European data space
for smart communities

Deliverable 3.3

DS4SSCC Deliverable 3.3 Creating the Model Input: Dataset Validation, Digital Twin Finalisation and Model Data Delivery

WP 3 – Model usage phase

Authors: Corné Helmons (Rotterdam)
Reviewers: Niek Verschure ('s-Hertogenbosch), Anni Joela (Tampere), Margriet Visscher (Future Insight)
Delivery date: 7 Juli 2026
Dissemination level: Sensitive
Type: Report

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or of the granting authority. Neither the European Union nor the granting authority can be held responsible for them

Revision History

Author Name, Partner short name	Description	Date
Corné Helmons, Rotterdam	City Project Manager	7-Juli-2026

Abbreviations

Abbreviation	Definition	Abbreviation	Definition
DS4SSCC	European data space for smart communities (DS4SSCC)	DT-aCoGo	Digital Twinning as Common Good
OASC	Open & Agile Smart Cities & Communities	OUP	Open Urban Platform
WP	Work Package	PC	Pilot Coordinator
BIM	Building Information Modeling	WFS	Web Feature Service
OGC	Open Geospatial Consortium	ETL	Extract, Transform, Load
LOD	Level of Detail	UMG	Unreal Motion Graphics
REST	Representational State Transfer		

Table of contents

1. Executive Summary 4
2. Introduction 6
3. Problem Definition 8
4. Loading datasets from OUP in analysis platforms 10
4.1 Accessing datasets through OUP hubs 10
4.2 Model-specific integration approaches 10
4.2.1 Rana 10
4.2.2 DatMobility – Traffic model 12
4.2.3 Tygron – Heat stress model 14
4.3 Conclusion 16
5. Registering model APIs in OUP 17
5.1 General approach to API registration 17
5.2 Rana – API registration and usage 17
5.3 DatMobility – API registration and integration 18
5.4 Tygron – API registration and proxy integration 18
5.5 Role of OUP in API governance 19
5.6 Summary 20
6. Conclusions and next steps 21
6.1 Conclusions 21
6.2 Key challenges identified 21
6.3 Next steps 22
6.4 Final reflection 23
7. Annexes 24
A1: Rana_dtacogo_sdk_light_v4.pdf 24
A2: 20260629 Deliverable I - Gathering data and setting up model Part 2.pdf 24
A3: 20260629 Deliverable I - Flowchart From Data to Flood Models .pdf 24
A4: Dat.Mobility; https://www.goudappel.nl/expertises/beleids-en-gedragsonderzoek/monitoring-en-evaluatie/datagedreven-mobiliteitsbeleid 24
A5: Tygron; Wiki adding measures.docx 24
A6: Tygron; 20260624 DT-aCoGo Tygron Data descriptions.pdf 24
A7: Tygron; 20260624 DT-aCoGo Tygron Data to OUP Instructions.pdf 24
A8: Tygron; Expert mode 2 - Heat Stress Overlay tutorial - Tygron Support wiki.pdf 24

1. Executive Summary

This deliverable presents the finalisation of the DT-aCoGo modelling workflow, demonstrating how harmonised datasets, analytical models, and application interfaces are integrated into a coherent digital twin ecosystem. The work builds upon the architectural and interoperability foundations established in previous deliverables, translating these into an operational framework for model execution and reuse.

The central objective of this work is to enable the consistent and scalable deployment of analytical models within a federated urban data space. This is achieved through the combination of three key principles:

- the harmonisation of heterogeneous urban datasets into a Common Information Model (CIM);
- the integration of analytical models through standardised APIs exposed via the Open Urban Platform (OUP);
- and the application of lifecycle-based filtering to ensure temporal consistency across all components of the digital twin.

Within this framework, datasets are prepared, validated, and published in OUP as authoritative data services. Analytical models consume these datasets through defined interfaces, execute domain-specific simulations, and return results that are subsequently visualised in a unified digital twin environment. This approach ensures that data, analysis, and visualisation remain fully aligned.

The deliverable demonstrates the integration of three complementary analytical domains:

- flood modelling (Rana), representing a process-driven simulation approach;
- traffic modelling (DatMobility), based on data-driven network analysis;
- heat stress modelling (Tygron), using scenario-based evaluation of urban interventions.

While each platform follows a distinct modelling paradigm, all are connected through the same data foundation and integration approach. This confirms that heterogeneous analytical models can operate coherently within a shared ecosystem, provided that data harmonisation and interface standardisation are maintained.

A key outcome of this work is the validation of lifecycle-based temporal filtering as a cross-cutting mechanism. By embedding time as an intrinsic property of the data through attributes such as ValidFrom and ValidTo, all applications—analytical and visual—interpret the same temporal state of the digital twin. This ensures consistency between model inputs, analytical results, and visualisation, and supports exploration of both current and future scenarios.

