



HARMONY

Holistic Approach for Providing Spatial & Transport Planning Tools and Evidence to Metropolitan and Regional Authorities to Lead a Sustainable Transition to a New Mobility Era

D3.5 Transport and Spatial Data Warehouse Final Version

Submission date: 13/03/2023



[@Harmony_H2020](https://twitter.com/Harmony_H2020)

[#harmony-h2020](https://twitter.com/Harmony_H2020)



<https://www.linkedin.com/company/harmony-h2020/>



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 815269



Part of:



THE CIVITAS INITIATIVE
IS CO-FINANCED BY THE
EUROPEAN UNION

SUMMARY SHEET

PROJECT

Project Acronym:	HARMONY
Project Full Title:	Holistic Approach for Providing Spatial & Transport Planning Tools and Evidence to Metropolitan and Regional Authorities to Lead a Sustainable Transition to a New Mobility Era
Grant Agreement No.	815269 (H2020 – LC-MG-1-2-2018)
Project Coordinator:	University College London (UCL)
Website	www.harmony-h2020.eu
Starting date	June 2019
Duration	45 months

DELIVERABLE

Deliverable No. - Title	D3.5 Transport and Spatial Data Warehouse Final Version
Dissemination level:	Public
Deliverable type:	Report
Work Package No. & Title:	WP3 - Data collection tools, data fusion and warehousing
Deliverable Leader:	ICCS
Responsible Author(s):	Babis Magoutas, Nikos Papageorgiyo, Grigoris Mentzas, Efthimios Bothos (ICCS)
Responsible Co-Author(s):	Panagiotis Georgakis (UoW), Ilias Gerostathopoulos, Shakur Al Islam, Athina Tsimpa (MOBY X SOFTWARE)
Peer Review:	Panagiotis Georgakis (UoW)
Quality Assurance Committee Review:	Maria Kamargianni, Emmanouil (Manos) Chaniotakis (UCL)
Submission Date	13/03/2023

DOCUMENT HISTORY

Version	Date	Released by	Nature of Change
0.1	12/01/2023	ICCS	First draft, ToC
0.9	04/03/2023	ICCS	Version ready for internal review
1.0	09/03/2023	ICCS	Final Version

TABLE OF CONTENTS

EXECUTIVE SUMMARY.....	3
1 Introduction.....	4
1.1 Project Summary.....	4
1.2 Deliverable Objectives	4
1.3 Deliverable Structure	5
2 Data Management Framework of the TSDW	5
3 User groups, access rights and roles	6
3.1 User management definition	7
4 Transport Simulators registry.....	9
4.1 Components' input data	11
4.2 Components' Output Data	13
5 Modelling Templates, Projects and Scenarios	14
6 Scenario Execution Logs	17
7 Analytics and Insights.....	17
8 Data Warehouse Implementation.....	18
9 Conclusions	18

LIST OF ABBREVIATIONS

Abbreviation	Explanation
KPI	Key Performance Indicator
MS	Model Suite
O/D	Origin / Destination
OCC	Oxfordshire County Council
OFS	Operational Freight Simulator
RBAC	Role-Based Access Control
TFS	Tactical Freight Simulator
TSDW	Transport and Spatial Data Warehouse

EXECUTIVE SUMMARY

This deliverable, the final of WP3, describes the final version of the HARMONY's Transport and Spatial Data Warehouse (TSDW) and is the outcome of the work performed in tasks T3.3 "Data management mechanisms", T3.4 "Data security & Privacy" and T3.5 "TSDW Integration, and population".

The work performed in these tasks has resulted to the specification and implementation of the HARMONY Data Management Framework. The need for such a framework stems from the fact in HARMONY, the integration of simulators is performed at different levels (operational, tactical and strategic), which needs to efficiently communicate, exchange information and data in an automated manner, while the HARMONY platform has to be able to store and manage the required input and output data. Moreover, in HARMONY different users and user groups can have access to the functionalities as well as the stored information. Users access rights are required in order to ensure that sensitive information can be viewed only by particular users (including transport modelers of a city or an area), while parts of the information can be accessible by a wider set of stakeholders. Additionally, part of the stored data refers to analytics and insights that the platform generates based on the output of the simulations. Such insights support decisions of transport stakeholders with respect to the integration of new mobility services.

The HARMONY data management framework allows managing the end to end lifecycle of transport simulation scenarios, including their definition, configuration, execution, tracking, comparison etc. Transport modelers use the HARMONY framework to track transport simulations, organize their work for future reuse, and structure output data for insights and decision support. Teams of modelers use the framework to log and compare results across multiple users working on the same problem. They can try different simulators to tackle the same problem and then run the same simulations again on new data to compare results in the future. The data management framework supports and drives the development of the platform and provides the basis for making sense of transport simulations in the integrated HARMONY platform. Note that the HARMONY TSDW is the instantiation of the proposed framework and consists of data stored in database systems as well as artifacts (files) stored in the filesystem of the server where the HARMONY platform is deployed.

1 Introduction

1.1 Project Summary

HARMONY envisions to assist metropolitan areas with evidence-based decision making, by providing a state-of-the-art model suite that quantifies the multidimensional impact of various policies, investments and mobility concept applications, while simultaneously identifying the most appropriate solutions and recommending ways to exploit disruptive mobility innovations. HARMONY proposes an integrated approach through the development of the HARMONY Model Suite (HARMONY MS) which integrates new and existing sub-models. This integrated approach is necessary to understand if, how and to what extent new policies, investments and mobility concepts can produce results that are in line with the objectives set by authorities.

