

# Using OptaWeb Vehicle Routing

The OptaPlanner Team

Version 7.37.0-SNAPSHOT

# Table of Contents

1. Introduction .....	2
1.1. What is OptaWeb Vehicle Routing? .....	2
2. Quickstart .....	3
2.1. Install Java 8 or higher .....	3
2.2. Download OptaWeb Vehicle Routing .....	3
2.3. Run OptaWeb Vehicle Routing .....	4
3. Run the OptaWeb Vehicle Routing using the <code>runLocally.sh</code> script .....	6
3.1. Getting started .....	6
3.2. Interactive mode .....	6
3.3. Run OptaWeb Vehicle Routing with a selected region non-interactively .....	7
3.4. Run OptaWeb Vehicle Routing in air distance mode .....	7
3.5. Tweak the data directory .....	8
4. Run OptaWeb Vehicle Routing without the script .....	9
4.1. Download routing data .....	9
4.2. Create the OptaWeb Vehicle Routing data directory structure .....	9
4.3. Run OptaWeb Vehicle Routing using the <code>java</code> command .....	10

As a developer, you can use OptaWeb Vehicle Routing to optimize your vehicle fleet deliveries. In this guide, you will create and run a sample OptaWeb Vehicle Routing application.

# Chapter 1. Introduction

## 1.1. What is OptaWeb Vehicle Routing?

The main purpose of many types of businesses is to transport various types of cargo. The goal of these businesses is to deliver a piece of cargo from the loading point to a destination and use its vehicle fleet in the most efficient way. This type of optimization problem is referred to as the [vehicle routing problem](#) (VRP) and has many variations.

[OptaPlanner](#) can solve many of these vehicle routing variations and provides solution examples. OptaPlanner enables developers to focus on modeling business rules and requirements instead of learning [constraint programming](#) theory. OptaWeb Vehicle Routing expands OptaPlanner's vehicle routing capabilities by providing a reference implementation that answers questions such as these:

- Where do I get the distances and travel times?
- How do I visualize the solution on a map?
- How do I build an application that runs in the cloud?

# Chapter 2. Quickstart

You can get up and running with OptaWeb Vehicle Routing in just a few steps. In this chapter you will download the OptaWeb Vehicle Routing distribution archive containing a binary build of OptaWeb Vehicle Routing. You will use a Bash script to run the binary without having to build the project.

## 2.1. Install Java 8 or higher

**Java SE 8 or higher** must be installed on your system before you can use OptaWeb Vehicle Routing.

### NOTE

It is recommended that you install Java SE Development Kit (JDK) because it is necessary in order to build OptaWeb Vehicle Routing from the source. However, if you have a binary distribution of OptaWeb Vehicle Routing, you only need the Java SE Runtime Environment (JRE).

### Procedure

1. To verify the current Java installation, enter the following command:

```
java -version
```

2. If necessary, install OpenJDK 8.

- To install OpenJDK 8 on Fedora, enter the following command:

```
sudo dnf install java-1.8.0-openjdk-devel
```

- To install OpenJDK on other platforms, follow instructions at <https://openjdk.java.net/install/>.

## 2.2. Download OptaWeb Vehicle Routing

Download the OptaWeb Vehicle Routing distribution archive, available from the OptaPlanner website, to quickly evaluate OptaWeb Vehicle Routing without having to set up build tools.

### NOTE

If you want to modify OptaWeb Vehicle Routing and build it yourself or contribute to upstream, see the Development chapter.

### Procedure

1. Go to <https://www.optaplanner.org/download/download.html> and click the **OptaWeb Vehicle Routing** tab.
2. Click **Download OptaWeb Vehicle Routing 7.37.0-SNAPSHOT**.

