

interTwin

D7.7 Final version of the thematic modules for the environment domain

Status: UNDER EC REVIEW
Dissemination Level: public



Funded by the
European Union

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


Abstract

Key Words

Digital Twins, Thematic modules, environment domain, development, integration, software release

This report describes the status of the development of the final version of thematic modules identified in the project to support the development of Digital Twins in the environmental domain.



Document Description			
D7.7 Final version of the thematic module for the environment domain			
Work Package number 7			
Document type	Deliverable		
Document status	Under EC Review	Version	1.0
Dissemination Level	Public		
Copyright Status	 <p>This material by Parties of the interTwin Consortium is licensed under a Creative Commons Attribution 4.0 International License.</p>		
Lead Partner	CMCC		
Document link	https://documents.egi.eu/document/4177		
DOI	https://zenodo.org/records/14918025		
Author(s)	<ul style="list-style-type: none"> • Donatello Elia (CMCC) • Davide Donno (CMCC) • Emanuele Donno (CMCC) • Sandro Fiore (UNITN) • Massimiliano Fronza (UNITN) • Bjorn Backeberg (DELTA RES) • Willem Tromp (DELTA RES) • Alexander Jacob (EURAC) • Michele Claus (EURAC) • Iacopo Ferrario (EURAC) • Christian Pagè (CERFACS) • Atef Ben Asser (IPSL) • Mathieu Vrac (IPSL) 		
Reviewers	<ul style="list-style-type: none"> • Estíbaliz Parceros Iglesias (UPV) • Kostas Koumantaros (GRNET) 		
Moderated by:	<ul style="list-style-type: none"> • Donatello Elia (CMCC) • Charis Chatzikyriakou (EODC) • Andrea Anzanello (EGI) 		
Approved by	AMB		

Revision History			
Version	Date	Description	Contributors
V0.1	04/12/2024	Created template	Charis Chatzikyriakou (EODC)
V0.2	09/12/2024	ToC revised	Donatello Elia (CMCC)
V0.3	16/12/2024	Update common and T7.4 sections	Donatello Elia (CMCC)
V0.4	05/02/2025	All sections updated	All contributors
V0.5	24/02/2025	Internal revision provided and comments addressed	Reviewers & all contributors
V1.0		Final	

Terminology / Acronyms	
Term/Acronym	Definition
API	Application Programming Interface
CMIP	Coupled Model Intercomparison Project
CNN	Convolutional Neural Network
COG	Cloud Optimized GeoTIFF
CSV	Comma Separated Value
CVAE	Convolutional Variational Auto-Encoder
DestinE	Destination Earth
DID	Data Identifier
DT	Digital Twin
DTE	Digital Twin Engine
ECMWF	European Centre for Medium-Range Weather Forecasts
EO	Earth Observation
ESGF	Earth System Grid Federation
FESOM	Finite-Element/volumE Sea ice-Ocean Model
FIAT	Fast Impact Assessment Tool
GNN	Graph Neural Network
GPU	Graphics Processing Unit
IBTrACS	International Best Track Archive for Climate Stewardship
icclim	Index Calculation for CLIMate
JSON	JavaScript Object Notation
ML	Machine Learning
SFINCS	Super-Fast INundation of CoastS
STAC	SpatioTemporal Asset Catalog
TC	Tropical Cyclone
VGG	Visual Geometry Group