The HARMONY MS enables end-users such as planners, decision makers, researchers and transport operators/providers to couple/link independent models and analyse a portfolio of regional and urban interventions for both passenger and freight mobility. The main objective behind the model system's architecture is to enable the evaluation of such interventions with regards to their impact on land-use, economic growth, transportation networks, energy, vehicular noise and emissions, while, at the same time, provide recommendations for Sustainable Urban Mobility Plans (SUMP) of the new mobility era.

HARMONY follows a multi-scale approach (described in detail in D1.3) consisting of the **Strategic Level** (Long-term), the **Tactical Level** (Mid-Term) and the **Operational Level** (Short-term) models.

- **Strategic Level:** It is mainly composed of regional economic, demographic forecasting, land-use, spatial freight interaction and long-term mobility choice models. It operates on a long-term horizon (year-to-year) and is mainly responsible for generating i) disaggregate household and firm population and the locations for different types of activities such as employment, housing, and education, ii) aggregate commodity flows between employment sectors and iii) long-term mobility choices of individuals (agents) including car-ownership or subscriptions to different mobility services.
- **Tactical Level:** it is a fully agent-based passenger and freight demand model and it consists of two sub-models one for passenger and one for freight which model agents' choices on a day-to-day level.
- **Operational Level:** it represents the transport supply and demand interactions at high granularity (e.g. second to second, minute to minute). It can be characterised as a multimodal network assignment model system that is responsible for loading the demand into different types of networks, while simultaneously capturing travellers' route choices and dynamic schedule re-evaluation choices due to supply conditions. It also includes dedicated modules that emulate disruptive new mobility service operations and their interactions with agents (e.g. traveller, vehicles) of the system. This simulator is mainly responsible for generating traffic volumes and impedance measures of the form of skim matrices (e.g. travel time, cost distance) per model and spatial unit of analysis.

1.2 Deliverable Objectives

This deliverable describes the final prototype of HARMONY's Transport and Spatial Data Warehouse (TSDW) and is the outcome of the work performed in tasks T3.3 "Data management mechanisms", T3.4 "Data security & Privacy" and T3.5 "TSDW Integration, and population". It is an extended version of D3.3 "Transport and Spatial Data Warehouse 1st Prototype" and emphasizes on the development and utilization of a tool that has been developed to manage the warehouse as part of the operation of the HARMONY MS.

The objective of this deliverable is to describe the above stated management tool, that allows users of the HARMONY MS to control the TSDW, and demonstrate its application with one of the models (the operational freight simulator) developed as part of the HARMONY MS. The description of .

1.3 Deliverable Structure

The deliverable is structured as follows. Section 2 provides an overview of the HARMONY Data Management Framework while Sections 3-7 focus on describing the data entities that comprise the framework:

- Section 3 focuses on the description of user groups, access rights and roles.
- Section 4 describes the transport simulators registry, a data entity which is used to keep track of the different simulators that are part of the HARMONY MS.
- Section 5 explains how modelling templates, projects and simulation scenarios are specified in the data warehouse.
- Section 6 describes the metadata used to describe the log data generated by the harmony simulators.
- Section 7 shows how key performance indicators inferred from the output data of the simulators are stored.

The final two sections (8 and 9) present the framework used to develop the TSDW management tool and the deliverables' conclusions respectively.

2 Data Management Framework of the TSDW

The need for the HARMONY Data Management Framework stems from the fact that in HARMONY, the integration of simulators is performed at different levels (operational, tactical and strategic), which have to efficiently communicate, exchange information and data in an automated manner, while the HARMONY platform has to be able to store and manage the required input and output data. Moreover, in HARMONY different users and user groups can have access to the functionalities, as well as the stored information. User access rights are required, in order to ensure that sensitive information can be viewed only by particular users (e.g. transport modelers of a city or an area), while parts of the information can be accessible by a wider set of stakeholders (e.g. transport modelers of a different city). Additionally, part of the stored data refers to analytics and insights that the platform generates based on the output of the simulations. Such insights support decisions of transport stakeholders with respect to the integration of new mobility services.

The HARMONY data management framework and its position within the HARMONY platform is shown in Figure 1.

The HARMONY data management framework has been designed such that it captures all needed information and adheres to the needs of the transport modellers developing the simulators as well as the transport stakeholders who need to access, analyse and understand the simulation outputs.

The final version of the HARMONY TSDW implements the proposed Data Management Framework and comprises of all data and metadata stored in the HARMONY database as well as artifacts (files) stored in the HARMONY server.

