H2020-MG-2014-2015

MOBILITY FOR GROWTH 2014-2015

H2020-MG-2014_TwoStages

Sustainable **U**rban **C**onsolidation
Centr**ES** for con**S**truction

Database Structure

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

Dissemination level	***Public***
Workpackage and task	***WP2 / Task 2.3***
Author(s)	***UNIMORE***
Contributor(s)	***LIST***
Due date of deliverable	**29/02/2016**
Submission date of deliverable	**31/10/2016**
Status (F: final, D: draft)	**F**

Document Control Sheet

Project number	**633338**		
Project Acronym	**SUCCESS**		
Work Package	**WP3**		
Version	2.0		
Issue	**Version**	**Dates**	**Content**
	0.1	11/02/2016	First draft: how we defined the DB structure
	0.2	18/02/2016	Second draft: DB structure
	0.3	22/02/2016	More detailed explanation of partners access and of import-export, how to compute KPIs.
	0.4	26/02/2016	Implementation of the last comments and improvements
	1.0	29/02/2016	Final version
	1.1	17/09/2016	Integration+List comments
	1.2	30/09/2016	Revision UNIMORE
	2.0	31/10/2016	Final Version

Classification of this report:

Draft	
Final	**x**
Confidential	
Restricted	
Public	**x**

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

LIST project coordinator:

5 Avenue Des Hauts Fourneaux, Esch Sur Alzette 4362, Luxembourg

Contact details:

FERRERO Francesco, +352 275 888 22 27, francesco.ferrero@list.lu
GUERLAIN Cindy, +352 275 888 67 46, cindy.guerlain@list.lu
SCHWARTZ Thomas, +352 425 991 25 21, thomas.schwartz@list.lu

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

Summary

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

Introduction

In this document we report how we provided the Databases (DB) structure for the SUCCESS project. We firstly describe the DB inputs. We describe the DB components: tables, attributes, and relationships. We report the used software and the servers on which we set the DB, providing the different access type to the different stakeholders. We briefly describe the import and export functions. Moreover, we show how to perform the quality check with a dedicated database and the computation of the KPIs thanks to the main DB and to the GIS database. Finally, we report some graphical output based on data.

This new version of the document (2.0) became necessary to take into account the adjustments to the list of KPIs requested by the consortium on the basis of the first feedbacks from the data collection phase and described in version 2.0 of D2.2 KPIs and methodologies for construction logistic.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

1 DB Input: the Data Collection

The SUCCESS Database (DB) has an important role in the SUCCESS project being the place where all the data collected in the Task 2.4 (Data Collection) will be stored. In Task 2.3 (Database Structure) we define the DB structure and provide the DB to be used in the next steps of the project.

To define the DB structure we started from the DB input, which is represented by the following data collection files, designed in Task 2.4 (Data Collection):

- *SUCCESS_Delivery_Pick up tracking board template.xlsx*
- *SUCCESS_Activity monitoring template.xlsx*
- *SUCCESS_Posteriori analysis.xlsx*

All these files are Excel files that are made of more than one sheet and tables. In the following we will present each of them in detail.

1.1 SUCCESS_Delivery_Pick up tracking board template.xlsx

The *SUCCESS_Delivery_Pick up tracking board template.xlsx* designed is an Excel file made of two sheets representing the two kinds of trips that can occur in the construction sites, namely, *delivery* and *pickup*, for deliveries of materials and pickups of wastes and materials.

Each sheet is divided into General Information, Trip Characteristics, Truck Characteristics, Loading/Unloading Characteristics, and Cause and Impact of Delay. Each of these set of information are separated into many columns providing numerical and non-numerical data. For each datum the type or measure unit is also specified. In Figure 1, we provide a highlight of the file, for a more specific description we address the interested reader to the Excel file: *SUCCESS_Delivery_Pick up tracking board template.xlsx*

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

General information														
ound trip or Direct trip	Date	Material name	Supplier name	Number of product	Number of items to manipulate	Item packaging	Material weight		Haulier name	Address of the loading location				Ad
roundtrip or direct)	dd/mm/yyyy	character string	character string	Number	Number	character string	kg	m3 (if concrete)	character string	Street	Postcode	City	Country	

Figure 1 Highlight of the Delivery tracking board Excel file.

1.2 SUCCESS_Activity monitoring template.xlsx

SUCCESS_Activity monitoring template.xlsx is an Excel file made of five sheets. The sheets are the following:

- Material Installation
- Storage Monitoring
- Logistic Activity Monitoring
- Several handling
- Congestion Monitoring
- Site Supervisor Interview

In the following we describe briefly each of the sheets. We do not report any figure of the data template files because they are similar to the Delivery tracking board template.

Material installation takes into account the time needed to install materials with respect to efficient work and non-efficient work. It is divided into the following sections: General Information, Efficient work, Setup on construction site, Setup on material, Rework, Waiting and Looking and transport.

Storage Monitoring collects data about the storage of materials, with its date, the storage area, the quantity of materials, etc.

Logistic Activity Monitoring is divided into the 5 days of the week and is needed to collect the information in time that is dedicated to all the selected logistic activities, such as order management, material validation, deliveries issues,

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

management of the site requirements, manage the site movements and storage area and reverse logistics management.

Several handling collects the information of the occurrence of several handling of the same material from one place to another of the construction sites, the reason of moving, the duration and the used equipment.

Congestion Monitoring helps in detecting when congestion episodes occur on site, and the concerning information such as date, reason of congestion, etc.

Site Supervisor Interview is made of two different tables: the first one is related to the *Accident monitoring*, whose meaning is to collect information on accidents on site, and the second one regards the *Material waste* and the related information.

1.3 SUCCESS_Posteriori analysis.xlsx

SUCCESS_Posteriori analysis.xlsx is also an Excel file made of two sheets, the *Cost of unsorted bins* and the *Haulier route*. The first sheet considers mainly the type and the costs of transporting and treating unsorted bins. The *Haulier route* sheet takes into account the entire path followed by the haulier, considering also the supplier, the production location, etc.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

2 The Entity-Relationship Models

To produce the DB structure, we thus started from the Excel files described above. First of all we decided to provide a database for each of set of data that we presented in the previous chapter. Namely:

- Delivery and Pickup
- Material Installation
- Storage Monitoring
- Logistic Activity Monitoring
- Several handling
- Congestion Monitoring
- Accident Monitoring
- Material waste
- Cost of unsorted bins
- Haulier route

For each of those DB we decomposed the data in relevant tables and we inserted, for each table, the data of the Excel file that will represent an attribute, and thus a column of the table. Among the attributes, we detected those that could be the key of the tables. The foreign keys have been detected to provide links among the tables. All this action have been performed and included in the entity-relationship (ER) scheme, reported in the following.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

2.1 The Delivery and Pickup Database

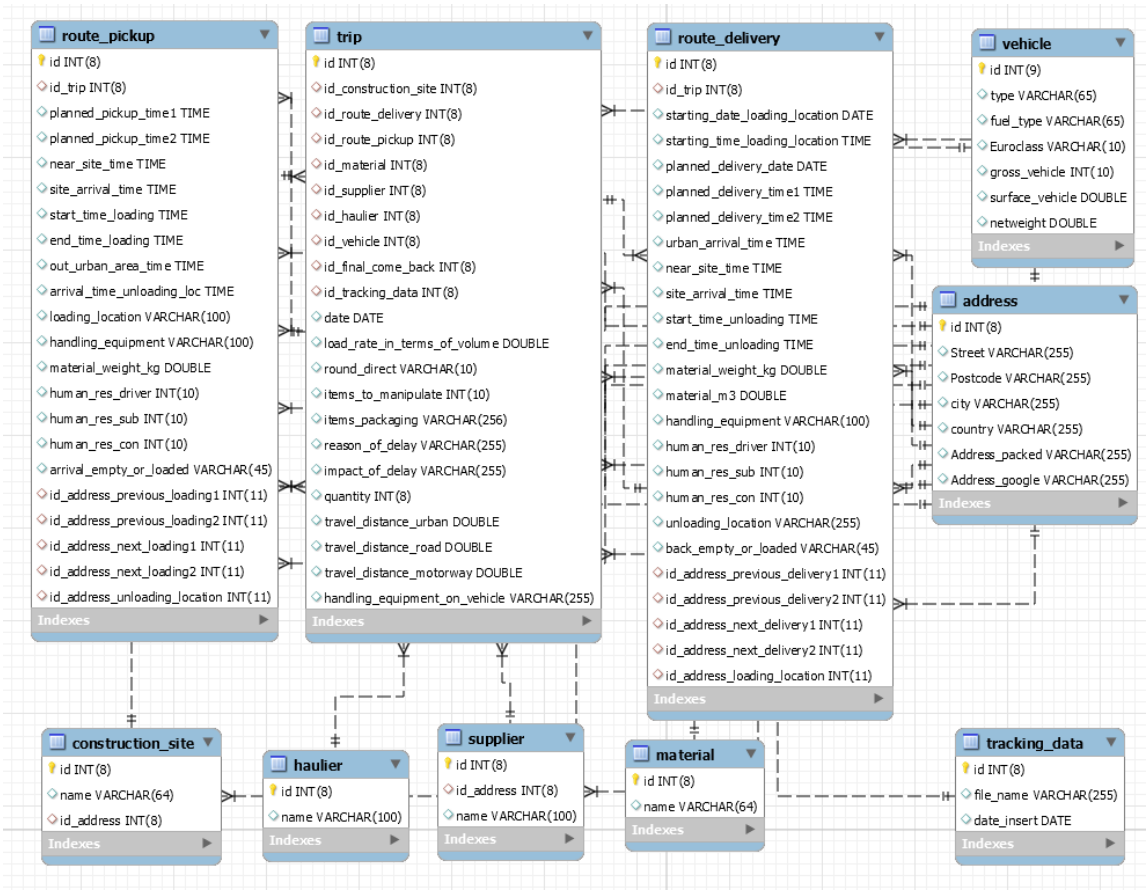

Figure 2 A highlight on the ER scheme of the Delivery and Pickup Database

To better represent data, we separated and aggregated them in a set of Tables. We considered Table Trip as the main table of the DB. All the tables are namely:

- Table Address,
- Table Construction_site,
- Table Haulier,
- Table Material,
- Table Route_delivery,
- Table Route_pickup,
- Table Supplier,
- Table Trip,
- Table Vehicle,
- Table Tracking_data.

In the following, we provide all the information for each one of the tables, reporting the attributes (with its datatypes), the primary keys, and the foreign keys (with the type of relationships between the tables). Finally a short description of each attribute is provided.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

This DB and its structure have been modified with respect to the previous release for the following reasons:

- To better represent *the SUCCESS_Delivery_Pick up tracking board template.xlsx* and provide a more direct understanding of all the attributes names.
- To ease the DB structure in order to provide a more consistent computation of the KPIs provided by the views. With this aim we moved some attributes from the *Table Trip* to the *Table Route_pickup* and *Route_delivery* and vice-versa (e.g. *address_previous_delivery*, *address_next_loading*, etc.).
- To include some modification that had been applied into the data collection file *SUCCESS_Delivery_Pick up tracking board template.xlsx* (see, e.g., the introduction of *planned_delivery_date*).
- To provide an improved representation of the addresses into the database: the addresses are now represented in a more homogenous way, and it is now easier to put them on a map (see, e.g., attributes *address_packed,* and *address_google*).

2.1.1 Table Trip

The Table Trip defines the trips entering the construction site, the so-called deliveries, and the trips exiting the construction site, the so-called pickups. This is the most important table of the Delivery and Pickup DB and it includes the majority of the information of each trip.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Unique identification number
Id_construction_site	Integer		Referred to *construction_site.id* (many to one)	Referred to *construction_site.id* (many to one)
Id_route_delivery	Integer		Referred to *route_delivery.id* (one to one)	Identification number of the route delivery
Id_route_pickup	Integer		Referred to *route_pickup.id* (one to one)	Identification number of the route pickup
Id_material	Integer		Referred to	Identification

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

		material.id (many to one)	number of the transported material
Id_supplier	Integer	Referred to *supplier.id* (many to one)	Identification number of the supplier
Id_haulier	Integer	Referred to *haulierl.id* (many to one)	Identification number of the haulier transporting the material
Id_vehcile	Integer	Referred to *vehicle.id* (many to one)	Identification number of the vehicle used
Id_final_come_back	Integer	Referred to *address.id* (many to one)	Identification number of the address of final come back place
Id_tracking_data	Integer	Referred to *tracking_data.id* (many to one)	Identification number that defines the data upload
Date	Date		Date of the trip
Load_rate_in_terms_of_volume	Double		Volume load rate on the vehicle
Round_direct	Varchar		Defines the type of trip, round or direct
Items_to_manipulate	Integer		Number of items to manipulate
Items_packaging	Varchar		Include the items packaging
Reasons_of_delay	Varchar		Reasons of trip delay
Impact_of_delay	Varchar		Impact of delay
Quantity	Integer		Quantity of transported material

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

Travel_distance_urban	Double	Distance travelled in urban area for the whole trip in km
Travel_distance_road	Double	Distance travelled in rural area for the whole trip in km
Travel_distance_motorway	Double	Distance travelled on motorways for the whole trip in km
Handling_equipment_on_vehicle	Varchar	Type of handling equipment on vehicle

2.1.2 Table Construction_site

The Table Construction_site includes the main characteristics of the construction sites.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Unique identification number of the construction site
Name	Varchar			Name of the construction site
Id_address	Integer		Referred to *address.id* (many to one)	Identification number of address of the construction site

2.1.3 Table Haulier

Table Haulier includes the characteristics of the hauliers.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Unique identification number of the haulier
Name	Varchar			Name of the haulier

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

2.1.4 Table Material

Table Material includes the characteristics of transported materials.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Unique identification number of each material
Name	Varchar			Name of the material

2.1.5 Table Route_delivery

The Table Route_delivery reports the information that is typical of a delivery trip.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Unique identification number of the route delivery
Id_trip	Integer		Referred to *trip.id* (one to one)	Identification number of the trip
Starting_date_loading_location	Date			Starting date at the last loading location
Starting_time_loading_location	Time			Starting time at the last loading location
Planned_delivery_date	Date			Date of delivery planned
Planned_delivery_time1	Time			First time of delivery planned
Planned_delivery_time2	Time			Second time of delivery planned
Urban_arrival_time	Time			Arrival time at the urban area
Near_site_time	Time			Time of arrival near the construction site
Site_arrival_time	Time			Arrival time at the construction site

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

Start_time_unloading	Time		Starting time of unloading
End_time_unloading	Time		Ending time of unloading
Material_weight_kg	Double		Weight of materials in kg
Material_m3	Double		Weight of materials in cubic meters
Handling_equipment	Varchar		Equipment used for unloading
Human_res_driver	Integer		Tells if driver is involved in the unloading
Human_res_sub	Integer		Number of people from the subcontractor involved in the unloading
Human_res_con	Integer		Number of people from the contractor involved in the unloading
Unloading_location	Integer		Unloading location inside the construction site
Back_empty_or_loaded	Varchar		Defines if the vehicle comes back is empty loaded
Id_address_previous_delivery1	Integer	Referred to *address.id* (many to one)	Identification number of the address of the previous delivery
Id_address_previous_delivery_2	Integer	Referred to *address.id* (many to one)	Identification number of the address of the second last delivery
Id_address_next_delivery1	Integer	Referred to *address.id* (many to	Identification number of the address of the next delivery

