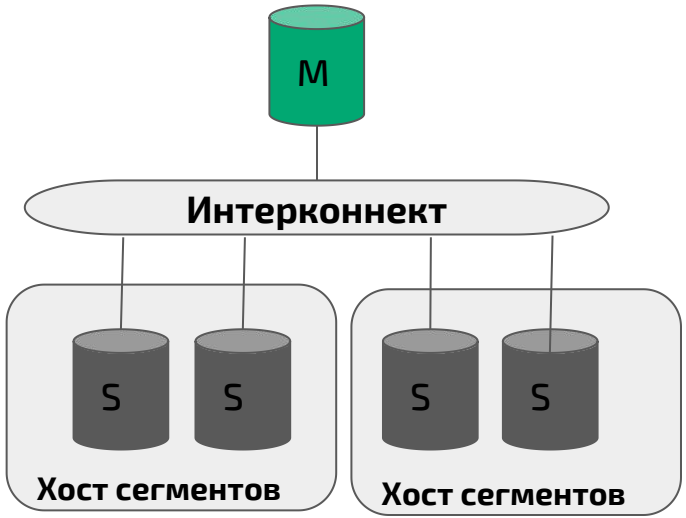
Средства Greenplum для работы с внешними данными

Урсегов Дмитрий
Руководитель разработки Arenadata

Что такое Greenplum

- PostgreSQL (6.x - 9.4, 7.x - 9.6)
- Аналитическая нагрузка
  - Распределённая система (MPP)
  - Колоночное хранение
  - Пропускная способность,
    а не время отклика

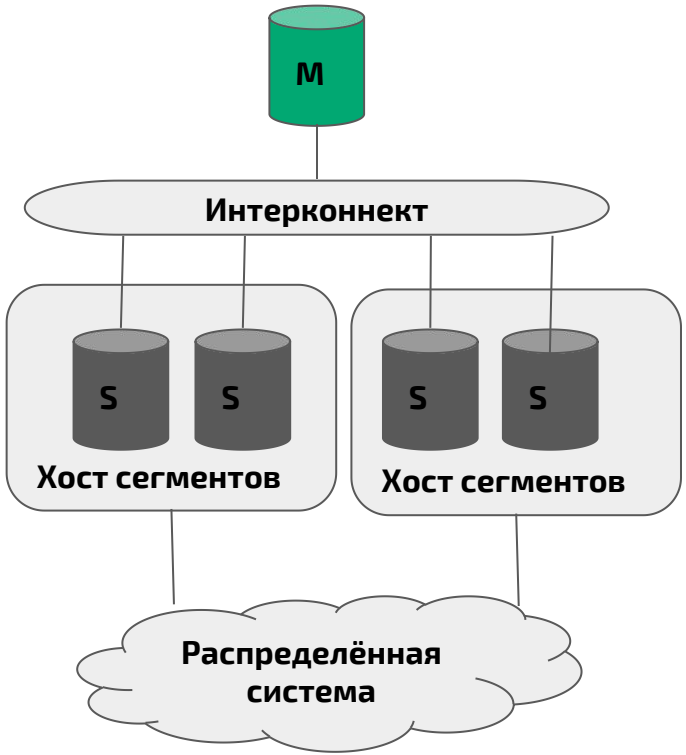
- Роли:
  - Сегменты - работают со своими данными
  - Мастер - контролирует их работу и собирает результаты



Задача загрузки и выгрузки данных

Большое количество данных и необходимость параллельной работы

- **Общая задача:** работа с внешними данными, как с локальными таблицами
- **Частная задача:** выгрузить/загрузить данные целиком решается общим способом через Insert into t1 select * from t2;