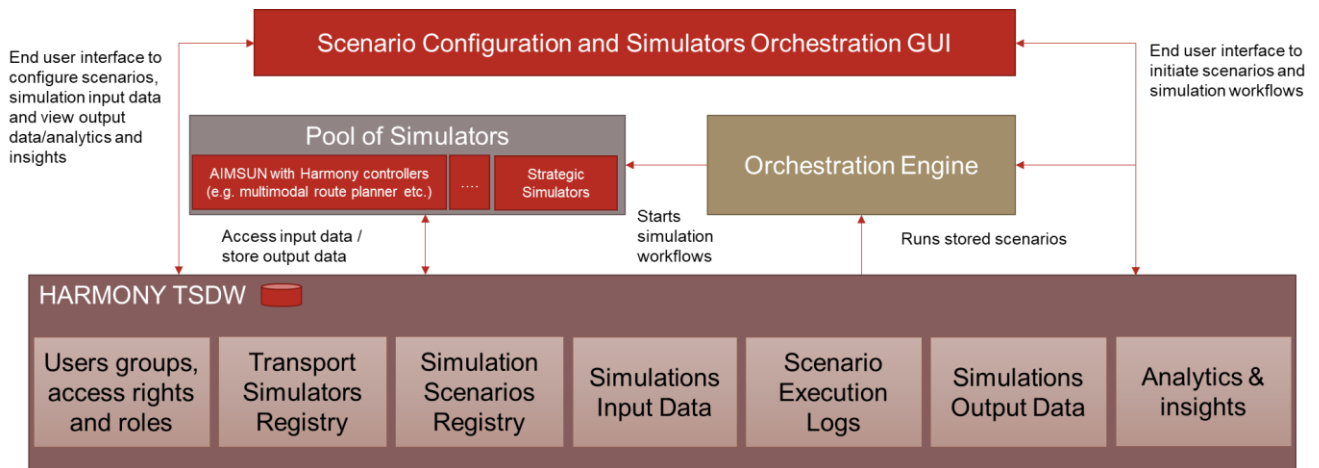


Figure 1: The HARMONY data management framework as part of the TSDW and its position within the HARMONY platform.

It consists of seven main data entities as follows:

- **User groups, access rights and roles** that encompasses data related to access control and security.
- **Transport Simulators registry** that maintains data and metadata related to the available HARMONY MS components and their versions. Each component incorporates metadata about inputs and outputs required to run the **simulations**.
- **Modelling Templates, Project and Scenarios** registry that stores data related to configured scenario templates and related scenarios that users execute. A scenario template provides a simulation workflow. A template can be instantiated in different projects with a set of input parameters for a specific modelling need. Furthermore, projects can instantiate different scenarios that can evaluate variations of a specific project. For example, a dynamic freight scheduling template can define an execution workflow where the tactical and operational freight simulators are run. Based on the template, a project that evaluates freight logistics with low emission zones can be defined. For this project, two scenarios may be realised, one where the logistics fleet is composed of electric vehicles, and a second one where the logistics fleet is composed of electric vehicles and bicycles.
- **Scenario Execution Logs** that hold logs which are produced when the simulation run. These can be useful especially for transport modellers implementing the simulators.
- **Analytics and Insights** that infers KPIs and stores them in a format that can be readily used by the HARMONY front-end applications so that end users can analyse the results and take transport planning decisions.

The following sections described the aforementioned data entities and how they are managed in HARMONY's TSDW.

3 User groups, access rights and roles

The HARMONY platform is accessed by different users, with varying interests regarding the functionalities and information stored in the platform. The platform needs to be aware of the access level of users and allow access to the functionalities and information relevant for the particular user. For this purpose, the 'user groups, access rights and roles' data entity stores related information.

A Role-based access control (RBAC) approach is followed, which establishes permissions based on groups (defined sets of users) and roles (defined sets of actions). Individuals can perform any action that is assigned to their role and may be assigned multiple roles as necessary. Users are not permitted to change the level of access control that has been assigned to their role.

The RBAC model allows HARMONY administrators to create roles for various job functions and assign permissions to perform certain operations to specific roles. Members or staff (or other system users) are assigned particular roles, and through those role assignments acquire the permissions needed to

perform particular system functions. With RBAC users are not assigned permissions directly, but acquire them through their role (or roles), which means that the management of individual user rights becomes a matter of simply assigning appropriate roles to the user's account; this simplifies common operations, such as adding a user, or changing a user's department.

Three main roles are foreseen for HARMONY at this point as follows:

- Admin users: This role has access to all functionalities and data of the platform. The admin user can configure user accounts, manage the platform and have global access to the data which are available in the platform.
- Transport modellers: This role has access to all functionalities and data for a specific area i.e. read and write access (change the network, introduce new service types, etc.) and can configure and run simulation scenarios.
- Policy makers: Policy makers and observers are mainly interested in the results of transport simulation and access related output data and generated analytics and insights. These users cannot configure or run simulations. Their main aim is to check and understand scenarios and respective results for decision making within the area/city.

3.1 User management definition

The HARMONY TSDW platform provides three pages for defining users, their roles and their access privileges. Figure 2, displays the page that can be used for the definition of a new organisation, while Figure 3, illustrates the page that allows the specification of new roles/groups. It has to be noted that the latter requires the assigning of the role/group to an organisation, and therefore one, or more organisations must have been created in advance. Lastly, Figure 4 displays the page that can be used for the creation of a new user. Each user must be assigned a role/group and therefore a similar interdependency as the one described above exists in this case between a role/group and a user. Hence, roles/groups must have been defined before a user can be created.

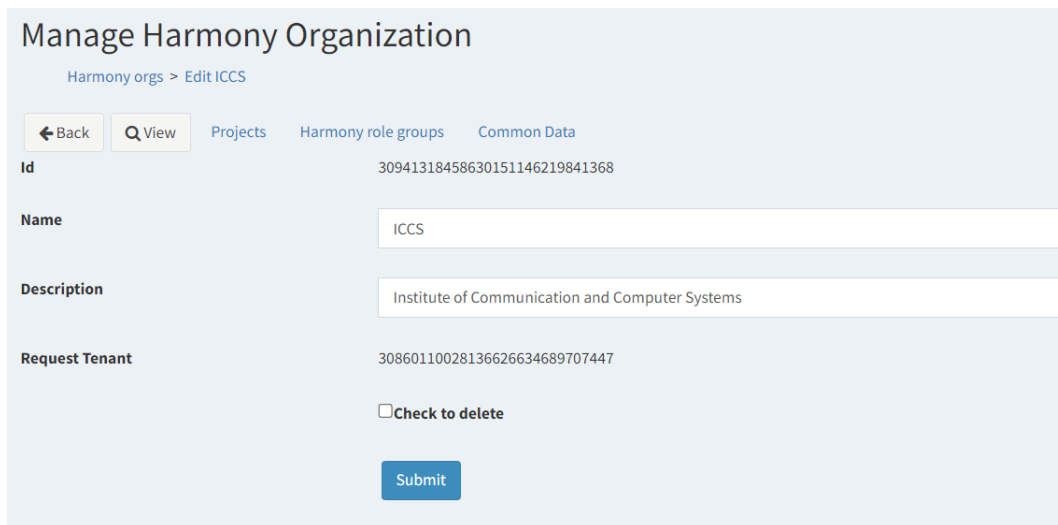


Figure 2: Organisation definition page



Manage Harmony Role/Group

Harmony role groups > New Harmony role group

[Back](#)