The results further demonstrate that API-based integration via OUP provides a scalable mechanism for connecting models to the digital twin. By registering model functionality as reusable services, analytical platforms can be accessed in a controlled and standardised manner, while maintaining independence of internal architectures. At the same time, OUP ensures governance, authentication, and discoverability of these services within a federated environment.

Across the pilot cities, the approach has proven to be technically feasible and adaptable to different contexts. At the same time, the work highlights key dependencies, particularly related to data quality, the definition of temporal scenarios, and the variability in model integration capabilities.

In conclusion, the DT-aCoGo framework establishes a robust and replicable approach for integrating analytical models into a shared digital twin ecosystem. By combining harmonised data, standardised APIs, and consistent temporal filtering, the solution enables scalable deployment across cities and domains, supporting data-driven decision-making within the broader European data space for smart communities.

2. Introduction

This deliverable is developed within the European Data Space for Smart Communities (DS4SSCC) project, as part of Pilot G: Digital Twin as a Common Good (DT-aCoGo). The pilot establishes a scalable and interoperable digital twin ecosystem that enables municipalities to run domain-specific analytical models from within a shared visualisation environment.

The DT-aCoGo architecture is built on a strict separation between data services, analytical platforms and visualisation environments, as illustrated in Figure 1. Within this structure, the Open Urban Platform (OUP) serves as the central data space, while Purple Polar Bear (PPB) provides the semantic harmonisation layer that transforms heterogeneous city datasets into a Common Information Model (CIM). Analytical models consume CIM-aligned data from OUP through standardised APIs, and results are published back to OUP for consumption by the Unreal Engine visualisation environment.



Figure 1: Schematic architecture DT-aCoGo

This deliverable builds directly upon the foundations established in the two preceding WP3 deliverables:

- **D3.1 — Interoperability between OUP and sectoral models.** D3.1 defined the Common Information Model, the role of PPB as a semantic interoperability layer, the Replicable Recipe for ecosystem configuration, and the integration approach for the three analytical models (Rana, DatMobility, Tygron). It established the architectural and semantic foundations upon which model input creation depends.
- **D3.2 — Timelapse viewing slider validated.** D3.2 confirmed the lifecycle-based filtering approach within the Unreal Engine environment, validating that temporal attributes (ValidFrom, ValidTo) can be used to ensure consistency between visualisation and analysis. The temporal mechanism described in D3.2 is directly applied in this deliverable when defining the 'points in time' for each analytical scenario.

Where D3.1 focused on the architectural design and technical connectivity of the ecosystem, this deliverable focuses on the operational process of preparing model inputs — the concrete steps required to move from registered source datasets to analytically ready, temporally scoped model inputs that are delivered to each model provider.

The scope of this deliverable encompasses:

- The dataset verification and quality assessment process across three pilot cities
- The definition and configuration of temporal scenarios ('points in time') for each analytical domain
- The creation of the definitive Digital Twin within OUP, establishing the authoritative data state for model execution
- The delivery of harmonised, CIM-aligned data to the Rana, DatMobility and Tygron platforms
- The validation of model input across all three cities and domains

The three analytical domains addressed in this deliverable correspond to the key urban resilience themes of the DT-aCoGo pilot: flooding resilience (Rana), heat stress (Tygron) and traffic management (DatMobility). Each city contributes datasets covering all three domains, enabling cross-city comparison of model inputs and results.

3. Problem Definition

Within the DT-aCoGo ecosystem, the preparation and delivery of model input data does not start from heterogeneous and unaligned datasets, but from a situation in which datasets are already semantically harmonised and made available through the Open Urban Platform (OUP). While this significantly reduces complexity compared to traditional data integration scenarios, challenges remain in the way analytical models connect to and utilise these datasets.

3.1 Availability of semantically aligned datasets in OUP

All datasets provided by the participating municipalities have been semantically aligned using the Common Information Model (CIM), based on the information requirements of the analytical models. These harmonised datasets are registered and published within the OUP, specifically within the hubs operated by Purple Polar Bear (PPB) for each city.

Through this setup, datasets are made available as standardised data services, allowing model providers to discover and access the information they require. The hub-based architecture ensures that each city maintains its own controlled data environment, while enabling consistent data access across the ecosystem.

3.2 Model access to OUP data services

Model providers can establish connections to the relevant OUP hubs in order to retrieve the required datasets. Through these connections, the models are able to identify and consume the available data services that match their input requirements.

In addition, the lifecycle-based time filtering mechanism ensures that datasets are interpreted consistently across applications. By applying the shared time filter logic, each model is able to select and process the correct temporal state of the data, without requiring additional data transformation steps.