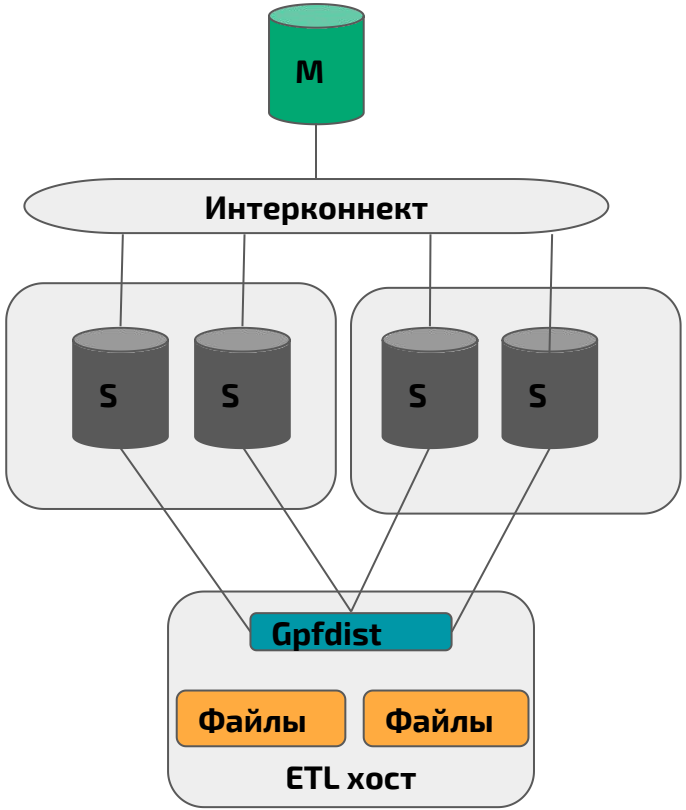


Основные подходы

- Фреймворки external tables (6.x) / foreign tables (6.x / 7.x)
  - Реализовать свой протокол
  - Gpfdist
  - PXF
    - Модули для Hadoop и других систем
    - Реализовать свой модуль
- Специализированное расширение

Работа с файлами через gpfdist

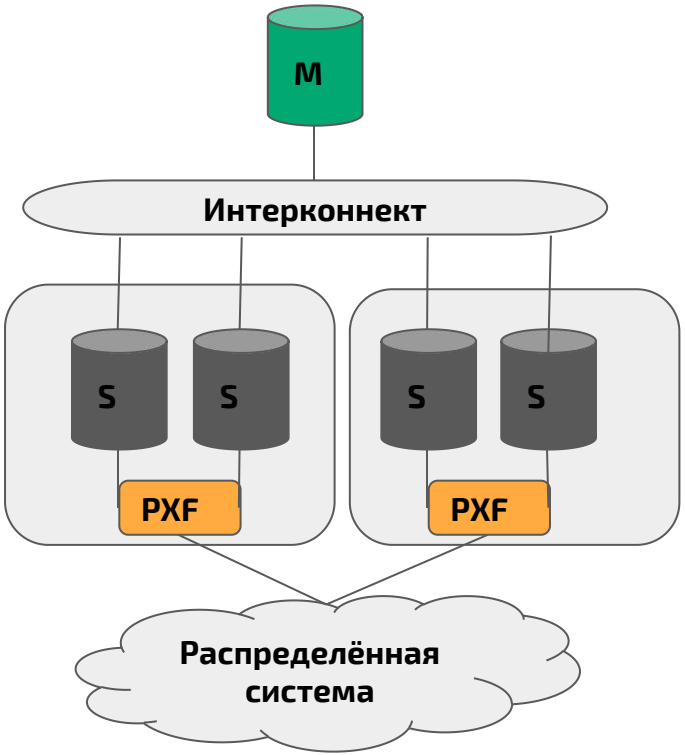
- Веб-сервер
- Работа с текстовыми файлами



Platform Extension Framework

- Отдельный Java сервис
- Java API к внешним системам (Hadoop и другие)
- Позволяет расширять функциональность через модули

- Для каждого сегмента свой поток обработки
- Поддержка filter pushdown и column projection