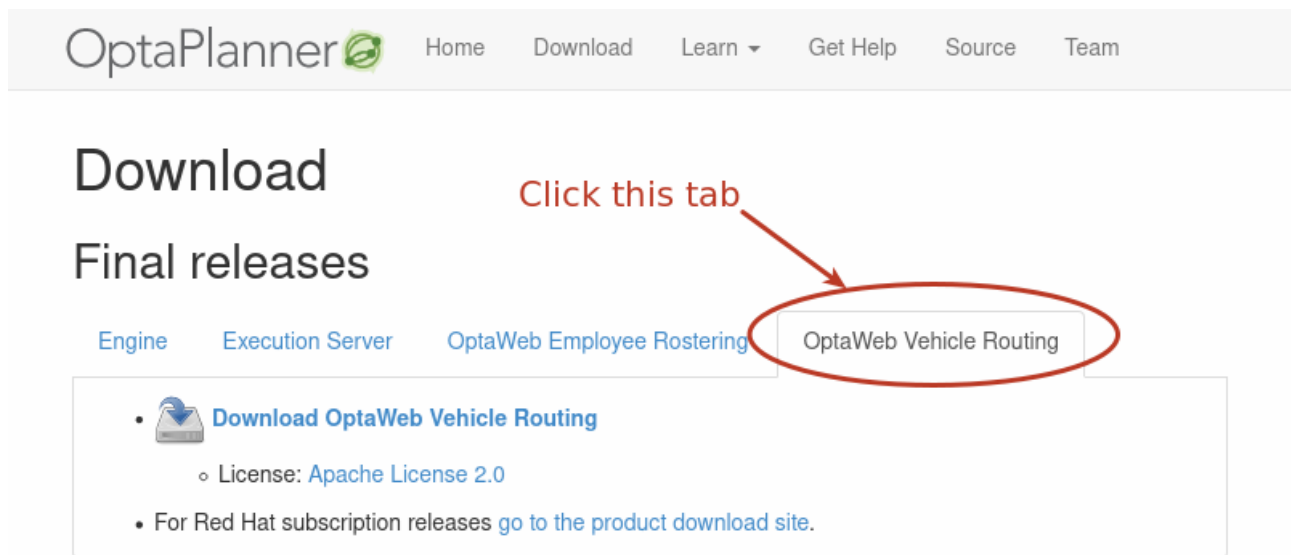


Figure 1. OptaPlanner download page

3. Extract the downloaded distribution ZIP file. The archive contains source files and a binary build of OptaWeb Vehicle Routing as well as the OptaWeb Vehicle Routing documentation.

