Entity-Relationship Diagram

- **Bus type** — *number of seats*
  - 1,N — means of transport
- **Service** — attributes: *dep.time*, *arr.time*, *cost*
  - 1,1 — means of transport (to Bus type)
  - 1,N — frequency — **Days** (1,N)
  - 1,1 — origin — **Bus Stop** (1,N)
  - 1,1 — destination — **Bus Stop** (1,N)
  - 0,N — milestone — **Town or city** (0,N)
  - 1,1 — carrier — **Company** (1,N)
- **Bus Stop** — attributes: *GPS coord.*, *address*
  - 1,1 — location — **Town or city** (1,N)
  - 1,N — boarding point — **Company** (1,N)
- **Company**

Arrows connect classes and stand for relations between them. The label on them expresses the meaning of the relation and the arrow shows the way it should be read out :

The relation Service -> Company has the label Carrier on it, which means that
• An instance of Service HAS A Carrier, which is an instance of Company
• An instance of Company IS THE carrier OF an instance of Service.

In this case, the cardinalities show that an instance of Service has a single instance of Company as its carrier, but that an instance of Company may be the Carrier of any number of instances of Service.

Rectangles on classes stand for attributes. When the value of a property cannot be listed out conveniently, instead of expressing the property by a relation between classes, attributes are used :

• Companies have a reduced and predictable number of instances, as have Services. That is why the property of a Service (having a carrier that is a company) and the inverse property (a company being the carrier of a service) can be expressed by a relation between the classes Company and Service.
• Departure times or GPS coordinates have a range that is too wide to be expressed as instances of a class. They are thus modelled as attributes. the value of an attribute is entered manually while entering data, whereas instances acting as values are chosen from a list.

We said that the meaningful stops are the beginning and the end ones. These two are user-defined and therefore not part of the data model.

We also said that the trip (combined legs or single leg) might be chosen among the different options that appear after calculation because of two main factors:

• For multiple legs, the shortest waiting time between legs and the possibly related shortest distance between the arrival of a leg and the start of the next one.
• The fact that the access area of the final stop includes the location where the passenger actually wants to go (his home, for instance).

Both these are calculated by the app and thus do not appear in the data model.