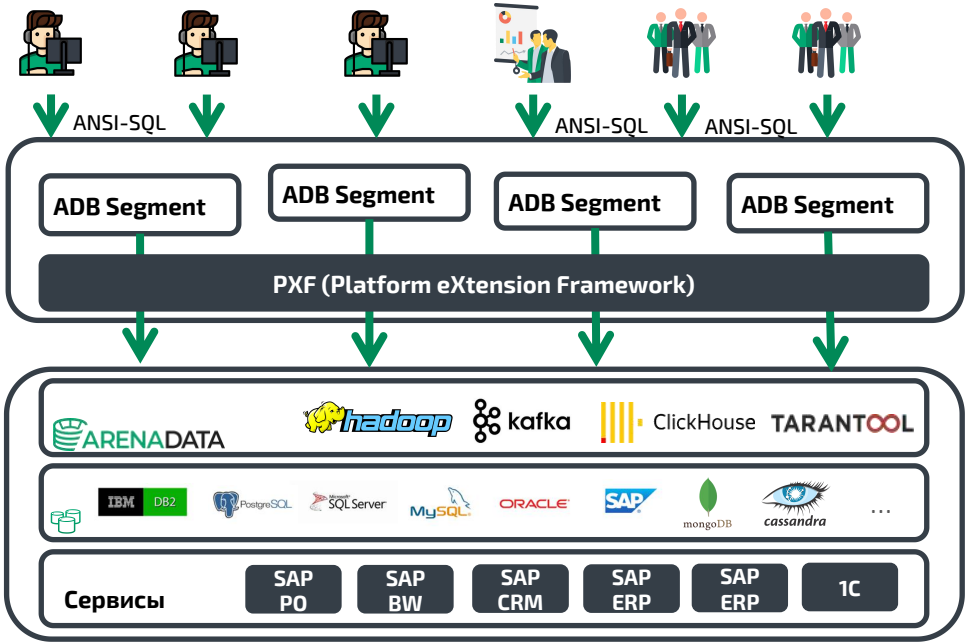


Поддержка FDW в Greenplum

- **Ограничения Greenplum 6.x**
  - Чтение со всех сегментов, запись только через master
  - Поддерживается только планировщик postgres
  - Стандартные расширения пока используют фреймворк внешних таблиц
- **Полная поддержка Greenplum 7.x**
  - От внешних таблиц останется только интерфейс для обратной совместимости
  - Стандартные расширения перенесут на интерфейс FDW, в т.ч. Gpfdist и PXF

Специализированное расширение

- Логика на внешних таблицах (штатная трансформация данных)
- Дополнительный процесс (PXF, streaming server)
- Подключение к сегментам напрямую



Пример: коннектор Greenplum->ClickHouse

- **Greenplum предназначен для построения больших хранилищ данных**
  - + Поддержка соединений больших таблиц
  - + Развитый синтаксис ANSI SQL
  - + Распределённые транзакции
  - +/- Планировщик с оценкой стоимости плана
  - - Движок, работающий со строчками
- **ClickHouse - доступ к построенным широким таблицам с макс. скоростью**
  - + Очень быстрый колоночный движок
  - - Для соединений данные таблиц и результат должны помещаться в память
  - - Отсутствие транзакций