3.3 Variability in integration capabilities across models

Despite the availability of harmonised data services through OUP, not all analytical platforms were able to implement a full technical integration with the platform. Differences in system architecture, development flexibility, and available interfaces resulted in varying levels of integration across the three model providers.

In the most advanced case, the Rana platform achieved a full integration with OUP. Users are able to directly connect to the relevant hub from within the application, select the required datasets, and configure the calculation model based on the available data services.

In other cases, full integration was not feasible within the project scope. As a result, alternative approaches were applied, such as hardcoding dataset links within the application. While this approach reduces flexibility, it ensures that the models can still access up-to-date data from OUP and remain operational within the overall workflow.

3.4 Implications for model input preparation

The combination of semantically aligned datasets and partially integrated analytical platforms leads to a hybrid integration landscape. While the data layer is standardised and centrally accessible through OUP, the way in which models consume this data varies between platforms.

This introduces challenges in maintaining consistency, flexibility, and scalability of model input preparation. Although the underlying data is harmonised, differences in integration approaches require additional consideration in configuring model connections and ensuring reliable data usage.

As a result, model input preparation is not only a matter of data availability, but also of aligning the integration approach of each analytical platform with the OUP-based data services, ensuring that all models can consistently access and interpret the same datasets.

4. Loading datasets from OUP in analysis platforms

Following the creation of the definitive Digital Twin within the Open Urban Platform (OUP), the next step in the workflow is the retrieval and use of these datasets by the analytical models. In this phase, model providers connect to the OUP hubs to access the required data services and configure their calculations accordingly.

4.1 Accessing datasets through OUP hubs

All datasets required for analytical modelling are made available as CIM-aligned data services within the OUP hubs. These services represent the authoritative and versioned state of the Digital Twin, including the applied lifecycle attributes defining the temporal scope.

Model providers establish connections to the relevant hubs in order to discover and retrieve the datasets needed for their analyses. Through standardised APIs, each model can request the required data subsets in a format aligned with its input specifications.

The lifecycle-based filtering mechanism ensures that datasets are interpreted consistently across platforms. By applying a shared time parameter, each model retrieves the data corresponding to the selected point in time, ensuring alignment between analytical results and the Digital Twin state.

4.2 Model-specific integration approaches

Although all models access the same OUP-based data services, the way in which datasets are loaded and used varies per analytical platform. These differences are primarily driven by the technical capabilities and integration maturity of each model provider.

4.2.1 Rana

The Rana platform demonstrates a fully integrated approach to model construction within the DT-aCoGo ecosystem. Users can directly connect to the relevant OUP hub from within the Rana environment and select the datasets required for model input.

Within Rana, calculation models are developed within a project-based environment. Each project acts as a structured container in which datasets, processes, and simulation results are organised. This approach ensures that modelling workflows are reproducible, traceable, and suitable for collaboration.

The construction of flood models follows a structured and reproducible workflow, as described in the flowchart “From data to flood models” (see annex). This workflow provides a guideline for converting harmonised datasets from OUP into executable flood simulations.

The process consists of the following main phases:

- Selection of the modelling scenario

The workflow starts by determining whether the model represents the current situation or a future scenario. This choice determines which datasets are selected and how they are interpreted within the modelling process.

- Selection and preparation of input data

Relevant datasets are selected from the available data services and prepared for modelling. This preparation step includes:

- defining the study area and clipping datasets accordingly;
- generating a digital elevation model suitable for hydrological analysis;
- deriving surface characteristics such as infiltration and friction based on soil type and land use.

These preparation steps transform raw input data into model-ready datasets that can be used in hydrodynamic simulations.

- Configuration of the model environment

Within the Rana project environment, datasets are organised and linked to automated processes. These processes consist of predefined computational steps that take selected input data and transform them into model components. This structured setup allows users to consistently configure and execute modelling workflows.

- Creation of the flood model

Based on the prepared datasets, a 2D hydrodynamic flood model is generated. This model represents the interaction between rainfall, surface runoff, infiltration, and flow across the terrain, forming the basis for further analysis.

- Simulation of rainfall scenarios

Using the constructed flood model, rainfall simulations are executed. These simulations typically consist of:

- a rainfall phase with a constant intensity applied across the model area;
- a subsequent dry phase to analyse drainage, retention, and water flow behaviour.

This approach enables the analysis of flooding under controlled and reproducible conditions.

- Execution and interaction with the model

The workflow supports different modes of interaction, including:

- direct simulation execution initiated from the digital twin environment;
- more advanced configuration and analysis performed by expert users within the Rana platform.

This dual approach supports both operational use cases and expert-driven scenario analysis.

Through this structured workflow, Rana provides a consistent method for transforming OUP datasets into executable flood models. The integration with OUP ensures that input data remains up to date, while the process-based modelling approach supports reuse across cities and scenarios.

