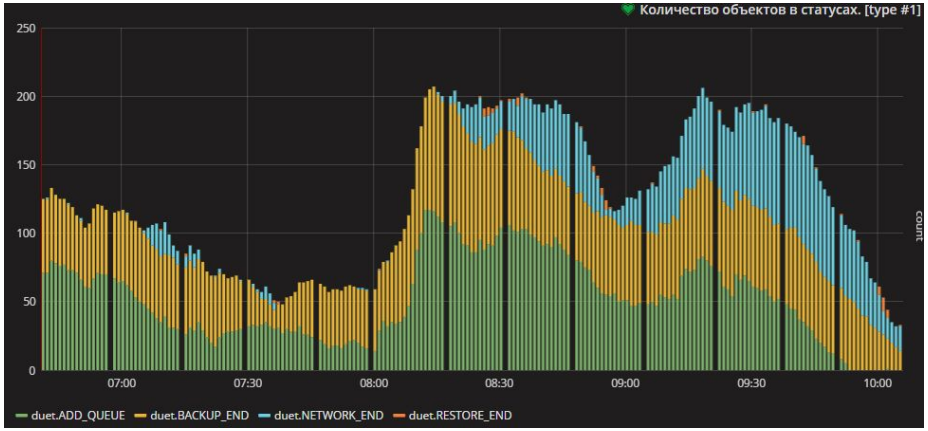


Собственный DR



Собственный DR

- Подерживает DR Greenplum в актуальном состоянии с задержкой в пиках до 3-х часов
- 30 Tb несжатых данных в сутки
- 2700 таблиц в сутки
- Нагрузка на сеть между Дата Центрами высокая – 4 гбит/сек в пиках до 10 гбит/сек. Выделенный канал на DR.



- Наличие DR позволяет выносить BI нагрузку на Greenplum в резервном Дата Центре

Альтернативы Greenplum

- В 2016 году искали альтернативную базу (In-Memory) для быстрой отчетности
- https://habrahabr.ru/company/tinkoff/blog/310620/
- Проект заморожен до конца года из-за того что не удалось решить проблему быстрого копирования данных

- Но был сделан ряд выводов
  - Exasol – **очень** достойный представитель In-memory MPP
  - Cloudera Impala пока не дотягивает до лидеров, но это временно. Через 1-2 года картина вероятно поменяется
  - Greenplum на RAM – тоже достойный вариант In-Memory базы данных
  - Все остальные In-Memory MPP крайне слабы

Что дальше?

- Сейчас 24 сервера. Будет 32. Как дальше масштабироваться?
- Pivotal? EMC? Dell? У семи нянек Greenplum «без глаза». На наш взгляд по сравнению с конкурентами Greenplum слабо развивается.
- Современные технологии и Greenplum? SSD? In-memory?
- Open Source? Что делать с фактом попадания GP в OS?

Open Source

- Есть надежда, что Greenplum будет активнее развиваться
- Можно и нужно участвовать самим в развитии.

  Разрабатывать нужный именно тебе функционал и править критичные именно для тебя ошибки

  Тратим $$$ на собственную экспертизу – капитальные вложения
- Есть опыт участия в доработке под себя Apache Zeppelin
- Вариант с Pull Request в Master выгоднее, чем собственный Branch
- Обязательно напрямую или опосредованно участвовать в PMC

Тинькофф

Дальше действовать будем мы!

Tinkoff.ru