Name ICCS TSDW Group

Role Type Modeler

Organization ICCS

Description The team of ICCS responsible for modelling

Access Rights

- Execute
- Execute All
- Read
- Read All
- Write

Request Tenant 30860110028136626634689707447

[Submit](#)

Figure 3: Role/Group definition page

Manage Harmony Users

Harmony users > New Harmony user

[Back](#)

Name John Modeller

Username jmodeller

password_hash d3de87ff28ca513477b595dfbd7a22b6

password_salt f1nd1ngn3m0

Date 2022-02-03

Role/Group ICCS TSDW Group

Request Tenant 30860110028136626634689707447

[Submit](#)

Figure 4: User definition page

In terms of implementation, the HARMONY deployment can be multitenant, which means that a single server hosts users and data coming from different areas or single tenant which means that the platform is deployed in the premises of an organization (e.g. a municipality) and is used by the transport modelers of that specific organization. In the case of multi-tenant deployments where more than one organizations make use of the same server, different user groups can be configured having access to the specific data of their organization.

For example, if we consider the Oxfordshire City Council (OCC), a pilot partner of HARMONY, the HARMONY administrator configures a group of Transport modelers and another of Policy makers. When a new user account is configured in this instance of the HARMONY platform, the user is allocated to one of the preconfigured groups depending on her/his role. The respective definitions for this example, in a JSON format used to store them in the HARMONY TSDW can be seen bellow.

```
//HARMONY platform roles / user groups for a typical deployment
{
  "id": "d290f1ee-6c54-4b01-90e6-d701748f0851",
  "name": "Admin",
```



```

"type": "admin",
"organization": "HARMONY",
"description": "This is a group for the admin users",
"access_rights": ["read_all", "write_all", "execute_all"],
}
{
  "id": "d290f1ee-6c54-4b01-90e6-d701748f0852",
  "name": "Modeller_OCC",
  "type": "modeler",
  "organization": "OCC",
  "description": "This is a group for the transport modelers of OCC",
  "access_rights": ["read", "write", "execute"], //users of this group can
view /write data created within the OCC organization and run simulation
scenarios
}
{
  "id": "d290f1ee-6c54-4b01-90e6-d701748f0852",
  "name": "Policy_maker_OCC",
  "type": "policy_maker",
  "organization": "OCC",
  "description": "This is a group for the policy / decision makers modelers
of OCC",
  "access_rights": ["read"], //users of this group can view data created
within the OCC organization
}
{
  "id": "d290f1ee-6c54-4b01-90e6-d701748f0853",
  "name": "External users",
  "type": "external",
  "description": "This is a group for users who can view public HARMONY
data",
  "access_rights": ["read_public"], //users of this group can view public
data only
}
//HARMONY platform example user for a typical deployment
{
  "id": "d290f1ee-6c54-4b01-90e6-d701748f0854",
  "name": "Patrick Mueller",
  "username": "pmueller@gmail.com",
  "password_hash": "234234sfdefwf23fe",
  "password_salt": "234232efef",
  "date": "2020-10-18T16:00:00Z",
  "group": "Modeler_OCC",
}

```

4 Transport Simulators registry

The HARMONY TSDW allows the definition and storage of metadata related to the specification of different simulators (components). The modeller can define a new simulator by entering metadata as shown in Figure 5. In addition, the specifications for the inputs and outputs for each simulator can be defined (Figure 6). This allows the generation of a component specification file that can be utilised from the HARMONY MS for the definition of templates, projects and scenarios. In addition to the definition of new components together with their inputs and outputs, the TSDW management tool allows editing functionality for all elements created (Figure 7). This allows users of the tool to edit or delete information about the components.





Manage Components

Components > Edit Operational Freight Simulator

[← Back](#) [Q View](#) [Component Inputs](#) [Component Versions](#) [Component Outputs](#)

Id 30934763097277423887709448024

Name Operational Freight Simulator

Description This component is the operational freight controller of the HARMONY MS

Model ID OFS-1

Updated 2022-03-21 10:49:24

Simulation Level Operational

Request Tenant 30860110028136626634689707447

☐ Check to delete

[Submit](#)

Figure 5: Component definition in TSDW

Manage Components

Components > Operational Freight Simulator > Component Inputs > New Component Input

[← Back](#)

Key UseCase

Description The name of the use case to be executed.