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

		one)	
Id_address_next_delivery_2	Integer	Referred to *address.id* (many to one)	Identification number of the address of the second next delivery
Id_address_loading_location	Integer	Referred to *address.id* (many to one)	Identification number of the address of the loading location

2.1.6 Table Route_pickup

The Table Route_pickup reports the information that is typical of a pickup trip.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Unique identification number of the route pickup
Id_trip	Integer		Referred to *trip.id* (one to one)	Identification number of the trip
Planned_pickup_time1	Time			First time of pickup planned
Planned_pickup_time2	Time			Second time of pickup planned
Near_site_time	Time			Time of arrival near the construction site
Site_arrival_time	Time			Arrival time at the construction site
Start_time_loading	Time			Starting time of loading
End_time_loading	Time			Ending time of loading
Out_urban_area_time	Time			Arrival time at the end of the urban area
Arrival_time_unloading_loc	Time			Time of arrival at the unloading location

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

Loading_location	Varchar		Loading location inside the construction site
Handling_equipment	Varchar		Equipment used for loading
Material_weight_km	Double		Weight of materials in kg
Human_res_driver	Integer		Tells if the driver is involved in the loading
Human_res_sub	Integer		Number of people from the subcontractor involved in the loading
Human_res_con	Integer		Number of people from the contractor involved in the loading
Arrival_empty_or_loaded	Varchar		Defines if the vehicle arrives empty loaded
Id_address_previous_loading1	Integer	Referred to *address.id* (many to one)	Identification number of the address of the previous loading
Id_address_previous_loading2	Integer	Referred to *address.id* (many to one)	Identification number of the address of the last second loading
Id_address_next_loading1	Integer	Referred to *address.id* (many to one)	Identification number of the address of the next loading
Id_address_next_loading1	Integer	Referred to *address.id* (many to one)	Identification number of the address of the second next loading
Id_address_unloading_location	Integer	Referred to *address.id* (many to one)	Identification number of the address of the unloading location

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

2.1.7 Table Supplier

Table supplier includes the information regarding the suppliers.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Unique identification number of the supplier
Id_address	Integer		Referred to *address.id* (many to one)	Identification number of address of the construction site
Name	Varchar			Name of the supplier

2.1.8 Table Vehicle

Table Vehicle reports the information of the vehicles used for the trips.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Unique identification number of the vehicle
Type	Varchar			Type of vehicle
Fuel_type	Varchar			Fuel type of the vehicle
Euroclass	Varchar			Type of euroclass of the vehicle
Gross_vehicle	Integer			Gross vehicle weight rating
Surface_vehicle	Double			Surface of the vehicle
Netweight	Double			Net weight of the vehicle

2.1.9 Table Address

The Table Address includes the address with its specific components, when needed.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Unique identification number

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

		of the address
Street	Varchar	Street and number
Postcode	Varchar	Zip code
City	Varchar	City of the address
Country	Varchar	Country of the address
Address_packed	Varchar	Complete address
Address_google	Varchar	Address as reported on google

2.1.10 Table Tracking_data

In Table Tracking_data we store the information on the files uploaded on the database. For each upload a row of the Table Tracking_data is created, in this way the file name and the date of upload of the file to the DB are tracked. In order to keep track of data, we also store the position of each datum uploaded in a mirror database.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
File_name	Varchar			Name of the imported file
Date_insert	date			Date of import

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

2.2 Material Installation Database

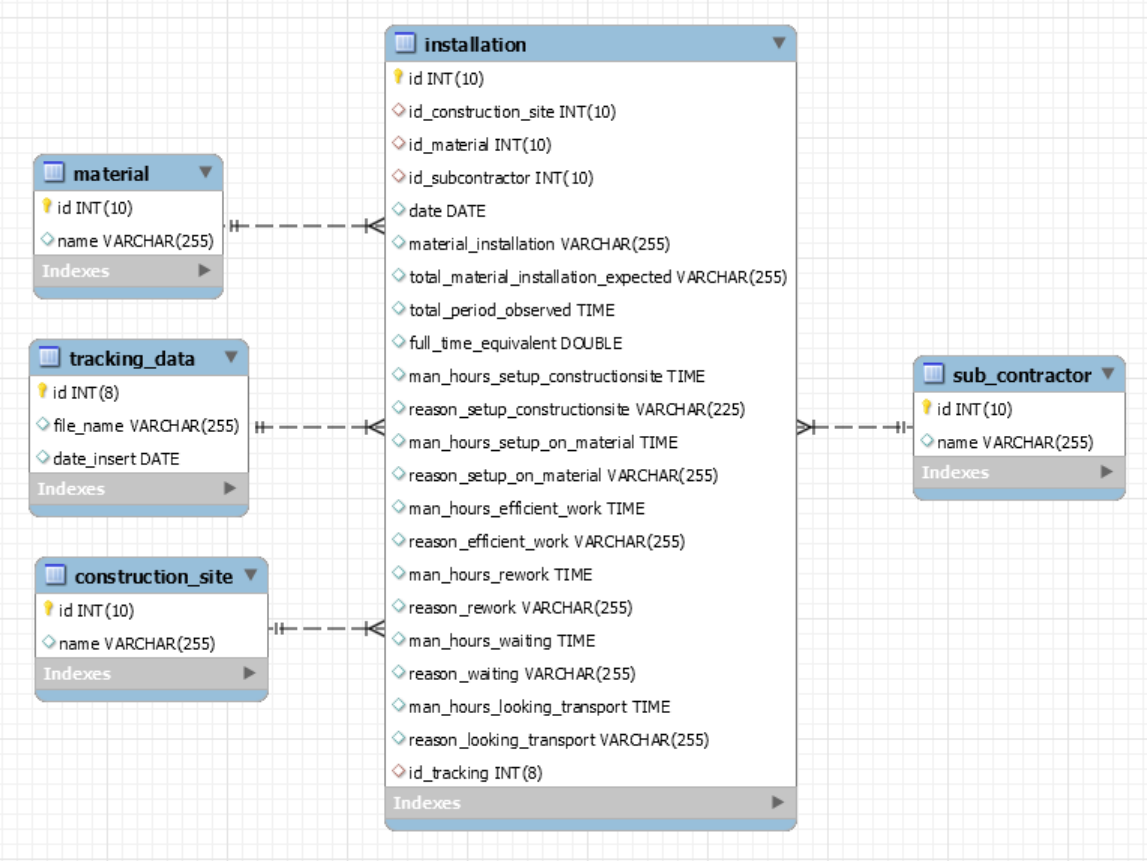

Figure 3 ER scheme of the Material Installation DB.

The Material Installation DB is structured into 5 tables, namely:

- Table Installation
- Table Material
- Table Sub_contracor
- Table Construction_site
- Table Tracking_data

We describe those tables in detail in the following.

2.2.1 Table Installation

This table represents the data deriving from the observation of works on site, and the subdivision of the work with respect to time.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Id_construction_site	Integer		Referred to	Identification

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

		construction_site.id (many to one)	number of the Table *construction_site*
Id_material	Integer	Referred to *material.id* (many to one)	Identification number of the Table *material*
Id_subcontractor	Integer	Referred to *sub_contractor.id* (many to one)	Identification number of the Table *sub-contractor*
Date	Date		Date of observation
Material_installation	Varchar		Material to be installed or installation
Total_material_installation_expected	Varchar		Total quantity of installation material
Total_period_observed	Time		Total period observed
Full_time_equivalent	Double		Full time equivalent
Man_hours_setup_constructionsite	Time		Man-hours of setup on the construction site for the material installation
Reason_setup_constructionsite	Varchar		Reason of the setup on the construction site
Man_hours_setup_on_material	Time		Man-hours of setup on the material for the material installation

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

Reason_setup_on_material	Varchar		Reason of the setup on the material
Man_hours_efficient_work	Time		Man-hours for the efficient work
Reason_efficient_work	Varchar		Reason of efficient work
Man_hours_rework	Time		Man-hours of rework
Reason_rework	Varchar		Reason of rework
Man_hours_waiting	Time		Man-hours spent waiting
Reason_waiting	Varchar		Reason of waiting
Man_hours_looking_transport	Time		Man-hours spent looking for material and for transport
Reason_looking_transport	Varchar		Reason of looking and transport
Id_tracking	Integer	Referred to *tracking_data.id* (many to one)	Identification number of the Table *tracking_data*

2.2.2 Table Material

Table Material includes the characteristics of the materials.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number of the material

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

Name	varchar	Name of the material

2.2.3 Table Sub_contractor

Table Sub_contractor includes the characteristics of sub-contractors.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number of the subcontractor
Name	Varchar			Name of the subcontractor

2.2.4 Table Construction_site

Table Construction_site includes the characteristics of the construction site.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Name	varchar			Name of the construction site

2.2.5 Table Tracking_data

In Table Tracking_data we keep track of the imported data.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
File_name	Varchar			Name of the imported file
Date_insert	date			Date of import

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

2.3 Storage Monitoring Database

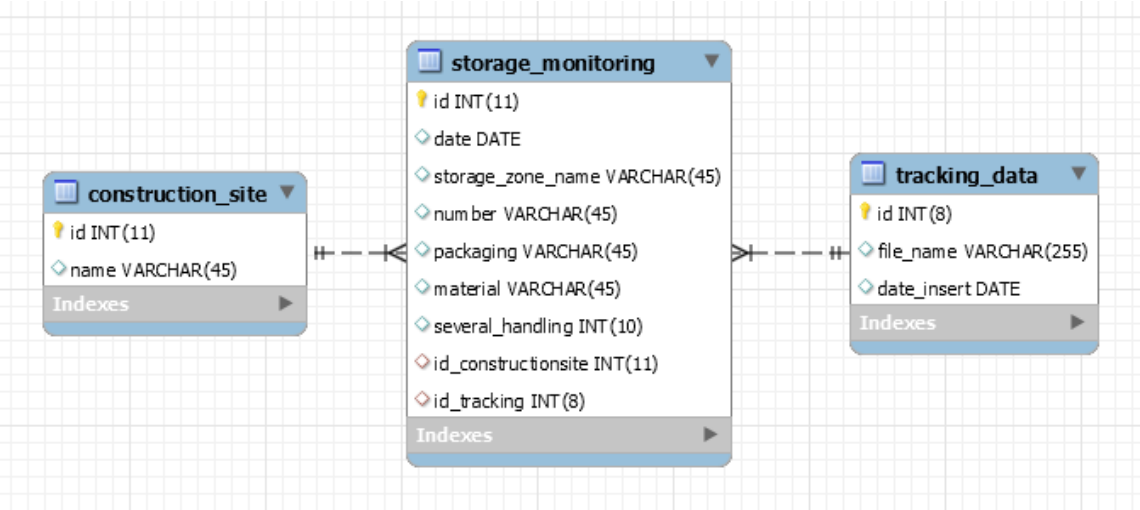

Figure 4 ER scheme of the Storage Monitoring DB

The database Storage Monitoring is made of 3 tables that are explained in the following, namely:

- Table Storage_monitoring
- Table Construction_site
- Table Tracking_data

2.3.1 Table Storage_monitoring

This Table reports the storage monitoring information.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Date	Date			Date of observation
Storage_zone_name	Varchar			Name of the storage area
Number	Varchar			Number of objects
Packaging	Varchar			Type of packaging
Material	Varchar			Type of material
Several_handling	Integer			Number of handling occurrences
Id_constructionsite	Integer		Referred to *construction_site.id* (many	Identification number of Table *construction_site*

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

		to one)	
Id_tracking	Integer	Referred to *tracking_data.id* (many to one)	Identification number of Table *tracking_data*

2.3.2 Table Construction_site

Table Construction_site includes the characteristics of the construction site.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Name	varchar			Name of the construction site

2.3.3 Table Tracking_data

In Table Tracking_data we keep track of the imported data.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
File_name	Varchar			Name of the imported file
Date_insert	date			Date of import

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

2.4 Logistic Activity Monitoring Database

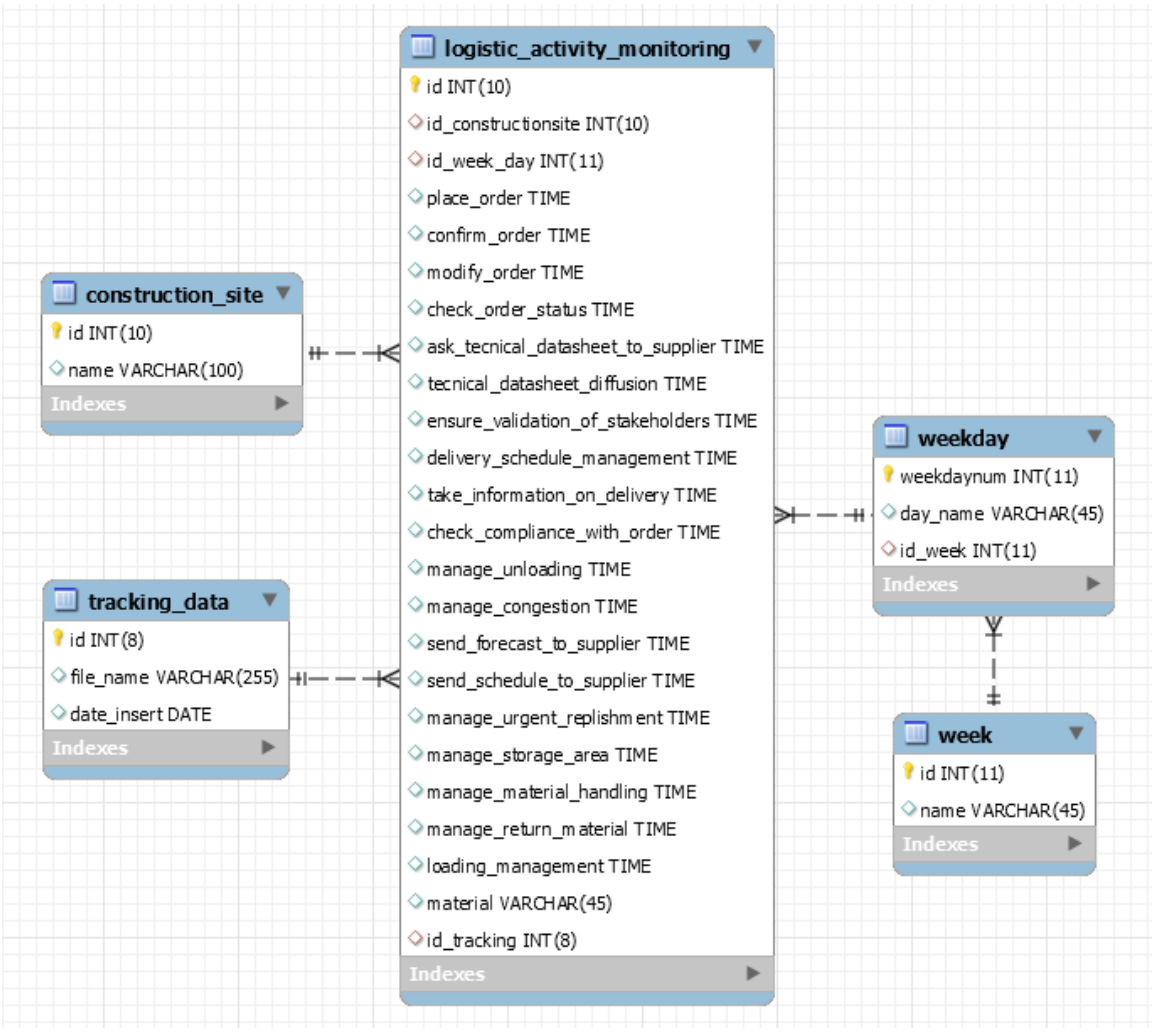

Figure 5 ER scheme of the Logistic Activity Monitoring DB

The Logistic Activity Monitoring DB is made of 5 tables, namely:

- Table Logistic_activity_monitoring
- Table Weekday
- Table Week
- Table Construction_site
- Table Tracking_data

That we explain in detail in the following.

