Dat.mobility – traffic calculation module

Content

Dat.mobility – traffic calculation module 1

Specification of required input 2

Fixed scenarios 2

The area 2

Additional Social Economical Data 2

Runtime modifications 3

Unreal implementation 3

Expert mode 3

Registration in OUP (Municipality side) 4

Registration in OUP (Dat.mobility side) 4

Technical documentation 7

Specification of required input

As Dat.mobility, we're capable of consuming almost any kind of traffic model format: OmniTRANS, PTV Visum, CUBE, Vissim, Aimsun,.. We can use this data to construct the base scenario for the Goudappel Mobility Twin.

If you don't have a traffic model yet, we can create one based on publicly available data, extended with local data (Social Economical Data, like number of inhabitants, jobs, scholars) per zone (e.g. postal codes).

Starting from here, we have 2 possible paths to follow:

A) make fixed scenarios
B) make runtime modifications

Fixed scenarios

For Dt-aCoGo, we used the first path, where municipalities delivered information to modify their 'base scenario'.

For now, we have agreed to deliver 2 data sets to create a new scenario: the area that will be modified AND new figures (Social Economical Data), containing the 'delta': what should be added/removed from the base scenario.

The area

The data set to describe the area must be a valid GeoJSON. It can be a (multi)polygon, just describing the 'study' area. The MobilityTwin calculation module will extend this area with 'influence' and 'boundary' zones.

Additional Social Economical Data

To make a fixed scenario, we use on these fields:

- expected_public_parking_spaces
- new_homes
- residents_forecast_min
- residents_forecast_max
- commercial_space
- commercial_space_unit (ignored for now)
- jobs_forecast_min
- jobs_forecast_max
- students_forecast_min
- students_forecast_max

When it is expressed in a JSON file like this, it can be consumed by out processes.

```
{
```

```
"header": {
"description": "This JSON contains planned values for two areas within the Spoorzone development context. It includes housing, resident forecasts, commercial space, job forecasts, and the expected number of public parking spaces."
},
"data": [
{
"area": "Spoorzone",
"expected_public_parking_spaces": 380,
"new_homes": 39,
"residents_forecast_min": 82,
"residents_forecast_max": 82,
"commercial_space": 2739,
"commercial_space_unit": "m²",
"jobs_forecast_min": 80,
"jobs_forecast_max": 80,
"students_forecast_min": 0,
"students_forecast_max": 0
}
]
}
```


When fields are lacking, we will estimate them where possible, otherwise they will be ignored. For instance, residents based on the new_homes,

Runtime modifications

When your Digital Twin allows you to e.g. close roads, or modify the figures above, the calculation models of Dat.mobility can recalculate the whole network, using the appropriate endpoints. A new scenario will be made within 10-60 seconds, depending on the size of the network.