Label Name of the scenario to be executed

Type TextBox

Scalar Type String

Semantic Type N/A

☐ Required

Default Value baseline

Associated Component Operational Freight Simulator

Request Tenant 30860110028136626634689707447

[Submit](#)

Figure 6: Component Input/Output definition in TSDW

Manage Components

Components > Operational Freight Simulator > Component Inputs

[+ Add Record](#) [Search](#) [Clear](#)

Key	Label	Scalar Type	Type	Required	
UseCase	Name of the scena...	string	textbox	<input checked="" type="checkbox"/>	Input Options Q View Edit Delete
SettingsFile	Settings File	None	file	<input checked="" type="checkbox"/>	Input Options Q View Edit Delete
AimsunModelFile	Aimsun Model File	None	file	<input checked="" type="checkbox"/>	Input Options Q View Edit Delete
PathAssignmentFile	Path Assignment File	None	file	<input checked="" type="checkbox"/>	Input Options Q View Edit Delete

Export: [CSV](#) [CSV \(hidden cols\)](#) [HTML](#) [JSON](#) [TSV \(Spreadsheets\)](#) [TSV \(Spreadsheets, hidden cols\)](#) [XML](#)

Figure 7: Component specification editing options in TSDW



For each simulator, several metadata are configured in the HARMONY warehouse and are regularly updated to reflect the simulators' current status. These include:

- `model_id`: Identifier of the simulator as defined by its creator.
- `name`: A descriptive name of the simulator.
- `description`: Short description of the simulator highlighting its main purpose.
- `versions`: An array with metadata about the different versions of the simulator as created over time.
- `inputs`: an array with descriptions of the inputs required to run the simulator.
- `outputs`: an array with descriptions that are generated by the simulator.

The json representation of a component as stored in the TSDW and used by the HARMONY MS can be seen below.

```
[
  {
    "modelId": "ofs",
    "name": "Operational Freight Controller",
    "description": "This component is the operational freight controller of the HARMONY MS.",
    "versions": [
      {
        "dockerContainerName": "ofs",
        "versionDescription": "Version September 2022",
        "versionId": "v1",
        "versionName": "Version 1",
        "versionTopicName": "ofs"
      }
    ],
    "inputs": [
      ...
    ],
    "outputs": [
      ...
    ]
  }
]
```

4.1 Components' input data

Each component/model of the HARMONY MS requires certain input data in order to function. The input data may vary from network geometries (i.e., nodes and links) to demand data (i.e. activity schedules) as well as data for the configuration and initialization of related functionality (i.e. which internal modules of the component need to be executed). The related data are stored in the HARMONY TSDW, as part of the database or as data artifacts in the HARMONY server filesystem.

The following metadata properties are considered per input data instance:

- `Description`: A brief description of the data input.
- `Fieldtype`: The input type for generating the relevant UI component on the HARMONY MS web interface.
- `Key`: A unique identifier for the data input.
- `Label`: A name for the data input to be displayed to the user.
- `Required`: A boolean variable for determining whether the input is required, or not.
- `ScalarType`: The data type of the input if it is a single value (ie, string, number, etc.)
- `Value`: A default value for the input.

As an example, a subset of the inputs for the operational freight controller can be seen below:



```
"inputs":
[
  {
    "description": "The name of the use case to be executed.",
    "fieldType": "textbox",
    "key": "UseCase",
    "label": "Name of the scenario to be executed",
    "required": true,
    "scalarType": "string",
    "value": "baseline"
  },
  {
    "description": "The settings file for the ofs.",
    "fieldType": "file",
    "key": "SettingsFile",
    "label": "Settings File",
    "required": true,
    "value": "settings.json"
  },
  {
    "description": "The aimsun model file.",
    "fieldType": "file",
    "key": "AimsunModelFile",
    "label": "Aimsun Model File",
    "required": true,
    "value": "Rotterdam.ang"
  },
  {
    "description": "The path assignment file.",
    "fieldType": "file",
    "key": "PathAssignmentFile",
    "label": "Path Assignment File",
    "required": true,
    "value": "PathAssignment_276047.apa"
  }
]
```

The TSDW offers a file management interface for managing input files on the HARMONY MS' file system. This can be shown in Figure 8 below.