2.4.1 Table Logistic_activity_monitoring

This is the main Table of the DB. Where the time dedicated to all the activities is stored.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Id_constructionsite	Integer		Referred to *construction_site.id* (many to one)	Identification number of Table *construction_site*
Id_weekday	Integer		Referred to *weekday.weekdaynum* (many to one)	Identification number of Table *weekday*
Place_order	Time			Time dedicated to place orders during the selected week
Confirm_order	Time			Time dedicated to the confirm orders during the selected week
Modify_order	Time			Time dedicated to the modify orders during the selected week
Check_order_status	Time			Time dedicated to the check orders status during the selected week
Ask_technical_datasheets_to_supplier	Time			Time dedicated to ask technical datasheet to suppliers during the selected week
Technical_datasheet_diffusion	Time			Time dedicated to technical datasheet diffusion during the selected week
Ensure_validation_of_stakeholders	Time			Time dedicated to ensure validation of

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

		the stakeholders during the selected week
Delivery_schedule_management	Time	Time dedicated to the delivery schedule management during the selected week
Take_information_on_delivery	Time	Time dedicated to take information on deliveries during the selected week
Check_compliance_with_order	Time	Time dedicated to check compliance with the orders during the selected week
Manage_unloading	Time	Time dedicated to manage the unloading during the selected week
Manage_congestion	Time	Time dedicated to manage congestion during the selected week
Send_forcast_to_supplier	Time	Time dedicated to send forecasts to suppliers during the selected week
Send_schedule_to_supplier	Time	Time dedicated to send schedules to suppliers during the selected week
Manage_urgent_replenishment	Time	Time dedicated to manage urgent replenishment during the selected week

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

Manage_storage_area	Time		Time dedicated to manage storage areas during the selected week
Manage_material_handling	Time		Time dedicated to manage handlings during the selected week
Manage_return_material	Time		Time dedicated to manage return materials during the selected week
Loading_management	Time		Time dedicated to manage loadings during the selected week
Material	Varchar		Possible additional information
Id_tracking	Integer	Referred to *tracking_data.id* (many to one)	Identification number of Table *tracking_data*

2.4.2 Table Week

Attributes	Data Type	Primary Key	Foreign Key	Description
weekdaynum	Integer	x		Identification number
Day_name	Varchar			Name of the day
Id_week	Integer		Referred to *week.id* (many to one)	Identification number of the week

2.4.3 Table Weekday

Attributes	Data Type	Primary Key	Foreign Key	Description

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

Id	Integer	x	Identification number
Name	varchar		Name of the week

2.4.4 Table Construction_site

Table Construction_site includes the characteristics of the construction site.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Name	varchar			Name of the construction site

2.4.5 Table Tracking_data

In Table Tracking_data we keep track of the imported data.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
File_name	Varchar			Name of the imported file
Date_insert	date			Date of import

2.5 Several Handling Database

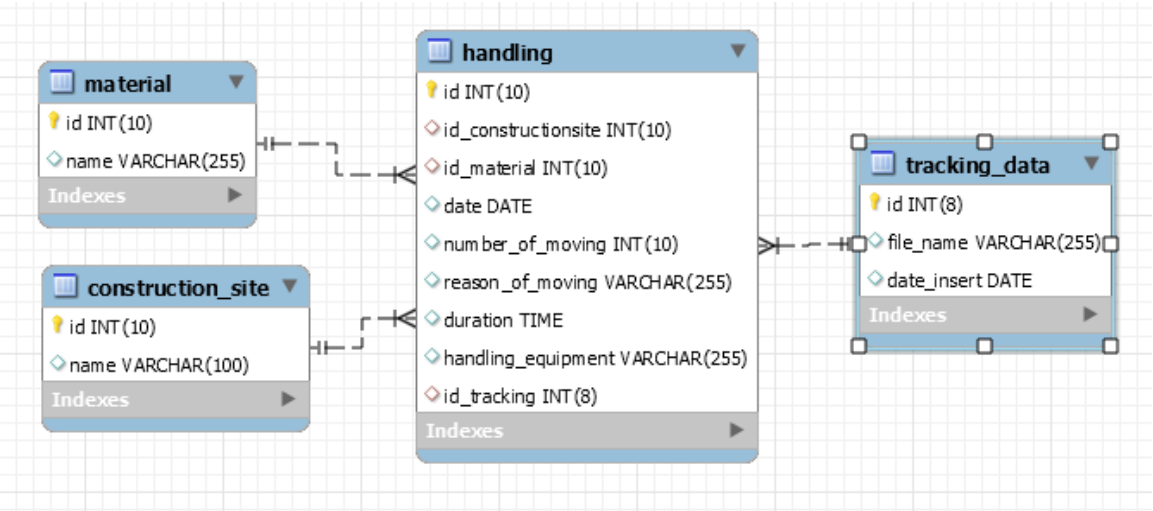

Figure 6 ER scheme of Several Handling DB

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

The Several Handling DB considers the occurrence of a number of moves and handlings of materials inside the construction site and the related characteristics. The DB is made of 4 tables, namely:

- Table Handling
- Table Material
- Table Construction_site
- Table Tracking_data

2.5.1 Table Handling

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Id_constructionsite	Integer		Referred to the *construction_site.id* (many to one)	Identification number of the construction site
Id_material	Integer		Referred to the *material.id* (many to one)	Identification number of the material handled
Date	Date			Date of occurrence
Number_of_moving	Integer			Number of moving
Reason_of_moving	Varchar			Reason of moving
Duration	Time			Duration of moving
Handling_equipment	Varchar			Equipment used for handling
Id_tracking	Integer		Referred to the *tracking_data.id* (many to one)	Identification number to track data

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

2.5.2 Table Material

This table reports the characteristics of the handled material.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Name	Varchar			Name of the material

2.5.3 Table Construction_site

Table Construction_site includes the characteristics of the construction site.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Name	varchar			Name of the construction site

2.5.4 Table Tracking_data

In Table Tracking_data we keep track of the imported data.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
File_name	Varchar			Name of the imported file
Date_insert	date			Date of import

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

2.6 Congestion Monitoring Database

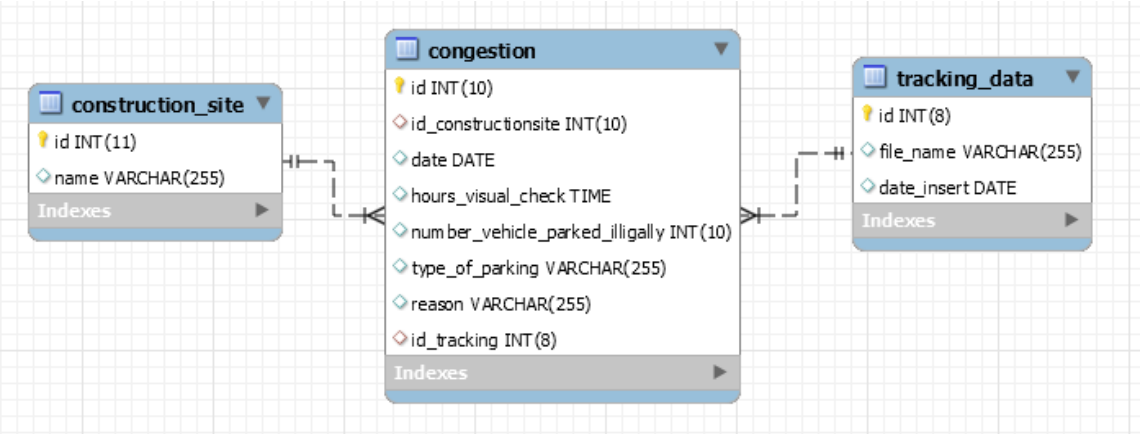

Figure 6 ER scheme of the Congestion Monitoring DB

The Congestion Monitoring DB is made of 3 tables, namely:

- Table Congestion
- Table Construction_site
- Table Tracking_data

2.6.1 Table Congestion

Table Congestion collects all the information on congestion in the construction site.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
ID_constructionsite	Integer		Referred to *construction_site.id* (many to one)	Identification number of the related construction site
Date	Date			Date of monitoring
Hours_visual_check	Time			Hours of visual check
Number_vehicles_parked_illigally	Integer			Number of vehicles parked illegally
Type_of_parking	Varchar			Description of parking

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

reason	Varchar		Reason of illegally parking
Id_tracking	Integer	Referred to *tracking_data.id* (many to one)	Identification number to track data

2.6.2 Table Construction_site

Table Construction_site includes the characteristics of the construction site.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Name	varchar			Name of the construction site

2.6.3 Table Tracking_data

In Table Tracking_data we keep track of the imported data.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
File_name	Varchar			Name of the imported file
Date_insert	date			Date of import

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

2.7 Accident Monitoring Database

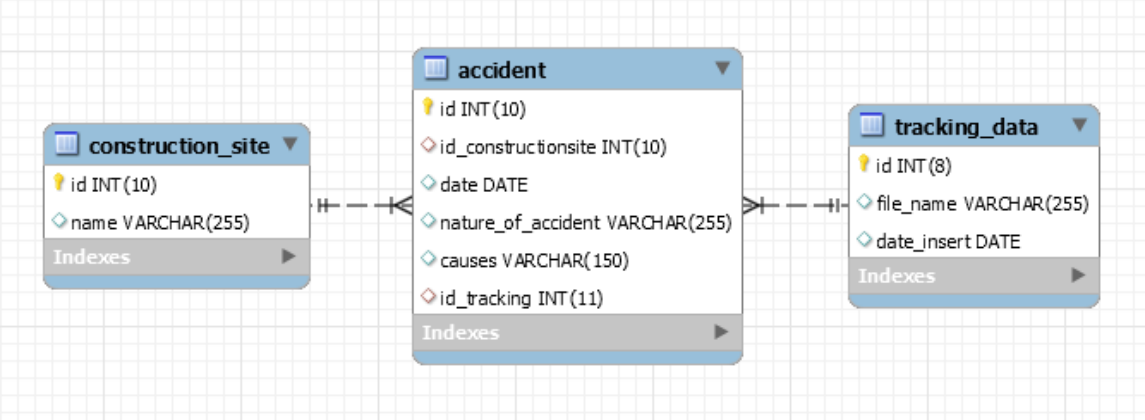

Figure 7 ER scheme Accident Monitoring DB

The Accident Monitoring DB is made of 3 tables, namely:

- Table Accident
- Table Construction_site
- Table Tracking_data

2.7.1 Table Accident

The Table Accident collects the information regarding the accident monitoring, the accident date of occurrence, their causes and nature.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Id_constructionsite	Integer		Referring to contruction_site.id (many to one)	Identification number of the related construction site
Date	Date			Data of the accident
Nature_of_accident	Varchar			Nature of the accident
Causes	Varchar			Cause of the accident
Id_tracking	Integer		Referring to tracking_data.id	Identification number of

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

	(many to one)	the tracking data

Table Accident is related to the Table Construction_site and to Table Tracking_data to track the inserted data.

2.7.2 Table Construction_site

Table Construction_site includes the characteristics of the construction site.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Name	varchar			Name of the construction site

2.7.3 Table Tracking_data

In Table Tracking_data we keep track of the imported data.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
File_name	Varchar			Name of the imported file
Date_insert	date			Date of import

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

2.8 Material Waste Database

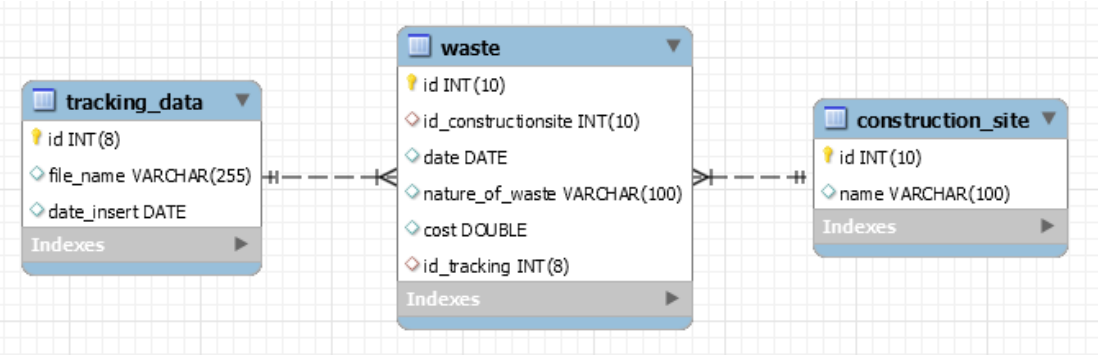

Figure 8 ER scheme of the Material Waste DB

The Material Waste DB is composed by 3 tables, namely:

- Table Waste
- Table Construction_site
- Table Tracking_data

2.8.1 Table Waste

This Table includes all the information regarding the wastes.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Id_constructionsite	Integer		Referred to *construction_site.id* (many to one)	Identification number of the Table *construction_site*
Date	Date			Date of occurrence
Nature_of_waste	Varchar			Information regarding the nature of the waste
Cost	Double			Cost of the waste
Id_tracking	Integer		Referred to *tracking_data.id* (many to one)	Identification number of the Table *tracking_data*

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

2.8.2 Table Construction_site

Table Construction_site includes the characteristics of the construction site.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Name	varchar			Name of the construction site

2.8.3 Table Tracking_data

In Table Tracking_data we keep track of the imported data.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
File_name	Varchar			Name of the imported file
Date_insert	date			Date of import

