

# Ruteo con Open Source Routing Machine (OSRM).

# Índice

Índice .....	2
Objetivos .....	3
Introducción .....	3
¿Qué es OSRM? .....	4
¿Cómo funciona OSRM? .....	4
Puesta en funcionamiento OSRM .....	6
└─ Instalación de docker .....	6
└─ Generar los archivos para el Back-end .....	6
└─ Entorno de Trabajo .....	6
└─ Generación de archivos .....	6
└─ Correr el Front-end .....	7
└─ Visualizar el Front-end .....	7

# Objetivos

El objetivo de este documento es que el usuario pueda utilizar la red ruteable que produce la Unidad de Sistemas de Información Geográfica (USIG) del Gobierno de la Ciudad. La información aquí presentada puede servir para generar nuevos productos, análisis e investigaciones, o simplemente para facilitar la movilidad en la Ciudad y el Gran Buenos Aires. Los ruteos están diseñados para desplazamientos en auto, bicicletas o a pie.

# Introducción

La red más importante para realizar este ruteo es el Callejero de CABA, la cual está compuesta por todas las fuentes geográficas oficiales provenientes del Gobierno de la Ciudad de Buenos Aires.<sup>1</sup> Por fuera de esta jurisdicción, los datos son obtenidos a través de OpenStreetMap, una base de datos geográficos abierta y colaborativa.<sup>2</sup>

La cobertura de la red ruteable que ponemos a disposición comprende todo el Área Metropolitana de Buenos Aires (**AMBA**) conformada por CABA y los siguientes 43 municipios de la Provincia de Buenos Aires: Almirante Brown, Avellaneda, Berazategui, Berisso, Brandsen, Campana, Cañuelas, Ensenada, Escobar, Esteban Echeverría, Exaltación de la Cruz, Ezeiza, Florencio Varela, General Las Heras, General Rodríguez, General San Martín, Hurlingham, Ituzaingó, José C. Paz, La Matanza, Lanús, La Plata, Lomas de Zamora, Luján, Marcos Paz, Malvinas Argentinas, Moreno, Merlo, Morón, Pilar, Presidente Perón, Quilmes, San Fernando, San Isidro, San Miguel, San Vicente, Tigre, Tres de Febrero, Vicente López, Zárate, Lobos, Mercedes y Navarro.

<sup>(1)</sup> El Callejero CABA está incluido BA Data, el Portal de Datos Públicos de la Ciudad: <https://data.buenosaires.gob.ar/dataset/calles>

<sup>(2)</sup> [https://wiki.openstreetmap.org/wiki/ES:P%C3%A1gina\\_principal](https://wiki.openstreetmap.org/wiki/ES:P%C3%A1gina_principal)



Fuente: elaboración propia en base a QGIS.

## ¿Qué es OSRM?

Open Source Routing Machine es una máquina de enrutamiento de código abierto y de alto rendimiento para las rutas de carretera. OSRM es un servicio de red gratuito, compatible con las plataformas Linux, FreeBSD, Windows y Mac OS X.

## ¿Cómo funciona OSRM?<sup>3</sup>

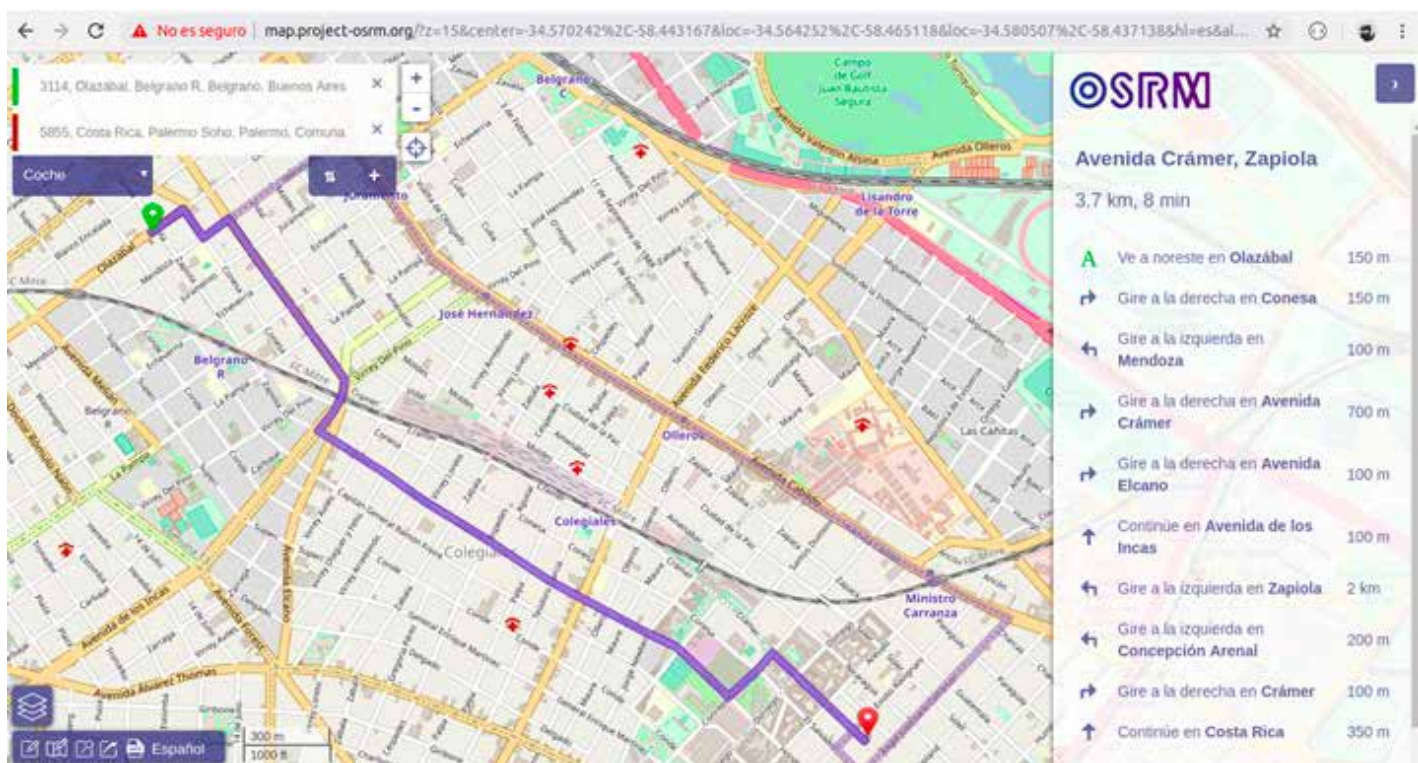
Combina sofisticados algoritmos de enrutamiento con los datos de la red de carreteras abierta y gratuita del proyecto OpenStreetMap (OSM).