Unreal implementation

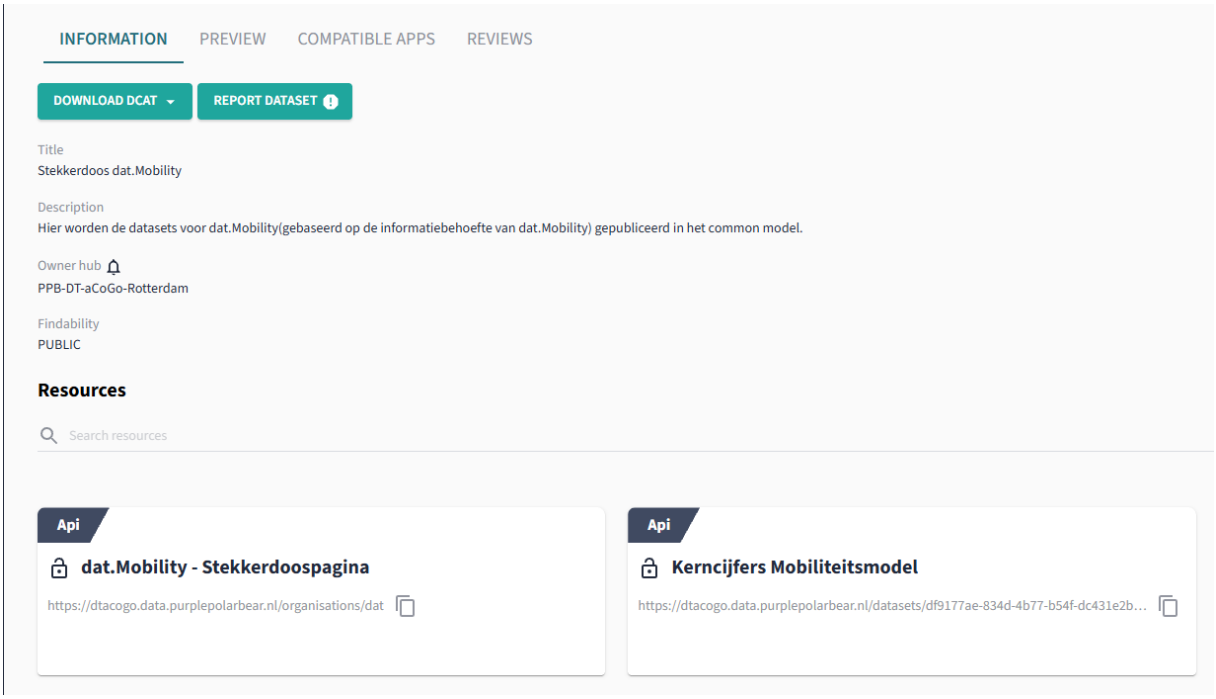
The current implementation within Unreal allows you to navigate through time, and the datasets enlisted in the OUP have tags containing a start-time and end-time. These are taken into account when using the time slider. There are three scenarios registered per municipality the ‘before’ situation, the ‘during’ and the 'after’ scenario. Each of these will be shown on your Digital Twin on the appropriate time.

Expert mode

Unfortunately, the expert mode is not available, except when you do have a subscription/contract with Goudappel, to maintain your traffic model. In that case, you already have your model available in OmniTRANS (Powered by Bently).

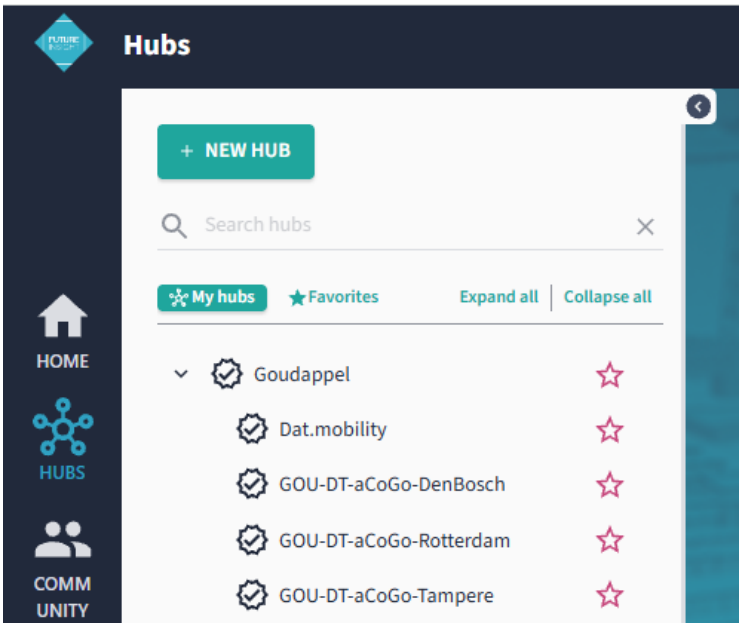
Registration in OUP (Municipality side)

The municipal can register their data set in the OUP, in order to publish them to Goudappel. When using the correct format (see above, the JSON example) and notifying Dat.mobility there is a new data set available the process can be started. Besides the registration of the 'delta' values, contact is required to exchange the traffic model. It can be done using the OUP, but in this project we choose not to.

