D7.5 Updated report on requirements and thematic modules functionalities for the environment domain

Status: UNDER EC REVIEW
Dissemination Level: public

Funded by the European Union

Abstract

Key Words	Earth Observation, Digital Twin, Environment, FAIR, open-source code libraries

interTwin Work Package 7 provides a set of reusable components, called thematic modules, for the interTwin Digital Twin Engine (DTE).

In this document we provide a summary of the current state of the environment thematic modules and of their requirements, the access points to the current versions of the source code, an overview of still pending tasks, and a summary of the requirements of these thematic modules.

This document represents an update of D7.1 report that has been submitted within the interTwin Work Package 7.

Document Description

Updated report on requirements and thematic modules functionalities for the environment domain

Work Package number 7

Document type	Deliverable		
Document status	UNDER EC REVIEW	**Version**	1.0
Dissemination Level	Public		
Copyright Status			
Lead Partner	TU Wien		
Document link	https://documents.egi.eu/document/4096		
DOI	https://doi.org/10.5281/zenodo.13365939		
Author(s)	• Matthias Schramm (TU Wien) • Clay Harrison (TU Wien) • Martin Schobben (TU Wien) • Donatello Elia (CMCC) • Emanuele Donno (CMCC) • Michele Claus (EURAC) • Iacopo Ferrario (EURAC) • Suriyah Dhinakaran (EURAC) • Björn Backeberg (Deltares) • Willem Tromp (Deltares) • Christian Pagé (CERFACS) • Anne Durif (CERFACS) • Mathieu Vrac (CNRS) • Massimiliano Fronza (UNITN) • Sandro Fiore (UNITN) • Andrea Manzi (EGI.eu) • Christoph Reimer (EODC)		
Reviewers	• Ivan Rodero (EGI) • Thomas Geenen (ECMWF)		
Moderated by:	• Matthias Schramm (TU Wien)		
Approved by	Andrea Manzi (EGI) on behalf of TCB		

Revision History

Version	Date	Description	Contributors
V0.1	20/06/2024	ToC, designing first layout, identifying thematic modules	Matthias Schramm (TU Wien), Clay Harrison (TU Wien)
V0.2	26/06/2024	Re-designing scope according to WP lead	Matthias Schramm (TU Wien), Clay Harrison (TU Wien)
V0.3	08/07/2024	Rewriting thematic modules, section 2.1.3	Donatello Elia (CMCC), Matthias Schramm (TU Wien), Emanuele Donno (CMCC), Mathieu Vrac (CNRS), Massimiliano Fronza (UNITN), Christian Pagé (CERFACS), Anne Durif (CERFACS)
V0.4	24/07/2024	Adding the state of all thematic modules and requirements	Donatello Elia (CMCC), Emanuele Donno (CMCC), Björn Backeberg (Deltares), Willem Tromp (Deltares), Michele Claus (EURAC), Iacopo Ferrario (EURAC), Suriyah Dhinakaran (EURAC), Matthias Schramm (TU Wien), Clay Harrison (TU Wien), Martin Schobben (TU Wien), Christian Pagé (CERFACS), Anne Durif (CERFACS), Mathieu Vrac (CNRS), Massimiliano Fronza (UNITN), Sandro Fiore (UNITN), Christoph Reimer (EODC), Andrea Manzi (EGI)
V0.5	26/07/2024	Aligning text modules for internal review	Matthias Schramm (TU Wien)
V0.6	05/08/2024	Internal review	Thomas Geenen (ECMWF), Ivan Rodero (EGI), Matthias Schramm (TU Wien)
V0.7	19/08/2024	Version approved by TCB	Andrea Manzi (EGI)
V1.0	27/08/2024	**Final**	

Terminology / Acronyms

Term/Acronym	Definition
AI	Artificial Intelligence
API	Application Programming Interface
BIC	Bayesian Information Criterion
CEMS	Copernicus Emergency Mapping Service
CMIP6	Coupled Model Intercomparison Project, Phase 6
CWL	Common Workflow Language
CNN	Convolution Neural Network
DoA	Description of Action
DNN	Deep Neural Network
DT	Digital Twin
DTE	Digital Twin Engine
FAIR	Findable, Accessible, Interoperable, and Reusable
GNN	Graph Neural Network
ML	Machine Learning
PLIA	Projected Local Incidence Angle
STAC	SpatioTemporal Asset Catalogs
VGG	Visual Geometry Group (a standard deep CNN)
WP	Work Package

Terminology / Acronyms: https://confluence.egi.eu/display/EGIG

Table of Contents

List of tables

List of figures

Executive summary

Work Package 7 (WP7) focuses on developing various thematic modules for the Digital Twin Engine, addressing environmental and physics-related topics. Several workflows have been outlined in the Description of Action (DoA) to be implemented as Digital Twins (DTs) within WP4. The project partners have identified the necessary features to be developed within specific thematic modules to support these workflows and their application as DTs. Preliminary descriptions of the required features for these thematic modules are provided in Deliverables D7.1 ("Report on requirements and thematic modules definition for the environment domain" **[R1]**) and D7.2 ("Report on requirements and thematic modules definition for the physics domain" **[R2]**).

This report updates the features of the thematic modules within the environmental domain as initially described in D7.1. It details the current state of these features and outlines remaining tasks, thus tracking the development process. Additionally, it describes input data requirements and processing capabilities for the thematic modules. The report covers these aspects for all thematic modules under the following interTwin tasks:

- T7.4: Climate Analytics and Data Processing
- T7.5: Earth Observation Modelling and Processing
- T7.6: Hydrological Model Data Processing

1 Introduction

1.1 Scope

This deliverable outlines the current status of thematic modules essential for implementing DTs within the interTwin environment domain, as described in the DoA. Its purpose is to update the information from Deliverable D7.1 [R1] and to provide the current state of the features described there for thematic modules in the following areas:

- Climate Analytics and Data Processing (interTwin task T7.4)
- Earth Observation Modelling and Processing (T7.5)
- Hydrological Model Data Processing (T7.6)

Since the publication of [R1], the structure of thematic modules in T7.4 has been revised, as detailed in Deliverable D7.3 ("First version of the thematic modules for the environment domain" [R3]). Therefore, the current descriptions of these thematic modules reference D7.3 instead of D7.1.

This document serves the following objectives:

- **Thematic modules**: Collecting all thematic modules for the environmental sector, as agreed upon by the responsible project partners throughout the project.
- **Feature Listing**: Listing all required features and functionalities for these thematic modules.
- **Status Update**: Describing their current state and outlining aspects still under development to provide project partners with a basis for monitoring the final development.
- **Requirement Update**: Updating the requirements for input data and processing capabilities for the thematic modules, as initially listed in D7.1. This will enable external users to recreate the necessary environment and allow project partners to align their requirements for joint DTs.

The document reflects the current state of the project; changes and adaptations throughout the project's funding period are expected.

1.2 Document structure

Due to this document's scope of updating the information delivered in D7.1, it also follows a similar document structure.

Section 2 is organised into three subsections, representing the three environmental areas and their associated tasks.

- T7.4: Climate Analytics and Data Processing
- T7.5: Earth Observation Modelling and Processing
- T7.6: Hydrological Model Data Processing

Each subsection begins with an updated overview of its area's overarching scope. Afterwards, the related thematic modules are described and their access interfaces are provided. A table provides the features for each thematic module (that have not already declared as being completed in D7.1), the current state of the thematic modules, and required next steps.

Section 3 characterises the requirements for running the thematic modules, increasing their reproducibility by allowing external users to rebuild a needed infrastructure environment. It covers requirements regarding the input data, storage needs and hardware and system preferences.

2 Progress in the design of thematic modules in the environment domain

2.1 T7.4 Climate analytics and data processing

2.1.1 Overview of progress regarding tasks for DTs for extreme events (T4.5, T4.6, T4.7)

The aim of this task is to develop and to integrate key components for the flexible design and implementation of DTs focused on extreme weather and climate events, as outlined in WP 4 (T4.5, T4.6, and T4.7). The primary focus is on supporting DTs for storms and wildfires that use Machine Learning techniques, as well as other climate extremes (see also **[R1]**).

Integration of the components from the thematic modules is being addressed by different DT applications at the level of WP4. Integration requirements with WP5 and WP6 are described in **section 3.2**.

In most cases, the related thematic modules will be containerized in order to enable the portability of workflows in the DTE infrastructure. It is also worth mentioning that in most cases data collection is not handled by the thematic modules, as these are quite static and dataset-specific. Common datasets required by multiple components, such as those from CMIP6 are being collected by using a specific software component (*esgpull*). A specific configuration for the set of needed data is being defined jointly with the DT applications developers. More details on data requirements are specified in **section 3.1**.

