

# Using LocatingEngineTimeouts

## What are timeouts used for?

Timeouts are used to limit the search time.

If a timeout occurs during the geocoding process, the geocoder will return all results found so far as quickly as possible. However, these results may differ from the ones you would get without a timeout. This means that maybe a smaller list is returned, or that only towns, but no streets are returned. It may also mean that the desired result will not be returned, because it had not been found until then.

## How do I configure timeouts?

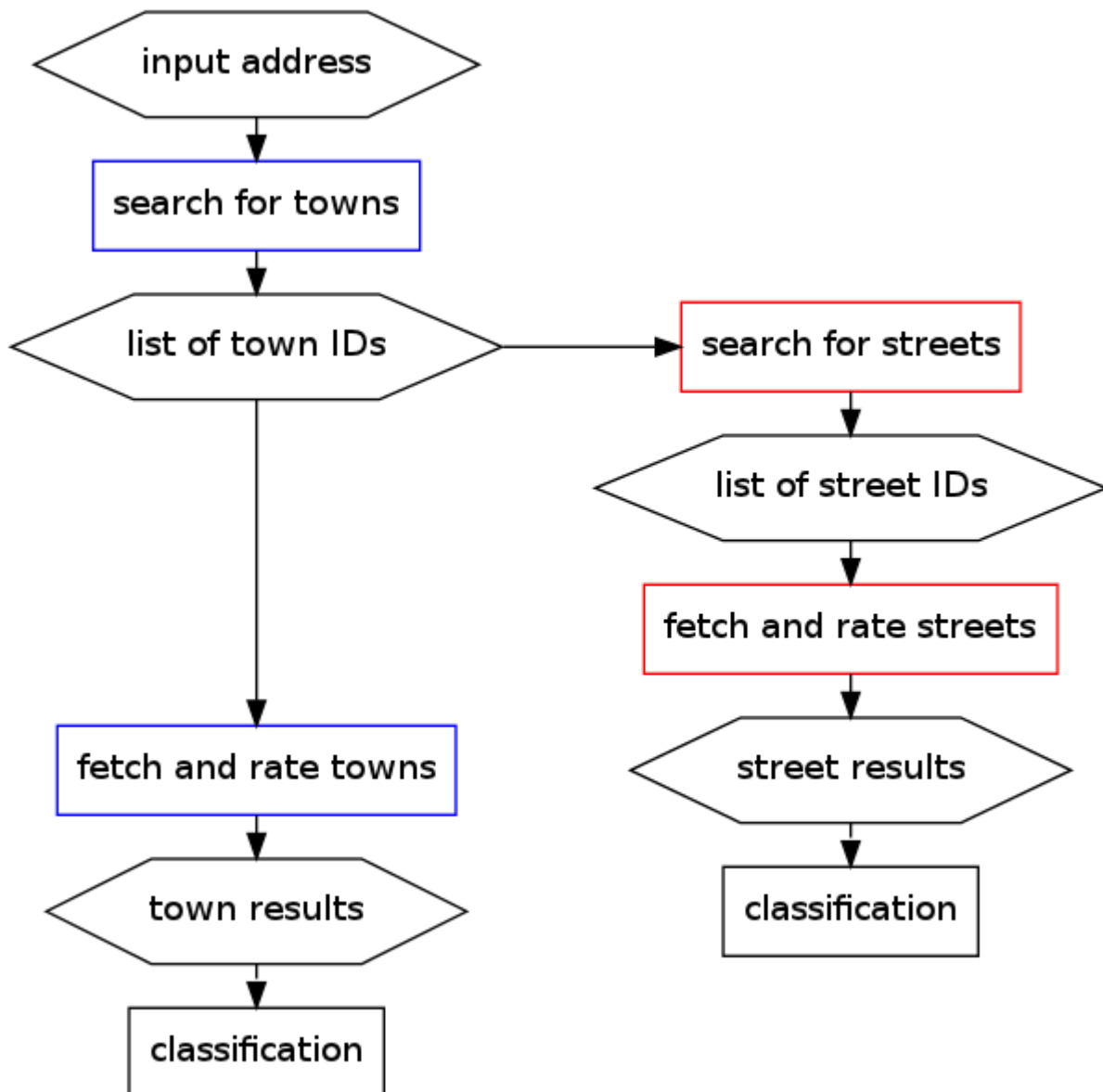
To configure the timeouts, there are two profile parameters [Engine.CitySearchTimeout](#) and [Engine.StreetSearchTimeout](#) that allow to specify separate timeouts for town and street searches.

If they are set to -1, no timeout is used. A positive value indicates a timeout period in milliseconds.

Usually, the [Engine.CitySearchTimeout](#) is equal or greater than the [Engine.StreetSearchTimeout](#) value. This will be explained below.

## How do timeouts work internally?

To understand the timeout you have to consider the following simplified geocoder workflow:



The search starts by searching the index for possible towns. This yields a list of town IDs, which is required as input for the street search, and also for the town processing. At this point, the street search is preferred, so the processing of towns is suspended until it is clear if a good street has been found. Once a good street has been found, it is returned. In case no good street is found, the geocoder will resume the town processing, and return those results.

The timeout functionally now works as follows: At the beginning a timer is started. Whenever the geocoder starts executing one of those tasks denoted by blue and red boxes, it checks if the timer exceeded one of the timeouts. For the blue boxes on the left side, it checks the [Engine.CitySearchTimeout](#), and for the red boxes on the right, it checks the [Engine.StreetSearchTimeout](#).

So, if, say, the street timeout occurs, no tasks on the right hand side are executed any more, and we either get a truncated list of streets, or the "fetch and rate towns" task is started,

yielding a list of towns. If it can complete before the town timeout occurs, we get a list of towns, otherwise the list of towns might be truncated, or we get an empty result.

Keep in mind that it also takes some time to execute the classification and to read out the results, so the geocoding will not return immediately after the timeout has expired.

## How should I set the timeout values?

From the last section we see:

- If you don't want to use any timeouts, set [Engine.CitySearchTimeout](#) and [Engine.StreetSearchTimeout](#) to -1. This way all tasks will be executed.
- If you want to abort the search after a specified time and are not interested in any results, set [Engine.CitySearchTimeout](#) and [Engine.StreetSearchTimeout](#) to the same value. This way, neither town nor street search tasks are executed after the timeout and the geocoder will return as quickly as possible.
- If you'd like to have the towns returned in case the search for streets takes too long, then choose the [Engine.CitySearchTimeout](#) larger than the [Engine.StreetSearchTimeout](#). This way, after the street search timeout occurred, the geocoder will spend some extra time to process the towns and return them. The overall time is still limited by the [Engine.CitySearchTimeout](#).
- If you want to have towns in any case, you can even set the [Engine.CitySearchTimeout](#) to -1.
- It makes little sense to set the [Engine.CitySearchTimeout](#) to a value smaller than the [Engine.StreetSearchTimeout](#), because this way you cannot be sure that the list of town ids can be determined successfully.

## Why do I sometimes get a truncated list of streets, but no towns?

- Even if you specify the [Engine.CitySearchTimeout](#) as -1, you do not always get a complete list of towns.
- This is due to the switch `PreferStreetOnCityStreetConflicts`, which can be turned on or off for the town search, but is always active in the singlefield searches.
- If the switch is on, no town results will be processed as soon as a street result was found.
- So, if the street timeout occurs after one or more streets were found, those streets will be returned.
- If the timeout occurs before a street result was found, all town tasks can be executed and you will get a complete list of towns.