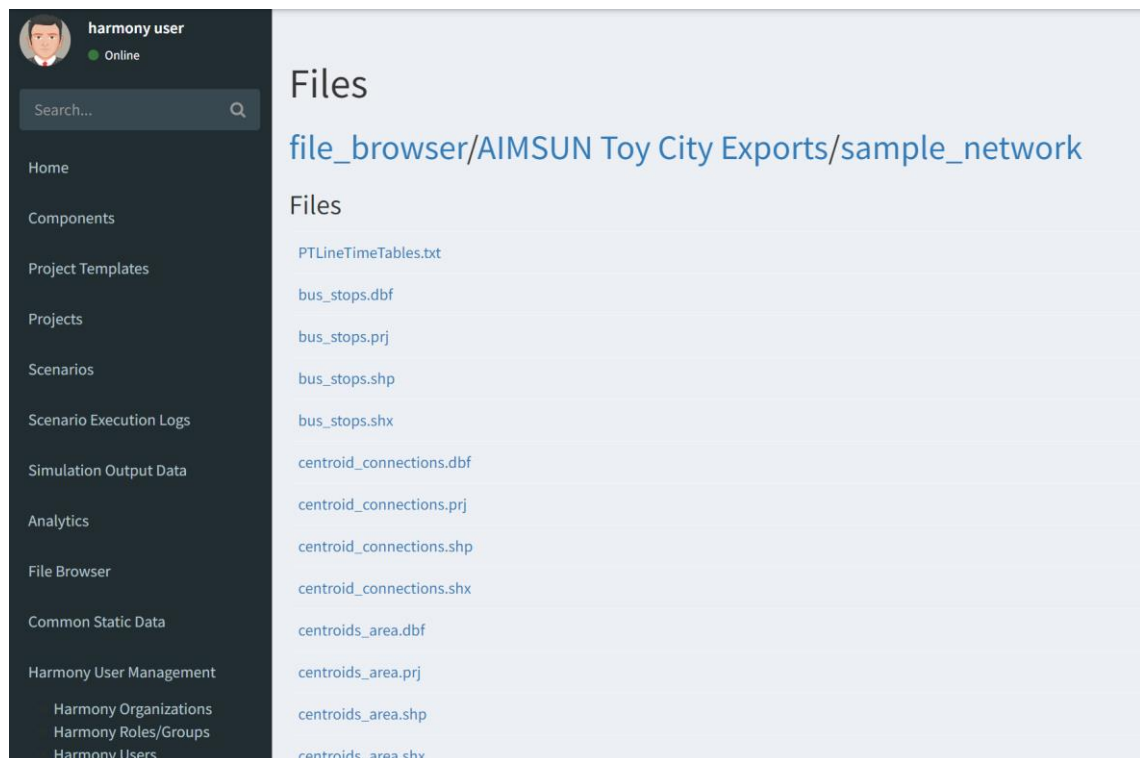


Figure 8: File management in TSDW

4.2 Components' Output Data

In a similar manner, TSDW can manage and store output data which result from the execution of a simulation scenario. These data are mainly used for the calculation of scenario KPIs. The metadata which are associated with the output data of a component include:

- Description: A brief description of the data output.
- FieldType: The output type for generating the relevant UI component on the HARMONY MS web interface.
- Key: A unique identifier for the data output.
- Label: A name for the data output to be displayed to the user.
- Value: A default value for the output.

As an example, a subset of the output data from the operational freight controller can be seen below:

```
"outputs":
[
  {
    "description": "A file with the outputs vehicle statistics",
    "fieldType": "file",
    "key": "VehicleStats",
    "label": "Vehicle Stats",
    "value": "VehicleStats.csv"
  },
  {
    "description": "A file with the outputs network statistics",
    "fieldType": "file",
    "key": "SectionsStats",
    "label": "Network Stats",
    "value": "SectionsStats.csv"
  }
]
```

5 Modelling Templates, Projects and Scenarios

In HARMONY MS, simulations are configured through modelling templates. Such a template addresses a specific modelling need and is composed of a processing workflow that is executed by running different simulators which interact and finally provide the simulation output that in turn supports policy decisions. A template can be instantiated in a project and a scenario with specific input data, calibrated and defined for a city/area of interest. Transport modelers configure projects for their modeling needs (e.g. for examining zero emission zones and green districts or for planning consolidation centers). Projects can be run with different input configuration data (scenarios) in order to allow transport modelers to compare different “treatments” and understand the effects of different configurations and designs.

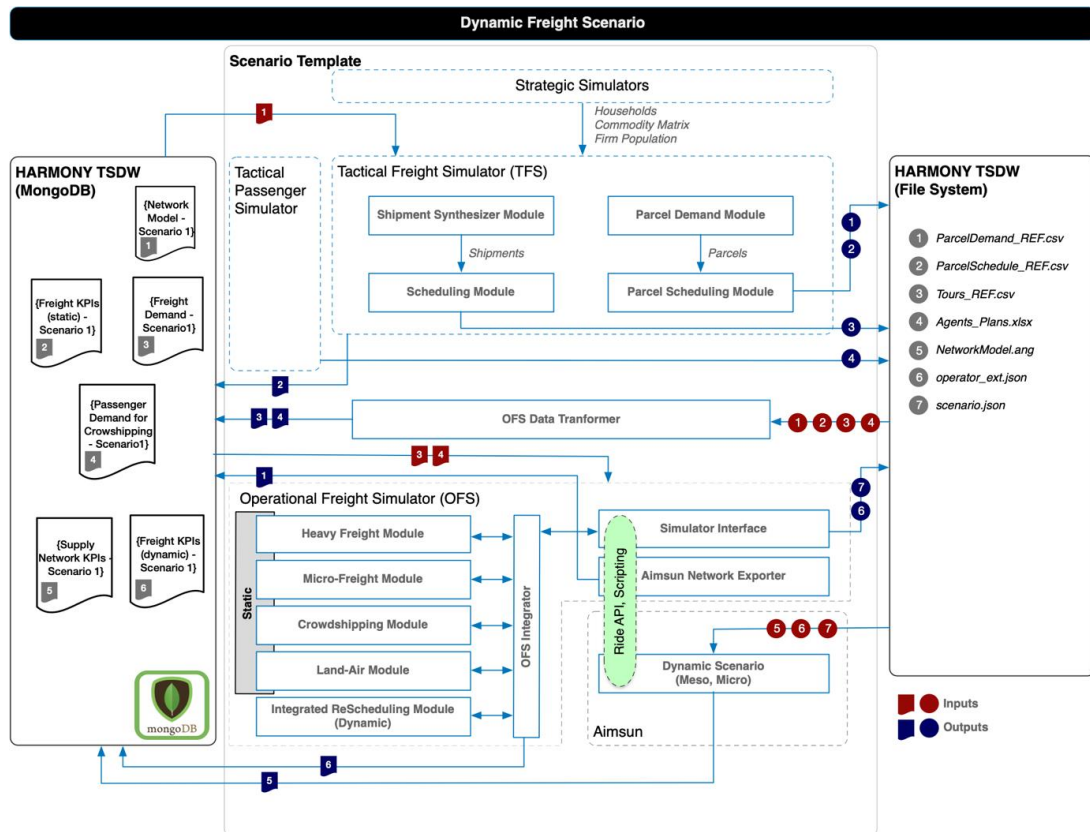


Figure 9: The template for dynamic freight simulation and the data configurations which result in a related scenario.

Figure 9 shows an instantiation of a template that allows the integration of the tactical and operational freight simulators. The workflow commences with the generation of freight demand (heavy freight and parcels) using the different modules included in the Tactical Freight Simulator (TFS). This involves two synthesizers (one for shipments and one for parcels) which generate the demand in the form of O/D matrices, and two scheduling modules which convert the O/D matrices into tours that different types of vehicles need to undertake. The demand and schedules from the tactical layer are fed into the Operational Freight Simulator (OFS) which simulates vehicle tours using dynamic traffic assignment in the supply simulator.