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

2.9 Cost of Unsorted bins Database

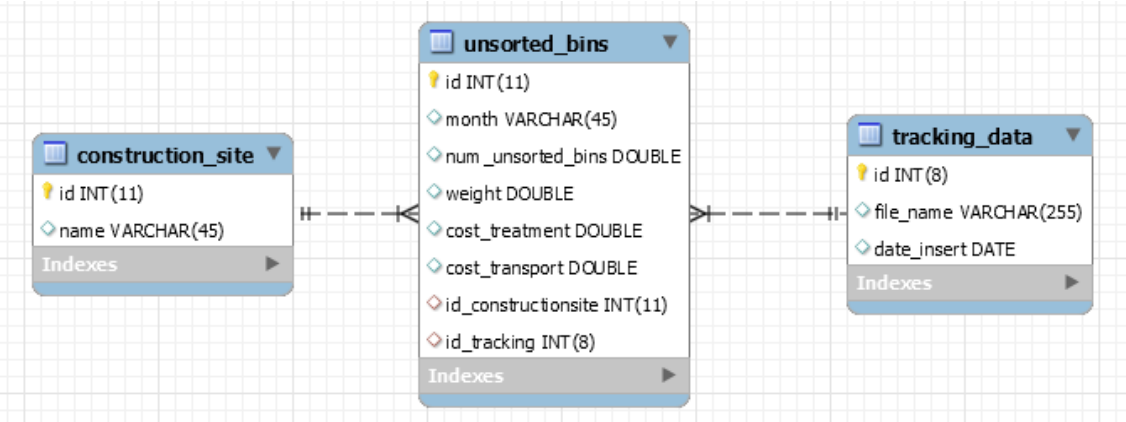

Figure 9 ER scheme of the Cost of unsorted bins DB

The Cost of Unsorted bins DB is structured into 3 tables:

- Table Unsorted_bins
- Table Construction_site
- Table Tracking_data

2.9.1 Table Unsorted_bins

Table Unsorted_bins reports the information regarding the unsorted bins.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Month	Varchar			Month of occurrence
Num_unsorted_bins	Double			Number of unsorted bins for the selected month
Weight	Double			Weight of the unsorted bins
Cost_treatment	Double			Cost of treatment of the unsorted bins
Cost_transport	Double			Cost of transport of the unsorted bins

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

Id_constructionsite	Integer	Referred to *construction_site.id* (many to one)	Identification number of the related *construction_site*
Id_tracking	Integer	Referred to *tracking_data.id* (many to one)	Identification number to track the data

2.9.2 Table Construction_site

Table Construction_site includes the characteristics of the construction site.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Name	varchar			Name of the construction site

2.9.3 Table Tracking_data

In Table Tracking_data we keep track of the imported data.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
File_name	Varchar			Name of the imported file
Date_insert	date			Date of import

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

2.10 Haulier Route Database

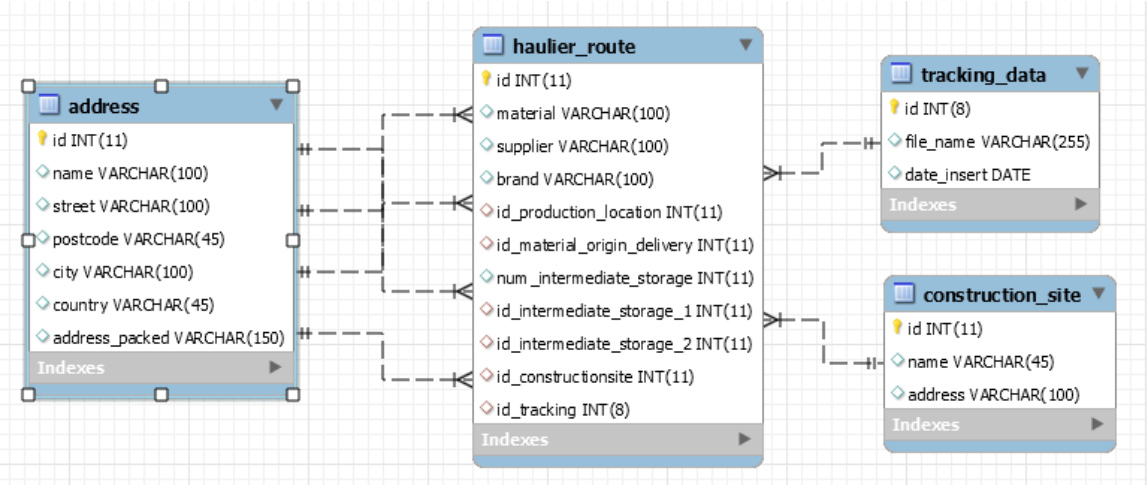

Figure 10 ER scheme of Haulier route DB

The Haulier Route DB shows the route of the hauliers and the related characteristics. The DB is made of 4 tables, namely:

- Table Haulier_route
- Table Address
- Table Construction_site
- Table Tracking_data

2.10.1 Table Haulier_route

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Material	Varchar			Material considered
Supplier	Varchar			Supplier of the material
Brand	Varchar			Brand
Id_production_location	Integer		Referred to *address.id*	Identification number of the production location address
Id_material_origin_delivery	Integer		Referred to *address.id*	Identification number of the material origin

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

			delivery
Num_intermediate_storage	Integer		Number of intermediate storage
Id_intermediate_storage_1	Integer	Referred to *address.id*	Identification number of the first intermediate storage (if existing)
Id_intermediate_storage_2	Integer	Referred to *address.id*	Identification number of the second intermediate storage (if existing)
Id_constructionsite	Integer	Referred to *construction_site.id*	Identification number of the related construction site
Id_tracking	Integer	Referred to *tracking_data.id*	Identification number for the tracking data to track data insertion

2.10.2 Table Address

This table includes the information on the addresses.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Name	Varchar			Name
Street	Varchar			Street
Postcode	Varchar			Postcode
City	Varchar			City
Country	Varchar			Country

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

Address_packed	Varchar	Entire addressed packed in one line

2.10.3 Table Construction_site

Table Construction_site includes the characteristics of the construction site.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
Name	varchar			Name of the construction site

2.10.4 Table Tracking_data

In Table Tracking_data we keep track of the imported data.

Attributes	Data Type	Primary Key	Foreign Key	Description
Id	Integer	x		Identification number
File_name	Varchar			Name of the imported file
Date_insert	date			Date of import

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

3 Controlling data quality: the quality database

Deliveries and pickups are the largest datasets collected by the project and the dataset collected over the largest period (up to 10 months for some pilots). Therefore the project management board took the decision to automate quality controls on these data.

The data quality controls are performed on a replica of the delivery and pickup database. The replica database operates on a PostgreSQL 9.1 server internal to LIST. This replica is composed of the same tables than described in section 2.1 plus a set of additional tables described hereafter and managed in a separate database schema.

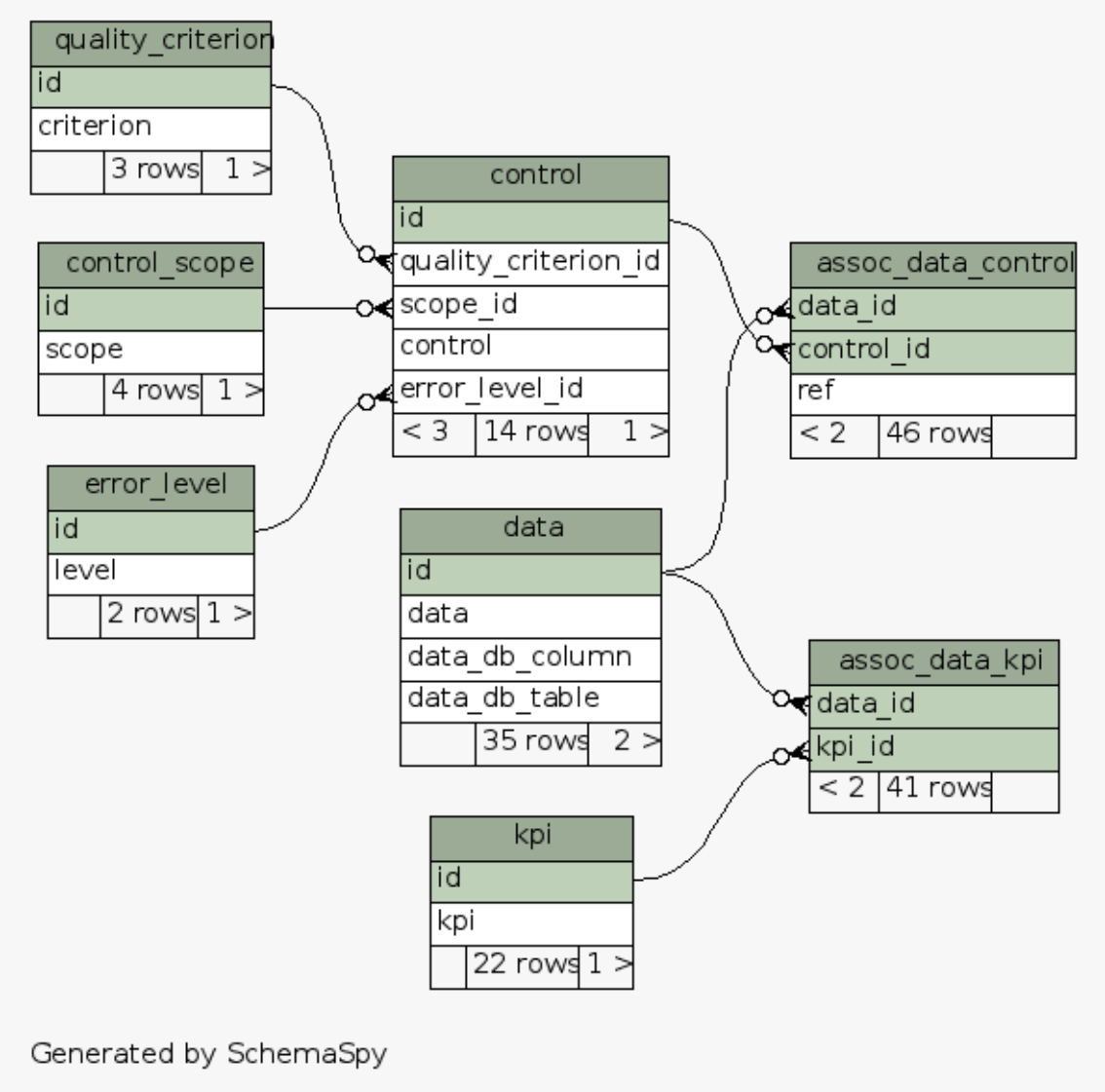

Figure 11 Overview of the quality database E-R model

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

A specific view, implemented in the quality replica of the deliveries and pickup database, makes use of these quality tables to apply the controls to the data collected.

This view is used to generate automated quality reports (see deliverable D2.5 *Open Data Accessibility and treatment* for a detailed description of the data quality principles, data quality processes and controls).

3.1 Table data

The *data* table is used to reference the deliveries and pickup data on which quality controls are operated.

Attributes	Type	Primary Key	Foreign key	Description
id	serial	X		Identification number
data	varchar			Name of the data column in the data collection Excel file
data_db_column	varchar			Name of the DB column in the DB
data_db_table	varchar			Name of the DB table

3.2 Table kpi

The *kpi* table is used to reference the KPI computed with the deliveries and pickups data collected.

Attributes	Type	Primary Key	Foreign key	Description
id	serial	X		Identification number
kpi	varchar			Name of the KPI

3.3 Table assoc_data_kpi

The *assoc_data_kpi* table is an associative table to make the relation between data and KPI

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

Attributes	Type	Primary Key	Foreign key	Description
data_id	int4	X	X	References data table
kpi_id	int4	X	X	References kpi table

3.4 Table control

The '*control*' table is used to reference the quality controls operated on the data.

A control operated on a data corresponds to a scope, answers to a quality criterion and raises an error level if failed.

Column	Type	Primary Key	Foreign key	Description
id	serial	X		Identification number
quality_criterion_id	int4		X	References quality criterion table
scope_id	int4		X	References scope table
control	varchar			Description of the control
error_level_id	int4		X	References error_level table

3.5 Table quality_criterion

The *quality_criterion* table is used to reference the quality criterion of the quality controls (see deliverable D2.5 for a detailed explanation of the quality criterion).

Attributes	Type	Primary Key	Foreign key	Description
id	serial	X		Identification number
criterion	varchar			Description of the criterion

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

3.6 Table control_scope

The *control_scope* table is used to reference the scope of the quality controls. So far only one scope was used: "data".

Attributes	Type	Primary Key	Foreign key	Description
id	serial	X		Identification number
scope	varchar			Description of the scope

3.7 Table error_level

The *error_level* table is used to reference the error level of the quality controls. The error level qualifies the criticality of a failed control. So far only two levels were used: "error" and "warning".

Column	Type	Primary Key	Foreign key	Description
id	serial	X		identification number
level	varchar			Description of the error level

3.8 Table assoc_data_control

The *assoc_data_control* an associative table to makes the relation between data and quality controls.

Attributes	Type	Primary Key	Foreign key	Description
data_id	int4	X	X	References the data table
control_id	int4	X	X	References the control table
ref	serial			identification number

4 Computing travel distances: the GIS database

The data collected by pilots contains, among others, addresses used by suppliers and hauliers before and after visiting the constructions sites for deliveries and pickups. The project management board took the decision to exploit these data to compute in a uniform way the distances travelled by suppliers and hauliers by making use of a Geographical Information System (GIS).

The GIS used for the project is the GIS extension of PostgreSQL (PostGIS) and the routing extension PGRouting. The GIS computations are performed on the replica of the delivery and pickup database. This replica is composed of the

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

same tables than described in section 2.1 plus a set of additional tables described hereafter and managed in a separate database schema.

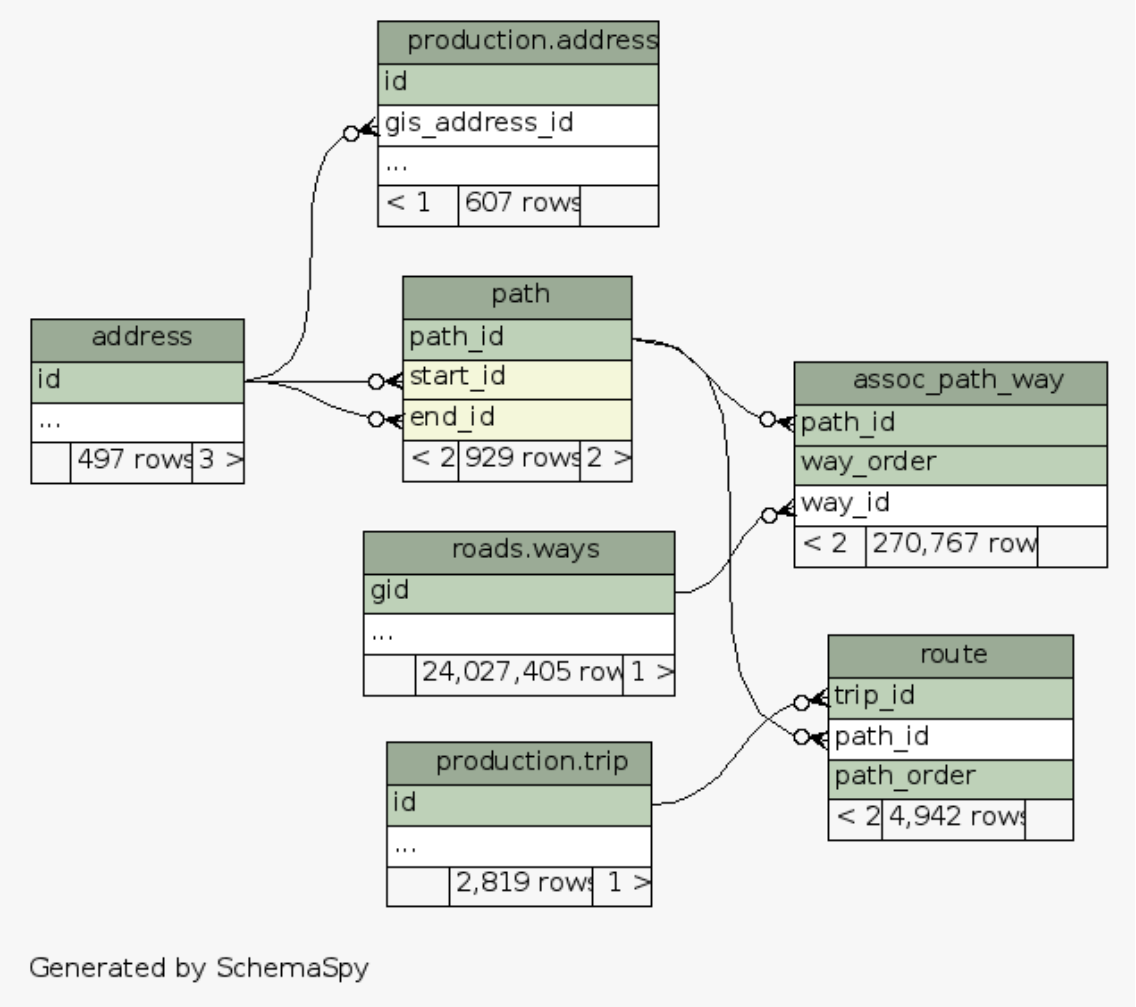

Figure 12: Overview of the E-R model for the GIS database

4.1 Table address

The *address* table is used is this schema to store unique addresses for geocoding. Data from the address table in the delivery and pickup replica database are filtered to ensure their unicity and then loaded in this table. The relationship between the original address and the filtered address is kept with a foreign key in the original table.