A comprehensive update on the progress of the planned features, as outlined in **[R3]**, is provided in **subsection 2.1.2**.

Clarification:
The set of functionalities within T7.4, as described in D7.1 **[R1]** has been reorganised into more specific modules, while integrating multiple capabilities from the initial design. Similarly, the initial set of thematic modules has been restructured to better fit evolving DTs requirements. Specifically, the set of capabilities originally identified in D7.1 is now provided across three distinct thematic modules that are tailored to application-specific tasks (storms, eddies - ocean currents, and fires). Additionally, new thematic modules have been introduced to include data-driven models for downscaling, as well as compound event analysis. These new functionalities are documented in D7.3 **[R3]**. The following sections refer to the current status and progress of the newly defined thematic modules.

2.1.2 Progress regarding thematic modules for climate analytics and processing

The feature state of following thematic modules is described in this section:

- ML TC detection: processing and analysis of tropical cyclones - related data and data driven models
- ML4Fires: Processing and analysis of wildfires - related data
- eddiesML: Processing oceanic mesoscale eddies - related data
- xtclim: Generic detection and characterization of climate extreme changes and impacts in the future climate projections
- downscaleML: Downscaling Climate Data
- CompEvPoEToE: Detection of time or periods of emergence for compound events

The first release of these thematic modules has been described in [R3]. All of their features that have not been reported as finalised in D7.1 are described in the following subsections.

ML TC detection

Processing and analysis of tropical cyclones-related data and data-driven models
Homepage: **https://www.intertwin.eu/article/thematic-module-ml-tc-detection**
GitHub repository: **https://github.com/CMCC-Foundation/ml-tropical-cyclones-detection/**

The thematic module ML TC detection supports the DT on the analysis of tropical cyclones on future projection data (interTwin task T4.5). Specifically, it provides a set of Python modules for supporting processing and analysis of TC-related data and data-driven models. The modules provide features for gathering and pre-processing data, training ML models and post-processing the results. Pre-processing features include the capabilities for splitting the input gridded data into non-overlapping patches. Furthermore, it provides functions for deterministic tracking and ML model ensemble. Multiple types of ML models are supported, in particular CNN and GNN.
The porting of the module from the first version in Tensorflow to PyTorch is almost completed. The final version will be fully written in PyTorch, allowing also the use of GNN with PyTorch Geometric. The multi-node and multi-GPU modes are available through the Fabric framework. The first version was released and described in [R3]. For the CNN-based approach, a set of data-driven models based on the VGG model were trained. An ML ensemble of VGG neural networks has also been implemented.

A multi-model thematic module will be also developed to support ensemble detection over multiple CMIP6 data.

Following is described the state and possible future steps of the planned features for this Thematic Module that have not already been declared as finalised in D7.1.

Table 1 – Features of the thematic module ML TC detection that have been described in D7.1 or that have been planned afterwards, their current status and planned next steps. Colour coded status: green: COMPLETED; yellow: IN PROGRESS; red: PENDING

Table 1 - Features of the thematic module ML TC detection

#	Planned feature	Status + next steps
1	Filtering of historical IBTrACS data	• Manual download of IBTrACS csv dataset (collection of IBTrACS data is not handled by the module since the size is quite low) • A Jupyter Notebook that filters the TC tracks data (time period and geographical domain of interest) based on the TC detection setup Next steps: • The Jupyter Notebook will be converted to a Python script • Steps for the manual download will be described in the module documentation
2	Gathering of ERA5 tropical cyclones fields based on IBTrACS	• Download ERA5 tropical cyclones' predictors fields (as NetCDF files) associated to filtered historical IBTrACS data on the selected subregion • The feature is fully implemented and available in the package's "*library*" submodule.
3	Input data selection on user-defined conditions and queries.	• Users can specify input variables from a configuration file used for the model's training Next steps: • Develop a Jupyter Notebook that allows the user to select input data and a trained ML model during inference. The Notebook automatically performs a prediction over the given dataset and provides both TC tracks.
4	Pre-processing: subsetting & patches generation on gridded data for TC-detection applications	• NetCDF files are processed by extracting patches (each associated with a TC center) and saved to annual Zarr archives related to training and validation. During the inference phase, NetCDF files are read, tiled in patches, and fed to the model. • The feature is fully implemented and available in the package's "*library*" submodule.
5	Conversion between 40x40 gridded patches and PyTorch Geometric graphs	• Conversion from 40x40 patches to PyTorch Geometric graphs implemented in the package's "*library*" submodule.

		• Conversion from PyTorch Geometric inference graphs to 40x40 coordinates in progress Next steps: • Systematically convert the inference graphs to the 40x40 gridded coordinates and save them to CSV files ready for the post-processing
6	Data augmentation (rotation and flipping of images) and feature scaling (MinMax or Standard normalisation)	• Implemented in the package's *"library"* submodule.
7	Configurable data-driven models (DNN): usage of configuration files for a more organised workflow	• Configuration file in .toml format describes paths to input and output directories, run-specific arguments, data configuration, and model/training hyperparameters. Next steps: • Integrate GNN approach in the same configuration file
8	Hyperparameter tuning script and notebook for results visualisation in the GNN use-case	• Configurations are created from a .toml file and training+validation is performed on each permutation of hyperparameters in a grid search fashion, producing a series of metrics that can be visualised in the notebook Next steps: • Code refinement
9	Post-processing functions: geo-referencing storm data to reconstruct the original maps	• ML model's results are georeferenced with the corresponding lat-lon coordinates and the detection results are saved in csv format • Detections can be visualised on a geographic map Next steps: • Convert the inferenced PyTorch graphs to 40x40 coordinates and save them to CSV files • Make sure the link between graph-derived CSV files and the post-processing functions work
10	Multi-model climate inference	• Several models have been trained • Initial version available in older Tensorflow implementation Next steps:

		• Porting the ensemble class from Tensorflow to PyTorch • Support ensemble detection over multiple CMIP6 data Make sure the link between graph-derived CSV files and the post-processing functions work
11	Deterministic tracking of TC centres	• Deterministic tracking function implemented to link detections from ML models • Initial version available in older Tensorflow implementation
		Next steps: • Integrate deterministic tracker into the current PyTorch version

ML4Fires

Processing and analysis of wildfires - related data

Homepage: **https://www.intertwin.eu/article/thematic-module-ml4fires**
Github repository: **https://github.com/CMCC-Foundation/ML4Fires**

The ML4Fires thematic module is designed to support the generation of wildfire danger maps, including burned areas maps also on future climate projections. This module provides a comprehensive suite of Python tools aimed at facilitating the processing and analysis of wildfire-related data. Specifically, a Machine Learning pipeline has been developed for running the components of the thematic module. The capabilities include subsetting SeasfireCube **[R4]** data in Training, Validation and Testing, after a preliminary selection of drivers and target variables. The ML model used for this version is a standard UNet++ network **[R5]**; users can customise the network depth. The module includes model training and data visualisation utilities. The ML4Fires thematic module's primary value proposition is to enhance wildfire analysis and prediction by providing tools allowing users to pre-process data, choose model architecture and train the model, post-process results (basic visualisation). The module has been fully ported from the initial version described in deliverable D7.3 from Tensorflow to PyTorch.

Details on the features are found in Table 2.

Table 2 – Features of the thematic module ML4Fires that have been described in D7.1 or that have been planned afterwards, their current status and planned next steps. Colour coded status: green: COMPLETED; yellow: IN PROGRESS; red: PENDING

Table 2 - Features of the thematic module ML4Fires

#	Planned feature	Status + next steps
1	Selection and filtering of input data according to user-defined metadata	• Filtering on variables and temporal extent possible via configuration files (toml format) as well as via interface to xarray Dataset object when scripting. • Zenodo DOI for SeasFire Cube v0.3 provided in module documentation, user should download it and place it in the "data" folder. • Automatic download is not needed since the process is quite smooth. Nevertheless, a command for download is provided in documentation <u>Next steps:</u> • Add functionalities for selecting projection scenarios (i.e., in the case of climate projection data)
2	Preprocessing and creation of training/inference dataset	• Input files are processed (e.g., subsetting, upsampling) to create dataset for model training/inference and saved to Zarr archives. <u>Next steps:</u> • Integrate features for reading, preprocessing (e.g., remapping) and feeding, during the inference phase, NetCDF files (e.g., CMIP) to the model.
3	Data augmentation and feature scaling	• Functionalities are implemented in the package's "*augmentation*" and "*_scalers*" submodules.
4	Configurable data-driven models (DNN)	• DNNs can be configured per experiment via configuration files in .toml format (U-Net++ model is currently supported) <u>Next steps:</u> • Implemented support for additional ML architectures (e.g, DNN/GNN) • Apply final models to climate projection data with different CMIP6 scenarios
5	Post-processing and visualisation functions	• Jupyter notebook that implements multiple functionalities to visualise predictions on test set has been developed <u>Next steps:</u> • Extended visualisation functions for climate projections (CMIP6 data)

		• Add functionalities to compare real data with predictions on test set
6	Multi-model climate inference	• First design of pipeline defined Next steps: • Support prediction over an ensemble of CMIP6 models

eddiesML

Processing and analysis of eddies-related data and data-driven models

Homepage: **https://www.intertwin.eu/article/thematic-module-eddiesgnn**
Github repository: **https://github.com/HPCI-Lab/eddiesML**

The eddies detection thematic module supports the DT on the analysis of oceanic mesoscale eddies. Specifically, it provides a set of Python modules for supporting processing and analysis of eddy-related unstructured data from the FESOM2 model. The core software is provided by AWI as part of a collaboration with an external institution willing to leverage the software solutions developed in interTwin to develop DT applications. The modules provide features for pre-processing data, training, and testing DL models. Pre-processing features include both interpolation operations between unstructured and structured grids, and the generation of segmentation masks that are used as the ground truth during both training and testing.