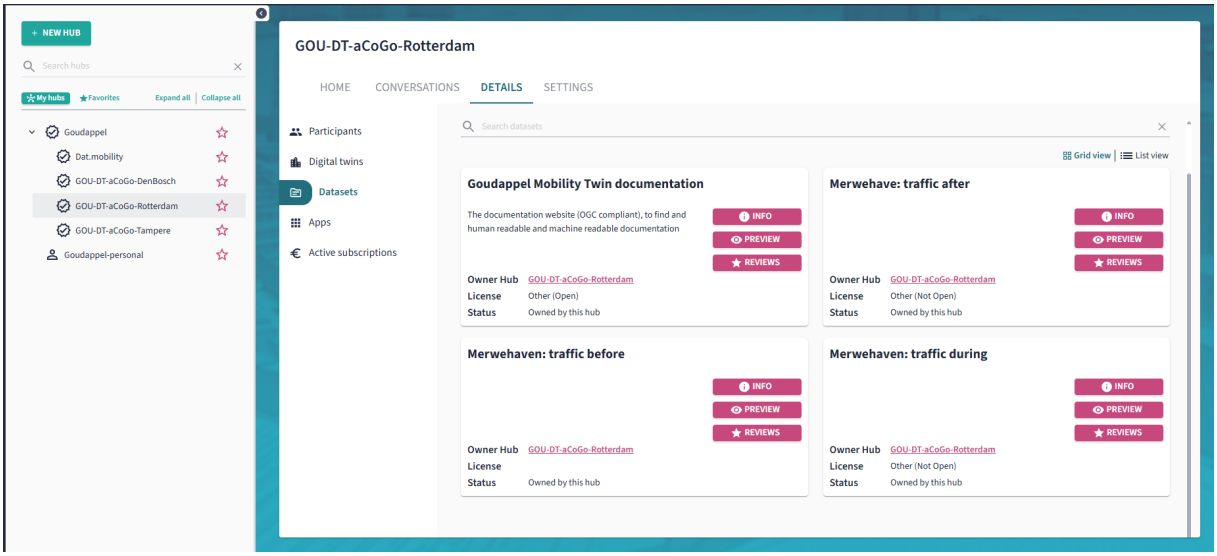


Registration in OUP (Dat.mobility side)

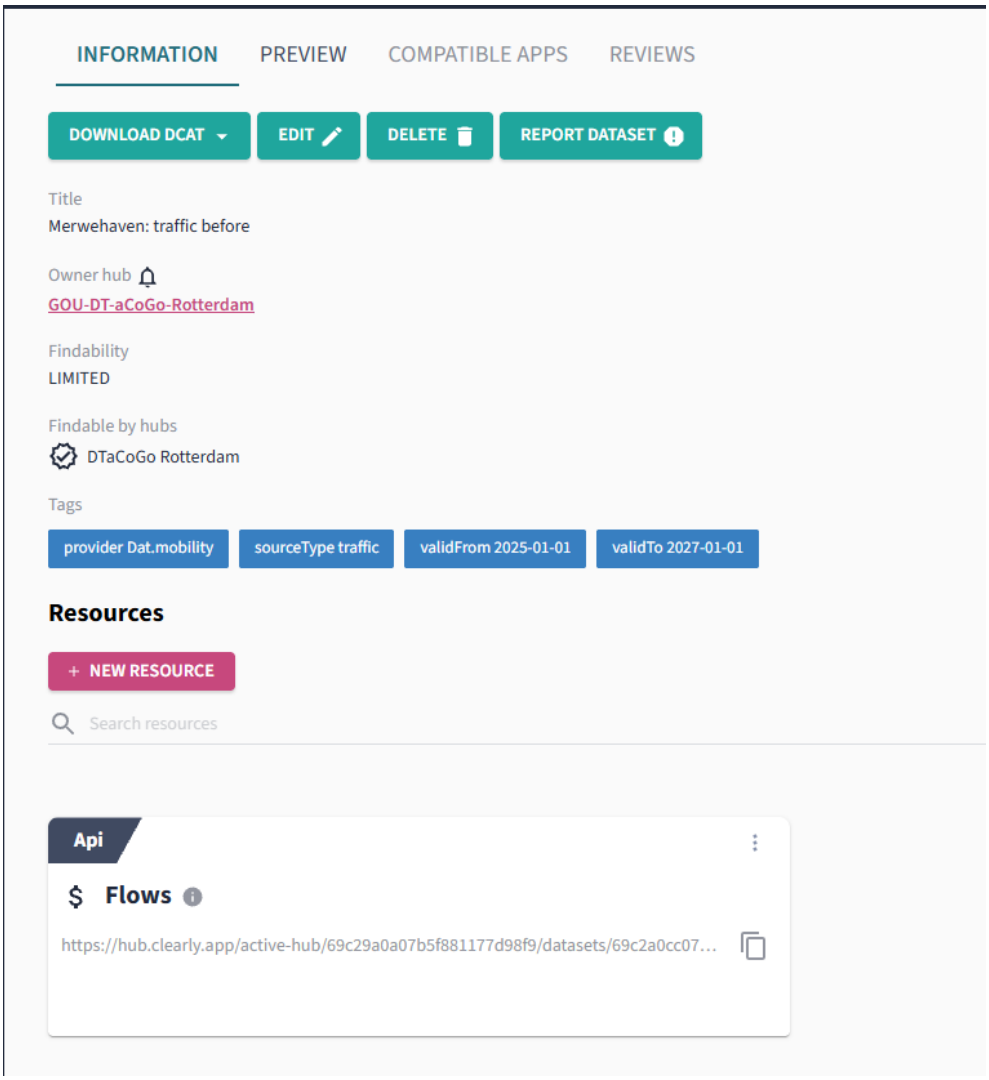
The OUP has registered a hub, specific for Goudappel, where the expert-mode output (the fixed scenarios) will be registered per municipal.