Detailed technical instructions on model configuration, API usage, and simulation parameters are provided in:
A1: Rana_dtacogo_sdk_light_v4.pdf
A2: 20260629 Deliverable I - Gathering data and setting up model Part 2.pdf
A3: 20260629 Deliverable I - Flowchart From Data to Flood Models .pdf

4.2.2 DatMobility – Traffic model

The DatMobility platform is used within the DT-aCoGo ecosystem to simulate traffic flows and evaluate mobility-related impacts under different urban conditions. The platform operates as a data-driven modelling environment in which transport networks, travel demand, and spatial conditions are combined to generate traffic simulations.

Unlike Rana, which is strongly process-driven, and Tygron, which is scenario-based through Measures, DatMobility focuses on constructing a consistent mobility model based on datasets describing infrastructure, travel behaviour, and spatial context.

Model construction approach
The construction of calculation models in DatMobility consists of the following core steps:

- Selection and preparation of mobility datasets

The model is based on datasets describing road networks, traffic demand, and spatial characteristics of the urban environment. These datasets are retrieved from OUP and aligned with the model's input structure.

The datasets typically include:

- road network definitions and topology;
- travel demand patterns (origins and destinations);
- contextual data influencing mobility, such as land use and urban density.

- Configuration of the transport model

Within the DatMobility platform, the selected datasets are used to configure a traffic model that represents the behaviour of the mobility system. This includes:

- defining the structure of the transport network;
- linking demand data to the network;
- configuring model parameters related to travel behaviour and routing.

This configuration step establishes the baseline model representing the current mobility situation.

- Integration of scenario inputs

The DatMobility model supports the evaluation of different scenarios by modifying input datasets or model parameters. These changes may include:

- adjustments in infrastructure (e.g. road closures or new connections);
- changes in demand or traffic patterns;
- impacts from external factors such as flooding or urban development.

In the DT-aCoGo context, such scenario inputs can originate from other models (e.g. flood results from Rana), enabling cross-domain analysis.

- Execution of traffic simulations

Once configured, the model executes simulations to calculate traffic flows across the network. The outputs include metrics such as:

- traffic intensity and congestion levels;
- travel times and accessibility;
- network performance under different conditions.

These outputs enable evaluation of how the mobility system responds to changes in the urban environment.

The DatMobility platform retrieves datasets from OUP through predefined connections to data services. Due to integration constraints, these connections are typically configured in advance, rather than dynamically selected at runtime.

This approach ensures that:

- the model consistently uses harmonised and validated datasets;
- updates in OUP data services can be propagated to the model without manual data import;
- the model remains aligned with the shared DT-aCoGo data environment.

Within the DT-aCoGo ecosystem, datasets include lifecycle attributes (ValidFrom, ValidTo), enabling time-aware interpretation of data. These temporal attributes ensure that the mobility model operates on data corresponding to the selected state of the digital twin.

Although the DatMobility integration is less dynamic than in platforms such as Rana, the time-filter logic ensures that:

- input datasets reflect the correct temporal scenario (e.g. current or future situation);
- traffic simulations are consistent with other analytical models and the visualisation environment;
- cross-domain scenarios (e.g. combining flooding and mobility impacts) can be evaluated coherently.

This enables DatMobility to participate in integrated digital twin workflows, where traffic simulations are aligned with spatial and environmental changes over time.

The DatMobility modelling approach is characterised by:

- data-driven construction of transport networks and demand models;
- scenario evaluation through modification of input datasets and parameters;
- integration with OUP-based data services;
- and alignment with lifecycle-based temporal filtering.

This results in a modelling environment capable of analysing mobility behaviour within a federated digital twin ecosystem, supporting both standalone traffic analysis and combined multi-domain scenarios.

Detailed technical instructions on data preparation, model configuration, and simulation workflows are provided in:

A4: Dat.Mobility; https://www.goudappel.nl/expertises/beleids-en-gedragsonderzoek/monitoring-en-evaluatie/datagedreven-mobiliteitsbeleid

4.2.3 Tygron – Heat stress model

The Tygron platform is used within the DT-aCoGo ecosystem to simulate heat stress and evaluate the impact of urban interventions. The platform operates as a scenario-based simulation environment, in which data, calculations, and future developments are integrated within a single project context.

Within Tygron, calculation models are configured using a combination of geospatial datasets, simulation overlays, and scenario-based "Measures". A project environment forms the basis for all modelling activities, allowing datasets, scenarios, and results to be managed in a structured and reproducible way.

Model construction approach
The construction of calculation models in Tygron consists of the following core steps:

- **Import and alignment of datasets**
  Datasets originating from OUP are imported into Tygron and aligned with the platform's data requirements. These datasets must be georeferenced and structured appropriately to support simulation overlays and scenario evaluation.

