BIGDATA TEAM

HW #10: Data Layout

автор задания:
- Драль Алексей, aadral@bigdatateam.org

редакторы задания[1]:
- Николай Попов*, Игорь Лазарев**
- Big Data Mentor @ BigData Team
- *Data Engineer @ inDriver
- **Big Data Analyst

[1] Хочешь стать ментором и оставить след в истории Big Data? Тогда хорошо учись, помогай другим и дай нам знать о своем желании. Смело пиши преподавателям и менеджерам учебных курсов.

1. Описание задания

В этом задании будем оптимизировать производительность хранилища и скорость выполнения аналитических запросов с помощью правильного выбора Data Layout. Нужно решить **4 задачи**. Для решения используем Hive.

Сами задания несложные, но на выходе вы получите полезные скрипты, которые сможете применять для оптимизации работы с вашими данными на работе.

В рамках решения ДЗ по Hive, у вас появилась таблица с логами пользователей новостных сайтов `logs`. Вам предлагается решить следующие задачи (отработать задачи на семплах _S, _M и получить решение или оценки роста производительности для полного датасета). Рекомендуется использовать Managed таблицы и перезаписывать `logs_` с помощью запроса `INSERT OVERWRITE`.

Полезные материалы:

- stackoverflow: использование конструкции --hivevar;

1.1. Описание данных

`logs_raw`:

- Путь на кластере: полный датасет - **`/data/user_logs/user_logs_M`**
- Семпл (для тестирования): **`/data/user_logs/user_logs_S`**
- Формат: текст
- В каждой строке находятся следующие поля, разделенные знаком табуляции (**иногда не одним**):
    1. ip STRING - ip-адрес, с которого пришел запрос,
    2. date STRING - время запроса,
    3. request STRING - пришедший с ip-адреса http-запрос,
    4. page_size INT - размер переданной клиенту страницы в байтах,
    5. http_status INT - http-статус запроса.
    6. user_agent STRING - User Agent, информация о клиентском приложении, с которого осуществлялся запрос на сервер, в том числе информация о браузере.

Пример:

```
135.124.143.193                 20150601013300
http://newsru.com/4712386       235    412    Firefox/5.0 (compatible; MSIE
9.0; Windows NT 6.1; Win64; x64; Trident/5.0)n
```

Важно:

- разделитель между IP и временем запроса состоит из 3 символов табуляции;
- Будем считать, что информация о браузере содержится в начале 6-ого поля лога - символы с нулевой позиции до позиции первого пробельного символа.
    - пример User Agent:
    - `Chrome/5.0 (compatible; MSIE 9.0; Windows NT 8.0; WOW64; Trident/5.0; .NET CLR 2.7.40781; .NET4.0E; en-SG)`
    - тогда браузером будет: `Chrome/5.0`

Подсказка:

- поскольку нас не интересует оставшаяся часть User Agent, то получить тип браузера пользователя можно с помощью правильного регулярного выражения в период чтения `logs_raw`.

1.2. Задание #1, Task ID: hive.optimize_storage

Переложите данные `logs_raw` в таблицу `logs_orc`, где будет использоваться формат хранения данных `ORC`. С помощью параметров `TBLPROPERTIES` найдите оптимальный набор параметров, чтобы получить максимальное сжатие данных.

Проверка будет производиться на датасете _M с помощью следующего кода:

```
CREATE TABLE logs_orc
STORED AS orc
TBLPROPERTIES (
    <content of your HQL is here>[2]
)
AS SELECT *
FROM logs_raw;
```

Балл за задачу складывается из:

- **0%** - правильное решение задачи
- **0%** - поддерживаемость и читаемость кода
    - в общем случае см. Clean Code и Google Python Style Guide

[2] Таким образом, вам нужно сохранить в HiveQL файл только свойства ORC файла для DDL

- **100%** - эффективность решения. Для ориентира - размер данных в HDFS в эталонном решении на порядок меньше, чем объем данных в датасете _M (проверяем размер с помощью `hdfs dfs -du -s /path/to/table`).

Вопрос для самостоятельной проработки: какой оптимизации пространства удалось добиться для датасетов `_S`, `_M` и `_full`? Сохраняется ли динамика между _M и _full?

1.3. Задание #2, Task ID: hive.speedup_query

Придумайте аналитические запросы, которые должны работать быстрее за счет использования ORC. Проверьте скорость выполнения таких запросов на таблицах `logs_raw` и `logs_orc`. Какая оптимизация по скорости выполнения получена в зависимости от типа запроса? Сделайте релевантные таблицы для датасетов `_S`, `_M` и `_full` и сравните наблюдения. Производительность решения будет проверяться на датасете _full.