A U-Net was chosen as the CNN model to be used both in training and testing, and the DL framework is Tensorflow.

Details on the features are found in Table 3.

Table 3 – Features of the thematic module eddiesML that have been described in D7.1 or that have been planned afterwards, their current status and planned next steps. Colour coded status: green: COMPLETED; yellow: IN PROGRESS; red: PENDING

Table 3 - Features of the thematic module eddiesML

#	Planned feature	Status + next steps
1	Interpolation of SSH data from FESOM2 from unstructured to gridded, both for train and test sets	• Data exists, interpolation scripts are hosted on a public repository Next steps: • Download unstructured FESOM2 data from AWI and prepare the interpolation scripts
2		• Code is already written and hosted on a public repository

	Generation of ground truth segmentation masks via py-eddy-tracker	Next steps: • Prepare the environment with py-eddy-tracker and tweak the scripts to produce the ground truth masks
3	Training of a U-Net model with Tensorflow	• Base code is already written and hosted on a public repository
		Next steps: • Setting up the hyperparameters and the configuration file • Start training and fix possible issues with the computation • Possibly, moving from Tensorflow to PyTorch
4	Testing of the U-Net model to produce the segmentation masks with the eddies	• Base code is already written and hosted on a public repository
		Next steps: • Setting up the hyperparameters and the configuration file • Take the trained model and produce the testing results

xtclim

Generic detection and characterization of climate extreme changes and impacts in the future climate projections

Homepage: **https://www.intertwin.eu/article/thematic-module-xtclim**
Github repository: **https://github.com/cerfacs-globc/xtclim**

The *xtclim* module supports the DT on the climate change impacts of climate extremes. It implements a generic method that can detect anomalies in atmospheric fields. This method is based on a Deep Learning technique called a Convolutional Variational AutoEncoder. In theory it can be applied to any gridded field, but it has to be configured and trained properly. It could even use the output of the *downscaleML* module to have higher spatial resolution data as input. In the context of interTwin the module implementation is targeting CMIP6 data over a region of 32x32.

Following are described the state and possible future steps of the planned features for this Thematic Module that have not already been declared as finalised in D7.1.

Table 4 – Features of the thematic module xtclim that have been described in D7.1 or that have been planned afterwards, their current status and planned next steps. Colour coded status: green: COMPLETED; yellow: IN PROGRESS; red: PENDING

Table 4 - Features of the thematic module xtclim

#	Planned feature	Status + next steps
1	Retrieval of global climate model gridded data (temperature, precipitation, winds, etc.) for the historical and future climate	• Completed for data stored locally, skeleton for rucio based data access. Still waiting for interTwin data lake access to be completed (search API using STAC). Next steps: • Implement data access with search API using STAC • Implement user configurable data selection • Multi-model support
2	Pre-processing: subsetting & standardisation of input data	• Standardisation of input data done • Subsetting still hard-coded Next steps: • Further update to have better user configuration options • Geographical user defined region configuration • General configurations Implement data access with search API using STAC
3	An AI-based model to characterise the change compared to the historical period, in several independent climate models.	• Done for temperature and 32x32 grid. • Exploration to extend it (to precipitation and winds) and to make it more configurable for end users. Next steps: • General configurations • Tests with precipitation and winds • Extract spatial information
4	Combining results of this Thematic Module for an uncertainty assessment for the expected changes.	• Multi-scenarios statistics currently designed, still to be implemented. Next steps: • Multi-scenarios data analysis
5	Post-processing and visualisation functions	• Done for individual model output Next steps: • Extending it to spatial information, and uncertainty information

downscaleML

Downscaling Climate Data

Homepage: **https://www.intertwin.eu/article/thematic-module-downscaleml**
Github repository: **https://github.com/interTwin-eu/downScaleML**

This Python package is designed to streamline the process of climate data downscaling using machine learning techniques. It offers an automated workflow tailored for downscaling seasonal forecast climate variables, specifically temperature and precipitation, with a particular emphasis on addressing climate extremes.

Table 5 – Features of the thematic module downscaleML that have been described in D7.1 or that have been planned afterwards, their current status and planned next steps. Colour coded status: green: COMPLETED; yellow: IN PROGRESS; red: PENDING

Table 5 - Features of the thematic module downscaleML

#	Planned feature	Status + next steps
1	Selection of input variables for the corresponding target variables, pre-processing chain, stacking of independent variables against dependent variables - specific to 2m-temperature and precipitation	• Users can modify the script */scripts/run_model.sh* to control important parameters and model output. • Brief document available Next steps: • Improve the documentation
2	Downscaling framework based on ensemble learning method	• Implemented in python package.
3	Application of the downscaling framework to ensemble SEAS5 seasonal forecast - for 2m-temperature and precipitation	• Implemented in python package.
4	Experimentation and feasibility testing on downscaling CMIP6 using downscaleML	• Depends on to be defined requirements from project partners that will be formulated after this report's submission Next steps: • Investigation on the scientific validity of the downscaling approach

		• Final test with the DT providers, after requirements have been defined by the project internal users
5	Linking input and output to STAC	• A similar task will be realised within the thematic module *raster2stac* (see Table 7). Once that is finished, it will be adapted to also be usable here.
		Next steps: • Containerizing the entire package environment using Docker • Establishing a streamlined pipeline for managing input and output data through openEO processes (see also Table 8)

CompEvPoEToE

Detection of time or periods of emergence for compound events

Homepage: **https://www.intertwin.eu/article/thematic-module-compevpoetoe**
Github repository: **https://gitlab.in2p3.fr/ipsl/espri/espri-mod/intertwin**

CompEvPoEToE analyses if "periods of emergence" (PoE) and/or "times of emergence" (ToE) of compound events occur in the bivariate climate time series used as inputs. Specifically, it provides an R package for fitting the most appropriate joint (2d) distribution based on separate modelling of the margins (i.e., univariate distributions) and copula functions (allowing to characterise the dependence between the variables). The full time period (e.g., 75 years) is cut into smaller ones (e.g., 20 years) and the probability of the compound event of interest (default is the couple of the two 90% percentiles of the variables) is computed for each sub-period and compared to the first period (reference). This package thus allows investigating/detecting if climate change is visible in terms of compound events. It is specifically designed to work on "heavy rainfall & strong wind" compound events but other types of (bivariate) events could be considered.

Table 6 – Features of the thematic module CompEvPoEToE that have been described in D7.1 or that have been planned afterwards, their current status and planned next steps. Colour-coded status: green: COMPLETED; yellow: IN PROGRESS; red: PENDING

Table 6 - Features of the thematic module CompEvPoEToE

#	Planned feature	Status + next steps
1	Selection and download of variables depending on compound event definition	• Application to storms, implying the use of daily precipitation and wind intensity (reanalyses or climate simulations)

		• The code will not download nor prepare any data, this should be handled by the DTE prior to the module run. Next steps: • Providing documentation on data requirements (e.g., filetype, required content and structure, naming conventions)
2	For each time period, fit of a set of parametric univariate statistical distributions for precipitation and wind intensity and selection of the most appropriate one according to BIC (Bayesian Information Criterion)	• Includes a large set of marginal distribution functions for precipitation and wind intensity. • Various model selection criteria for the marginals are calculated and provided as outputs • The (default) selection criterion is based on the BIC Next steps: • Providing relevant documentation
3	For each time period, fit of a set of copula functions to link precipitation and wind intensity and selection of the most appropriate copula function according to BIC	• Fit of a large set of Archimedean copula functions • Various model selection criteria for the copulas are calculated and provided as outputs • Selection of the copula model based on BIC (default) Next steps: • Providing relevant documentation
4	Monitoring the change of compound event probability.	• Detection of Periods of Emergence and Time of Emergence of strong “rainfall & wind” compound event probability • Computing the contributions of the univariate and dependence evolutions to change of compound event probability for each time period Next steps: • Providing relevant documentation

2.2 T7.5 Earth Observation modelling and processing

This subsection covers the progress and adaptations that have been made by the project consortium since the finalisation of D7.1 [R1] for developing thematic modules to run DTs based on EO data, with openEO [R6] as the driving technology.