- **Configuration of simulation overlays**
  The analytical logic in Tygron is implemented through overlays. For heat stress, a Heat Stress Overlay is configured, which calculates indicators such as the Physiological Equivalent Temperature (PET) based on environmental parameters and spatial characteristics such as buildings, vegetation, and weather conditions.

- **Definition of Measures (urban scenarios)**
  A key mechanism within Tygron is the use of "Measures". Measures represent future urban development plans and consist of a set of spatial and functional changes to the environment, such as adding buildings, modifying land use, or introducing vegetation.

  Measures are defined as structured data objects that describe a future scenario. They can be created manually or imported from geospatial plan data (e.g. GeoJSON or DXF files), provided that these datasets contain clearly defined features and attributes.

  Once defined, a Measure remains stored within the project and can be activated or deactivated at any time, enabling consistent and repeatable evaluation of the same urban scenario.

- Activation of scenarios (Test Run mode)

When a Measure is activated, the Tygron platform enters a Test Run mode. In this mode:

- the original state of the urban environment is preserved as the “current” situation;
- a second representation, the “planned” or future state, is generated based on the defined Measure;
- all calculation models, including overlays, are recalculated to reflect the updated spatial configuration.

This mechanism enables direct comparison between the current and planned situation within the same project environment.

- Evaluation of results

The impact of the Measure is evaluated by comparing overlay results (e.g. heat stress maps) between the original and planned scenario. Because all calculations are rerun when a Measure is activated, changes in the urban environment are directly reflected in the analytical outputs.

Within the DT-aCoGo architecture, Tygron integrates with OUP through API-based data exchange. Datasets retrieved from OUP include lifecycle attributes (ValidFrom, ValidTo), enabling time-aware interpretation of data.

By applying the lifecycle-based filtering mechanism (as defined in D3.2), Tygron ensures that:

- only data valid for the selected point in time is used as model input;
- scenario-based Measures are evaluated against the correct temporal baseline;
- analytical results align with the selected state in the digital twin (e.g. current vs future scenarios).

This creates a direct alignment between the Tygron scenario mechanism (Measures) and the DT-aCoGo time slider, allowing users to analyse both planned interventions and temporal development within a consistent framework.

The Tygron modelling approach combines:

- harmonised datasets retrieved from OUP;
- overlay-based analytical calculations;
- scenario definition through Measures;
- and lifecycle-based temporal filtering.

This results in a flexible modelling environment in which urban interventions can be defined, activated, and evaluated in a controlled and reproducible way, while maintaining consistency with the broader digital twin ecosystem.

Detailed technical instructions on data requirements, Measures configuration, and API integration are provided in:
A5: Tygron; Wiki adding measures.docx
A6: Tygron; 20260624 DT-aCoGo Tygron Data descriptions.pdf
A7: Tygron; 20260624 DT-aCoGo Tygron Data to OUP Instructions.pdf
A8: Tygron; Expert mode 2 - Heat Stress Overlay tutorial - Tygron Support wiki.pdf

4.3 Conclusion

This chapter has demonstrated how analytical models within the DT-aCoGo ecosystem are constructed using harmonised datasets provided through the Open Urban Platform (OUP). While each platform follows its own modelling paradigm, a shared foundation is established through the Common Information Model (CIM), standardized data services, and lifecycle-based temporal filtering.

The three analytical models illustrate complementary approaches to model construction:

- Rana applies a process-based workflow, in which datasets are systematically transformed into hydrodynamic flood models through a sequence of automated steps.
- DatMobility follows a mostly manual data-driven approach, constructing transport models by combining network data, demand patterns, and external influences.
- Tygron operates as a scenario-based environment, where urban interventions are defined as Measures and evaluated through simulation overlays.

Despite these differences, all models rely on the same core principles:

- datasets are centrally harmonised and published via OUP;
- models retrieve their input data through standardised services;
- temporal consistency is ensured through lifecycle attributes (ValidFrom, ValidTo);
- analytical results are aligned with the selected state of the digital twin.

This approach confirms that a federated, data-driven ecosystem can support heterogeneous analytical models while maintaining consistency and interoperability across domains. By separating data, models, and visualisation environments, the DT-aCoGo architecture enables each platform to operate independently while still contributing to a coherent and integrated digital twin.

As a result, the ecosystem supports both domain-specific analysis and cross-domain workflows, such as the combined evaluation of flooding, traffic, and heat stress scenarios. The use of a shared data foundation ensures that model outputs remain comparable across cities and use cases, facilitating scalable deployment and reuse within the broader European data space for smart communities.

5. Registering model APIs in OUP

Following the construction and execution of calculation models within the analytical platforms, the next step in the DT-aCoGo workflow is the registration of model APIs within the Open Urban Platform (OUP). These APIs enable structured interaction between the digital twin environment and the analytical models, supporting data exchange, model execution, and retrieval of results.