The address components (addr_street, addr_postcode, addr_city, addr_country) are duplicated instead of being referenced to in order to allow small corrections on the components (for better geocoding results) without altering the original inputs from pilots.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

This filtered table is then sent to Open Street Map geocoder nominatim[1] that throws geocoding results. When the geocoding succeeds, geographical information (latitude, longitude) are extracted from the geocoding outputs. These pieces of information are used to build a GIS geometry object representing the address.

Column	Type	Primary Key	Foreign Key	Description
id	serial	X		Unique identification number of the address
addr_street	varchar			Street and number
addr_postcode	varchar			Zip code
addr_city	varchar			City of the address
addr_country	varchar			Country of the address
geocoder_output	varchar			Outputs form the geocoding engine
lat	float8			Latitude extracted from the geocoding engine
lon	float8			Longitude extracted from the geocoding engine
the_geom	geometry			GIS object built on Latitude and Longitude

4.2 Table ways

The '*ways*' table contains the complete road network for France, Belgium, Netherlands, Luxembourg, Germany, Italy and Spain extracted from Open Street Map as of October 2015. This table is the biggest table of the database in number of records as it counts more than 24 million records.

This table has been generated by the OSM2PgRouting tool. Each row represents a road segment. For detailed description, refer to PGrouting documentation. References can be found on Chapter 5.

Column	Type	Primary Key	Foreign Key	Description
gid	int4	X		Unique identification number

[1] http://nominatim.openstreetmap.org/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

class_id	int4	X	References the road category
length	float8		Length of the road segment
name	text		The name of the road
x1	float8		Segment start longitude
y1	float8		Segment start latitude
x2	float8		Segment end longitude
y2	float8		Segment end latitude
reverse_cost	float8		The cost for the reverse traversal of the road segment
rule	text		A string with a comma separated list of ids which describes a rule for turning restriction
to_cost	float8		A cost of restricted passage (can be very high in a case of turn restriction or comparable with an edge cost in a case of traffic light)
maxspeed_forward	int4		Maximum speed allowed on the road segment in the forward way
maxspeed_backward	int4		Maximum speed allowed on the road segment in the backward way
osm_id	int8		Identifier of the original Open Street Map object
priority	float8		Number allowing to rank roads when selecting the optimal

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

			path
the_geom	geometry		GIS object
source	int4	X	Source identifier for the routing network
target	int4	X	Target identifier for the routing network

4.3 Table path

The *path* table is an associative table that relates an origin address to a destination address.

Column	Type	Primary Key	Foreign Key	Description
path_id	serial	X		Unique identification number
start_id	int4		X	Reference to origin address
end_id	int4		X	Reference to the destination address

Note that a unicity constraint is set on the tuple start_id, end_id.

4.4 Table assoc_path_way

The *assoc_path_way* table is an associative table that relates the ways that compose a path in an ordered way.

This table is filled with the results of the routing algorithm.

Column	Type	Primary Key	Foreign Key	Description
path_id	int4	X	X	Reference to a path
way_order	int4	X		Order of the way in the path [2]
way_id	int4		X	Reference to a way

4.5 Table route

The route table is an associative table that relates the paths of a trip in an ordered way. This table is filled by retrieving all addresses of a trip, ordering these addresses according to the trip characteristics (delivery vs. pickup, direct

[2] from 1 to n in a path composed of n ways

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

vs. roundtrip) and building the couples of origin/destination addresses based on this order.

Column	Type	Primary Key	Foreign Key	Description
trip_id	int4	X	X	Reference to a trip
path_id	int4		X	Reference to a path
path_order	int4	X		Order of the path in the trip

5 The Software and the Servers

5.1 MySQL

The main databases to store data collected have been developed on the free and open source relational database management system MySQL, in particular by using MySQL Workbench 6.3. MySQL Workbench is a visual database design tool that integrates SQL development, administration, database design, creation and maintenance into a single integrated development environment for the MySQL database system.

Figure 13 The MySQL Workbench icon.

All the Project Partners need to have at least read access to the DB, and thus we set our DB on a server. We thus decided to use the Amazon RDS for MySQL that provides 20 GB of free storage.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

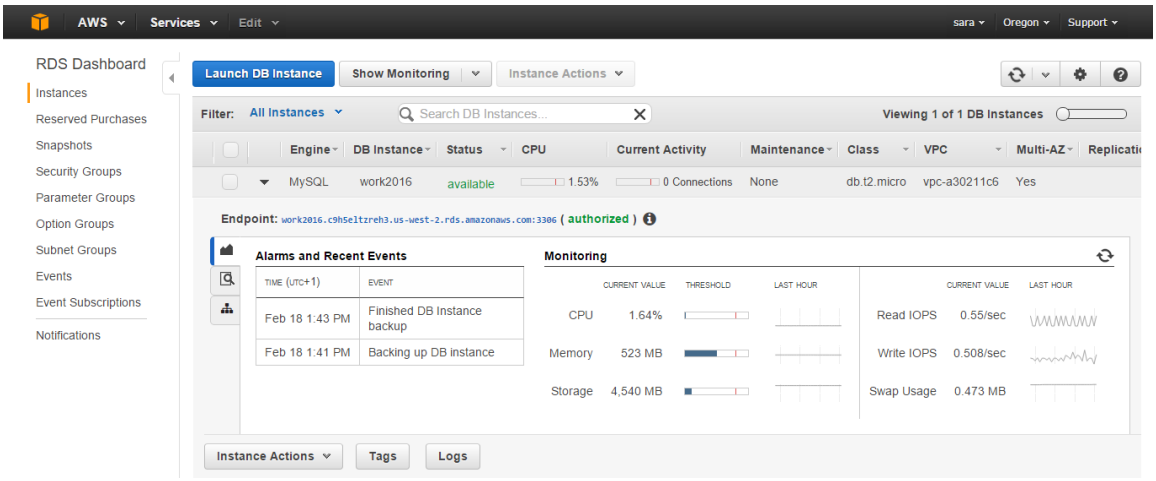

Figure 14 The Amazon RDS Dashboard

Connecting Workbench with Amazon RDS for MySQL is straightforward by inserting the right Hostname, Username, and Password as one can see in the Figure.

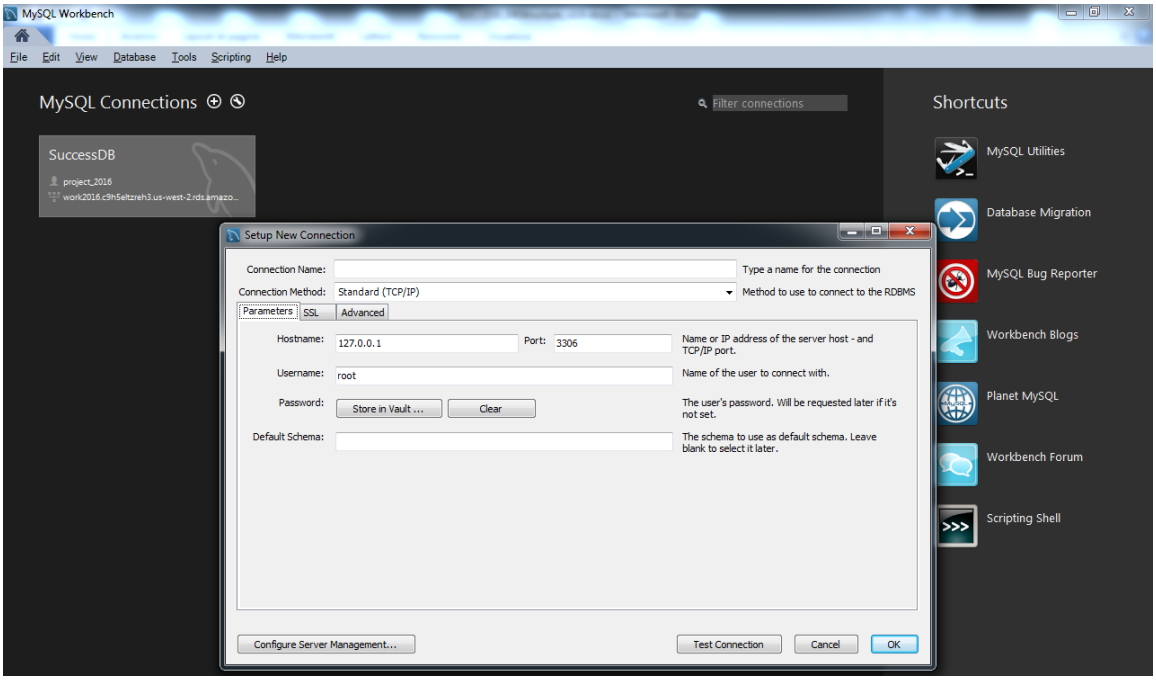

Figure 15 Connection between Workbench to Amazon RDS for MySQL.

Some of the partners will need the full access, while some others will need the reading access. To guarantee the different type of access to Partners we provided to each access type a User Account with different levels of privileges.

5.1.1 ReadingPartners Access Type

In the following we report the *ReadingPartners* type of access. To access, Partners should start a new connection on Workbench and then insert:

- Hostname: *ask to task leader*

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

- Port: 3306
- Username: ReadingPartners
- Password: *ask to task leader*

Then click on *Test Connection* and thus *OK*. After that action, Partners can click on the newly created connection to the DB. The ReadingPartners type of access is limited to SELECT and SHOW VIEW actions. We can provide an increased set of access privileges if partners will require it, in the future.

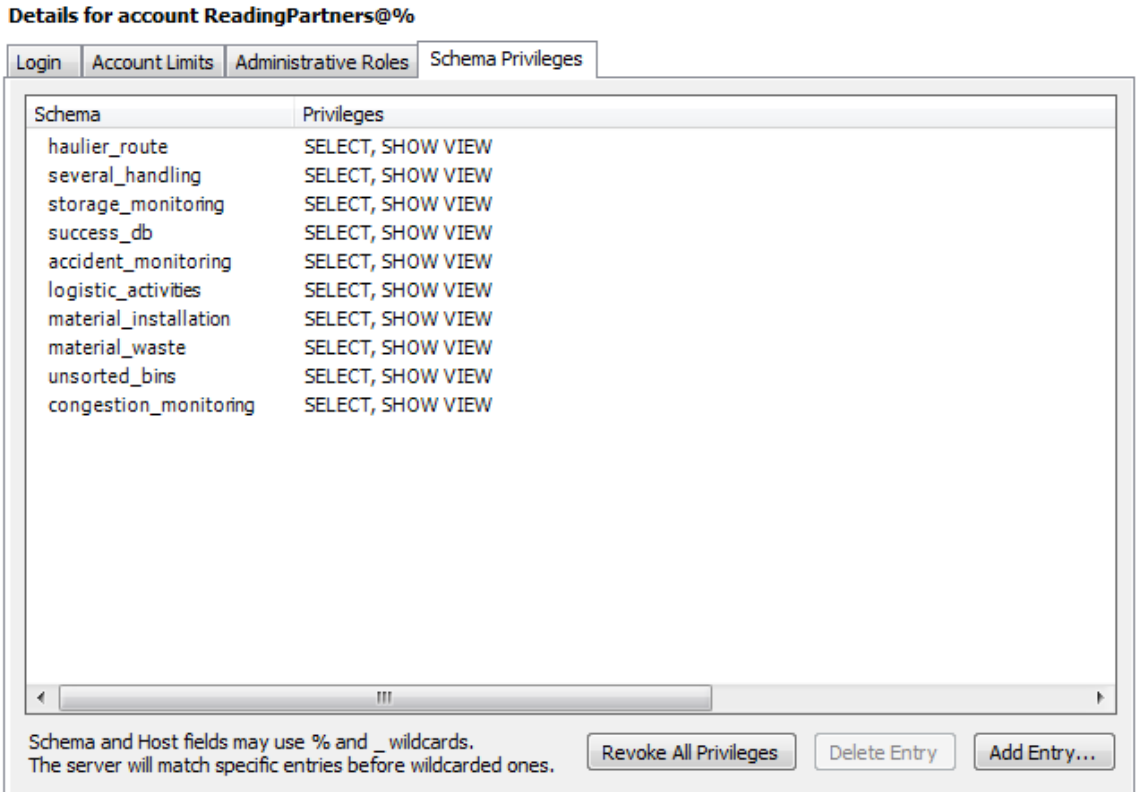

Figure 16 Privileges of the ReadingPartners type of access.

5.1.2 DataQuality Access Type

The partner in charge of data quality (Task 2.5) need a special access type that allows them the select access to all tables and views and the possibility of creating views. We refer to this type of access as to *DataQuality* access type.