2.2.1 Overview of the progress

Task 7.5 entails the development of several thematic modules that are necessary to run within T4.6 Digital Twins based on EO data, with openEO as the driving technology. The modules cover different aspects of developing flood and drought monitoring workflows, making use of raster data with the openEO API (see general architecture in Figure 1).

Clarification:
The workflows are presented here as a C4 Model - a set of hierarchical diagrams of different grades of detail. The 4 different levels are called 'Context', 'Container', 'Component', and 'Code'. The phrase container is not to be confused with Docker containers.

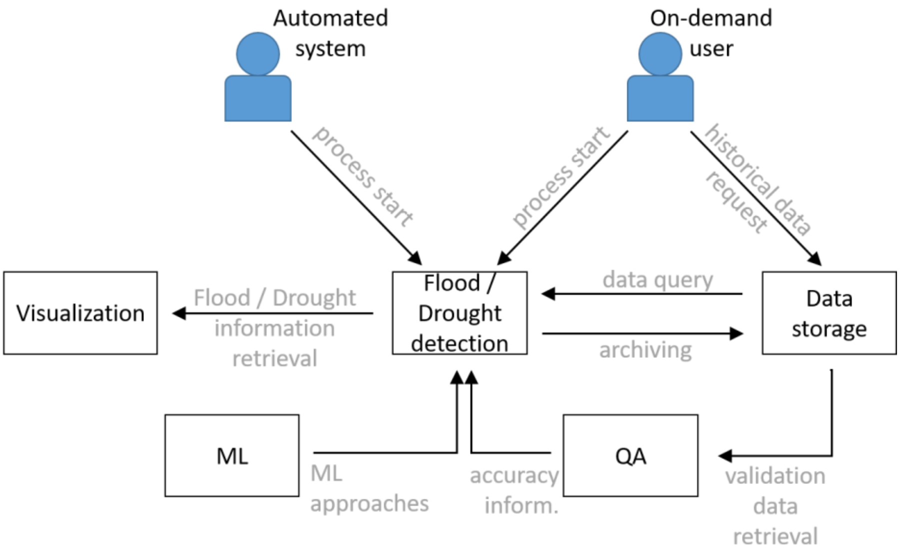

Figure 1 - Context of the planned flood and drought detection workflows that will be used in digital twins for Early Warning of Extreme Events (T4.6)

The Python library *openeo-flood-mapper-local* has been developed for detecting flood events from Sentinel-1 data, using the openEO syntax. It is publicly available under https://github.com/interTwin-eu/openeo-flood-mapper-local. Figure 2**Error! Reference source not found.** shows the different steps of the flood detecting workflow; the container Flood detection refers to the thematic module open-flood-mapper-local. The output will be used in the 'Flood early warning' DT; in Figure 4 it represents one of several Flood maps. If suitable, several stretched goals may be included into the workflow to increase the systems dynamic.

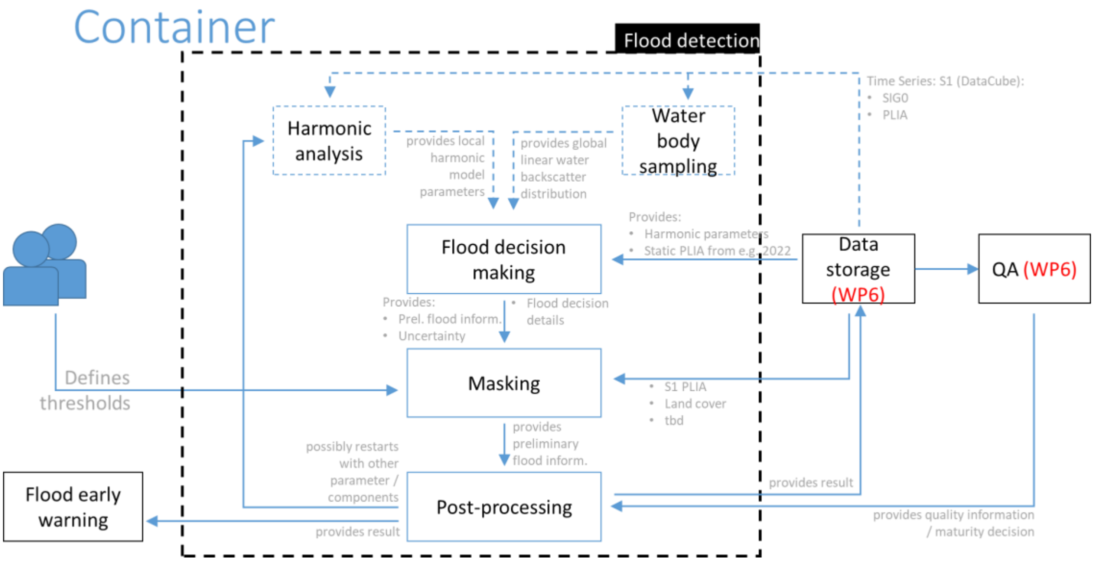

Figure 2 - Container of the detailed flood detection workflow that will be used within the DT for Early Warning of flood events (T4.6)

A drought prediction/forecasting tool for the Alpine region is currently under development; its structure is planned as shown in Figure 3.

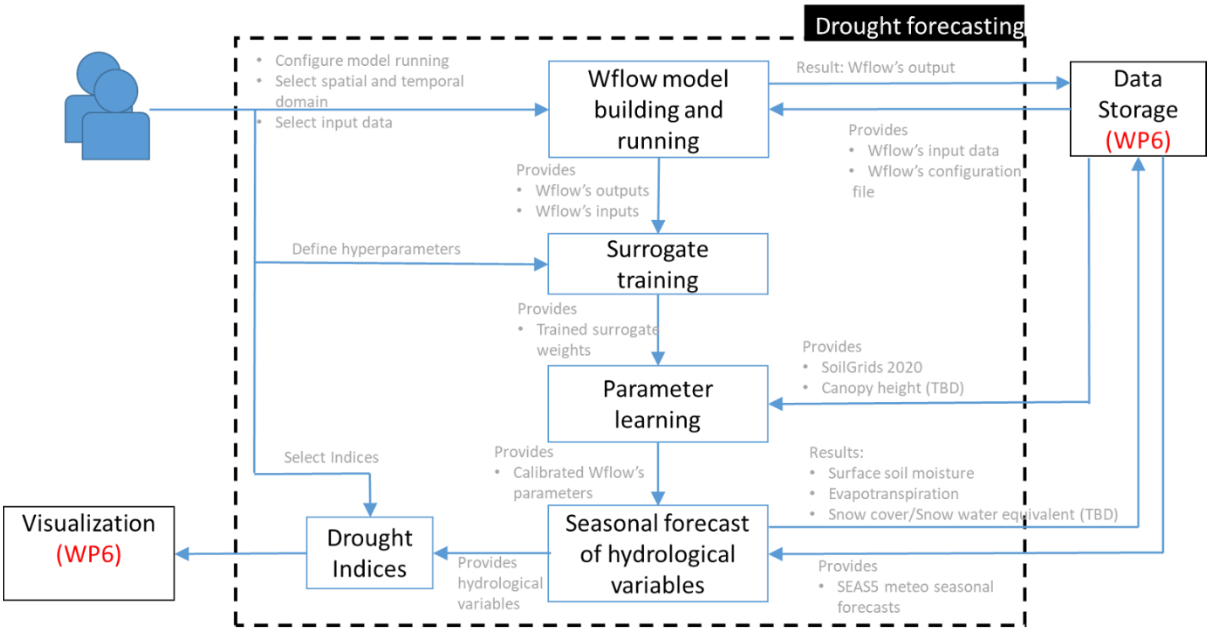

Figure 3 - Container of the detailed drought forecasting workflow that will be used within the DT for Early Warning of droughts (T4.6)

Details on the planned progress of the related thematic modules can be found in the following subsection.

