11.10.17 Инструкция по подготовке данных для службы построения маршрутов(Linux версия).

1) Сервис маршрутизации использует подготовленный граф дорог, который получается обработкой исходного файла в формате .pbf (обновляются файлы графа России и стран СНГ на ресурсе <http://gis-lab.info/projects/osm_dump/index.html>). Оттуда необходимо скачать свежий файл osm.pbf для России. Можно скачать файл для всех стран СНГ или для конкретного региона (на странице ресурса описано подробнее).

2) Скачанный файл RU.osm.pbf необходимо будет преобразовать в перечень файлов, которые сможет использовать наш текущий сервис маршрутизации, установленный на Ubuntu. Это можно сделать на развернутой виртуальной машине с ubuntu(x86,x64), на которой подготовлен и скомпилирован сервер маршрутизации. Поскольку для генерации используется большое количество оперативной памяти и памяти на жестком диске, нужно использовать виртуальную машину с размером жесткого диска около 100гб.

Если на виртуальной машине не была скомпилирована служба маршрутизации, то выполняем следующие шаги(если нет директории /home/user/GIS\_WEB60/RoutingService/RoadService, то создаем её ):

а) Устанавливаем необходимые пакеты для компиляции исходников службы маршрутизации https://github.com/Project-OSRM/osrm-backend/wiki/Building-on-Ubuntu

**sudo apt install build-essential git cmake pkg-config \**

**libbz2-dev libstxxl-dev libstxxl1v5 libxml2-dev \**

**libzip-dev libboost-all-dev lua5.2 liblua5.2-dev libtbb-dev**

б) Сборка из исходников(Сборка проекта из исходных текстов длится около 10-15 минут( x64 – около 12 минут) <https://github.com/Project-OSRM/osrm-backend/wiki/Building-OSRM>

* Если папка osrm-backend существует и в ней установлена старая служба, то удаляем эту папку
* копируем архив osrm-backend-5.12.0.tar.gz в папку /home/user/GIS\_WEB60/RoutingService и переходим в эту папку в консоли

**cd /home/user/GIS\_WEB60/RoutingService**

* распаковываем архив и созданную папку переименовываем в osrm-backend

**tar -xzf osrm-backend-5.12.0.tar.gz**

**mv osrm-backend-X.Y.Z osrm-backend**

в) Компиляция и установка в систему сервера маршрутизации

**cd osrm-backend**

**mkdir -p build**

**cd build**

**cmake .. -DCMAKE\_BUILD\_TYPE=Release**

**cmake --build .**

**sudo cmake --build . --target install**

1. Подготовка графа дорог для запуска сервера

С помощью утилиты osrm-extract, данные извлекаются из карты и сохраняются для дальнейшей загрузки. Перед тем как извлекать данные, необходимо указать файл профиля (profile.lua), в котором описано какие именно маршруты необходимы. Все профили лежат в папке profiles в корневой папке с исходниками и подключаются созданием ссылки на папку

* Возвращаемся в директорию службы маршрутизации

**cd ../../RoadService**

* Создаем ссылку на папку с профилями

**ln -s osrm-backend/profiles**

* Для извлечения данных потребуется много ресурсов, поэтому желательно создать swapfile в случае, если в консоли выдается подобная ошибка:

[info] RAM: peak bytes used: 4419612672

[error] [exception] std::bad\_alloc

[error] Please provide more memory or consider using a larger swapfile

* Если файл подкачки уже существует (/home/user/GIS\_WEB60/RoutingService/RoadServiceSwapFile), то нужно его удалить(либо не трогать. Предпочтительно не трогать)

**sudo swapoff /home/user/GIS\_WEB60/RoutingService/RoadServiceSwapFile**

**rm /home/user/GIS\_WEB60/RoutingService/RoadServiceSwapFile**

* Создаем файл подкачки

**fallocate -l 40G /home/user/GIS\_WEB60/RoutingService/RoadServiceSwapFile**

**chmod 600 /home/user/GIS\_WEB60/RoutingService/RoadServiceSwapFile**

**mkswap /home/user/GIS\_WEB60/RoutingService/RoadServiceSwapFile**

**sudo swapon /home/user/GIS\_WEB60/RoutingService/RoadServiceSwapFile**

* Дополнительно создаем файл в директории, где лежит программа экспорта, если его там нет

**nano ../osrm-backend/build/.stxxl**

**disk=/tmp/stxxl,40G,syscall**

* Удаляем все файлы в папке /home/user/GIS\_WEB60/RoutingService/RoadService. Там должна остаться только ссылка на папку profiles
* Копируем скаченный файл RU.osm.pfb в папку /home/user/GIS\_WEB60/RoutingService/RoadService
* Запускаем команду извлечения данных (занимает около 30 минут)

**osrm-extract RU.osm.pbf**

либо с прямым указанием профиля

**osrm-extract RU.osm.pbf -p profiles/car.lua**

* Запускаем команду для создания иерархии графа дорог(занимает около часа)

**osrm-contract RU.osrm**

Результатом будет создание 24-х файлов RU\*.\* общим размером около 5,5гб

1. Запуск службы(проверка работоспособности созданного графа)

* Запуск службы на порте 8484(по умолчанию порт 5000), -t=8 – количество потоков

**osrm-routed -p 8484 –t 8 RU.osrm**

* ждём сообщение в консоли, что служба готова к приёму запросов

[info] running and waiting for requests

* Проверяем, что служба работает с подготовленным графом дорог(указываем ip адрес, где запущена служба. 127.0.0.1 , если внутри линукса в firefox проверить)

[**http://192.168.1.126:8484/route/v1/driving/30.348887,60.010632;30.32459,59.873786?overview=false&steps=true**](http://192.168.1.126:8484/route/v1/driving/30.348887,60.010632;30.32459,59.873786?overview=false&steps=true)

1. Сохраняем 24 сгенерированных файла для дальнейшего использования при обновлении графа у клиентов