Разработка коннектора

1. Архитектура
2. Производительность
3. Консистентность

Архитектура

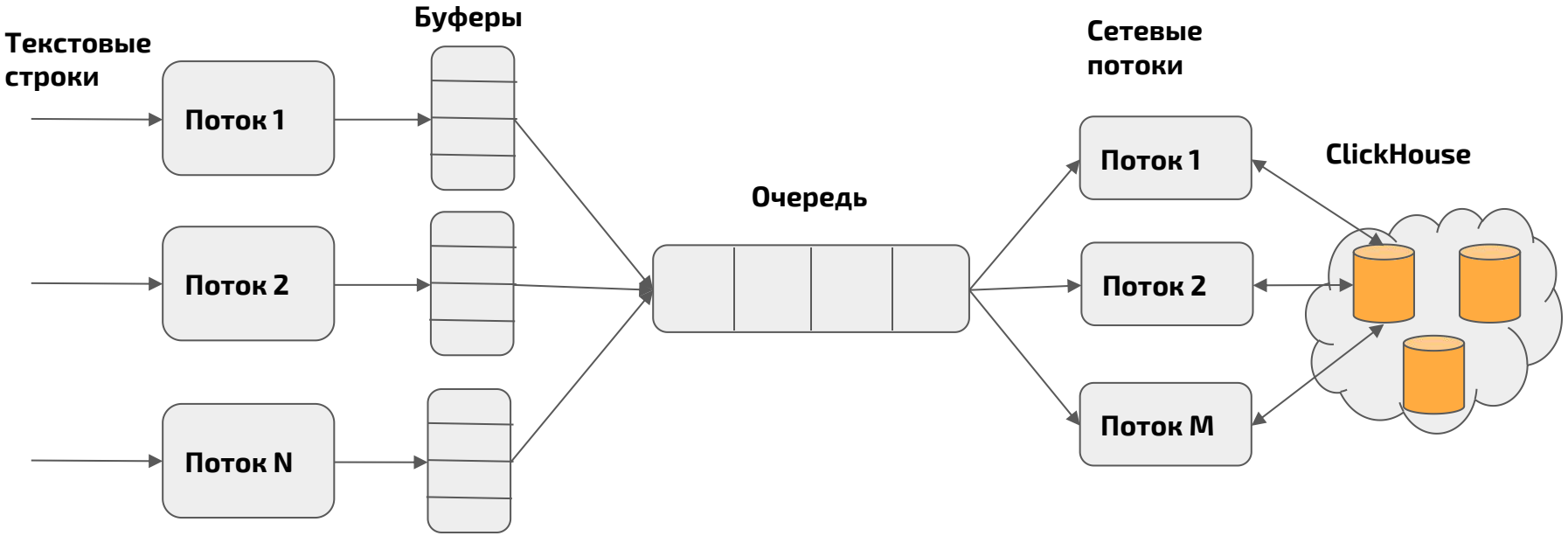
Особенности Clickhouse:

- Вставка большими блоками с ограниченной частотой и параллельностью запросов
- Скорость вставки до 200МБ/с

Работа с сегментами напрямую (свой протокол) - нет возможности синхронизировать отправку данных.

Модуль PXF позволяет регулировать поток, работая с данными всех сегментов в одном процессе.

Архитектура



Скорость

Архитектура позволяет смоделировать производительность

Реальность может отличаться:

- Ошибки конфигурации (top, nethogs)
- Конвертация/копирование данных (perf, async-profiler, flamegraphs)

После оптимизации:

- Скорость 15 Мб/с на сегмент
- Накладные расходы модуля PXF составляют 20%
- Что дальше? Shared memory, arrow

Консистентность

Двух распределённых систем

- Нет распределённых транзакций для external/foreign tables
- Нет транзакций в Clickhouse ;)

Эмуляция транзакций через расширение:

1. Копирование в промежуточный слой с аналогичной топологией
2. Проверка на совпадение по количеству строк
3. Переключение кусков данных средствами ClickHouse без копирования

Выводы

- Стоит учитывать переход на фреймворк FDW в Greenplum 7.x
- Модуль PXF или отдельный процесс для внешних сервисов со сложным API
- Скорость 15 Мб/с на сегмент и накладные расходы модуля PXF составляют 20%
- Дальнейшая оптимизация требует большой работы
- Можно получить почти полноценные транзакции для внешних данных, если очень хочется ;)

Вопросы?

Оцените потенциал Big Data и Open Source вместе с Arenadata
Скачайте бесплатно на **store.arenadata.io**