2.2.2 Progress regarding thematic modules

The feature state of the following thematic modules is described in this section:

- raster2stac: Index raster and vector data in openEO
- openeo-processes-dask: Running openEO process graph on backend

Features for these thematic modules that have not been reported as finalised in D7.1 are described in the following subsections.

raster2stac

Creating STAC compliant metadata for raster data

Homepage: **https://www.intertwin.eu/article/thematic-module-raster-to-stac**
Github repository: **https://gitlab.inf.unibz.it/earth_observation_public/raster-to-stac**
PyPi for latest release: https://pypi.org/project/raster2stac/

This component allows the creation of STAC Collection [R7] with Items and Assets starting from different kinds of raster datasets. It also allows the user to automatically upload the resulting files to an Amazon S3 Bucket, to make them publicly accessible and reachable worldwide. The goal is to make a dataset easily accessible, interoperable, and shareable.

Table 7 – Features of the thematic module raster2stac that have been described in D7.1 or that have been planned afterwards, their current status and planned next steps. Colour coded status: green: COMPLETED; yellow: IN PROGRESS; red: PENDING

Table 7 – Features of the thematic module raster2stac

#	Planned feature	Status + next steps
1	Data to be used in the openEO environment shall follow STAC specifications	• The flood monitoring workflow has included the possibility to query for STAC compliant metadata. • Inclusion of vector data is still pending.
		Next steps: • To discuss with project partners, whether scope of this thematic module shall be broadened to also include vector data.
2	Developing a module that creates required STAC metadata (JSON text files) for any raster data to be processed in the openEO environment, if not already available.	• The module allows the creation of STAC Collections with items and assets starting from different kinds of raster datasets. • It follows the STAC best practices [R8] to ensure the best interoperability.
		Next steps: • Add support ZARR dataset • Add support for climate data

3	Enabling users to automatically upload the resulting files to an Amazon S3 Bucket, to make them publicly accessible and reachable worldwide.	• This feature is implemented and ready to be used.

openeo-processes-dask

Running openEO process graph

Homepage: **https://www.intertwin.eu/article/thematic-module-openeo-processes-dask**
Github repository: **https://github.com/Open-EO/openeo-processes-dask**

This component allows users to remotely access the Sentinel-1 σ_0, PLIA and HPAR datasets and run the TU Wien flood monitoring workflow using the Xarray/Dask framework on any backend. This solution is implemented at the data storage provider and processing platform (EODC). Non-expert users can use openEO clients for Python, R, and JS, or browser-based solutions, to construct and deploy EO data analysis workflows at EODC. It is envisioned that expert users might benefit from a more refined control over the processing chain. Highly granular control can be achieved by writing solutions in Xarray/Dask syntax and offloading of processing graphs to a Dask cluster (e.g., at EODC). Henceforward it is planned to develop notebooks that showcase a pure Dask-based approach of flood monitoring.

Table 8 – Features of the thematic module openeo-processes-dask that have been described in D7.1 or that have been planned afterwards, their current status and planned next steps. Colour coded status: green: COMPLETED; yellow: IN PROGRESS; red: PENDING

Table 8 – Features of the thematic module openeo-processes-dask

#	Planned feature	Status + next steps
	Data pre-processing	
1	Sentinel-1 microwave data is provided at a cloud storage (interTwin partner) Analysis-Ready-Data (ARD): it is gridded, formatted and stored into a data cube.	• For ARD σ_0 backscatter data that is currently used to develop DTs, a STAC metadata catalogue has been created at EODC and has been filled. • openEO compatible flood monitoring workflows have been adapted to use this data / metadata as input.

2	Value added output data from openEO processing lines can be stored and made accessible to project partners	• Output data is stored at EODC • Data is accessible by project partners via Rucio <u>Next steps:</u> • Connection to the central datalake will be finalised
3	The pre-processed data can be queried, accessed, and processed from different storage platforms / at processing service providers in a comparable way.	• Can be used at EODC or locally with public STAC catalogues • Feature is developed in '*openeo-pg-parser-networkx*' and '*openeo-processes-dask*' <u>Next steps:</u> • Merge the load_stac improvements in the main repository
	Data Processing	
4	The preprocessed Sentinel-1 data can be used in a first step to compute backscatter information, defining several parameters (as e.g., the use of specific exclusion masks or of a specific digital elevation model)	• The backscatter information can be calculated via the openEO syntax, using the process '*sar_backscatter*' <u>Next steps:</u> • As an optional stretched goal, the choice between σ_0 and γ_0^T **[R9]** backscatter is possible. This will require the processing of suitable γ_0^T backscatter information for to-be-defined use cases that can serve as proof of concept.
5	Static information is provided on known exclusion masks	• Data is available at the EODC and is used for the flood and drought monitoring workflows. • Data has been manually stored at the cloud provider. • The stretched goal of developing a dynamic exclusion mask, using the openEO syntax has been **CANCELLED** due to limited usability / expected increase of its impact.
	Drought and flood monitoring workflows	
6	Drought monitoring workflow in the openEO environment.	• This feature has been **ADAPTED**: ○ A Sentinel-1 based drought product, developed within the openEO environment is not anymore planned to be developed due to its limited use for the follow up workflows. ○ Instead, a surface soil moisture product (RT0) has been developed with an optimised exclusion mask over the Alpine area. This product is used in

		follow-up drought monitoring workflows.
		Next steps: • Perform the parameter learning task of the surrogate model (WP4.6) using the surface soil moisture product (RT0) produced by TU Wien • Develop openEO/OGC API for training the surrogate model • Develop openEO/OGC API for parameter learning • Develop openEO/OGC API for running surrogate inference
7	Flood monitoring workflow in the openEO environment.	• The workflow is completed and published in GitHub • Discussions are ongoing to adapt its results to the requirements of input data for Thematic Module 'FloodAdapt'. • Jupyter notebooks for joint DT (see WP4) on this topic are under development.
		Next steps: • Adapting workflow results to requirements for input data in Flood impact DT • Finalising Jupyter notebooks for DT (see WP4 and Thematic Module 'FloodAdapt') • Extending development of flood monitoring workflow to make use of DASK

2.3 T7.6 Hydrological model data processing

This subsection covers the progress and adaptations that have been made by the project consortium since the finalisation of D7.1 for developing the necessary components to facilitate near-automatic setting up of local flood hazard and impact models anywhere on Earth.

2.3.1 Overview regarding the planned features' progress

This task involves developing thematic modules that allow the establishment of DTs for flood early warning systems and climate change impact in coastal and inland regions. The goal is to enable a DT Application, within WP4, to demonstrate that the architecture, tooling, and capabilities of the Digital Twin Engine (DTE; see WP5 and WP6) can support the implementation of DTs across various domains. Existing components are adapted for inter-compatibility, allowing for the creation of workflows with diverse input datasets (see Figure 4 and Figure 5).

