Интеграция в ClickHouse функциональности обработки HTTP User Agent

Филитов Михаил
ВШЭ ФКН ПМИ 166

ClickHouse

ClickHouse - столбцовая система управления базами данных предназначенная для работы с аналитическими запросами

Сценарии использования:

- Данные добавляются в базу данных, но не изменяются
- Большая часть запросов на чтение
- Данные обновляются большими частями

User-Agent

https://www.whoishostingthis.com/tools/user-agent/

User agent - это программное обеспечение (программный агент), которое действует от имени пользователя.

User-Agent Header - строка, описывающая User-Agent

- Mozilla/5.0 (iPad; U; CPU OS 3_2_1 like Mac OS X; en-us) AppleWebKit/531.21.10 (KHTML, like Gecko) Mobile/7B405

- Googlebot/2.1 (+http://www.google.com/bot.html)

Разбор

https://www.whoishostingthis.com/tools/user-agent/

Mozilla/5.0 (iPad; U; CPU OS 3_2_1 like Mac OS X; en-us) AppleWebKit/531.21.10 (KHTML, like Gecko) Mobile/7B405

- Строка, показывающая совместимость с движком для рендеринга
- Описание системы, на которой запущен браузер
- Используемый движок для рендеринга
- Детали платформы браузера
- Доступные на платформе улучшения

Разнообразие

SOFTWARE:

- Chrome (8,645,939)
- Facebook App (8,611,571)
- Instagram (2,154,833)
- Internet Explorer (1,520,148)
- UC Browser (979,519)
- Opera (689,399)
- Yandex Browser (686,455)
- RuxitSynthetic (579,157)
- Safari (498,491)

Browse all Software Names

OPERATING SYSTEMS:

- Android (16,424,246)
- iOS (6,634,985)
- Windows (3,213,596)
- Linux (242,606)
- Mac OS X (138,200)
- macOS (89,776)
- Symbian (40,274)
- Chrome OS (19,197)
- Fire OS (17,568)

Browse all Operating Systems

OPERATING PLATFORMS:

- iPhone (3,657,849)
- iPad (707,212)
- Pixel (361,868)
- Nexus 5 (269,372)
- Samsung SM-G900P (260,007)
- iPhone 6 (213,556)
- iPhone 6s (195,985)
- iPhone 7 (184,788)
- Moto G (182,862)

Browse all Operating Platforms

SOFTWARE TYPES:

- Web Browser (14,655,424)
- In-App Browser (10,105,517)
- Site Monitor (610,775)
- Application (188,688)
- Crawler (31,590)
- Media Player (7,212)
- Software Library (6,034)
- Analyser (4,358)
- Download Helper (1,845)

Browse all Software Types

HARDWARE TYPES:

- Phone (10,904,498)
- Mobile (8,908,089)
- Computer (3,404,523)
- Tablet (1,855,906)
- Server (650,270)
- Music Player (35,220)
- E-Book Reader (12,230)
- TV (7,729)
- Car (1,858)

Browse all Hardware Types

LAYOUT ENGINES:

- WebKit (13,498,618)
- Blink (8,297,812)
- Trident (1,736,030)
- Gecko (468,070)
- Presto (348,317)
- EdgeHTML (16,616)
- KHTML (3,820)
- NetFront (3,814)
- Goanna (3,312)

Browse all Layout Engine Names

Рис.1 Разнообразие UA

Похожие но разные

- Mozilla/5.0 (iPhone; CPU iPhone OS 12_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Mobile/15E148

- Mozilla/5.0 (iPhone; CPU iPhone OS 11_4_1 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Mobile/15G77

- Mozilla/5.0 (iPhone; CPU iPhone OS 9_2_1 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Version/9.0 Mobile/13D15 Safari/601.1

Запрос пользователей

Функционал для получения частей User-Agent из строки

SELECT extractOSFromUserAgent(url) FROM urls;

SELECT extractBrowserVersionFromUserAgent(url) FROM urls;

SELECT hasSameSiteSupport(url) FROM urls;

Существующие решения

Faisalman/ua-parser-js

- Stars - 4600
- Язык - JS

- Нельзя добавить новые данные
- Последовательный проход по заданным регулярным выражениям
- Требуется несколько раз обрабатывать строку для получения разных полей из UA