Within the DT-aCoGo architecture, analytical models are not directly embedded within the OUP. Instead, they are exposed as external services through APIs, which are registered in OUP as datasets or services. This approach ensures a clear separation between data, analytical logic, and visualisation, while enabling controlled and standardised access to model functionality.

5.1 General approach to API registration

The registration of model APIs in OUP follows a consistent approach across all model providers:

- model APIs are exposed externally by the model provider;
- API endpoints are registered within OUP as accessible resources;
- authentication and access control are configured through the OUP hub;
- model results are stored and published back into OUP as datasets.

This approach allows analytical models to be invoked from external applications (e.g. Unreal Engine or orchestration layers), while ensuring that all interactions remain traceable and governed within the OUP environment.

Depending on the capabilities of the model provider, APIs may be registered as:

- public endpoints, where authentication is handled externally;
- limited endpoints, where OUP injects authentication credentials;
- or provider-specific endpoints, configured per hub or organisation.

5.2 Rana – API registration and usage

The Rana platform provides a structured API for interacting with flood models and executing simulations. Within the DT-aCoGo ecosystem, this API is registered in OUP as a service that enables:

- retrieval of available flood models;
- execution of rainfall simulations;
- access to simulation results in formats such as 3Dtiles, WMS or OGC services.

Authentication is handled through API keys and multi-tenant configuration, allowing each organisation to interact with its own model environment.

The API registration process includes:

- publishing the API documentation as a dataset in OUP;
- linking the API to an application entry within the hub;
- configuring authentication settings and access rights;
- enabling model execution through external calls.

Once registered, the Rana API allows external applications to trigger simulations and retrieve results, which are subsequently visualised or reused within the digital twin.

Detailed implementation guidance is provided in:
Appendix X: rana_dtacogo_sdk_light_v4.pdf
Appendix X: 20260629 Deliverable V - Data publishing and retrieval.pdf

5.3 DatMobility – API registration and integration

The DatMobility platform exposes its functionality through APIs that support traffic simulation and data exchange. Within the DT-aCoGo context, these APIs are registered in OUP as part of the provider-specific connector environment.

The registration process focuses on:

- linking predefined model endpoints to OUP datasets;
- configuring service access per hub;
- enabling consistent data retrieval for traffic simulations.

Due to differences in integration capabilities, DatMobility APIs are typically configured as stable, predefined connections rather than fully dynamic services. This ensures consistent model behaviour and alignment with the underlying data structure.

The APIs support:

- ingestion of mobility datasets;
- execution of traffic simulations;
- retrieval of mobility indicators such as congestion and travel times.

These services can be combined with outputs from other models (e.g. flood data from Rana) to support cross-domain analyses.

Further technical details are provided in:
Appendix X: SDK Light - Dat.mobility.docx
Appendix X: Dat.Mobility; Datagedreven mobiliteitsbeleid.docx

5.4 Tygron – API registration and proxy integration

The Tygron platform provides a comprehensive API that allows external systems to trigger simulations and retrieve results. Due to the complexity of this API, a proxy-based approach is implemented within DT-aCoGo.

In this setup:

- a Proxy API is deployed as an intermediary between OUP and the Tygron platform;
- external applications interact with the simplified Proxy API;
- the Proxy API handles communication with the Tygron platform and returns aggregated results.

The API registration process involves:

- registering the Proxy API endpoint in OUP;
- configuring authentication tokens per organisation;
- linking the API to dataset and application entries;
- enabling request–response workflows for simulation execution.

The API supports:

- triggering heat stress calculations;
- monitoring simulation progress;
- retrieving results via WMS or similar services.

This proxy-based approach simplifies integration while maintaining access to the full capabilities of the Tygron platform.

Detailed documentation is provided in:
Appendix X: Tygron; 20260624 DT-aCoGo Tygron WebApp Proxy API Description.pdf
Appendix X: Tygron; 20260624 DT-aCoGo Tygron Data to OUP Instructions.pdf

5.5 Role of OUP in API governance

The Open Urban Platform acts as the central governance layer for all registered APIs. It ensures that:

- access to model services is controlled through authentication and hub-level permissions;
- API endpoints are discoverable and reusable across applications;
- data exchange between models and users is standardised;
- results are stored and shared in a consistent manner.

By registering APIs within OUP, model providers can expose their functionality in a controlled and interoperable way, while maintaining ownership of their internal systems. In figure 2 an example of registered API's and their documentation is given.