To access, the partner in charge of data quality should start a new connection on Workbench and then insert:

- Hostname: *ask to task leader*
- Port: 3306
- Username: DataQuality
- Password: *ask to task leader*

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

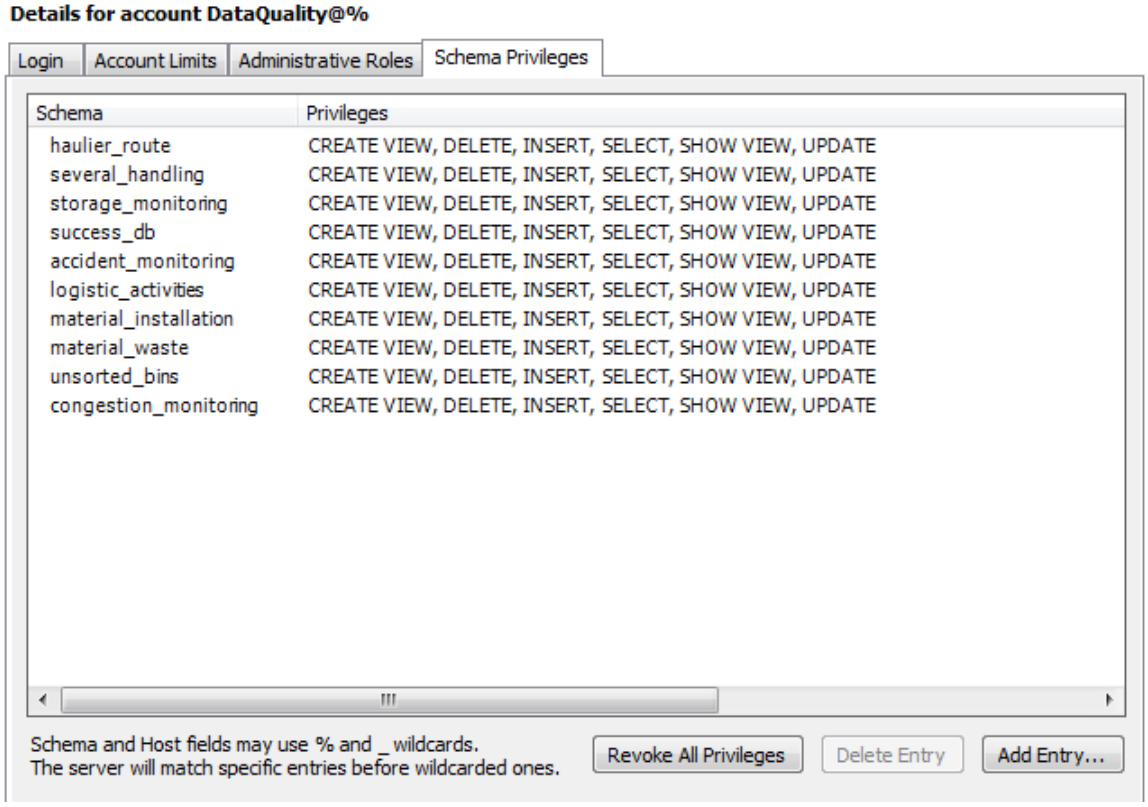

Figure 17 Privileges of the DataQuality access type

5.1.3 GIS Access Type

The *GIS* access type is needed because some additional information, such as distances, must be included in the DB. These data will be derived from the data imported in the DB and computed by making use of a GIS. The partner in charge of computing data from a GIS will need the possibility of inserting / updating / deleting the travel_distance_urban, travel_distance_road, and travel_distance_motorway columns in the Trip Table. At the moment we left them to access to all the attributes of the Tables, as you can see in the following Figure.

To access, the partner in charge of GIS use and computation should start a new connection on Workbench and then insert:

- Hostname: *ask to task leader*
- Port: 3306
- Username: GIS
- Password: *ask to task leader*,

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

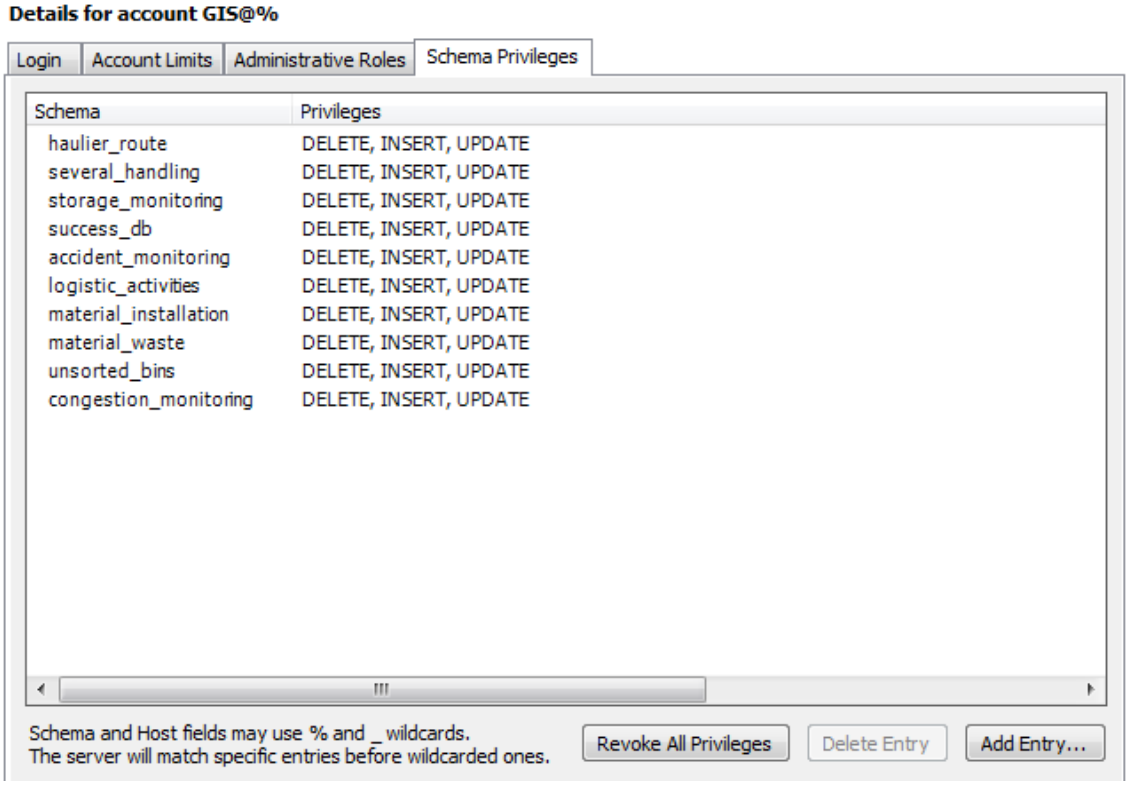

Figure 18 Privileges of the GIS access type

5.1.4 Administration Access

The *Administration* access is the access that allows all the actions and it is reserved only for the Task 2.3 (Database structure) Leaders.

5.2 PostgreSQL

Specific databases used for controlling data quality and computing geographical information have been developed using the free and open-source relational database management system PostgreSQL 9.1[3].

The geographical information system has been developed with the free and open-source GIS extensions of PostgreSQL: PostGIS[4] (for general GIS purpose) and PgRouting[5] (for routing purpose).

5.2.1 OSMOSIS and OSM2PgRouting

The specific free and open-source tools and scripts OSMOSIS[6] and OSM2PgRouting[7] have been used to extract road data from the community web site OpenStreetMap.

[3] https://www.postgresql.org/
[4] http://postgis.net/
[5] http://pgrouting.org/
[6] http://wiki.openstreetmap.org/wiki/Osmosis

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

More precisely, OSMOSIS has been used to extract SUCCESS-related countries' data from OpenStreetMap, to filter them so as to keep only roads data, and to merge country-specific files into a single one.

Then OSM2PgRouting has been used to load the single data file into the GIS database and to build a routable network for PgRouting operations.

5.3 Penthao Data Integration (ETL)

An Extract Transform and Load (ETL) software has been used to perform data extraction, data transformation, and data loading in between the various databases and systems. The free and open source ETL software Penthao Data Integration (Community edition version 6.1[8]) has been used in particular for:

- Managing the replication between MySQL databases and PostgreSQL databases.
- Chaining the SQL queries required to feed the GIS.
- Dynamically querying the OpenStreetMap Nominatim geocoding service and processing its results.
- Exporting KPIs computation views in Excel files for further detailed analysis by project partners.

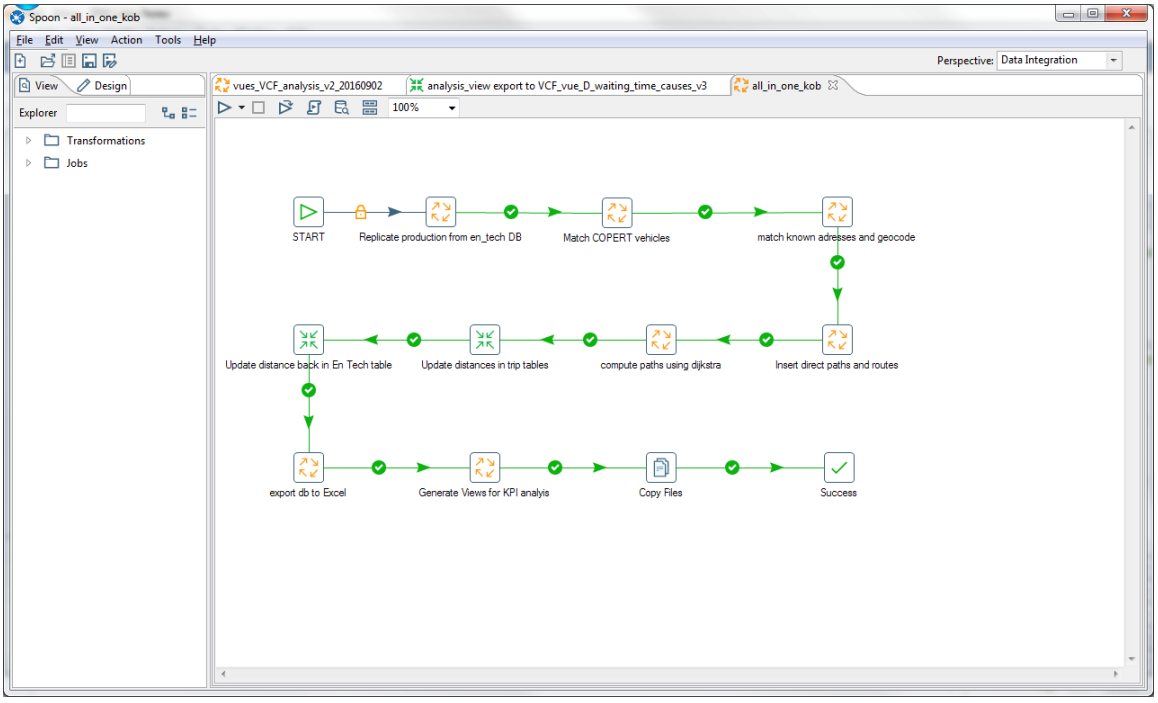

Figure 19: An overview of the global ETL script used to replicate the data collection databases, perform GIS computations and export KPI view to Excel files

[7] http://pgrouting.org/docs/tools/osm2pgrouting.html
[8] http://community.pentaho.com/projects/data-integration/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

6 Filling the DBs: the Import – Export Interface

The majority of the tables of the DBs will be filled, day by day, with the data included into *SUCCESS_Delivery_Pick up tracking board template.xlsx, SUCCESS_Activity monitoring template.xlsx, and SUCCESS_Posteriori analysis.xlsx.*

To ease the uploading process we provide an import interface for uploading data files, coded in php.

In this interface it is possible to import a *SUCCESS_Delivery_Pick up tracking board template.xlsx,* while doing it we must separate the Excel files into the Delivery and Pickup sheets and translate them in csv files. In the same way we must separate the *SUCCESS_Activity monitoring template.xlsx* into csv files representing the following set of data: Material Installation, Storage Monitoring, Logistic Activity Monitoring, Several handling, Congestion Monitoring, Accident monitoring, and Material Waste. The *SUCCESS_Posteriori analysis.xlsx* file also needs to be separated into two csv files representing the Cost of unsorted bins and the Haulier route.

We also report from which pilot site we are referring to by selecting the name of the pilot site from the dedicated menu. A highlight of the import/export page is shown in the Figure 17.

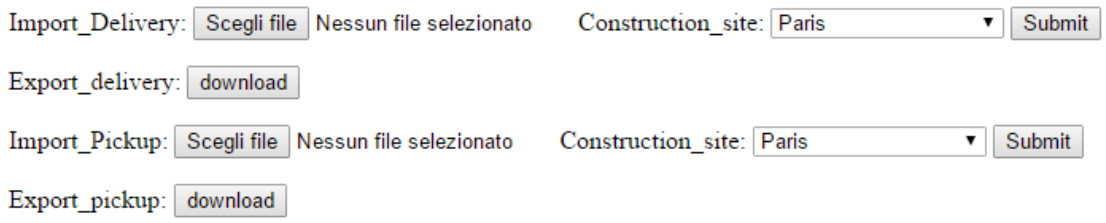

Figure 20 Interface for data import and export.

En&Tech is in charge of uploading the Excel/csv files to the DB.

By importing the files we can populate the DB. An example of the populated Table Material is given in Figure 18.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

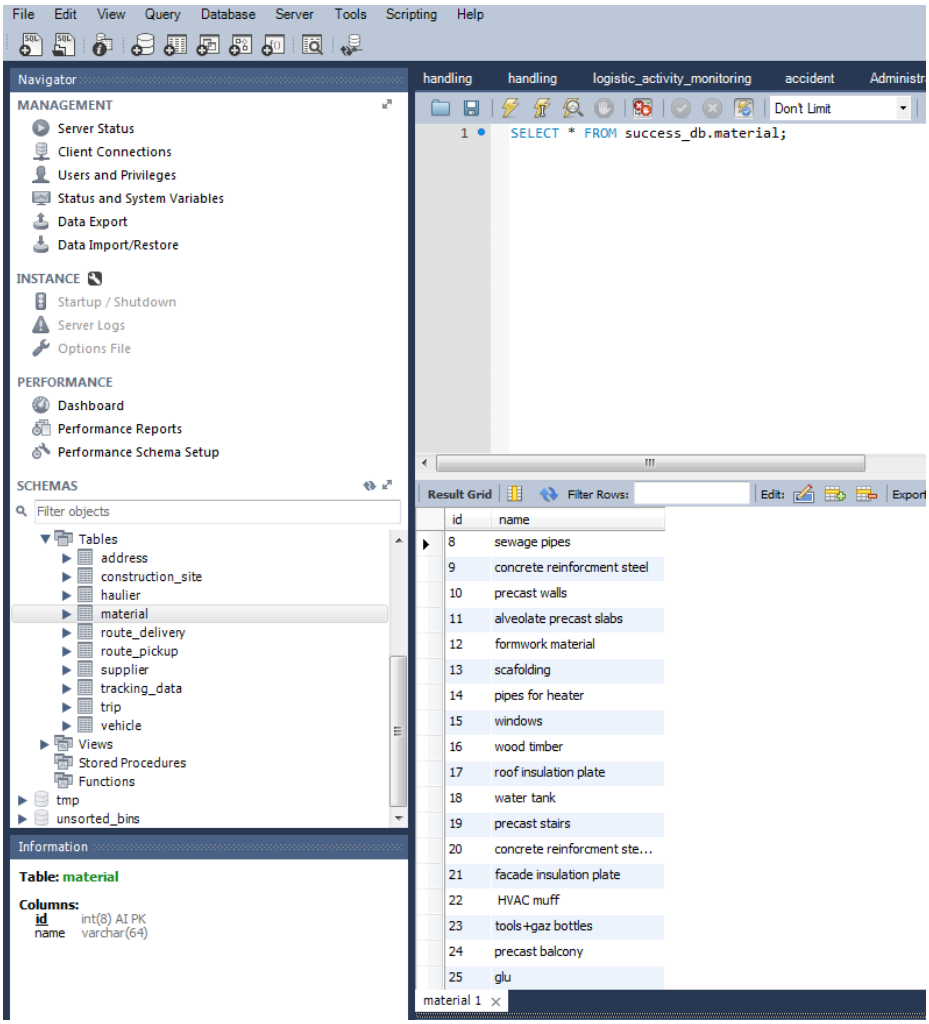

Figure 21 A view of the Table Materials.