For instance, if you look at the configuration of Rotterdam, you will see the 3 registered datasets, and an additional dataset describing the technical interface.



If you look at one of the datasets, you will find here the ‘validFrom’ and ‘validTo’, to specify the start- and end date of the dataset. In this case, the scenario ‘before’ of the Merwehaven, in Rotterdam.



These datasets can only be accessed by Rotterdam, since it has a unique authentication method (also registered in the OUP). The authentication process is implemented using the OAuth2-client credential flow. No-one else than Rotterdam can access these datasets.



Technical documentation

Generic description of the platform

The GMT (Goudappel Mobility Twin) is a fully OGC API Processes compliant restful API, containing a lot of functions. The one that is applied in this project is to simulate the 'flow' (=number of cars per hour) on every road segment in the network, depending on the weather, the temperature, modality, date and time.

There are other functions available (e.g. road closures, modifications of Social Economical Data), but these are not applied here.

Definition of information needs and translation towards a CIM.

The required input for this process is to create a fixed scenario:

a) a traffic model, including configuration details of the model (most traffic modelling software allows you to configure at lot, to tailor-make your solution)
b) Social Economical Data (as described before)

For the input to 'play' with the GMT (conditional flows), we need an API call containing a date + time, weather conditions, temperature conditions and the purpose of the travels (e.g. like 'work', or 'school'). This is of course not available when these purposes are not used in the traffic model.

Description of connecting to the OUP for data services.

See above.

Description of API connections.

The API description (OpenAPI) can be found here: https://mobilitytwin.goudappel.nl/api.

The used endpoint is "/collections/segments/items".

Description of the Asset Information Model.

The model used and published by Purple Polar Bear is the 'common data model' used. The JSON example in this document complies to it.

Overview of the overall configuration of the OUP including hub structure and metadata.

See above, in the screenshots.

Provision of a lightweight SDK for developers such as TPG.

{TBD}

User manual of the platform for setting up calculation models.

This is out of scope. The calculation model is maintained in a proprietary solution, mostly used by employees from the municipality or a consultancy firm.

User manual for input from the Unreal Digital Twin.

??

User manual for the Unreal Digital Twin and the plug-ins for the model providers.

??

Overview of the way IAM works in relation to OUP and Unreal

??