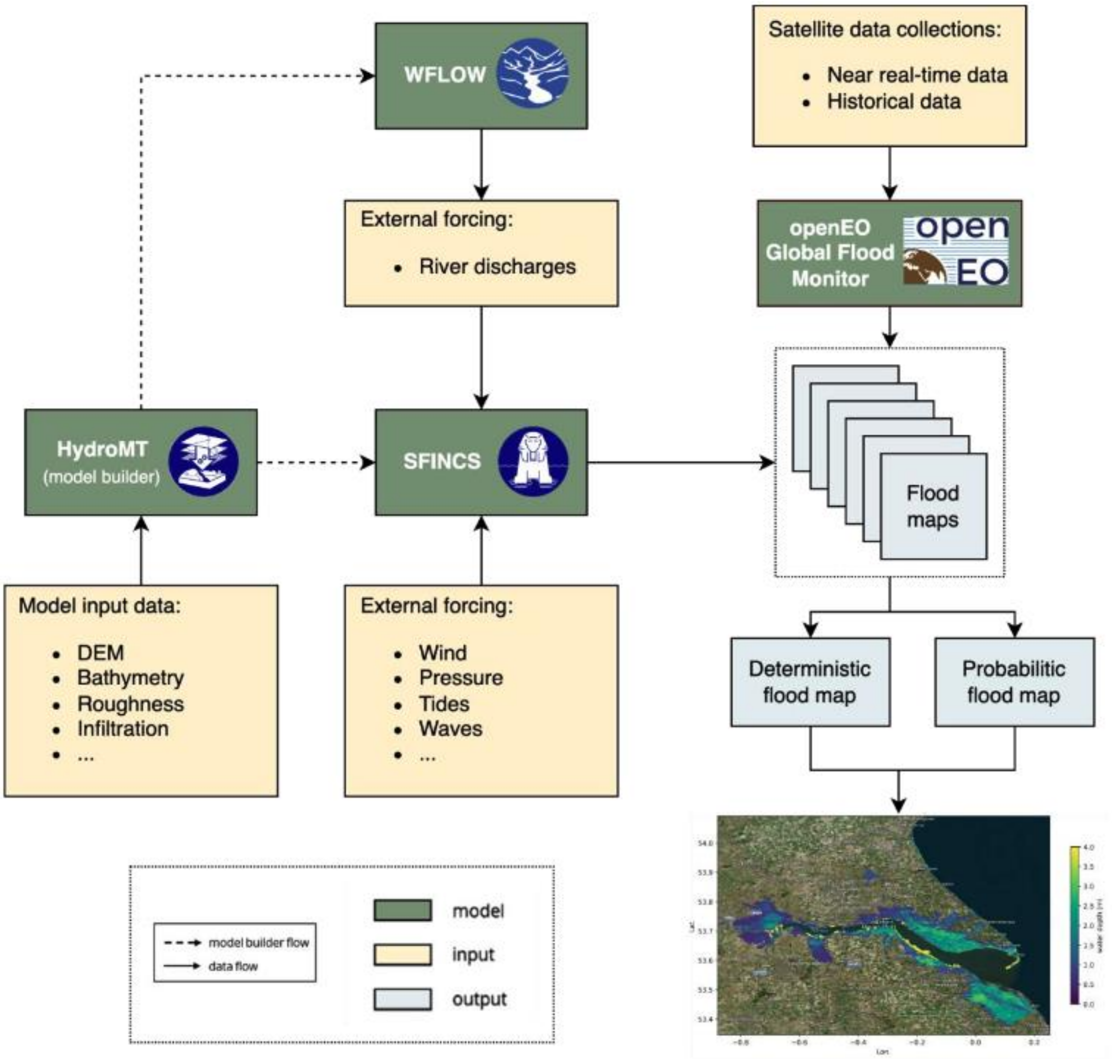

Figure 4 - High-level overview of the data and components of the Digital Twin for Flood early warning in coastal and inland regions

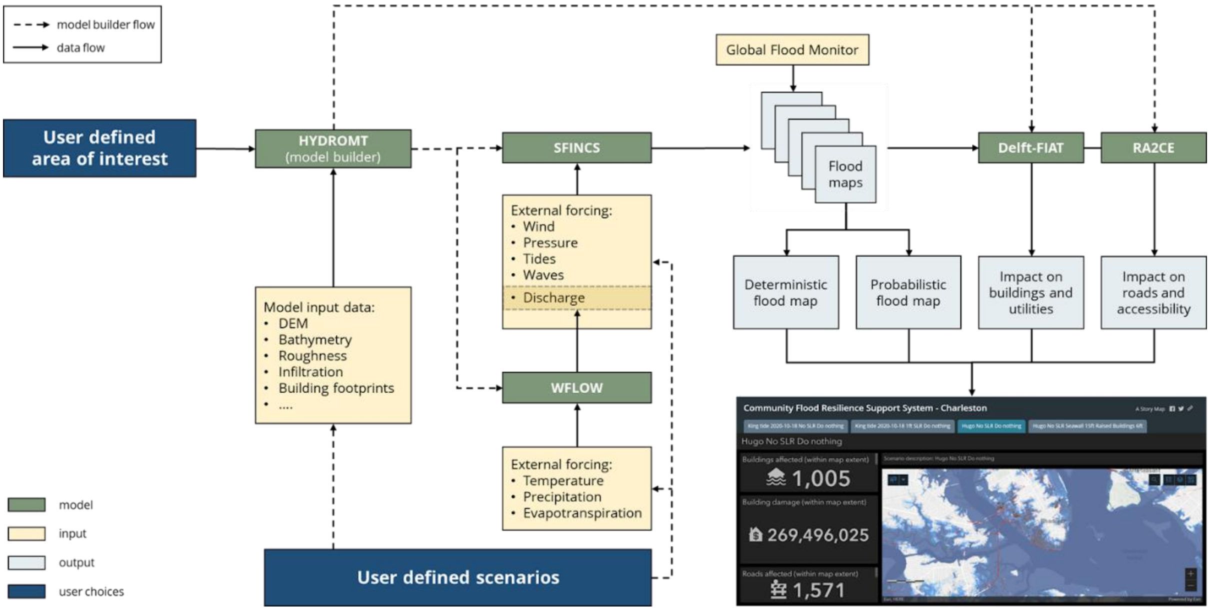

Figure 5 - High-level overview of the data and components of the Digital Twin for Flood climate change impact in coastal and inland regions

Once an area of interest and a few model-specific parameters are defined, the user triggers a workflow that will automatically set up the necessary models for the flood early warning or flood climate impact DTs. The flood early warning and flood climate impact DTs have similar but slightly different workflows and capabilities (see D4.3 [R10]). The flood early warning DT will focus on the generation of flood risk maps that can be used by early warning systems to trigger alerts when a flood is predicted, while the flood climate impact DT will focus on simulating (compound) flood events by producing flood hazard and risk maps and assessing their impact on building, utilities, roads, and accessibility.

The main interface to both DTs is a Jupyter Notebook, where users can define areas of interest, model specific parameters and scenarios to simulate. Once these have been defined, Common Workflow Language (CWL) workflows are triggered that automatically modify and update the models and execute these to provide the DT results. Subsequent post-processing steps will facilitate visualisation of the results and users can further analyse these in Jupyter Notebooks, which will also be provided within this task. The Notebooks will enable users to define mitigation measures or what-if scenarios and to rerun simulations. For example, users can access the impact of constructing flood protection in specific areas or analyse the impact of a theoretical scenario where twice the amount of rainfall occurs.

Clarification:
This task includes several thematic modules that represent amongst others open-source python packages and are to be used subsequently for processing the hydrological model data. These packages have been developed both independently from and within the interTwin project; the thematic modules have been adapted to the project's requirements as described in D7.1.

- FloodAdapt (https://www.intertwin.eu/article/thematic-module-floodadapt)
- WFLOW (to be published as interTwin thematic module)
- SFINCS (https://www.intertwin.eu/article/thematic-module-sfincs)
- Delft-FIAT (https://www.intertwin.eu/article/thematic-module-delft-fiat)
- RA2CE (to be published as interTwin thematic module)
- HydroMT-Core
    - HydroMT-WFLOW
    - HydroMT-SFINCS (https://www.intertwin.eu/article/thematic-module-hydromt-sfincs)
    - HydroMT-FIAT (https://www.intertwin.eu/article/thematic-module-hydromt-fiat)

These thematic modules have been described extensively in D7.1; this document focuses on describing the progress of work for these thematic modules towards publishing the second version of the thematic modules for the environment domain (D7.7). This document focuses on elaborating on the status of the capabilities of DTs (described in D4.3).

2.3.2 Progress regarding models and data

DT flood early warning and climate impact

Hydrological model data processing

The thematic modules associated with these DTs are meant for expert users to set up DTs for either flood early warning or climate impact to support decision makers and other end-users to make use of flood and flood impact modelling. Note that Deltares is in parallel to the interTwin project expanding the capabilities of FloodAdapt, SFINCS, WFLOW, Delft-FIAT, RA2CE and HydroMT and making these developments available to the interTwin thematic modules, these developments are described in rows 5-9 of Table 9.

Table 9 – Features of the thematic module on DT flood early warning and climate impact that have been described in D7.1 or that have been planned afterwards, their current status and planned next steps. Colour coded status: green: COMPLETED; yellow: IN PROGRESS; red: PENDING

Table 9 – Features of the thematic module on DT flood

#	Planned feature	Status + next steps
1	Jupyter Notebook interface that allows a Digital Twin developer / implementer to specify an area of interest,	• Jupyter Notebooks have been developed that allow users to define areas of interest, model parameters and specify scenarios to simulate.

	define model parameters, select scenarios to simulate and visualise results.	• Notebooks to set up SFINCS, Delft-FIAT and WFLOW as well as Notebooks to set up sites, configure scenarios and visualise results have been published at https://github.com/interTwin-eu/DT-flood <u>Next steps:</u> • Add a Notebook to set up RA2CE. • Extend the Notebook that visualises results to include easy-to-understand metrics of the simulated scenario. • Integrate Notebook with OSCAR to facilitate automatic, file availability based, triggering of CWL workflows to execute the models to produce the scenario results • Write simulation results to Rucio-based interTwin datalake in processing/visualisation-ready formats such as NetCDF, CSV, XLSX, and GeoPackage.