In addition, we provide an export interface that is also written in php code. The export interface can export the entire files as they were imported; this has been implemented to double check the import phase.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

7 KPIs computation and Views

In this section we report the KPIs detected in *Task 2.2* (Identification of KPIs and methodologies to elaborate data) and we describe how to compute them by making use of the data collected and stored in the DB. Most of the KPIs can be computed thanks to the data stored in the DB, but for some of them additional information or data could be required.

7.1 Trips travel distances computation

Trips travel distance were not collected from the drivers (for the ease of data collection) but were computed based on trips addresses and characteristics.

7.1.1 Data used for trips' travel distance computation

Characteristic of the trip:

- trip.round_direct

All addresses related to a trip in their logical order:

1. id_address_loading/unloading_location
2. id_address_previous_delivery/pickup_2 (in case of a round trip)
3. id_address_previous_delivery/pickup_1 (in case of a round trip)
4. construction_site.id_address (to represent in a quick way what more precisely would be: the address expressed in Table Address which has the same id as the one in construction site, where the construction site is the one defined by the id_contruction_site in Table Trip).
5. id_address_next_delivery_1 (in case of a round trip)
6. id_address_next_delivery_2 (in case of a round trip)
7. trip.id_final_come_back

7.1.2 Method used to compute trips' travel distance

1. Address of a trip are ordered in their logical order (see previous section)
2. Pairs of origin and destination addresses are built based on this order
3. These pairs are inserted in the path table
4. The order of the pairs is inserted in the route table with a back reference to the trip table
5. The PgRouting extension is used to retrieve the ways of each path and their order, with the following parameters:
    - Algorithm: Dijkstra's shortest path on directed network
    - Cost function: (length / maxspeed_forward) [9] * priority[10]
6. The results of the previous step are inserted in the assoc_path_way table

[9] i.e. time to travel a road segment at the maximum allowed speed
[10] Based on the road priority as described in the 'ways' table

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

7. The total lengths of the ways of the paths of a trip are summarized following the breakdown:
    - Ways where the maximum speed is < 50 km/h are considered as urban roads
    - Ways where the maximum speed is between 50 and 90 km/h are considered as standard roads
    - Ways where the maximum speed is over 90 km/h are considered as motorways
8. The total distance travelled on each road category is updated at each trip record (in the columns travel_distance_urban, travel_distance_road, travel_distance_motorway) and replicated back in the datacollection database.

7.2 KPIs computation

Note: the KPI computation data and methods are described for deliveries. Unless otherwise mentioned, the same methods apply for pickups but with pickup- related data (e.g. unloading location instead of loading location, etc.).

7.2.1 CO2 equivalent

7.2.1.1 Data used to compute CO2 equivalent

- route_delivery.back_empty_or_loaded/route_pickup.arrival_empty_or_loaded
- vehicle.Euroclass
- vehicle.type
- route_delivery.material_weight_kg / route_pickup.material_weight_kg
- vehicle.gross_vehicle
- vehicle.netweight
- vehicle.fuel_type
- address.country
- trip.travel_distance_urban
- trip.travel_distance_road
- trip.travel_distance_motorway

7.2.1.2 Method to compute CO2 equivalent

The data are aggregated by country, type of vehicle, type of fuel, category of gross weight and euroclass. The aggregated data are input in COPERT software that returns back the CO2 equivalent emissions.

Due to the fact that COPERT feeding and computation is not automatable, it was not possible to compute the KPI per trip, but only aggregated as described before.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

7.2.2 Congestion on construction site

7.2.2.1 Data used to compute Congestion on construction site

- vehicle.surface_vehicle
- route_delivery.site_arrival_time
- route_delivery.end_time_unloading

7.2.2.2 Method to compute Congestion on construction site

Congestion on construction site =

(route_delivery.end_time_unloading - route_delivery.site_arrival_time) * vehicle.surface_vehicle

7.2.3 Construction site punctuality

7.2.3.1 Data used to compute Construction site punctuality

- route_delivery.start_time_unloading
- trip.date (used in case the delivery date is different from the planned one)
- route_delivery.planned_delivery_date (used in case the delivery date is different from the planned one)
- route_delivery.planned_delivery_time1
- route_delivery.planned_delivery_time2

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

7.2.3.2 *Method used to compute Construction site punctuality*

When only planned_delivery time1 is input, it means that the schedule is **fixed** and the KPI is computed as

Construction site punctuality =
route_delivery.planned_delivery_time1 -
route_delivery.start_time_unloading

When both planned_delivery_time1 and planned_delivery_time2 are input, it means that the schedule is an interval and the KPI is computed as

When start_time_unloading < planned_delivery_time1

Construction site punctuality = route_delivery.planned_delivery_time1 - route_delivery.start_time_unloading

When planned_delivery_time1 ≤ start_time_unloading ≤ planned_delivery_time2

Construction site punctuality = '00:00'

When start_time_unloading > planned_delivery_time2

Construction site punctuality = route_delivery.planned_delivery_time2 - route_delivery.start_time_unloading

7.2.4 Distance from the production to the construction site

7.2.4.1 *Data used to compute Distance from the production to the construction site*

- haulier route table
- address table

7.2.4.2 *Method used to compute Distance from the production to the construction site*

The distance from the production to the construction site is computed following the same method as for the trips' travel distances (see 7.1.2):

1) For each record in the haulier route table, the addresses are ordered in their logical order:
   a) production location
   b) first intermediate storage (if existing)
   c) second intermediate storage (if existing)
   d) construction site
2) Pairs of origin and destination addresses are built based on this order
3) The PgRouting extension is used to retrieve the ways of pair and their order, with the following parameters:
   a. Algorithm: Dijkstra's shortest path on directed network

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

b. Cost function: (length / maxspeed_forward) [11] * priority[12]

4) The total lengths of the ways are summarized for each haulier route

7.2.5 Loading / unloading time

7.2.5.1 Data used to compute Loading / unloading time

- route_delivery.start_time_unloading
- route_delivery.end_time_unloading
- route_pickup.start_time_unloading
- route_pickup.end_time_unloading

7.2.5.2 Method used to compute Loading / unloading time

Unloading_time =

route_delivery.end_time_unloading - route_delivery.start_time_unloading

Loading_time =

route_pickup.end_time_unloading - route_pickup.start_time_unloading

7.2.6 Number of deliveries

7.2.6.1 Data used to compute Number of deliveries

- route_delivery.id

7.2.6.2 Method used to compute Number of deliveries

Count(route_delivery.id)

7.2.7 PM

7.2.7.1 Data used to compute PM

- vehicle.fuel_type
- route_delivery.back_empty_or_loaded/route_pickup.arrival_empty_or_loaded
- route_delivery.material_weight_kg / route_pickup.material_weight_kg
- vehicle.type
- vehicle.Euroclass

[11] i.e. time to travel a road segment at the maximum allowed speed

[12] Based on the road priority as described in the 'ways' table

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

- vehicle.netweight
- vehicle.gross_vehicle

7.2.7.2 Method used to compute PM

The data are aggregated by country, type of vehicle, type of fuel, category of gross weight and euroclass. The aggregated data are input in COPERT software that returns back the PM emissions.

Due to the fact that COPERT feeding and computation is not automatable, it was not possible to compute the KPI per trip, but only aggregated as described before.

7.2.8 Travel time inside the city centre (only deliveries)

Note that too few data was collected for pickups to be able to analyse consistently a travel time inside the city centre for pickups. Therefore only a global travel time was computed for pickups.

7.2.8.1 Data used to compute Travel time inside the city centre

- route_delivery.near_site_time
- route_delivery.urban_arrival_time

7.2.8.2 Method used to compute Travel time inside the city centre

Travel_time_inside_the_city_centre =

route_delivery.near_site_time - route_delivery.urban_arrival_time

7.2.9 Travel time outside the city centre (only deliveries)

Note that too few data was collected for pickups to be able to analyse consistently a travel time outside the city centre for pickups. Therefore only a global travel time was computed for pickups.

7.2.9.1 Data used to compute Travel time outside the city centre

- route_delivery.urban_arrival_time
- route_delivery.starting_time_loading_location

7.2.9.2 Method used to compute Travel time outside the city centre

Travel_time_outside_the_city =

route_delivery.urban_arrival_time - route_delivery.starting_time_loading_location

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

7.2.10 Travel time (only pickups)

7.2.10.1 Data used to compute Travel time

- route_pickup.arrival_time_unloading_location
- route_pickup.end_time_unloading

7.2.10.2 Method used to compute Travel time

Travel_time =

route_pickup.arrival_time_unloading_location - route_pickup.end_time_unloading

7.2.11 Truck punctuality

7.2.11.1 Data used to compute Truck punctuality

- route_delivery.near_site_time
- trip.date (used in case the delivery date is different from the planned one)
- route_delivery.planned_delivery_date (used in case the delivery date is different from the planned one)
- route_delivery.planned_delivery_time1
- route_delivery.planned_delivery_time2

7.2.11.2 Method used to compute Truck punctuality

When **only planned_delivery time1 is input,** it means that the schedule is **fixed** and the KPI is computed as

Truck_punctuality = route_delivery.near_site_time - trip.planned_delivery_time1

When **both planned_delivery_time1 and planned_delivery_time2 are input**, it means that the schedule is an **interval** and the KPI is computed as

When near_site_time < planned_delivery_time1

Truck_punctuality = route_delivery.near_site_time - trip.planned_delivery_time1

When planned_delivery_time2 ≤ near_site_time ≤ planned_delivery_time2

Truck_punctuality = '00:00'

When near_site_time > planned_delivery_time2

Truck_puntuality = route_delivery.near_site_time -

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

trip.planned_delivery_time2

7.2.12 Truck waiting time inside the site

7.2.12.1 Data used to compute Truck waiting time inside the site

- route_delivery.site_arrival_time
- route_delivery.start_time_unloading

7.2.12.2 Method used to compute Truck waiting time inside the site

Truck_waiting_time_inside_the_site =

route_delivery.start_time_unloading - route_delivery.site_arrival_time

7.2.13 Truck waiting time outside the site

7.2.13.1 Data used to compute Truck waiting time outside the site

- route_delivery.near_site_time
- route_delivery.site_arrival_time

7.2.13.2 Method used to compute Truck waiting time outside the site

Truck_waiting_time_outside_the_site =

route_delivery.site_arrival_time - route_delivery.near_site_time

7.3 Views on Delivery and Pickup DB

In order to compute KPI we developed the following views on the Delivery and Pickup DB. The name we gave to the views are self-explaining.

- Delivery_tracking_board_with_kpi
- Pickup_tracking_board_with_kpis
- Kpi_per_delivery_trip
- Kpi_per_pickup_trip

Then for the purpose of the KPI analysis in task T2.4, specific views based on these views were developed. In general in consisted in aggregating the KPI according the required analysis criteria[13]. The results were exported in Excel files for further processing and graph generation by the T2.4 team.

[13] For conciseness reasons the detail of each view is not detailed in this report, although the reader can have an overview of the views by reading the D2.4 deliverable

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

8 Graphical Output

In order to provide a better interpretation of the data some graphical output guaranteed by graphical queries was opportune. Thus we used the MOMIS system from DataRiver.

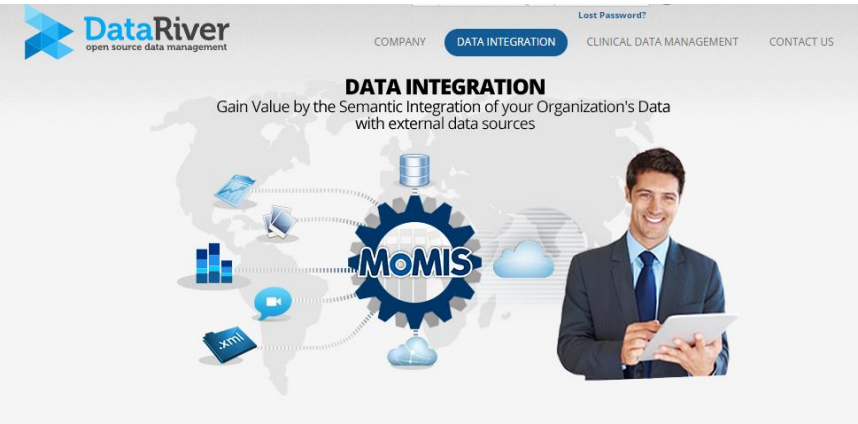

Figure 22 The MoMIS website

As from the Data River site:

> *"The Open Source MOMIS system (Mediator EnvirOnment for Multiple Information Sources) is a system able to aggregate data coming from heterogeneous data sources (structured and semi-structured) in a semi-automatic way, to bring out new information from apparently unrelated existing data. The discovery of relationships between the schemas of the information sources exploits the semantics of the data sources, clustering techniques and description logics inferences. The Open Source MOMIS system performs data integration following a virtual approach that preserves the autonomy and security of the original information sources."*

With a SUCCESS dedicated MOMIS dashboard we could provide different kind of graphical output and a use the dashboard in a flexible way to filter values, define exportable graphs, maps, and views.