Балл за задачу складывается из:

- **0%** - правильное решение задачи
- **0%** - поддерживаемость и читаемость кода
  - в общем случае см. Clean Code и Google Python Style Guide
- **100%** - эффективность решения:
  - **80%** - использование MapReduce CPU Time должно быть в разы меньше в случае использования logs_orc (эталонное решение работает эффективнее более чем в 4.7 раза).
  - **20%** - wall time выполнения задачи (эталонное решение работает в 1.25 раза быстрее)

Сохраните ваш запрос в HiveQL файле, где название таблицы `${table_name}` для работы будет передаваться через hivevar. Оптимизированная таблица logs - это данные в формате ORC со значениями TBLPROPERTIES по умолчанию.

1.4. Задание #3, Task ID: hive.skew

Для самостоятельного изучения

Попробуйте заменить в логах информацию о браузере таким образом, чтобы 90% данных содержало одинаковый браузер (или браузер “unknown”). Запишите результат

в таблицу `logs_broken`. Попробуйте посчитать запрос в задаче “identify browser sex”. Оцените время выполнения запроса. Для того, чтобы пофиксить проблему:

1. В реальной жизни рекомендуется сделать запрос в формате `TABLESAMPLE`, чтобы увидеть, по каким параметрам происходит перекос;
2. Теперь вы знаете, по каким данным происходит перекос. Представьте эту информацию в формате `SKEWED TABLE` для Hive.

Оцените скорость выполнения запроса для датасетов `_S`, `_M` и `_full`. Не забывайте отслеживать параметр числа редьюсеров, если их недостаточно для выполнения запроса.

1.5. Задание #4, Task ID: hive.optimize_aggregate

Для самостоятельного изучения

Придумайте запрос, содержащий конструкцию GROUP BY или JOIN, который можно выполнить на стадии Map с помощью правильной укладки данных. Под правильной укладкой данных подразумевается их бакетирование и сортировка. Сколько времени тратится на переукладку данных? Какова полученная оптимизация по скорости выполнения запроса?

2. Критерии оценивания

Веса задач:

1. 50%
2. 50%
3. 0% (для самостоятельного изучения)
4. 0% (для самостоятельного изучения)

Discounts (скидки и другие акции):

- **100%** за плагиат в решениях (всем участникам процесса)
- **100%** за посылку решения после hard deadline
- **30%** за посылку решения после soft deadline и до hard deadline
- **5%** за каждую дополнительную посылку в тестирующую систему (всего можно делать до 3х посылок без штрафа):

Пример работы системы штрафов:

День	Посылка	Штраф
День 1	Посылка 1	Без штрафа
День 1	Посылка 2	Без штрафа
День 1	Посылка 3	Без штрафа
День 1	Посылка 4	-5%
День 2	Посылка 5	-5%
День 3	Посылка 6	-5%
Итоговый штраф: -15%		

Для подсчета финальной оценки **всегда** берется **последняя** оценка из Grader.

3. Инструкция по отправке задания

Перед отправкой задания оставьте, пожалуйста, отзыв о домашнем задании по ссылке: https://rebrand.ly/bdmade2022q2_feedback_hw. Это позволит нам скорректировать учебную нагрузку по следующим заданиям (в зависимости от того, сколько часов уходит на решение ДЗ), а также ответить на интересующие вопросы.

Оформление задания:

