

GeoIP.jl:

Быстрое получение ip данных

Андрей Оськин (<https://github.com/Arkoniak>)



IP Addresses



Enter up to 25 IP addresses separated by spaces or commas. You can also [test your own IP address](#).

Submit

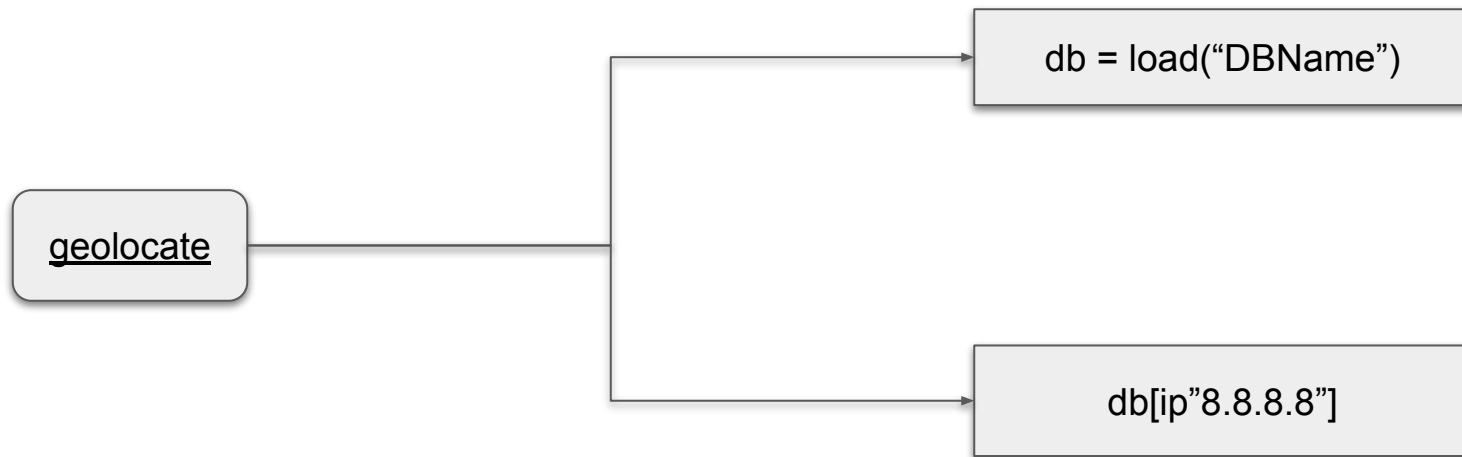
GeoIP2 City Results

IP Address	Country Code	Location	Network	Postal Code	Approximate Coordinates*	Accuracy Radius (km)	ISP	Organization	Domain	Metro Code
8.8.8.8	US	United States, North America	8.8.8.8/32		37.751, -97.822	1000	Google	Google		

Проблемы GeolIP.jl

1. Pre 1.0
2. Основной метод `geolocate` включал в себя загрузку и апгрейд базы
 - a. В 2018 году MaxMind поменял лицензию и загрузка стала невозможной
 - b. Нет поддержки IPv6 и локализаций
 - c. Глобальная переменная “база данных”
3. Результат возвращался в виде `Dict{String, Any}`
4. Низкая скорость работы: ~400 ms на один вызов

Первая итерация



Производительность: ~200 us или 5 000 запросов в секунду

Вторая итерация: Column oriented или Row oriented?

Column

<u>IPv4 Net</u>	<u>Continent</u>	<u>City</u>
1.0.1.0/24	Asia	Tokyo
1.0.2.0/24	Asia	Tokyo
1.0.8.0/24	Asia	Beijing
1.0.16.0/24	Oceania	Canberra

Row

<u>IPv4 Net</u>	<u>Continent</u>	<u>City</u>
1.0.1.0/24	Asia	Tokyo
1.0.2.0/24	Asia	Tokyo
1.0.8.0/24	Asia	Beijing
1.0.16.0/24	Oceania	Canberra

Вторая итерация: Hybrid approach (Key Value)

Hybrid

<u>IPv4 Net</u>	<u>Continent</u>	<u>City</u>
1.0.1.0/24	Asia	Tokyo
1.0.2.0/24	Asia	Tokyo
1.0.8.0/24	Asia	Beijing
1.0.16.0/24	Australia	Canberra

Поиск: searchsortedfirst (бинарный поиск)

Возвращаемое значение: по индексу из вектора структур

Производительность: ~1.2 миллиона запросов в секунду

Для сравнения:

<https://github.com/maxmind/GeoIP2-python/issues/78#issuecomment-535982396>

“50,000 to 150,000 lookups per second with the C extension.”

Планы

1. Добавить поддержку IPv6
2. Ускорить загрузку данных
3. Добавить поддержку mmdb
4. Выпустить версию 1.0 :-)

Некоторые наблюдения

1. Julia community построена на добровольных началах и многим проектам не хватает поддержки.
2. Хотя Julia код может быть очень быстрым, но не стоит забывать про обычную оптимизацию: Performance Tips + производительные алгоритмы.
3. Не стоит переживать из-за того, что существуют другие библиотеки, которые решают похожие задачи.

СПАСИБО!