2	HydroMT functionality to support automatic model building and updating	• All HydroMT plugins complete except RA2CE. RA2CE has opted not to make use of HydroMT, they rely on their own bespoke functionality (reported in row 8 of this table)
		• HydroMT plugins complete for automatic creation of model schematisations for:
		  ○ WFLOW (hydrological model),
		  ○ SFINCS (Super-Fast INundation of CoastS), except for SFINCS Quadtree grids. Quadtree grids allow for variable resolution grids within a single model, facilitating enhanced resolution where needed while maintaining computational efficiency.
		  ○ Delft-FIAT (Fast Impact Assessment Tool)
		
		<u>Next steps:</u>
		• The current hydromt-sfincs branch is causing issues with building subgrid models (Quadtree). Switching branches results in errors, and the workaround of not using subgrid models is not ideal. Significant progress has been made on the quadtree branch, but it requires more time for proper documentation, testing, and full support, which will be the focus of the work before the next thematic release in January 2025.

3	WFLOW and SFINCS produce flood maps, which will be augmented with flood maps from Copernicus Emergency Mapping Service (CEMS) (T7.5).	• WFLOW and SFINCS capabilities are in place, as well as EO based flood mapping workflow • A use case to pilot the augmentation has been defined, namely Storm Babet in Northern Europe on 20th October 2023 [R11]. • A local high-resolution DEM has been sourced for the region. • It is not yet clear if WFLOW is needed for this event. Next steps: • Augmentation of WFLOW/SFINCS maps with the latter data for the defined use case. • Determine if WFLOW is needed. • Set up SFINCS model for the area of interest. • WP5 to provide resources (JupyterHub, datalake access) • Explore out how to offload SFINCS simulations via interlink.
4	Create deterministic and probabilistic flood maps in a post-processing step to serve as input data for Delft-FIAT and RA2CE, determining the impact of flood events on buildings, utilities, roads, and accessibility	• Deterministic flood maps completed. • A Jupyter Notebook is being developed that provides a recipe for creating probabilistic flood maps. • Pre-processing steps for Delft-FIAT are in place Next steps: • A small script to convert netcdf output from SFINCS to a geotiff for RA2CE is needed for the flood maps to be used in RA2CE.
5	New developments to SFINCS since D7.1 made available to interTwin	• SFINCS is published open-source at **https://github.com/Deltares/SFINCS** • Green-infrastructure adaptation measures are implemented into HydroMT-SFINCS with some first working examples. • Conversion scripts for converting local data to the correct format are available. Their use will be demonstrated in the SFINCS - CEMS augmentation activity. • Validation of flood maps will be developed during the SFINCS - CEMS augmentation activity. Early developments of comparing water level predictions with observations can be found at **https://github.com/Deltares-research/cht_observations** and

		http://cosmos.deltares.nl/nopp_event_viewer/index.html
		Next steps: • Consolidate above capabilities in the DT-flood interTwin repository: https://github.com/interTwin-eu/DT-flood
6	New developments to Delft-FIAT since D7.1 made available to interTwin	• Delft-FIAT is published open-source at https://github.com/Deltares/Delft-FIAT • No major updates to report on
		Next steps: • Develop Docker container for Delft-FIAT • Provide example of using local data
7	New developments to WFLOW since D7.1 made available to interTwin	• WFLOW is published open-source at https://github.com/Deltares/Wflow.jl • A link with the leaf area index and land classes has been created. When changing land use or cover, the necessary parameters are automatically updated using *hydromt-wflow*. In terms of flooding, we can now test the impact of land cover changes on flooding
		Next steps: • Develop capabilities for re-infiltration of surface water • Support the development of STAC for HydroMT by reviewing, testing and documenting developments by T7.5.
8	New developments to RA2CE since D7.1 made available to interTwin	• RA2CE is published open-source at https://github.com/Deltares/ra2ce • Users can set up RA2CE anywhere on Earth by defining a bounding box. A Notebook exists that demonstrates capabilities. • Notebook examples have been developed that demonstrate how to use SFINCS output in RA2CE. • A Docker container is available. • Initial ideas for visualising RA2CE results are available
		Next steps: • A conversion script is required to convert SFINCS netCDF output to RA2CE GeoTIFF. • Need to develop CWL workflow for RA2CE.

		• Need to finalise RA2CE visualisation examples. • All of the above RA2CE capabilities need to be exemplified in Notebooks in the interTwin DT-flood repository: **https://github.com/interTwin-eu/DT-flood**
9	New developments to FloodAdapt since D7.1 made available to interTwin	• FloodAdapt is published open-source at **https://github.com/Deltares-research/FloodAdapt** • Ongoing efforts to make the hazard module in FloodAdapt more model agnostic. This will facilitate using different flood maps in FloodAdapt. • Ongoing efforts to restructure the FloodAdapt API to align with the needs of the interTwin workflows.
		Next steps: • Develop a FloodAdapt adapter for WFLOW • Develop a FloodAdapt adapter for RA2CE

3 Requirements for the thematic modules in the environment domain

3.1 Input data requirements

The Thematic Modules developed in this work package depend on various input datasets. Those datasets should be available to the Modules at runtime. In Table 10 , the required datasets for the Thematic Modules in the environment domain are listed, as they are needed in the project execution. Also, their current storage locations and methods of access are described here; this information refers to the current project state with possibly only requiring data for well-defined, local use cases. At the current state, not all aspects regarding storage location and access mechanisms have been finally decided; future adaptations are possible.

Table 10 – Currently used input data for running or testing thematic modules

Dataset	Related tasks	Storage / Access	Size	Present at storage?	Access mechanism	Responsible party / Data steward
CMIP6	T7.4, T7.6	Rucio DataLake	O(10) TB	In progress	Download via esgpull software component (IPSL) to Rucio DataLake planned; Access via STAC under discussion	CMCC, CNRS, Deltares, UNITN, CERFACS
IBTrACS	T7.4, T7.6	Project external repositories	O(100) MB	In progress	Download from provider on demand	CMCC, UNITN
SeasFire Cube	T7.4	Project external repositories	O(10) GB	In progress	Download from provider on demand	CMCC
ERA5	T7.4, T7.6	Local platforms	O(100) GB	In progress	Download from provider on demand	CMCC, EURAC, UNITN, Deltares
FESOM2	T7.4	Local platforms	O(10) GB	In progress	Download from provider on demand (to AWI)	UNITN
Sentinel-1 σ_0 data from 2016 onwards	T7.5	EODC	O(500) TB	Yes	openEO, STAC	TU Wien, EODC
Harmonic parameters of Sentinel-1 σ_0 data	T7.5	EODC	O(5) TB	Yes	openEO, STAC	TU Wien
PLIA of Sentinel-1 σ_0 data from 2020	T7.5	EODC	O(1) TB	Yes	openEO, STAC	TU Wien
Surface Soil Moisture from H-SAF service (ECMWF)	T7.5	Local platforms	O(10) GB	Yes	Download from provider on demand	TU Wien, EURAC

HydroLAKES	T7.4, T7.5, T7.6	Local platforms	O(2) GB	In progress	EURAC: Currently S3-bucket, shall be made available to one of the WP5 storage providers Deltares: Download from provider on demand	Deltares, EURAC
GRanD	T.74, T7.5, T7.6	Local platforms	O(100) MB	No	Download from provider on demand	EURAC
GlobCover	T7.6	Local platforms	O(5) GB	No	Download from provider on demand	Deltares
SoilGrids	T7.6	Local platforms	O(1) TB (used O(1) GB)	No	EURAC: Currently EURAC S3-bucket, Deltares:Download from provider on demand	Deltares, EURAC
Randolph Glacier Inventory	T7.6	Local platforms	O(500) GB	No	EURAC: Currently EURAC S3-bucket,	EURAC
MOD15A2	T7.6	Local platforms	O(4) TB	No	Download from provider on demand	Deltares
MCD15A3Hv061	T7.6	Local platform	O(500) MB	No	Currently S3-bucket, shall be made available to one of the WP5 storage providers	EURAC
SEAS5	T7.6	Local platforms	O(20) TB	No	Download from provider on demand	EURAC
MERIT Hydro	T7.4, T7.6	Local platform	O(1) TB	No	Deltares: Download from provider on demand; EURAC: Currently EURAC S3-bucket	Deltares, EURAC
Copernicus DEM	T7.5, T7.6	Local platforms	O(5) TB	No	Download from provider on demand	Deltares, TUW
GEBCO	T7.6	Local platforms	O(5) GB	No	Download from provider on demand	Deltares
GCN250	T7.6	Local platforms	O(2) GB	No	Download from provider on demand	Deltares
ESA worldcover	T7.6	Local platforms	O(250) GB	No	Download from provider on demand	Deltares
GTSM water levels	T7.6	Local platforms	O(8) TB	No	Download from provider on demand	Deltares
OpenStreetMap (OSM)	T7.6	Local platforms	O(60) TB	No	Download from provider on demand	Deltares

WorldPop	T7.6	Local platforms	O(35) GB	No	Download from providers on demand	Deltares
JRC Depth Damage Curves	T7.6	Local platforms	O(100) MB	No	Download from provider on demand	Deltares
E-OBS	T7.6	Local platforms	O(500) GB	No	Currently EURAC S3-bucket	EURAC
CORINE Land Cover (CLC2018)	T7.6	Local platforms	O(1) TB	No	Currently EURAC S3-bucket	EURAC

3.2 DTE Infrastructure and Core components integration requirements

DTE Infrastructure and Core Modules designed and developed in WPs 5 and 6 respectively, can be used to handle allocation of computing resources, composition of Thematic Modules into workflows, and data management.