- Код задания (Short name): **HW10:DataLayout**
- Выполненное ДЗ запакуйте в архив **BD_MADE_2022_Q2_**<Surname>_<Name>_HW#**.zip**, например, для Алексея Драля -- **BD_MADE_2022_Q2_**Dral_Alexey_**HW10**.zip. Проверяйте отсутствие пробелов и невидимых символов после копирования имени отсюда[3]. Если ваше решение лежит в папке my_solution_folder, то для создания архива hw.zip на Linux и Mac OS, выполните команду[4]:
  - `zip -r hw.zip my_solution_folder/*`
- На Windows 7/8/10: необходимо выделить все содержимое директории my_solution_folder/ нажать правую кнопку мыши на одном из выделенных объектов, выбрать в открывшемся меню "Отправить >", затем "Сжатая ZIP-папка". Теперь можно переименовать архив.
- По результатам решения ожидается отчет в формате PDF с описанием результатов оптимизации (ответов на поставленные исследовательские вопросы).
- HQL-скрипты для запуска решений следует называть по суффиксу Task ID задачи **task_<Surname>_<Name>_<#task_ID_suffix>.hql**:
  - например решение задачи 2 должно называться task_<Surname>_<Name>_speedup_query.hql и его можно запустить с помощью команды:
    `$ hive -v --database=${DB_NAME}`[5] `--hivevar table_name=${table_name} -f task_*_speedup_query.hql`
- Перед проверкой убедитесь, что дерево вашего архива выглядит так:
  - **| BD_MADE_2022_Q2_<Surname>_<Name>_HW10.zip**
  - **| ---- task_<Surname>_<Name>_optimize_storage.hql**
  - **| ---- task_<Surname>_<Name>_speedup_query.hql**
  - **| ---- task_<Surname>_<Name>_skew.hql (optional)**
  - **| ---- task_<Surname>_<Name>_optimize_aggregate.hql (optional)**

[3] Онлайн инструмент для проверки: https://www.soscisurvey.de/tools/view-chars.php
[4] Флаг -r значит, что будет совершен рекурсивный обход по структуре директории
[5] Это означает, что Вы не должны использовать “use <database_name>” внутри скриптов

    - При несовпадении дерева вашего архива с представленным деревом, ваше решение будет невозможно автоматически проверить, а значит, и оценить его.
- Для того, чтобы сдать задание, необходимо:
    - Зарегистрироваться и залогиниться в сервисе Everest
    - Перейти на страницу приложения: MADE BigData Grader
    - Выбрать вкладку Submit Job (если отображается иная).
    - Выбрать в качестве “Task” значение: **HW10:DataLayout**[6]
    - Загрузить в качестве “Task solution” файл с решением
    - В качестве Access Token указать тот, который был выслан по почте
- Если Вы видите надпись "You are not allowed to run this application" во вкладке Submit Job в Everest, то на данный момент сдача закрыта (нет доступных для сдачи домашних заданий, по техническим причинам или другое). Попробуйте, пожалуйста, еще раз через некоторое время. Если Вы еще ни разу не сдавали, у коллег сдача работает, но Вы видите такое сообщение, сообщите нам об этом.
- Ситуации:
  * система оценивания показывает оценку (Grade) < 0, а отчет (Grading report) не помогает решить проблему (пример помощи: в случае неправильно указанного Access Token система вернет -2 и информацию о том, что его нужно поправить);
  * система показывает 0 и в отчете (Grading report) не указано, какие тесты не пройдены. Если Вы столкнулись с какой-то из них, присылайте ссылку на выполненное задание (Job) на почту с темой письма “Short name. ФИО.”. Например: “**HW10:DataLayout**. Иванов Иван Иванович.”
  Пример ссылки: https://everest.distcomp.org/jobs/67893456230000abc0123def
  Внимание: Если до дедлайна остается меньше суток, и Вы знаете (сами проверили или коллеги сообщили), что сдача решений сломана, обязательно сдайте свое решение и напишите письмо, как написано выше, чтобы мы видели, какое решение Вы имели до дедлайна и смогли его оценить.

Любые вопросы / комментарии / предложения можно писать в телеграм-канал курса или на почту bd_made2022q2@bigdatateam.org.

Peace, love, обнимашки, интересности скидываем в общий чат курса ;)

[6] Сервисный ID: hive.layout_hw

4. FAQ (часто задаваемые вопросы)

Что в отчете Grader означает проверка X ?
Как читать отчет: Для каждого теста • Raw_score - балл за конкретный тест. Может быть как бинарным (1\0), так и находиться в интервале от 0 до 1 • Score - Raw_score*weight(вес теста в общей оценке). Вес указан для каждого теста ниже Итоговая оценка: смотрите строку Score (сумма Score всех индивидуальны тестов) внизу отчета. **Правильность решения задачи:** test_unzip_is_succesful (weight = 0) - ДЗ заархивировано в .zip архив и грейдер может его разархивировать **Эффективность решения:** test_hdfs_size_compression_ratio - улучшение сжатия данных на hdfs test_hive_speedup_query_message_helper - оптимизация скорости доступа к данным test_hive_query_mr_cpu_time_speedup_ratio - уменьшение времени работы ядер в map-reduce задаче test_hive_query_wall_time_speedup_ratio - уменьшение предельного времени выполнения запроса