In the following we report some examples of the obtained output.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

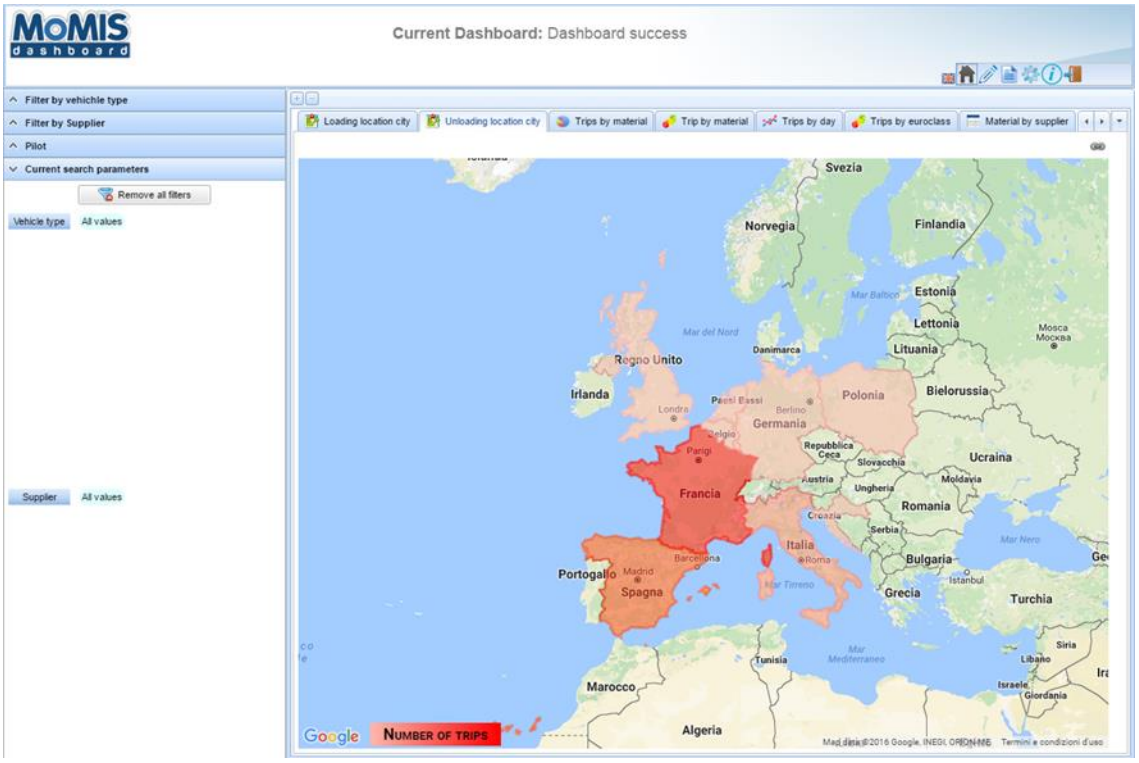

Figure 23 Map representing the number of Delivery routes per departure country

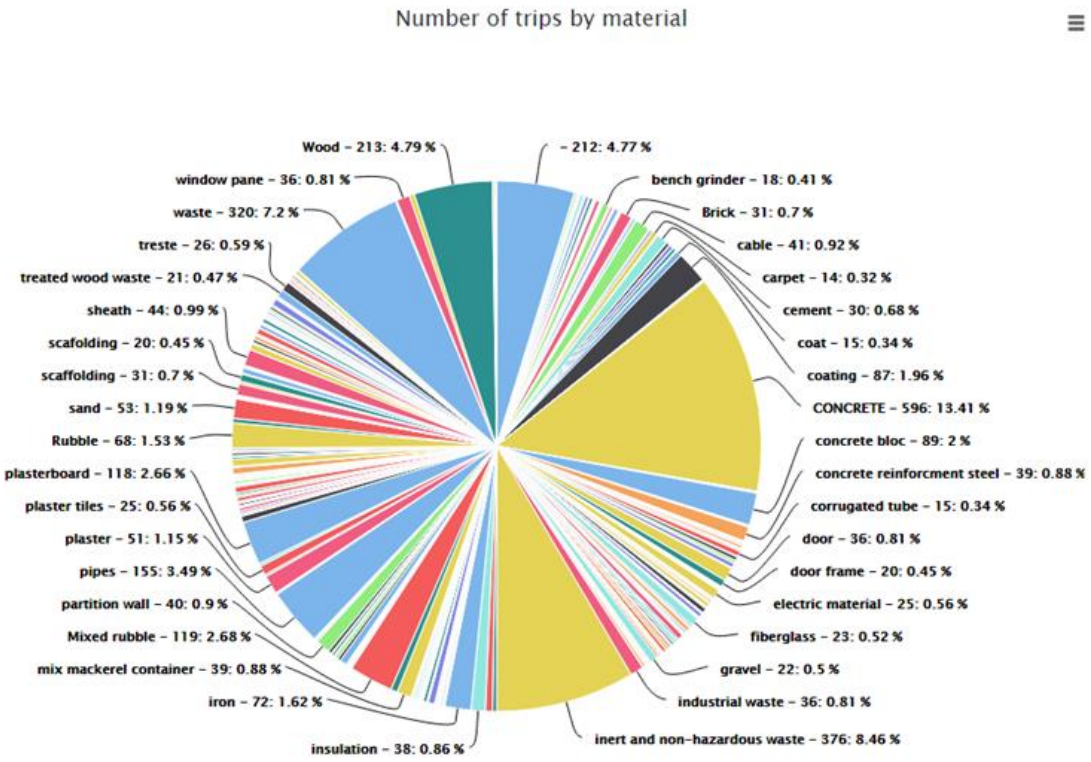

Figure 24Number of trips per material in the four pilot sites

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

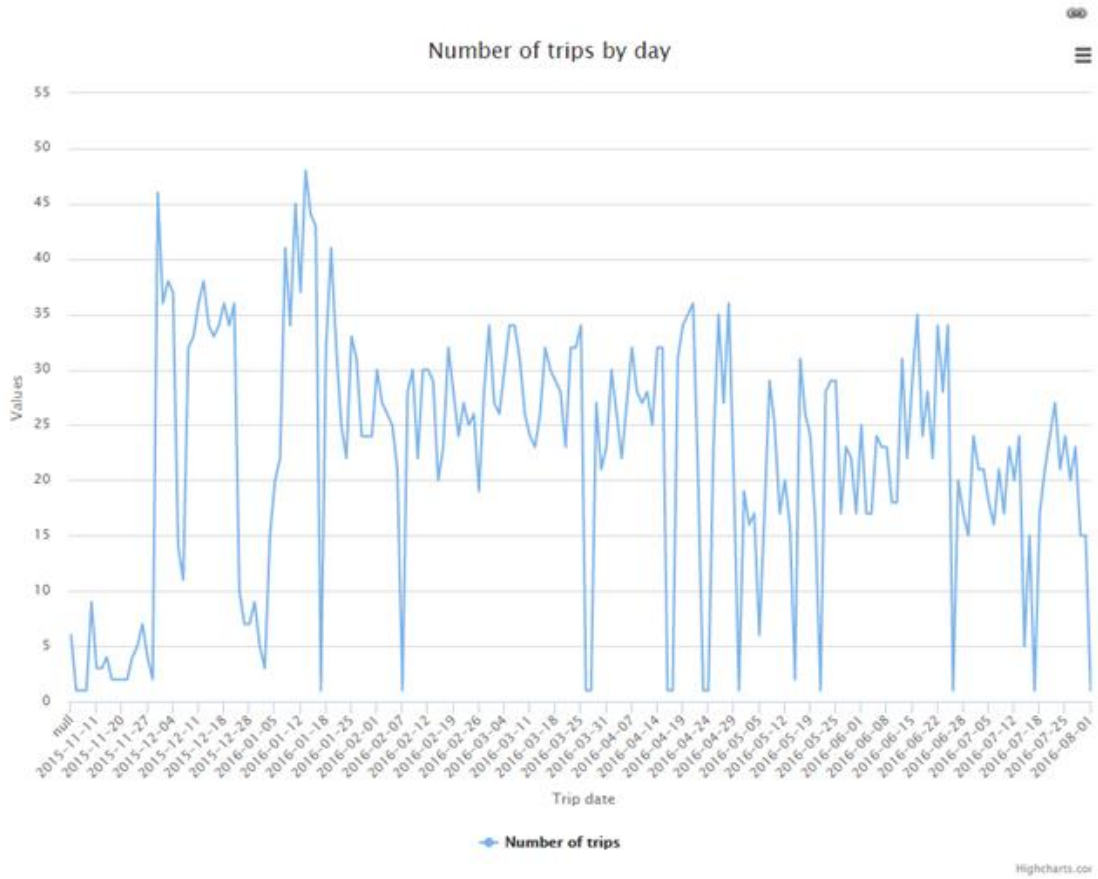

Figure 25Number of trips per day

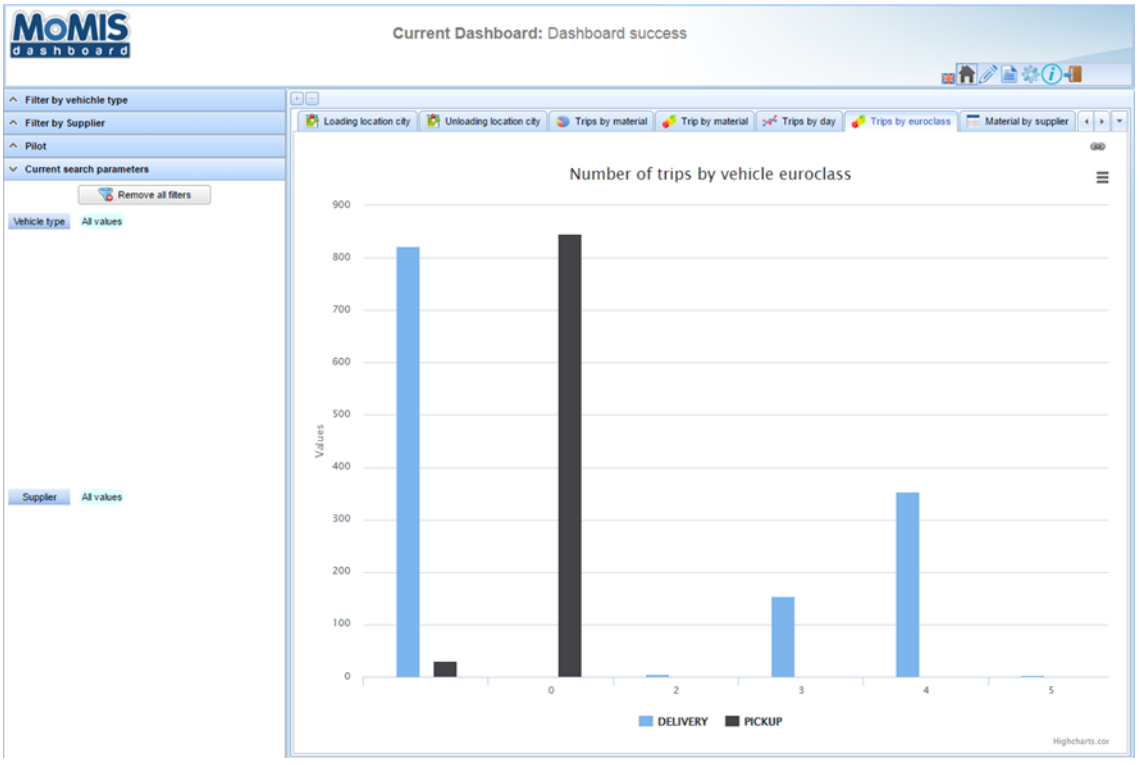

Figure 26Number of pickups and deliveries per euroclass

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

In the following graphs we selected one supplier to produce some output:

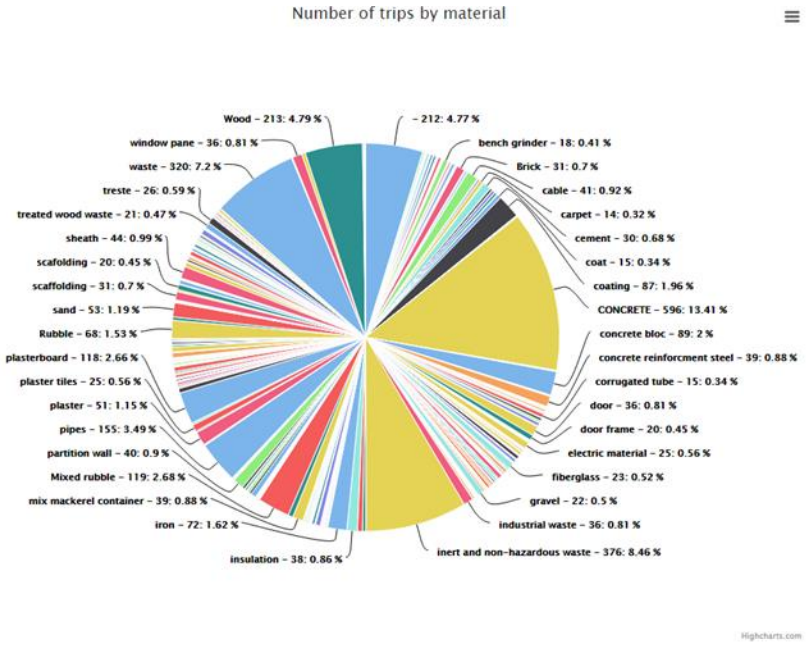

Figure 27 Trips by materials supplied by a supplier

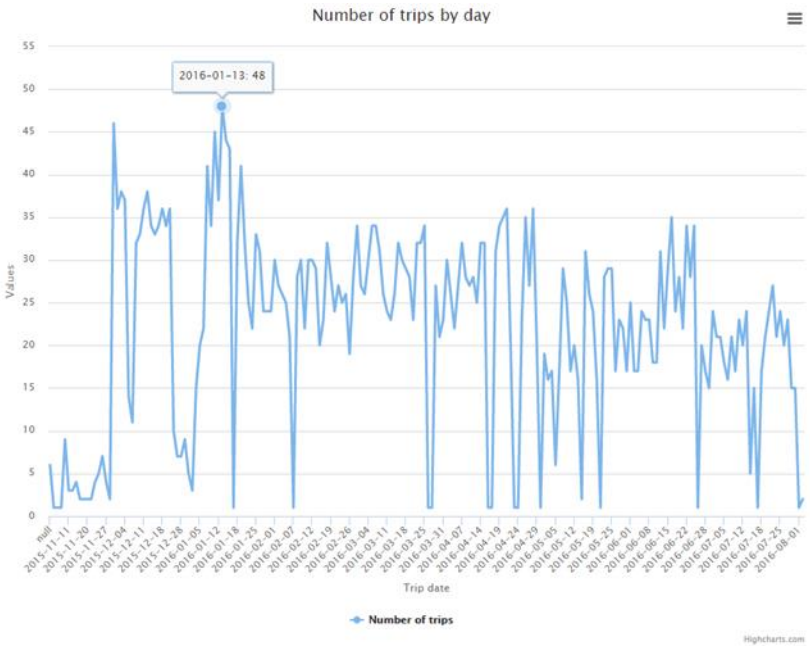

Figure 28Number of trips per day provided by a supplier

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.

9 Conclusions

Data are a core asset of the SUCCESS Project. Thus, databases where to store and analyse those data represent very important tools.

In this document we reported how we structured the databases with respect to the data collection files, showing the entity-relationship scheme, the tables, the included attributes and used datatypes, and the relationships among them. We also described the used software, the server, the different kind of access guaranteed to partners, and the functions created to perform data import and export.

As a single point of storage, the DB eases the data check as required in Task 2.5 (Quality of data and open data issues) and the common analysis of the data collected. Of course, the quality of collected data and storage are of primary importance. With this aim we developed a quality database on the most relevant data: the pickup and delivery ones. Thanks to this support database we could be able to provide checks and verifications on data.

A non-secondary output of the data collected is the computation of the KPI for the as-is analysis in Task 2.4, and in particular geographical information, in particular defining the paths between the pilot sites and the other stakeholders included into the data collected, and the related distances.

Those pieces of information will be needed as input for

- the computation of the KPIs detected during Task 2.2 (KPIs and methodologies for construction logistics) for the as-is analysis in Task 2.4;
- helping the target improvement setting (Task 4.1);
- the optimization tools that will be developed in Task 3.4 (Optimization Models, Algorithms and Tools);
- the definition of solutions in Task 4.3 (Solution Test and Simulation).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 633338.