Существующие решения

Matomo-org/device-detector

- Stars - 1700
- Язык - PHP
- Сложно добавлять новые данные
- Последовательный проход по заданным регулярным выражениям
- Требуется несколько раз обрабатывать строку для получения разных полей UA

Существующие решения

Ua-parser/uap-cpp

- Stars - 26
- Язык - C++
- Данные загружаются из Yaml файла
- Последовательный проход по заданным регулярным выражениям
- Одна строка обрабатывается один раз
- Есть оптимизация с поддержанием индекса из частей регулярных регулярных выражений

Существующие решения

Highpower/uatraits

- Stars - 18
- Язык - C++
- Данные загружаются из XML файла
- Поиск подходящих регулярных выражения с помощью Ахо-Корасик
- Одна строка обрабатывается один раз
- Не поддерживается с 2012 года
- Используются устаревшие конструкции (собственная реализация shared_ptr)

Существующие решения

Uatraits-fast (Yandex)

- Язык - C++
- Данные загружаются из XML файла
- Поиск подходящих регулярных выражения с помощью Ахо-Корасик
- Одна строка обрабатывается один раз
- Невозможно использовать как библиотеку

Существующие решения

Hyperscan (Intel)

- Движок для поиска регулярных выражений
- Поиск многих совпадений одновременно
- Потоковый режим
- Используется в ClickHouse

Финальное решение

- Hyperscan - быстрый поиск “сработавших” регулярных выражений

- Uatraits-fast - установление полей UA на основе совпадений регулярных выражений

От старта до результата

Старт сервера:

- Загрузка данных из XML
- Компиляция базы регулярных выражений Hyperscan
- Объект UserAgent

Запрос:

- Класс ExtractBrowserFromUserAgent(request, context)
- Метод detect(): Hyperscan -> uatraits-fast -> Agent
- Agent.getBrowser()

Результат

- Проведен анализ существующих решений, выбрано наиболее подходящее

- Реализованы функции, получающие части User-Agent, например, extractOSFromUserAgent

- Реализована связка uatraits-fast + hyperscan -> легкое добавление новых функций

- Код uatraits-fast переработан для использования в ClickHouse и может быть переиспользован в других частях проекта

- Реализованы тесты

Результат

```
SELECT *
FROM test.first

┌──────────d─┬─url──────────────────────────────────────────────────────────────────────────┐
│ 2020-05-25 │ Mozilla/5.0 (Macintosh; Intel Mac OS X 10.14; rv:76.0) Gecko/20100101 Firefox/76.0 │
└────────────┴──────────────────────────────────────────────────────────────────────────┘

┌──────────d─┬─url──────────────────────────────────────────────────────────────────────────┐
│ 2020-05-25 │ Mozilla/5.0 (Linux; Android 9; Pixel 2 XL Build/PPP3.180510.008) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.87 Mobile Safari/537.36 │
└────────────┴──────────────────────────────────────────────────────────────────────────┘

2 rows in set. Elapsed: 0.003 sec.

mfilitov-dev.sas.yp-c.yandex.net :) select extractOSFromUserAgent(url) as OS, extractBrowserFromUserAgent(url) as browser from test.first

SELECT
    extractOSFromUserAgent(url) AS OS,
    extractBrowserFromUserAgent(url) AS browser
FROM test.first

┌─OS─────┬─browser─┐
│ macos  │ Firefox │
└────────┴─────────┘

┌─OS─────┬─browser──────┐
│ android │ ChromeMobile │
└────────┴──────────────┘

2 rows in set. Elapsed: 0.003 sec.

mfilitov-dev.sas.yp-c.yandex.net :)
```

Рис.2 Пример обработки User-Agent

Дальнейшее развитие

- Реализация дополнительных функций, получающих информацию из User-Agent (extractOSVersionFromUserAgent)

- Реализация функций, косвенно получающих информацию из User-Agent (hasSamesiteSupport)

- Добавление версии с кэшированием наиболее популярных User-Agent

Спасибо за внимание!

Филитов Михаил

filitovme@gmail.com

8-985-974-79-59

Подробнее про User-Agent:

developers.whatismybrowser.com/useragents/explore