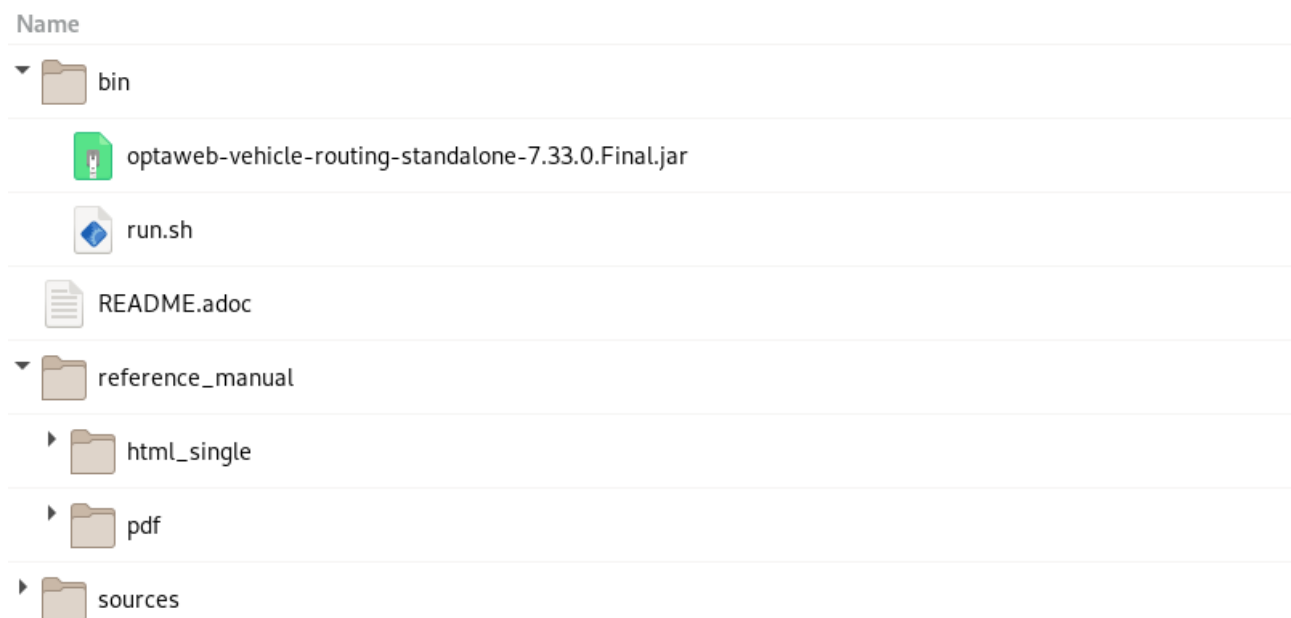


Figure 2. Content of the OptaWeb Vehicle Routing distribution archive

## 2.3. Run OptaWeb Vehicle Routing

After you download OptaWeb Vehicle Routing and extract the distribution archive, use the `runLocally.sh` script to run it.

- |             |   |
|-------------|---|
| <b>NOTE</b> | If the standalone JAR is not part of the distribution, build the project from source by using the <code>sources</code> directory. You can use the <code>sources</code> directory inside the distribution as if you have cloned the source repository from GitHub. |
| <b>NOTE</b> | If Bash is not available on your system, continue to <a href="#">Run OptaWeb Vehicle Routing without the script</a> .   |

### *Prerequisites*

- Internet access is available. When OptaWeb Vehicle Routing runs it uses third-party public services such as [OpenStreetMap](#) to display map tiles and provide search results.
- Java 8 or higher is installed.
- OptaWeb Vehicle Routing distribution archive is downloaded and extracted.

### *Procedure*

Enter the following command.

```
./bin/runLocally.sh
```

The script will download an OSM file that is needed to work with the sample data set that is included with the application. The script also has an interactive mode you can use to download additional regions. See [Run the OptaWeb Vehicle Routing using the runLocally.sh script](#) to learn more about the script.

# Chapter 3. Run the OptaWeb Vehicle Routing using the `runLocally.sh` script

Linux and macOS users can use a Bash script called `runLocally.sh` to run OptaWeb Vehicle Routing. The script automates some setup steps that would otherwise have to be carried out manually. The script will:

- Create the data directory.
- Download selected OSM files from Geofabrik.
- Try to associate a country code with each downloaded OSM file automatically.
- Build the project if the standalone JAR file does not exist.
- Launch OptaWeb Vehicle Routing by taking a single region argument or by selecting the region interactively.

## 3.1. Getting started

### *Prerequisites*

- `optaweb-vehicle-routing` repository is cloned on your computer.
- Internet access is available.
- Java 8 or higher is installed.

### *Procedure*

1. Change directory to the project root.
2. Run `./runLocally.sh`.
3. Confirm the download of the OSM file needed to work with the built-in data set.

The application starts after the OSM file is downloaded. Open <http://localhost:8080> in a web browser to work with OptaWeb Vehicle Routing.

### **NOTE**

The first start may take a few minutes because the OSM file needs to be imported by GraphHopper and stored as a road network graph. Subsequent runs will load the graph from the file system without importing the OSM file and will be significantly faster.

## 3.2. Interactive mode

Using the interactive mode, you can see the list of downloaded OSM files and country codes assigned to each region. You can use the interactive mode to download additional OSM files from Geofabrik without visiting the website and choosing a destination for the download.



### 3.2.1. Download a new region using the script

#### Procedure

1. Run `./runLocally.sh -i`.
2. Enter `d` to show the download menu.
3. Go to a region by entering its ID and then entering `e`.
4. Repeat the previous step until you see a list with the region you want to download.
5. Download a region by entering its ID and then entering `d`.

### 3.2.2. Select a region and run OptaWeb Vehicle Routing

#### Procedure

1. Run `./runLocally.sh -i`.
2. Select a region from the list of downloaded regions by entering its ID.
3. Confirm the project build if it hasn't been built yet.
4. Confirm starting OptaWeb Vehicle Routing using the selected region.

## 3.3. Run OptaWeb Vehicle Routing with a selected region non-interactively

Use the non-interactive mode to specify an existing region and start OptaWeb Vehicle Routing with a single command. This is useful for switching between regions quickly or when doing a demo.

#### Procedure

Run `./runLocally.sh <REGION>`.

## 3.4. Run OptaWeb Vehicle Routing in air distance mode

OptaWeb Vehicle Routing can work in air distance mode that calculates travel times based on the distance between two coordinates. Use this mode in situations where you need to get OptaWeb Vehicle Routing up and running as quickly as possible and do not want to use an OSM (OpenStreetMap) file. Air distance mode is only useful if you need to smoke-test OptaWeb Vehicle Routing and you do not need accurate travel times.

#### Procedure

Run the `runLocally.sh` script with `--air` argument to start OptaWeb Vehicle Routing in air distance mode:

```
./bin/run.sh --air
```

## 3.5. Tweak the data directory

- To use a different data directory, write its absolute path to the `.DATA_DIR_LAST` file at the project root.
- To change country codes associated with a region, edit the corresponding file under `DATA_DIR/country_codes/`.

For example, you could have downloaded an OSM file for Scotland, for which the script fails to guess the country code. In this case, set the content of `DATA_DIR/country_codes/scotland-latest` to `GB`.

- To remove a region, delete the corresponding OSM file from `DATA_DIR/openstreetmap/` and GraphHopper directory from `DATA_DIR/graphhopper/`.

# Chapter 4. Run OptaWeb Vehicle Routing without the script

Use this section if you cannot use [runLocally.sh](#) to run OptaWeb Vehicle Routing because Bash is not available on your system.

## 4.1. Download routing data

The routing engine requires geographical data to calculate the time it takes vehicles to travel between locations. You must download and store OSM (OpenStreetMap) data files on the local file system before you run OptaWeb Vehicle Routing.

**NOTE** The OSM data files are typically between 100 MB to 1 GB and take time to download so it is a good idea to download the files before building or starting the OptaWeb Vehicle Routing application.

### Procedure

1. Open <http://download.geofabrik.de/> in a web browser.
2. Click a region in the **Sub Region** list, for example **Europe**. The sub region's page opens.
3. In the **Sub Regions** table, download the OSM file (**.osm.pbf**) for a country, for example Belgium.

## 4.2. Create the OptaWeb Vehicle Routing data directory structure

OptaWeb Vehicle Routing reads and writes several types of data on the file system. It reads OSM (OpenStreetMap) files from the **openstreetmap** directory, writes a road network graph to the **graphhopper** directory, and persists user data in a directory called **db**. Create a new directory dedicated to storing all of these data to make it easier to upgrade to a newer version of OptaWeb Vehicle Routing in the future and continue working with the data you created previously.

### Procedure

1. Create the **openstreetmap** directory in your user account **home** directory, for example:

```
$HOME/.optaweb-vehicle-routing
└── openstreetmap
```

2. Move all of your downloaded OSM files (files with the extension **.osm.pbf**) to the **openstreetmap** directory.

The rest of the directory structure will be created by the OptaWeb Vehicle Routing application when it runs for the first time. After that, your directory structure will look similar to the following example:

```
$HOME/.optaweb-vehicle-routing
├── db
│   └── vrp.mv.db
├── graphhopper
│   └── belgium-latest
└── openstreetmap
    └── belgium-latest.osm.pbf
```

## 4.3. Run OptaWeb Vehicle Routing using the **java** command

### *Prerequisites*

- Internet access is available. When OptaWeb Vehicle Routing runs it uses third-party public services such as [OpenStreetMap](#) to display map tiles and provide search results.
- Java 8 or higher is installed.
- The data directory is created at `$HOME/.optaweb-vehicle-routing`.
- A subdirectory called `openstreetmap` with at least one OSM file exists.
- A country code to use in search queries is identified.

### *Procedure*

Enter the following command.

```
java -jar optaweb-vehicle-routing-standalone-7.37.0-SNAPSHOT.jar \  
--app.persistence.h2-dir=$HOME/.optaweb-vehicle-routing/db \  
--app.routing.gh-dir=$HOME/.optaweb-vehicle-routing/graphhopper \  
--app.routing.osm-dir=$HOME/.optaweb-vehicle-routing/openstreetmap \  
--app.routing.osm-file=belgium-latest.osm.pbf \  
--app.region.country-codes=BE
```