Figure 10 displays the screen that allows the definition of the aforementioned template in the TSDW.



Manage Project Templates
Project Templates > Edit Dynamic Freight Simulation

◀ Back Q View Workflow Tasks

Id 30935500294528154913969874776

Name

Description

Project Template ID

Simulation Levels
☐ Operational
☐ Strategic
☐ Tactical

Simulation Workflow
☐ Operational Freight Simulator
☐ Tactical Freight Simulator (TFS)
☐ Tactical-Operational Freight Controllers Data Transformations

Category

KPIs

Request Tenant 30860110028136626634689707447

☐ Check to delete

Figure 10: Modelling template definition in TSDW

For each modelling template the following metadata are configured:

- **Name:** Descriptive name of the modelling template.
- **Description:** Detailed information which shows the aim and scope of the scenario template.
- **Template ID:** A unique identifier for the template.
- **Simulation levels:** Involved strategic/tactical/operational simulators required to run scenarios generated using the template.
- **Simulation workflow:** The list of simulators that are part of this template. It refers to an array of IDs of simulators that the platform needs to run in order to execute a scenario template workflow.
- **Category:** Category of this scenario template
- **KPIs for evaluation:** list of KPIs that are calculated with the use of the output data of the simulation workflow.

And the corresponding (minimised) json format is as follows:

```
{
  "name": "Dynamic Freight Simulation",
  "description": "A template that runs the tactical and operational
simulators",
  "templateSaved": true,
  "id": "b5f97686-ecda-e5c0-1d83-646ea972569a",
  "components": [
    {
      "name": "Tactical Freight Simulator (TFS)",
      "description": "The TFS calculates freight demand at the tactical
level in the form of shipments and tours.",
      "modelId": "tfs",
      "versions": [
        ]
      ],
      "inputs": [

```

```

    ],
    "outputs": [
    ],
    "createdBy": "501d4c4e-e0ec-11ec-8e06-0242ac180005",
    "componentSaved": true,
    "componentId": "7be094f1-e5bd-41a1-ad8f-ec7f3c87375d"
  },
  {
    "description": "This component is the operational freight
controller of the HARMONY MS.",
    "inputs": [
    ],
    "modelId": "ofs",
    "name": "Operational Freight Controller",
    "outputs": [
    ],
    "versions": [
    ],
    "createdBy": "501d4c4e-e0ec-11ec-8e06-0242ac180005",
    "componentSaved": true,
    "componentId": "9e5c6644-698f-4953-8241-e90e8d7edcae"
  }
]
}

```

The modelling templates can be used to define projects that correspond to use cases where the workflows defined in the templates can be executed. Each project needs to be specified with a name and description in a similar manner to that of the template (Figure 11).

Create a new project from a template	
Name	Micro-consolidation centers assessment
Description	A project that assesses the impact of micro-consolidation centers on the performance of urban networks.
Organization	ICCS
Template	Dynamic Freight Simulation
Is Active	<input checked="" type="checkbox"/>
Created On	2023-03-08 14:48:06
Created By	harmony user (30954305413502596871368570572)
Modified On	2023-03-08 14:48:06
Modified By	harmony user (30954305413502596871368570572)
Request Tenant	30954305413502596871368570572

Figure 11: Project definition in TSDW

In addition, at the project specification stage the inputs required for each project (defined as part of the models/simulators that are included in the template) need to be uploaded to the HARMONY server. Scenarios are merely different instantiations of a project where inputs may change to allow the evaluation of variable configurations of a project.

6 Scenario Execution Logs

Each simulator generates logs that provide a detailed view of how the simulator runs. Depending on the implementation approach of each simulator the logs can be stored as part of the TSDW either on the HARMONY database or in the server's filesystem. When creating a log, the simulators provide the following metadata:

- AssociatedScenario: The scenario associated with the log.
- Format: Can be database object or file (artifact) stored in the filesystem.
- Access details: Can be a query string for fetching the data or a path in the filesystem.

An example record of an execution log entry can be seen below, while the execution log definition screen of the TSDW can be seen in :

```
{
  "id": "30fd231e342c128642dd0911d",
  "date": "2020-10-18T16:00:00Z",
  "format": "artifact",
  "access_details": "/harmony/network_models/logs/30fd231e342c128642de098ed/O2_log.csv",
  "associated_scenario": [ObjectId("30fd231e342c128642de098ed")],
}
```