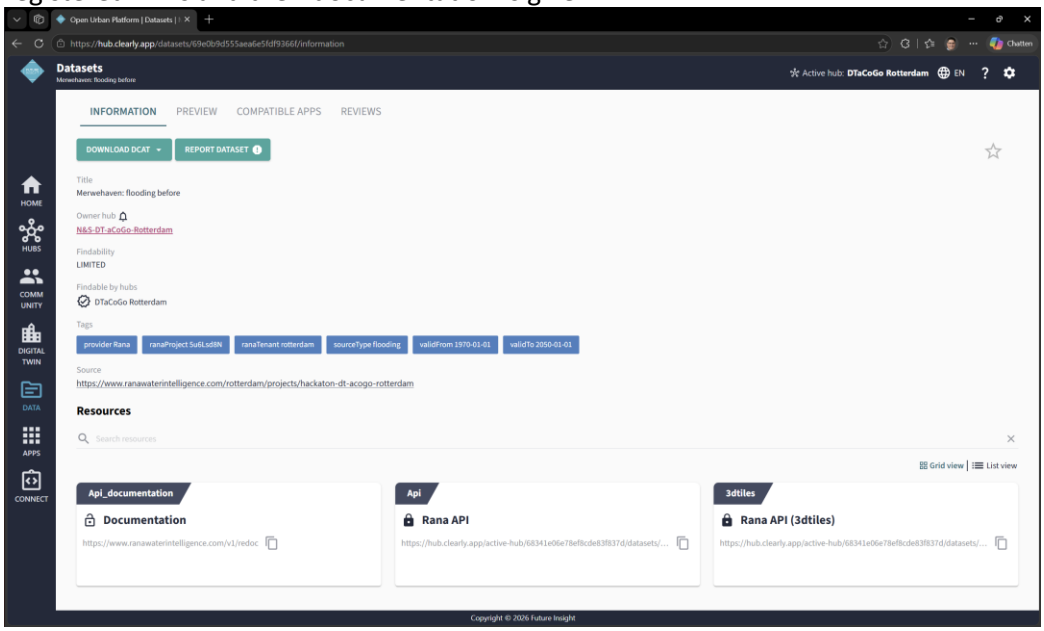


Figure 2: Registering API's in the OUP HUB

5.6 Summary

The registration of model APIs in OUP enables the integration of heterogeneous analytical platforms into a unified digital twin ecosystem. While each model provider follows its own technical approach:

- Rana uses a structured and fully integrated API;
- DatMobility relies on predefined service connections;
- Tygron applies a proxy-based API layer;

all approaches align with the same architectural principles:

- separation of data, models, and applications;
- standardised access through OUP;
- controlled authentication and governance;
- and reuse of model outputs within the digital twin.

This confirms that API-based integration is a scalable and flexible mechanism for connecting analytical models within a federated urban data space, supporting both standalone and cross-domain workflows.

6. Conclusions and next steps

6.1 Conclusions

This deliverable has demonstrated how a federated digital twin ecosystem can be realised through the integration of harmonised datasets, analytical models, and standardised data services within the Open Urban Platform (OUP).

The core objective of the DT-aCoGo approach—enabling interoperable, scalable, and reusable digital twins across cities—has been addressed through three key mechanisms:

- the harmonisation of heterogeneous datasets into a Common Information Model (CIM);
- the exposure of analytical models through standardised APIs;
- the implementation of lifecycle-based filtering to ensure temporal consistency.

Together, these elements establish a coherent workflow in which data is prepared centrally, processed by distributed analytical models, and visualised consistently across applications.

A key finding is that the separation between data, models, and applications is essential for scalability. By maintaining this separation:

- datasets remain authoritative and reusable;
- analytical models can operate independently of the data infrastructure;
- visualisation environments reflect results without introducing inconsistencies.

This architectural approach allows heterogeneous model platforms—such as Rana, DatMobility, and Tygron—to coexist within a single ecosystem while maintaining interoperability.

Another important conclusion is the effectiveness of lifecycle-based temporal filtering. By treating time as an intrinsic property of data, all applications—including analytical models and visualisation tools—can consistently interpret the state of the digital twin at any selected moment. This prevents discrepancies between analysis and visualisation and supports reliable decision-making.

The experience across the pilot cities confirms that the use of CIM and OUP significantly reduces integration complexity. However, the overall success of the approach remains strongly dependent on data quality and the correct definition of temporal scenarios.

6.2 Key challenges identified

Despite the successful implementation, several challenges have been identified:

- Variability in model integration capabilities
Not all model providers support the same level of API integration, resulting in differences between fully dynamic (Rana), partially integrated (Tygron), and more static (DatMobility) approaches.

- Data quality and availability
The reliability of model outputs is directly dependent on the completeness and accuracy of source datasets provided by the participating cities.

- Definition of temporal scenarios

While lifecycle attributes provide a robust technical solution, the governance process for defining and validating “points in time” requires further formalisation.

- **API management and authentication**
  Differences in authentication mechanisms and API configurations complicate the scalability and ease of use of model integrations across organisations.