<sup>(3)</sup> <http://project-osrm.org/>

Lo que hace el OSRM es calcular y generar una ruta más corta entre cualquier origen y destino en unos pocos milisegundos. La mayor parte del esfuerzo se dedica a anotar la ruta y transmitir la geometría a través de la red.

Dado que está diseñado en compatibilidad con OpenStreetMap, los archivos de datos OSM se pueden importar fácilmente.

En nuestro caso, para construir una red unificada para el ruteo con las fuentes de CABA y OSM, los datos son homologados mediante Postgres/Postgis, con la estructura que necesita OSRM para aplicar el algoritmo de ruteo. El archivo de esta red ruteable es el .pbf en sus distintos modos de transporte (caminata, bici y auto), estos archivos que disponibilizamos en BA DATA, permite generar los archivos faltantes para el Back-end del ruteo.



Fuente: <http://map.project-osrm.org>

# Puesta en funcionamiento OSRM

## Instalación de docker

Para poder trabajar con OSRM, es necesario instalar docker para poder ejecutar el back-end (generar los archivos para el ruteo) y el front-end (visualizador). La descarga del docker para todos los sistemas operativos se realiza en la siguiente página:

<https://docs.docker.com/get-docker/>

## Generar los archivos para el Back-end

### Entorno de Trabajo

Una vez instalado el docker, debemos generar los archivos para el Back-end. Para ello, debemos colocar el .pbf del modo de transporte que queramos visualizar en una carpeta "data", para el ejemplo utilizamos: [/disk1/volumenes/osrm/data](#)

Corremos el docker para que OSRM construya el algoritmo de ruteo en base al .pbf que colocamos dentro de la carpeta data. Esto se construye gracias al archivo de configuración que tiene velocidades y otras características propias del motor, denominado .lua, cada modo tiene su propio archivo. Ejemplo car.lua para el modo auto.

Todo esto puede ser consultado en el github de OSRM:

<https://github.com/Project-OSRM/osrm-backend>

### Generación de archivos

1. Corremos el docker y en la siguiente línea corremos el modo auto gba\_auto.pbf ubicado en la carpeta data para que lea su archivos:

```
docker run -t -v "/disk1/volumenes/osrm/data:/data" osrm/osrm-backend:v5.22.0  
osrm-extract -p /opt/car.lua /data/gba_auto.pbf
```

2. Luego generamos los archivos de partición:

```
docker run -t -v "/disk1/volumenes/osrm/data:/data" osrm/osrm-backend:v5.22.0  
osrm-part ition /data/gba_auto.osrm
```

3. Después de esto, generamos los archivos de clusterización:

```
docker run -t -v "/disk1/volumenes/osrm/data:/data" osrm/osrm-backend:v5.22.0  
osrm-customize /data/gba_auto.osrm
```

4. Finalmente, generamos el ruteo con los archivos anteriores en el puerto 500:

```
docker run --restart=always -t -i -p 5000:5000 -v "/disk1/volumenes/osrm/data:/data"  
osrm/osrm-backend:v5.22.0 osrm-routed --algorithm mld /data/gba_auto.osrm
```

## Correr el Front-end

Una vez concluido el proceso de Back-end, debemos levantar el front del OSRM.

Para hacer esto, se debe correr el siguiente código:

```
docker run -p 9966:9966 -e OSRM_BACKEND='http://127.0.0.1:5000'  
osrm/osrm-frontend
```

Para este paso, debemos tener en cuenta la ip de nuestro local y el puerto que le hayamos configurado anteriormente -en nuestro caso el puerto 500-.

## Visualizar el Front-end

Para empezar a utilizar la interfaz gráfica entrar a la siguiente URL:

```
http://localhost:9966/?z=13&center=-34.593791%2C-58.346586&hl=en&alt=0
```

**Secretaría de Innovación y Transformación Digital**  
Subsecretaría de Políticas Públicas Basadas en Evidencia