3.2.1 DTE Infrastructure components

As documented in deliverable D5.2 [R12], a first version of the DTE infrastructure components have been released. In particular, components from WP7 are interfacing directly towards data management components such as Rucio.

- **Rucio**
  - DT-flood Notebooks require read and write access to the datalake because it will write simulation results to Rucio-based interTwin datalake in processing/visualisation-ready formats such as NetCDF, CSV, XLSX, and GeoPackage.
  - DT applications from T4.5 (tropical cyclone detection and wildfire danger prediction) and T4.7 (weather extremes) will need access to the Rucio-based datalake for reading future climate projection datasets (CMIP6) in NetCDF format. In case of support for STAC integration, queries to the STAC catalog can simplify the access to the CMIP data;
  - openEO data loading requires metadata to be available as a STAC collection. Data access happens via provided URLs within a STAC asset. Integrating Rucio URLs into STAC for the use with openEO requires a software layer to automatically pull data out of the Rucio data lake to the given storage location where the openEO processing backend is running.

3.2.2 DTE Core components

As documented in deliverable D6.2 [R13], first versions of the DTE Core Components have been released. In particular the components from WP7 are integrating (or planning to integrate) with the following WP6 components:

- **Itwinai, common ML model**
  - Thematic modules, such as the TC ML detection and ML4Fires, are being integrated with itwinai and require a platform supporting well-known frameworks like PyTorch and distributed training based on PyTorch Lightning
- **Software quality as a service**
  - Thematic modules from T7.4 (either the TC ML detection or ML4Fires) will be integrated with the SQAaaS core module for performing a quality validation of the results and workflows.
- **OSCAR for event-based workflow triggering**

- The DT-Flood is working on integrating its Notebook with OSCAR to facilitate automatic file-availability based triggering of CWL workflows to execute the models for producing the scenario results.
- DT-Flood would benefit from CWL workflow composition and execution and infrastructure to offload heavy computations

3.3 Computing Requirements

In order to guarantee a working software environment on most machines, Thematic Modules will be containerized according to the OGC Application Package [R14] standard along with all necessary software dependencies. This enables portable and distributed execution of workflows on resources that are available to a particular user of a DT. However, some Thematic Modules have specific hardware and computing requirements that should be considered when deciding which resources to use.

The following computing requirements have been collected by the institutions developing the thematic modules:

ML TC Detection model training requirements

- Jupyter Notebook environment
- GPU: 2 NVidia A100 (40GB each)
- RAM: 128GB
- Storage: 500GB

Enes Dataspace configuration for GNN pipeline users

- CPU: 8 cores
- GPU: 1 NVIDIA A40
- RAM: 128GB
- Storage: 500GB

ML4Fires model training requirements

- Jupyter Notebook environment
- GPU: 4 NVIDIA A100 (40GB each)
- RAM: 128GB
- Storage: 200GB

CompEvPoEToE requirements

- R environment

DT-Drought requirements

- CPU: 24 vCPUs
- GPU: 4-8 NVIDIA, 20 GB each
- RAM: 64 GB
- Storage: 250 GB

EODC openEO configuration per user to run a job

- Adaptive scaling of worker from minimum 1 to maximum 6
- CPU configuration per worker: 8 vCPUs
- RAM configuration per worker: 12 GB RAM

DT-Extreme requirements

The requirements for our demonstrator are as follows. It must be noted that the estimations of CPU, GPU and memory is a very rough estimate at this time.

- Jupyter Notebook Environment: We are developing our demonstrator using Jupyter Notebook. Therefore, we need access to JupyterHub.
- Containerization: The virtual machine should have Docker installed, as we will use Docker for containerization.
- Data Access: We require access to data lake (access using STAC and Rucio)
- CPU: 12-24 vCPUs with a clock speed of 3.0 GHz or higher
- GPU: 4-8 NVIDIA, 20 GB each
- RAM: 128-256 GB
- Input data: Access to STAC and datalake
- Storage: Between 500 GB and 1 TB

3.4 Output storage requirements

Table 11 – Planned output data and its storage requirements

Dataset	Related task	Storage location	Size	Access mechanism	Responsibly party / Data steward
HydroMT's outputs	T7.6	N.A.	O(10) GB	Currently S3-bucket	EURAC
Wflow's outputs	T7.6	N.A.	O(50) GB	Currently S3-bucket	EURAC
DT-Flood scenarios	T7.6	N.A.	O(50) GB	Rucio datalake	Deltares
TUW flood retrieval	T7.5	EODC Storage service	O(100) GB	Rucio datalake (EODC RSE)	TU Wien, EODC
xtclim	T7.4	Accessible by the Jupyter Notebooks	O(1) GB	Posix CSV output of the DT	CERFACS

4 Conclusion

The requirements and functionalities of the thematic modules for the environment domain have been further clarified, developed and aligned with the workflows of WPs 4, 5, and 6. As these tasks are ongoing, the characteristics described here represent a snapshot in time.

The various thematic modules of Tasks T7.4, T7.5, and T7.6 have been further harmonised, and initial use cases towards DTEs (see WP4) have been successfully tested or are under development. The thematic modules are publicly available; and their requirements for data and computing infrastructure are well-defined to ensure their reproducibility. However, most of the module features identified in D7.1 as future work are still in progress. Since these tasks are expected to span the entire project's funding period, the publicly available modules will undergo further changes in upcoming versions.

5 References

Reference	
No	**Description / Link**
R1	**Report on requirements and thematic modules definition for the environment domain.** M. Claus et al. interTwin Deliverable D7.1. DOI: 10.5281/zenodo.8036991
R2	**Report on requirements and thematic modules definition for the physics domain.** K. Tsolaki et al. interTwin Deliverable D7.2. DOI: 10.5281/zenodo.8036997
R3	**First version of the thematic modules for the environment domain.** S. Fiore et al. interTwin Deliverable D7.3. DOI: 10.5281/zenodo.10224252
R4	**SeasFire Cube: A Global Dataset for Seasonal Fire Modeling in the Earth System.** L. Alonso et al. DOI: 10.5281/zenodo.8055879
R5	**UNet++: A Nested U-Net Architecture for Medical Image Segmentation.** Zhou et al. DOI: 10.48550/arXiv.1807.10165
R6	**The openEO API – Harmonising the Use of Earth Observation Cloud Services Using Virtual Data Cube Functionalities.** M. Schramm et al. In Remote Sensing, vol 13(6). DOI: 10.3390/rs13061125
R7	**SpatioTemporal Asset Catalogs.** https://stacspec.org/en/; Accessed 26/07/2024
R8	**STAC Best Practices.** https://github.com/radiantearth/stac-spec/blob/master/best-practices.md; Accessed 26/07/2024
R9	**Flattening Gamma: Radiometric Terrain Correction for SAR Imagery.** D. Small. In IEEE Transactions on Geoscience and Remote Sensing, vol 49(8). DOI: 10.1109/TGRS.2011.2120616
R10	**First version of the DTs capabilities for climate change and impact decision support tools.** C. Pagé et al. interTwin Deliverable D4.3. DOI: 10.5281/zenodo.11389801
R11	**Widespread flooding and destruction as Storm Babet hits Northern Europe | Euronews.** https://www.euronews.com/2023/10/20/widespread-flooding-and-destruction-as-storm-babet-hits-northern-europe; Accessed 26/07/2024
R12	**First DTE Infrastructure Software Release.** D. Spiga et al. interTwin Deliverable D5.2. DOI: 10.5281/zenodo.10224150
R13	**First release of the DTE core modules.** I. Campos et al. interTwin Deliverable D6.2. DOI: 10.5281/zenodo.10224213
R14	**OGC Best Practice for Earth Observation Application Package – v1.0.** P. Gonçalves (ed). Open Geospatial Consortium. Identifier: http://www.opengis.net/doc/BP/eoap/1.0