These challenges highlight that, in addition to technical solutions, organisational alignment and governance processes are critical for the success of a shared digital twin ecosystem.

6.3 Next steps

Building upon the results of this deliverable, several next steps are identified to further develop and scale the DT-aCoGo ecosystem:

- **Standardisation of API integration**
  Further alignment is required to standardise how analytical models expose their functionality, reducing variability between providers and simplifying integration within OUP.

- **Improvement of data governance and quality assurance**
  Processes for validating, maintaining, and updating datasets need to be strengthened to ensure reliable model inputs across cities.

- **Formalisation of temporal scenario management**
  A structured governance approach should be established for defining lifecycle attributes and “points in time”, involving both domain experts and city stakeholders.

- **Enhancement of authentication and access mechanisms**
  Improvements to credential management within OUP would enable more flexible and scalable use of APIs, reducing the need for provider-specific configurations.

- **Expansion of cross-domain workflows**
  The integration of multiple analytical domains (e.g. flood, traffic, heat stress) should be further developed into reusable workflows and “recipes” that can be applied across cities.

- **Scaling to additional cities and use cases**
  The validated approach should be extended beyond the current pilot cities, demonstrating its applicability in different urban contexts and supporting wider adoption within the European data space.

6.4 Final reflection

The DT-aCoGo approach confirms that a digital twin can function as a shared urban infrastructure when built upon open standards, harmonised data, and modular analytical services. By combining a federated data architecture with distributed model execution, the ecosystem enables flexible, scalable, and collaborative use of digital twins across domains and cities.

The results demonstrate that interoperability is not achieved through a single platform, but through the alignment of data, models, and governance. This provides a strong foundation for future development and positions the DT-aCoGo framework as a replicable model for digital twin implementations within the broader DS4SSCC initiative.

7. Annexes

A1: RANA_DTACOGO_SDK_LIGHT_V4.PDF
A2: 20260629 DELIVERABLE I - GATHERING DATA AND SETTING UP MODEL PART 2.PDF
A3: 20260629 DELIVERABLE I - FLOWCHART FROM DATA TO FLOOD MODELS .PDF
A4: DAT.MOBILITY; HTTPS://WWW.GOUDAPPEL.NL/EXPERTISES/BELEIDS-EN-GEDRAGSONDERZOEK/MONITORING-EN-EVALUATIE/DATAGEDREVEN-MOBILITEITSBELEID
A5: TYGRON; WIKI ADDING MEASURES.DOCX
A6: TYGRON; 20260624 DT-ACOGO TYGRON DATA DESCRIPTIONS.PDF
A7: TYGRON; 20260624 DT-ACOGO TYGRON DATA TO OUP INSTRUCTIONS.PDF
A8: TYGRON; EXPERT MODE 2 - HEAT STRESS OVERLAY TUTORIAL - TYGRON SUPPORT WIKI.PDF

About the European Data Space for Smart Communities

The European Data Space for Smart Communities (DS4SSCC-DEP) initiative is a pivotal deployment following the preparatory action for a Data Space for Sustainable and Smart Cities and Communities (DS4SSCC). Emphasising sustainability aspects and diversity in communities, DS4SSCC developed a multi-stakeholder data governance scheme, created a blueprint for the European DS4SSCC, delivered priority datasets, and developed a roadmap towards a mature DS4SSCC.

DS4SSCC-DEP's vision revolves around creating a well-governed federated territorial, place-based data space for smart communities available for developers and infrastructure providers, aligning with the prospects outlined in Europe's Digital Decade objectives. This approach distinguishes itself from sectoral data spaces by encompassing diverse domains, and underpinning governance across all levels of society. The key objectives of the project are:

- **Objective 1: Establishing a Federated and Innovative Data Space**. This phase focuses on creating a large-scale data space controlled by public data holders, ensuring alignment with the Smart Middleware Platform and broader data space ecosystems. It also aims to offer middleware service solutions facilitating data sharing and management while refining the blueprint based on gained experience.
- **Objective 2: Pilots and Refinement of the Data Space Blueprint**. Through 10-12 cross-sectoral data pilots in various EU communities, this objective aims to validate and refine the data space blueprint. Using common high-value datasets, it seeks to create added value by combining data from various domains, contributing to the sustainability plan, and minimising infrastructure investment.
- **Objective 3: Fostering Innovation and Ecosystem Alignment**. Engaging stakeholders, compliance with sector-specific legislation, establishing links with Horizon Europe missions, creating services available via trusted application catalogues, and enhancing ethical considerations and AI-enabled local solutions constitute the focus of this objective.

DS4SSCC-DEP, building upon established European networks, aims to significantly contribute to the sustainability goals of European citizens. It emphasises co-creation, standardisation, business models, strategies for data spaces, and envisions a federated platform.

DS4SSCC-DEP consortium includes the following partners: