Pivotal®

Решение средствами Pivotal HAWQ задачи COPM

А.Ермаков – Pivotal

Pivotal

Платформа Pivotal

Advanced Analytics

Pivotal Greenplum Database

Pivotal HAWQ

Apps at Scale

Pivotal GemFire

Redis

Rabbit MQ

Pivotal Labs & Data Science Labs

Pivotal Cloud Foundry

Data Processing

Spring XD

Spark

Pivotal HD & Open Data Platform

Commodity Hardware

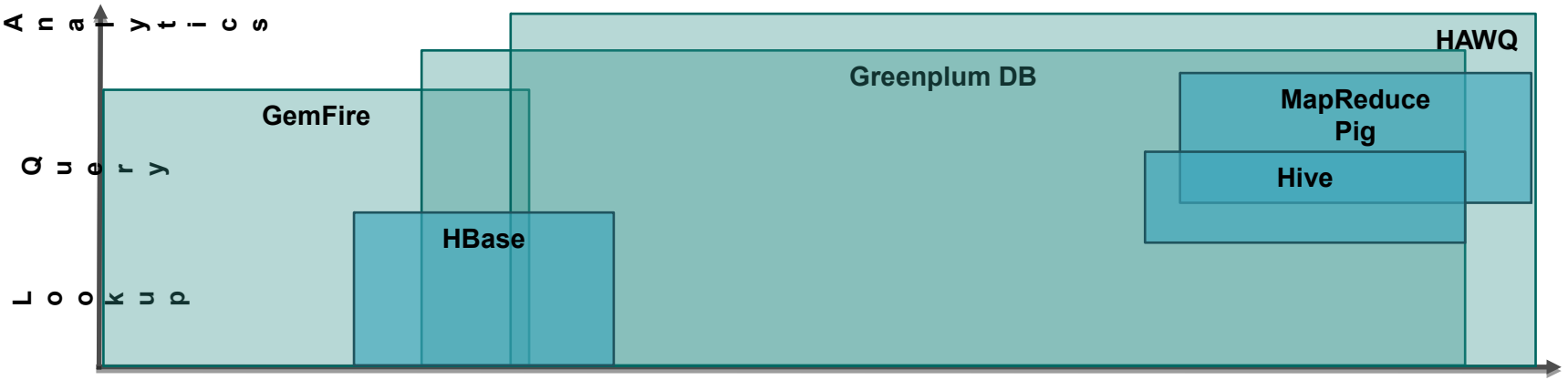
Appliance

Hybrid Cloud

Pivotal Cloud Foundry

Pivotal

Data Access



Real Time

- NoSQL Object Store
- Transaction Support

GemFire

Interactive

- MPP DB
- Polymorphic Data Storage

Greenplum

- SQL on Hadoop
- Query Optimizer

HAWQ

Batch

- Real-time looking
- Simple data queries

Hbase

Pivotal

СОРМ (Постановка задачи)

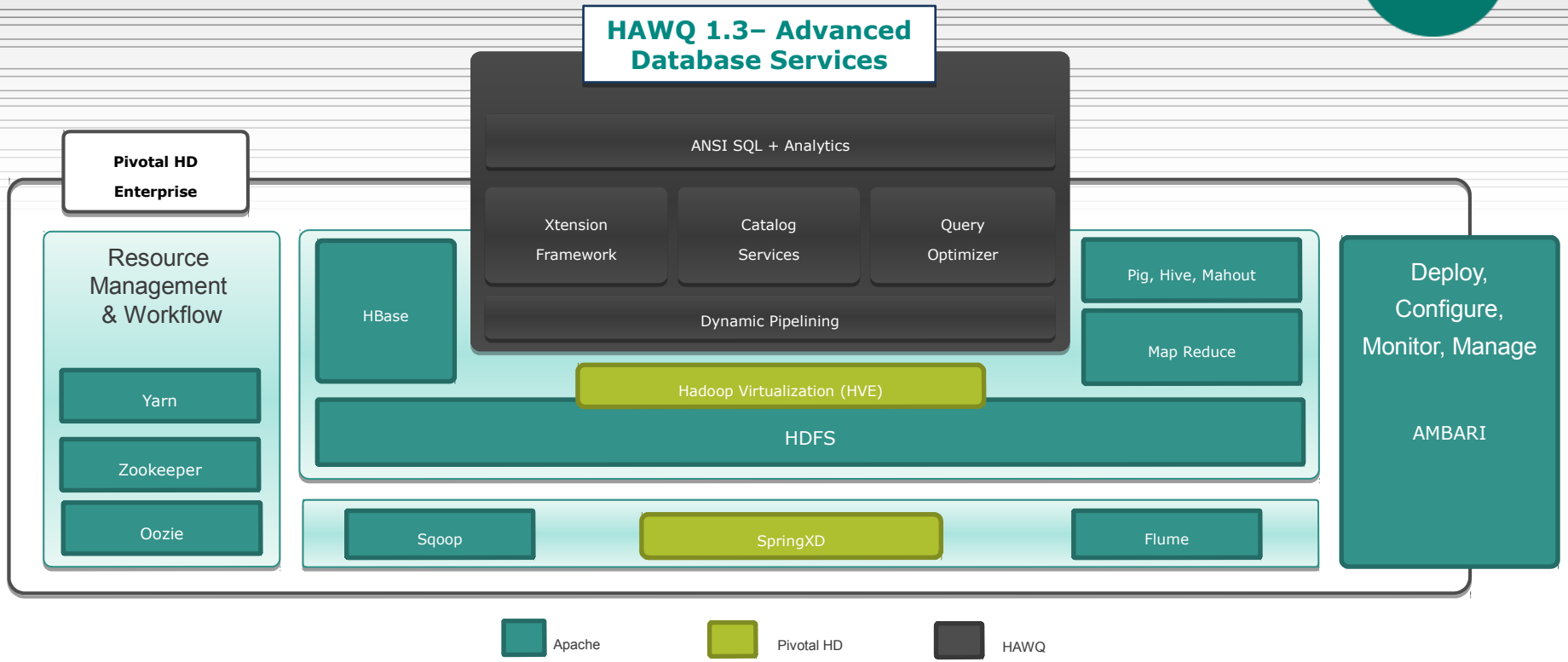
СОРМ-3 — система для обеспечения долгосрочного хранения и оперативного доступа к данным об абонентах оператора и оказанных им услугах связи.

Результатом работы системы является структурированный ответ, содержащий информацию об абоненте, номере телефона, звонках, посещенных сайтах, сессиях, прокси и др. по одному или нескольким параметрам.

Почему Hadoop?

- Экономическая эффективность (TCO, ROI)
- Производительность и масштабируемость
- Большие объёмы данных
- Хранение данных в исходном формате
- Возможность сжатия данных (zlib, quicklz, gzip)
- Устойчивость к отказам приложений
- Прозрачная масштабируемость приложений

Pivotal HD 3.0

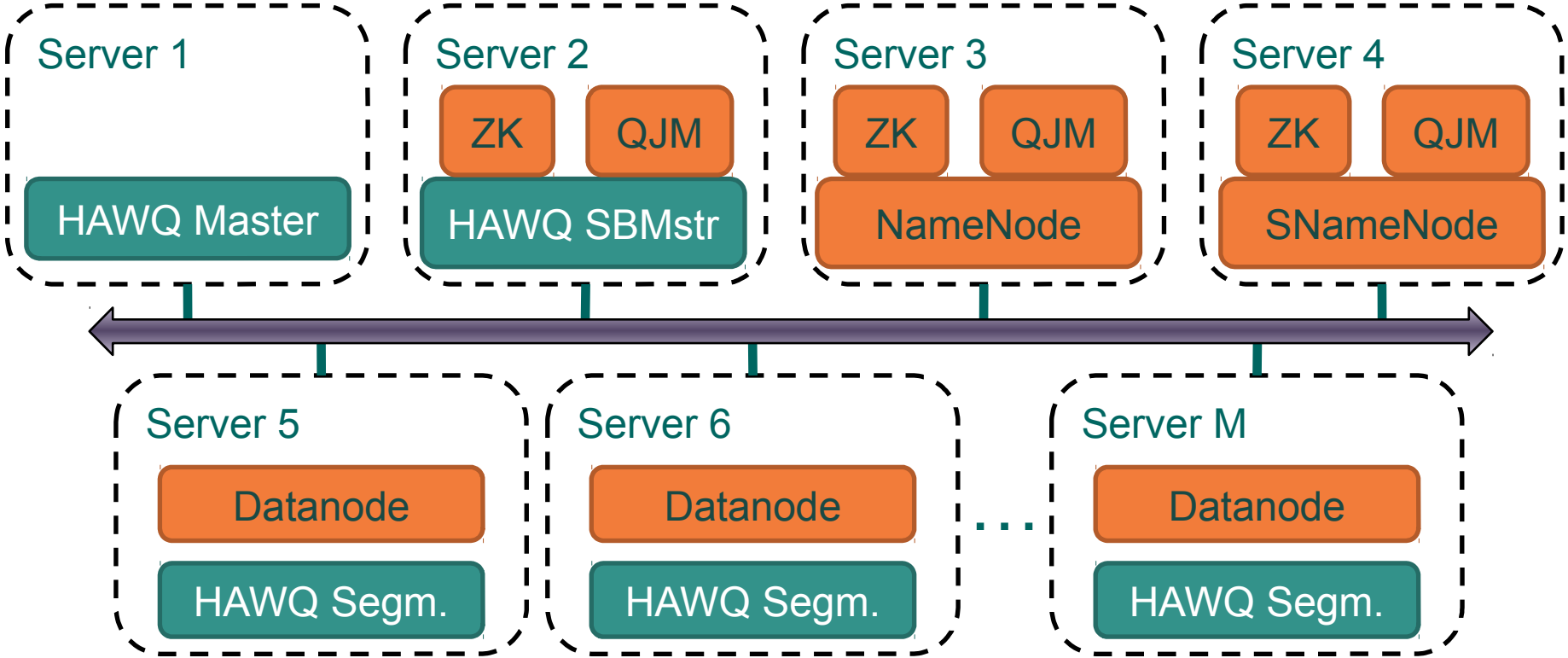


HAWQ

- Полная поддержка ANSI SQL
- Параллельная обработка запросов
- Лучший в классе оптимизатор
- Горизонтальное масштабирование
- Поддержка неструктурированных файлов
- Углубленная аналитика (MADLib)
- Поддержка шифрования данных и аутентификации пользователей

Pivotal

Компоненты Pivotal HAWQ



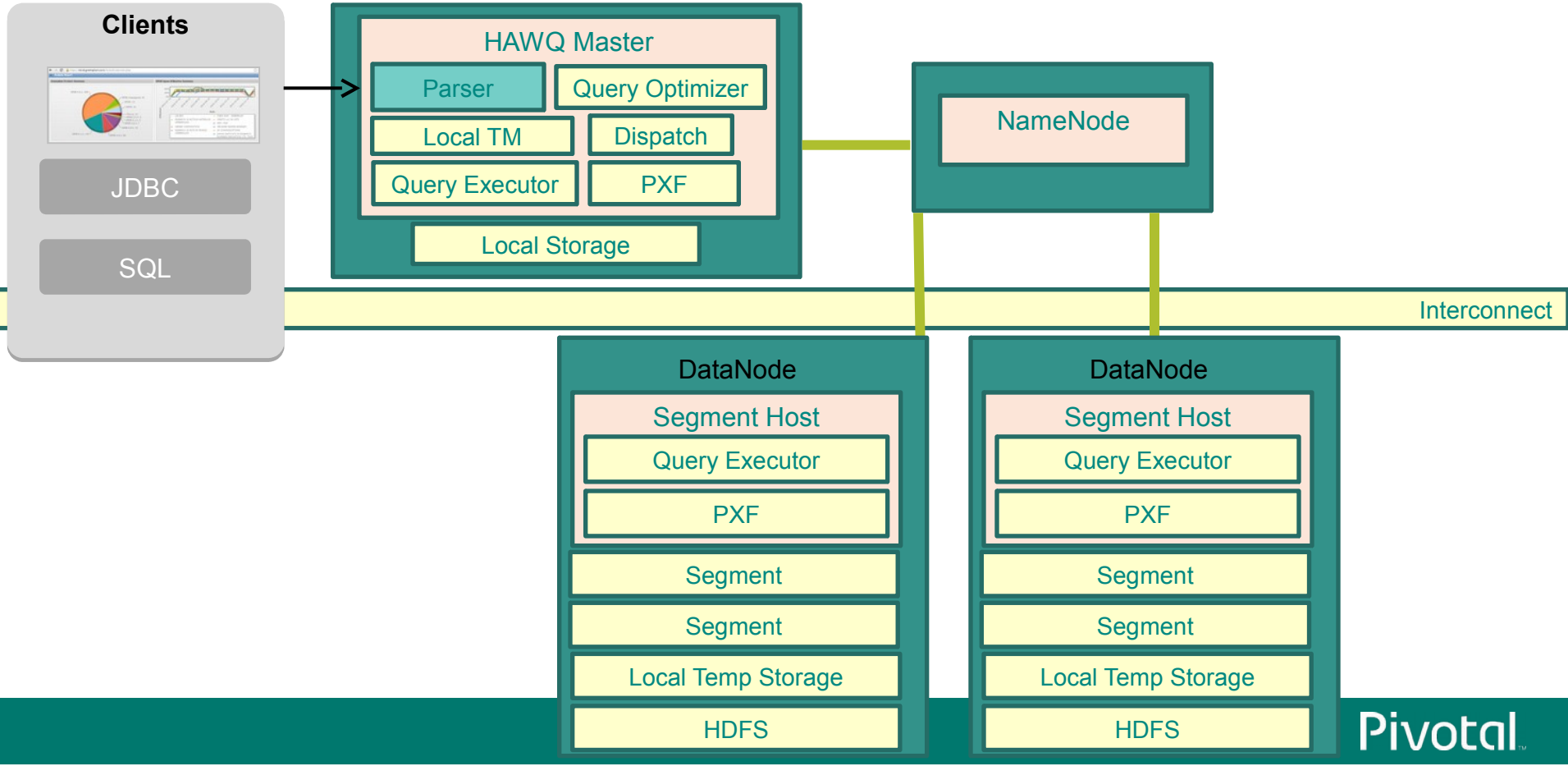
Компоненты Pivotal HAWQ

- И Master, и Segment'ы являются модифицированными процессами postgres
- При открытии подключения к Master-серверу – форк postmaser для работы с сессией
- При отправке процесса на выполнение – форк postmaster на сегментах
- План выполнения запроса разделяется на независимые блоки, slice, каждый из которых является отдельным процессом на сегменте

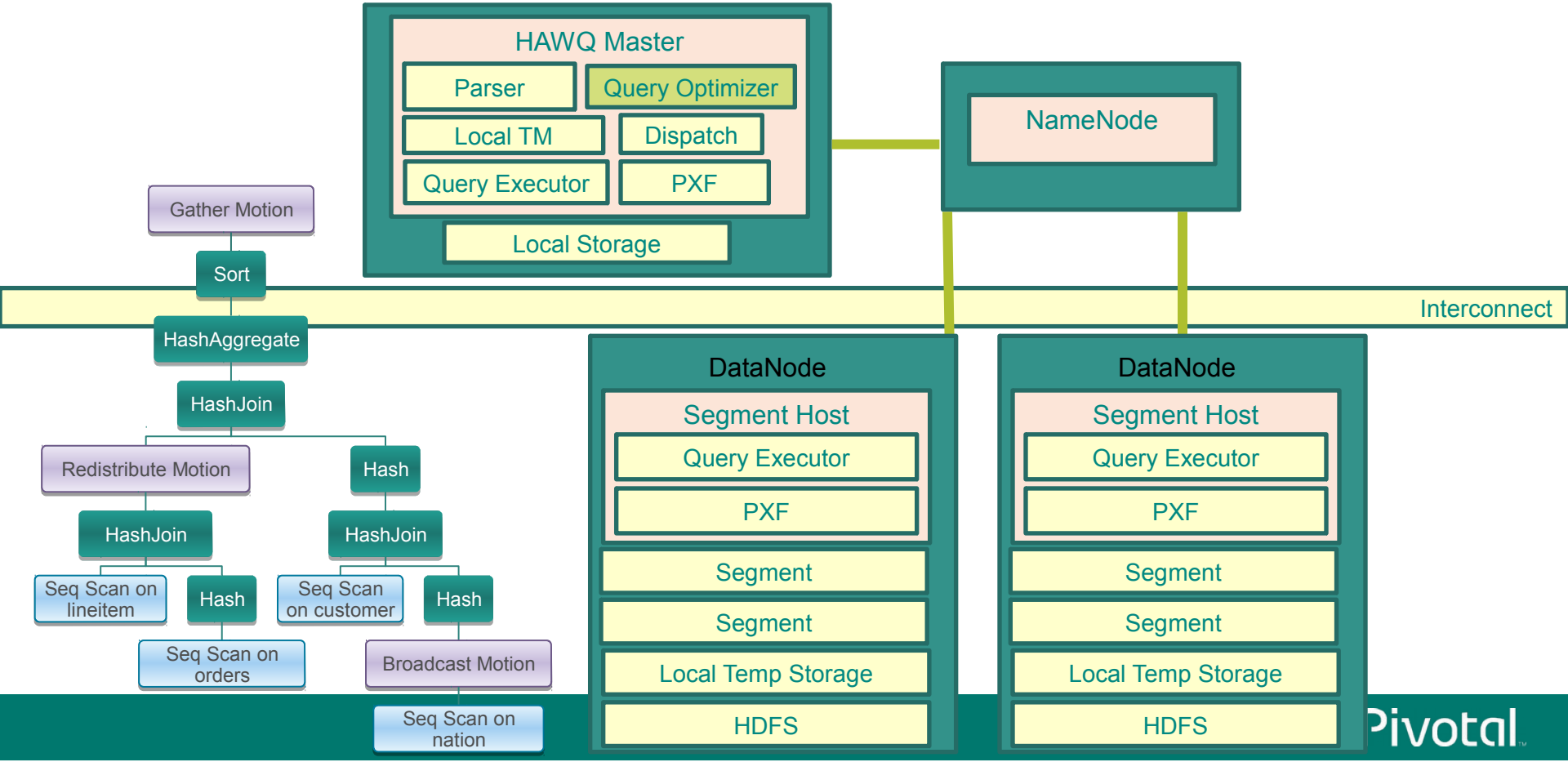
Компоненты Pivotal HAWQ

- Таблицы могут храниться как:
  - Row-oriented таблицы (сжатие quicklz, zlib)
  - Column-oriented таблицы (сжатие quicklz, zlib, rle)
  - Parquet таблицы
- Каждый сегмент имеет отдельную директорию на HDFS и отдельно хранит свой шард данных
- При поколоночном хранении каждый столбец предствляет собой 1 файл
- Parquet позволяет хранить таблицу по столбцам и при этом не нагружать NameNode множеством файлов

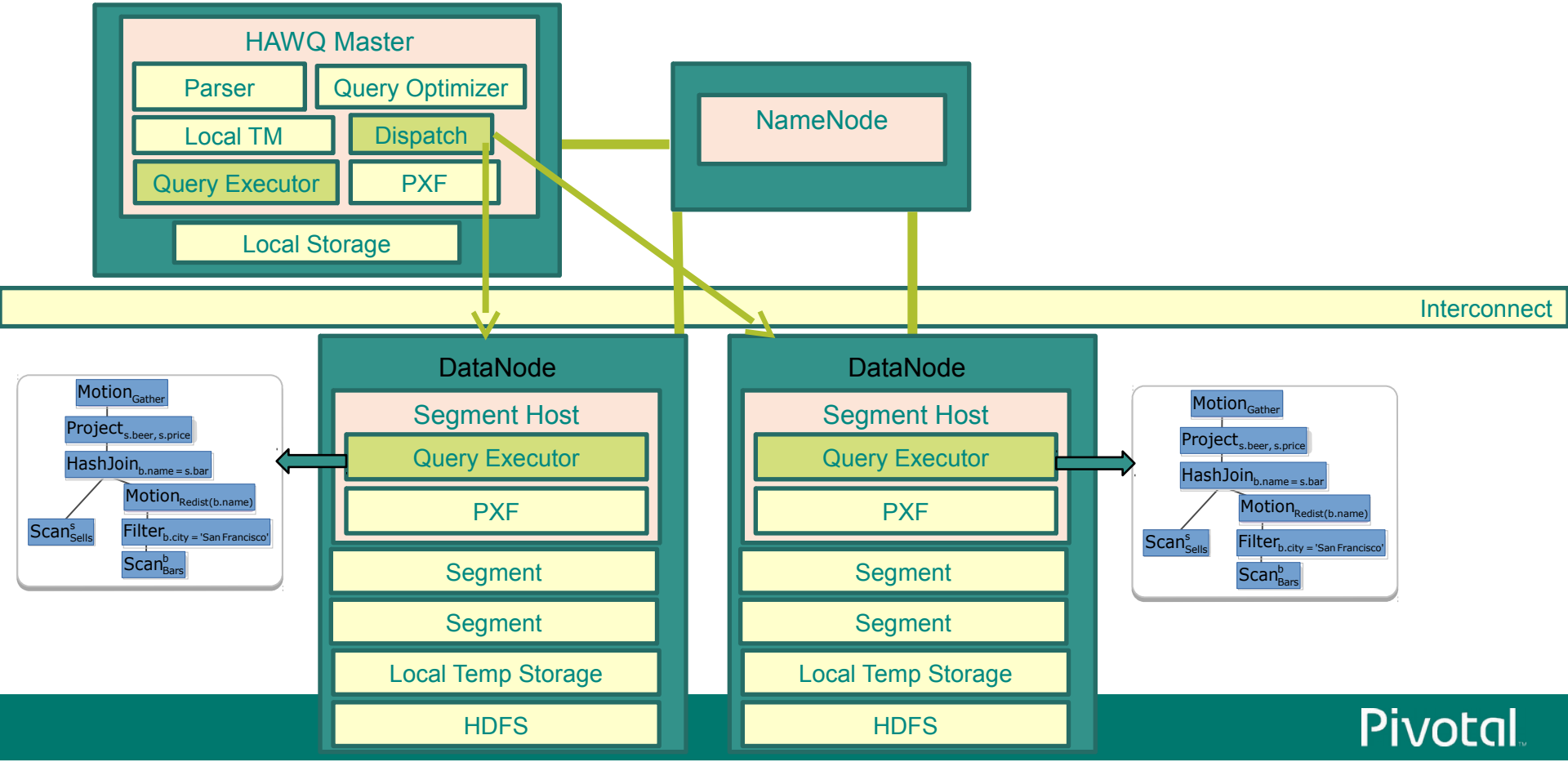
Выполнение запросов в HAWQ



Выполнение запросов в HAWQ

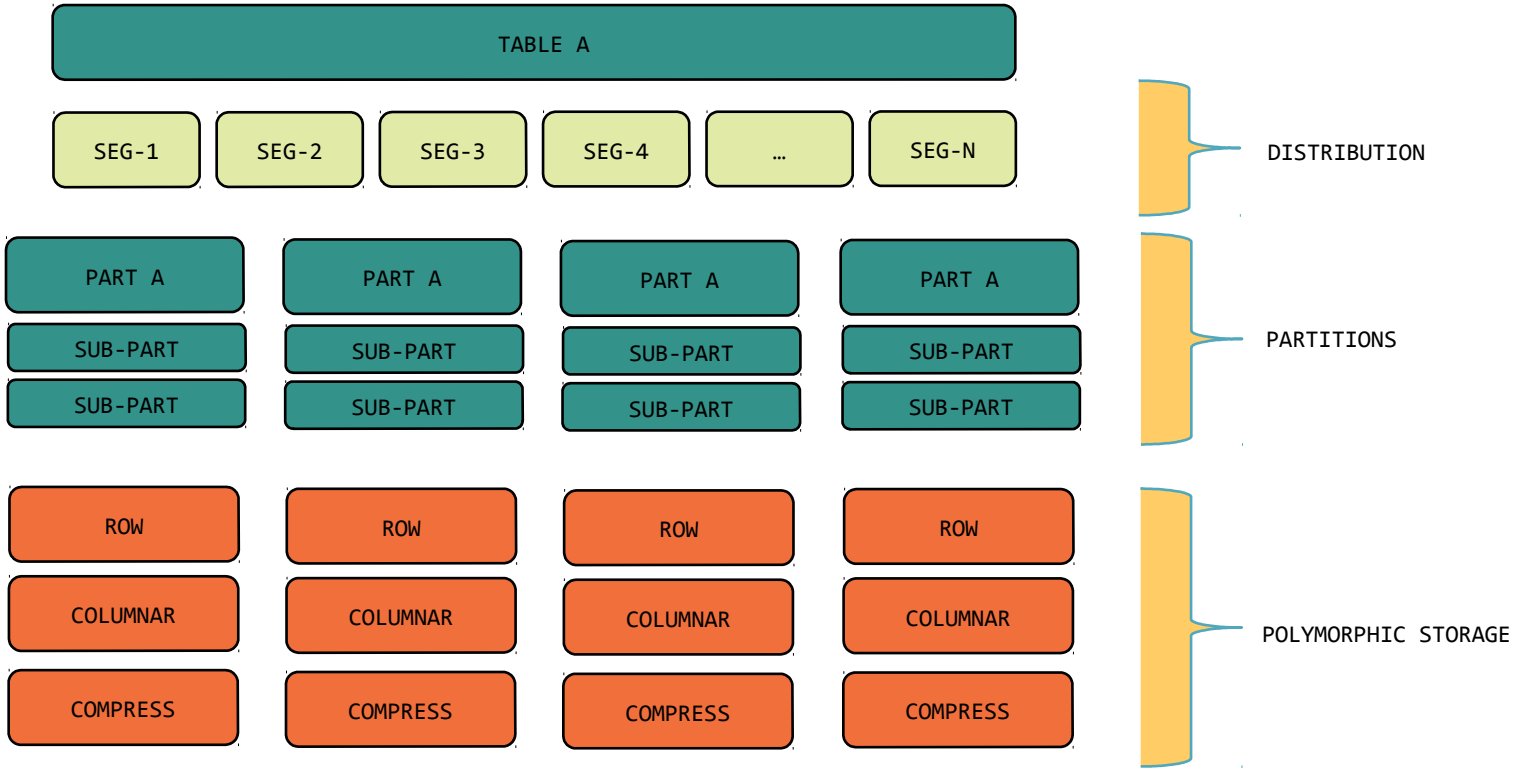


Выполнение запросов в HAWQ



Pivotal

HAWQ Хранение данных



COPM (Исходные данные)

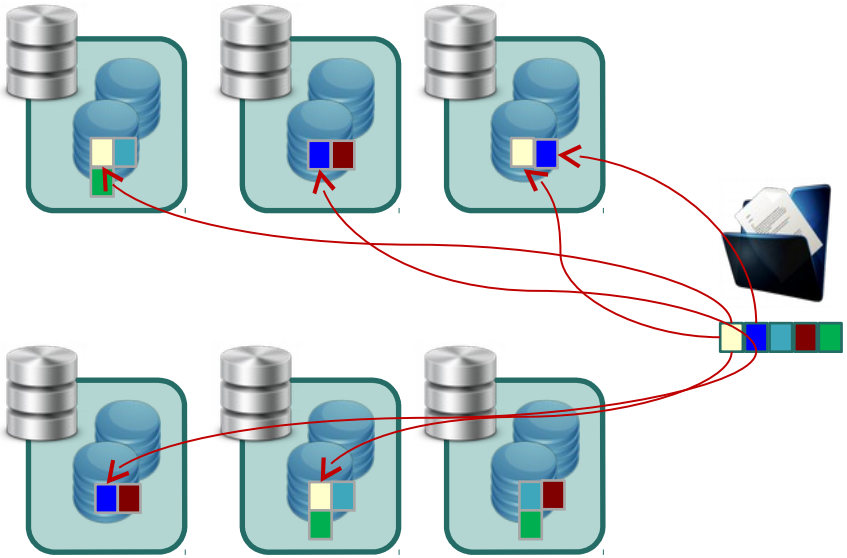
Предположим что в год требуется хранить и анализировать 200 TB информации (без учета сжатия);

Если принять продолжительность квартала 92 дня, в день прирост будет составлять ~ 540 GB.

Квартал	Объем данных, TB
Квартал 1	40
Квартал 2	45
Квартал 3	55
Квартал 4	60
Итого	200

Отказоустойчивость

3-х кратная репликация данных * (+ учет снижения производительности при наполнении более 70%)



*200 TB * 3 / 0.7 = 857 TB*

* - данный коэффициент является настраиваемым

Pivotal

Хранение данных #1

- Рекомендуемый объем данных на каждом их узлов не более *48 TB (4 ТБ диск x 12 сокетов).*
- Для хранения 857 TB данных потребуется:
  - *857 TB / 48 TB = 18 узлов*.
- Если рассматривать текущую ситуацию, при которой объем данных за один день составляет 543 GB, то на каждом из узлов будет находиться по:
  - *540 / 18 = 30.24 GB (Primary Data)*
- Пусть на каждом из узлов функционирует по 4 сегмента HAWQ, таким образом каждый из сегментов будет оперировать объемом:
  - *30,24 GB / 4 = 7.6 GB данных*

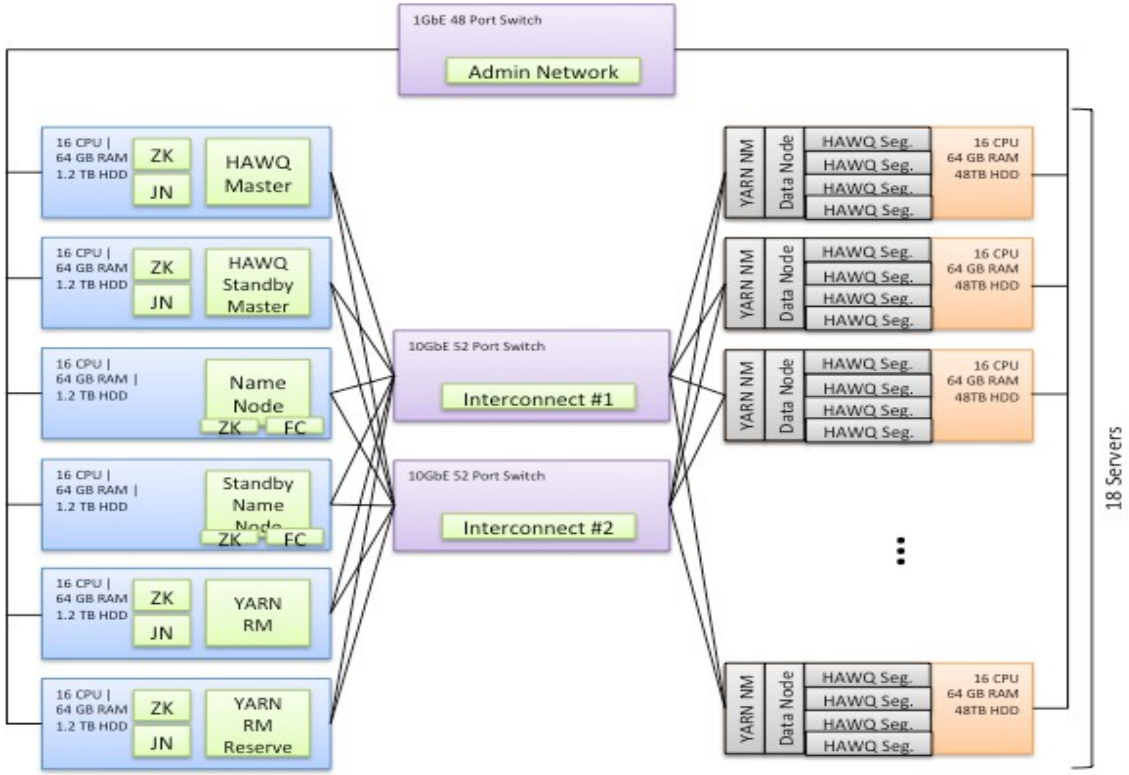
Хранение данных #2

- Предлагается разбить данные на интервалы по одному дню, а также разбить IMSI на 32 интервала по значению шести последних бит IMSI. Таким образом, каждый из дней будет разделен на 32 интервала, при этом каждому из интервалов будет соответствовать приблизительно:
  - *30.24 GB / 32 = 900 MB для узла в целом*
  - *7.6 GB / 32 = 200 MB для сегмента HAWQ (файл HDFS)*
- Таким образом, в файловой системе Hadoop будет находиться следующее количество файлов:
  - *18 узлов * 4 сегм. на серв. * 32 парт. * 92 дня в кварт. * 4 кварт. = 847’872 файлов*

Производительность

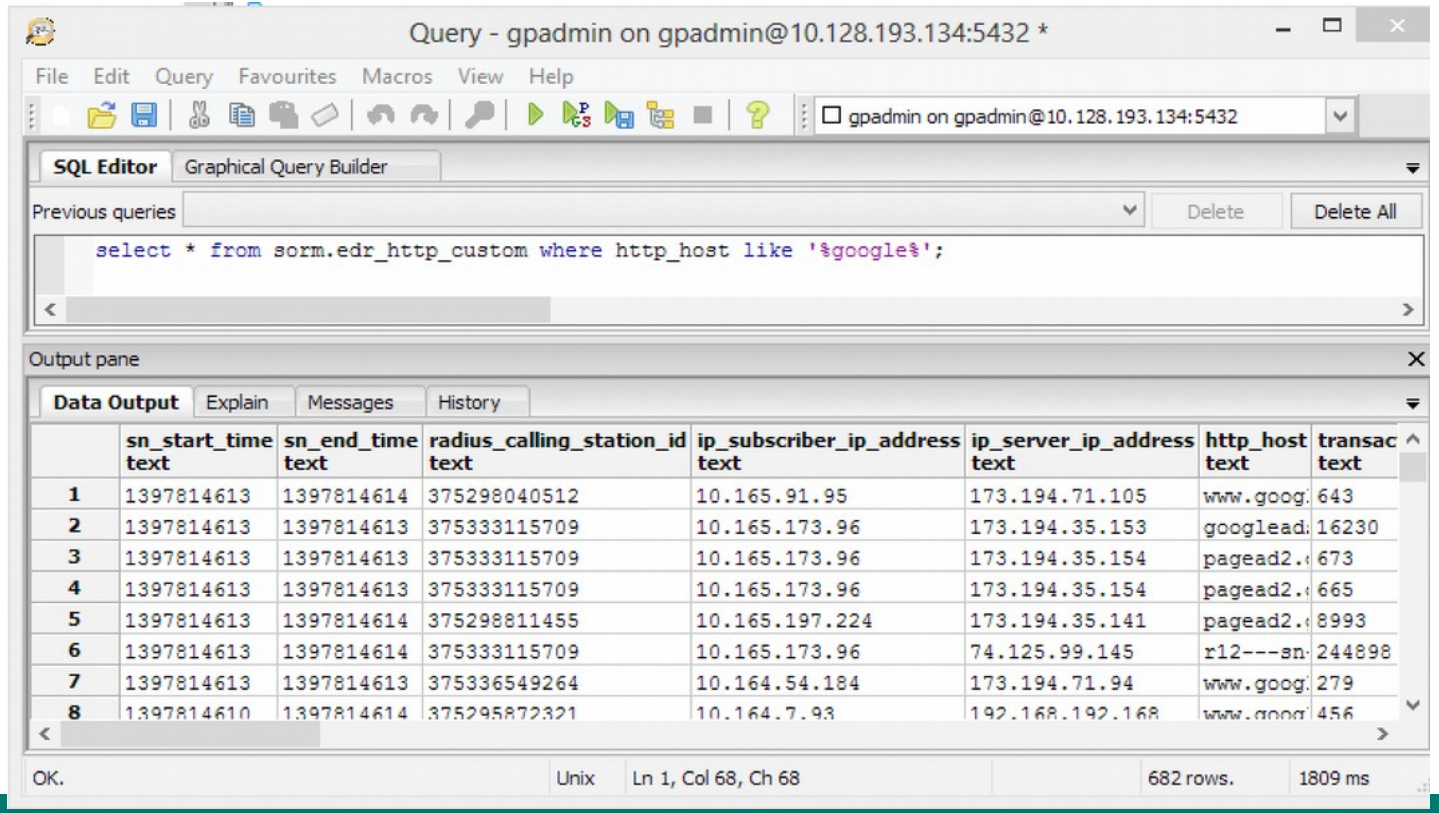
- Запрос по IMSI и периоду времени. Для анализа запроса за один день потребуется обработать 200 MB данных, что при размере блока в 64 MB составит 4 блока, то есть в среднем данные будут считываться с 4 дисков, каждый из которых сможет отдавать данные со скоростью 70 MB/sec, итого время отклика:
  - *200 MB / 70 MBps / 4 дисков = 0.7 секунды*
- При запросе данных за месяц работать будут все диски и время отклика должно составить:
  - *200 MB * 30 дней / 70 MBps / 10 дисков = 8.57 секунды*
- По транслированному IP и периоду времени. Этим запросам свойственен короткий временной период запроса, обычно минуты. Для обработки такого запроса необходимо поднять с дисков 8.1 GB данных на каждом из узлов что приведет к задействованности всех дисков, то есть время работы составит:
  - *7600 MB / 70 MBps / 10 дисков = 10.85 секунды*
- По IP назначения + периоду времени. Обрабатывается аналогично предыдущему запросу, соответственно время отклика составит 10.85 секунды

Аппаратное обеспечение (Пример)



- Общий объем 860 ТБ (с учетом резервирования и 30% свободного пространства)

Пример реализации (анализ EDR)



Вопросы?

Pivotal

aermakov@pivotal.io

www.pivotal.io