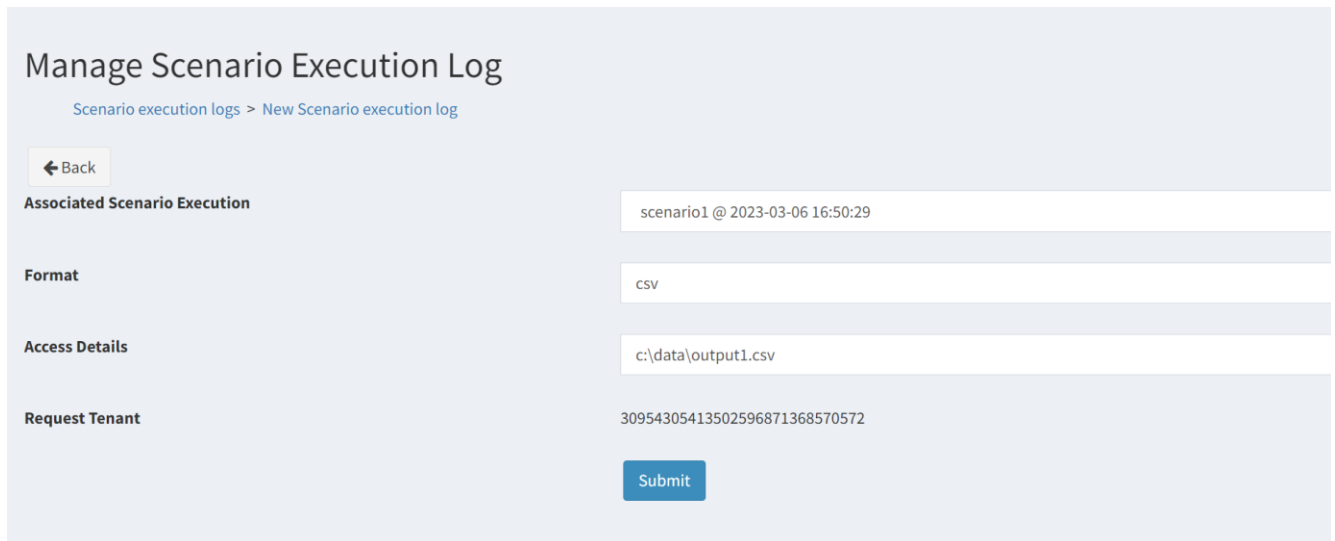


Figure 12: Execution log definition in TSDW

7 Analytics and Insights

This data entity is used to describe Key Performance Indicators (KPIs) associated with a scenario. The metadata describing the KPIs include:

- KPIName: Descriptive name of the KPI
- Date: Datetime object showing when the KPI was last updated.
- AssociatedScenario: A scenario for which the KPI is calculated.
- Type: This can be a time series or a single value.
- KPI calculated values: An object providing the calculated values of the KPI depending on its type.

As an example, a subset of the KPIs for a micro freight logistics with low emission zones scenario are the following:

```
{
```



```

    "id": "30fd231e342c128642de0911d",
    "KPIname": "average freight CO2 emissions",
    "date": "2020-10-18T16:00:00Z",
    "AssociatedScenario": "[ObjectId("30fd231e342c128642de098sd")]"
    "type": "single_value",
    "calculated_values": "105gr/km",
  }
  {
    "id": "30fd231e342c128642de0911d",
    "KPIname": "average hourly freight CO2 emissions",
    "date": "2020-10-18T16:00:00Z",
    "associated_scenario": "[ObjectId("30fd231e342c128642de098ed")]"
    "type": "time_series",
    "calculated_values": {"09:00": "100gr/km", "10:00": "105gr/km", "11:00":
    "110gr/km", "12:00": "110gr/km", "13:00": "105gr/km", "14:00": "100gr/km"}
  }

```

8 Data Warehouse Implementation

The HARMONY TSDW management tool has been implemented using the web2py open source web framework that enables the development of scalable, secure and portable web applications. A number of ‘controllers’ and ‘views’ have been developed to offer the functionality described in the aforementioned section (Figure 13).

web2py™ administrative interface

Site Edit About Errors Versioning Help Logout Debug

Edit application "TSDWDMFv3_0"

collapse/expand all models controllers views languages static modules private files plugins

Models

database administration sql.log graph model

Edit 0.py

Edit db.py

Edit db_models.py defines tables analytics, common_data, component, component_argument_scenario_value, component_argument_value, component_input, component_input_data_instance, component_input_data_scenario_instance, component_input_option, component_input_project_value, component_input_scenario_value, component_output, component_output_data, component_version, harmony_org, harmony_role_group, harmony_user, kpi_definition, project, project_category, project_template, scenario, scenario_execution, scenario_execution_log, workflow_task, workflow_task_gen_input, workflow_task_seq

Edit db_models_populate.py

Edit menu.py

Create

Controllers

test crontab

Edit appadmin.py exposes ccache, csv, d3_graph_model, download, hooks, index, insert, manage, select, state, update

Edit default.py exposes analytics_manage, call, common_data_manage, common_data_manage, component_input_data_manage, component_manage, dashboard, download, download2, error, file_browser, harmony_org_manage, harmony_role_group_manage, harmony_user_manage, index, kpi_definition_manage, project_category_manage, project_execution_manage, project_manage, project_template_manage, project_template_manage_test, scenario_execution_log_manage, scenario_manage, simulation_output_data_manage, user

Edit project.py exposes project_create, project_create_step2, project_create_step2_v1, project_create_step3

Edit services.py exposes api, call, echojson, export_data, getProjectTemplates, getProjectTemplatesv1_old, getStatus, import_data, insertModel, insertProjectTemplate, insertProjectTemplatev1_old, parseModel, parseProjectTemplate

Figure 13: TSDW application development page

9 Conclusions

This deliverable presented the work undertaken for the development of the TSDW, including the necessary data management and access mechanisms, as well as its integration with HARMONY MS. Furthermore, the web-based tool that can be utilised for its population was presented. The developed tool allows: (i) the definition of user groups and roles together with their respective access rights, (ii) the definition of components to be used as part of the HARMONY MS, (iii) the definition of modelling





templates, projects and scenarios which form the backbone of simulations run as part of the HARMONY MS, (iv) the definition of logs that can store outputs of simulations and (v) the definition of KPIs that can encapsulate simulation results. The developed tool can be used by users of the platform to manage the TSDW and also various metadata related to the aforementioned functionalities.



[@Harmony_H2020](https://twitter.com/Harmony_H2020)

[#harmony-h2020](https://twitter.com/Harmony_H2020)



<https://www.linkedin.com/company/harmony-h2020/>

For further information please visit www.harmony-h2020.eu



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 815269