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



Table of Contents

1	Introduction	9
1.1	Scope	9
1.2	Document Structure	9
2	Overview of Thematic Modules for the environmental domain	10
2.1	T7.4: Climate analytics and data processing	10
2.2	T7.5: Earth Observation Modelling and Processing	11
2.3	T7.6: Hydrological model data processing	11
2.4	Summary of supported Digital Twins	12
3	Thematic Modules	14
3.1	ML TC detection	14
3.1.1	Release notes	15
3.1.2	Future plans	15
3.2	ML4Fires	16
3.2.1	Release notes	16
3.2.2	Future plans	17
3.3	eddiesML	17
3.3.1	Release notes	18
3.3.2	Future plans	18
3.4	Xtclim	19
3.4.1	Release notes	19
3.4.2	Future plans	20
3.5	downscaleML: Downscaling Climate Data	20
3.5.1	Release notes	21
3.5.2	Future plans	21
3.6	emergence.compound	22
3.6.1	Release notes	23
3.6.2	Future plans	23
3.7	Esgpull_rucio	23
3.7.1	Release notes	24
3.7.2	Future plans	24
3.8	openeo-processes-dask	24
3.8.1	Release notes	25
3.8.2	Future plans	25
3.9	openeo-pg-parser-networkx	25
3.9.1	Release notes	26
3.9.2	Future plans	26
3.10	raster-to-stac	26
3.10.1	Release notes	27



D7.7 Final version of the thematic module for the environment domain

3.10.2	Future plans	27
3.11	FloodAdapt	27
3.11.1	Release notes	28
3.11.2	Future plans	29
3.12	HydroMT-SFINCS	29
3.12.1	Release notes	30
3.12.2	Future plans	30
3.13	HydroMT-FIAT.....	30
3.13.1	Release notes	31
3.13.2	Future plans	32
3.14	SFINCS	32
3.14.1	Release notes	33
3.14.2	Future plans	33
3.15	Delft-FIAT	33
3.15.1	Release notes	35
3.15.2	Future plans	35
3.16	WFLOW.jl.....	35
3.16.1	Release notes	37
3.16.2	Future plans	37
3.17	HydroMT-WFLOW.....	37
3.17.1	Release notes	39
3.17.2	Future plans	39
3.18	RA2CE	39
3.18.1	Release notes	40
3.18.2	Future plans	40
3.19	Hython Wflow_SBM Hydrological Model.....	40
3.19.1	Release notes	42
3.19.2	Future Work.....	42
4	Conclusions	44
5	References	46

List of Tables

Table 1: Summary of supported Digital Twins.....	12
Table 2: ML TC detection	14
Table 3: ML4Fires	16
Table 4: eddiesML	17
Table 5: Xtclim	19
Table 6: downscaleML: Downscaling Climate Data	20
Table 7: emergence.compound	22
Table 8: Esgpull_rucio	23
Table 9: openeo-processes-dask	24
Table 10: openeo-pg-parser-networkx.....	25



D7.7 Final version of the thematic module for the environment domain

Table 11: raster-to-stac.....	26
Table 12: FloodAdapt.....	27
Table 13: HydroMT-SFINCS.....	29
Table 14: HydroMT-FIAT	30
Table 15: SFINCS.....	32
Table 16: WFLOW.jl	35
Table 17: HydroMT-WFLOW	37
Table 18: RA2CE.....	39
Table 19: Hython Wflow_SBM Hydrological Model	40



Executive summary

This report describes the status of the development of the 19 thematic modules that have been identified within the interTwin project to support multiple Digital Twins (DTs) in the environmental domain. With respect to the previous release [\[R1\]](#), 4 additional thematic modules (i.e., `esgpull_rucio`, `WFLOW.jl`, `HydroMT-WFLOW`, and `RA2CE`) have been added to cover the requirements of the DT applications.

In particular, the report provides an overview of the final version of the different thematic modules being developed for supporting Climate analytics and data processing ([Task 7.4](#)), Earth Observation Modelling and Processing ([Task 7.5](#)), and Hydrological model data processing ([Task 7.6](#)) and how these are linked to the Digital Twins.

The 19 thematic modules are described according to a common template highlighting key information on the codebase as well as the release notes and future work. All the modules are released as open source software and are provided with user and technical documentation. The majority of the modules are implemented in Python language.

The report concludes with a short summary of the integration status of the different thematic components. Multiple integration activities with respect to the other components of the interTwin DTE (e.g., the core and infrastructure modules) have already started.

1 Introduction

1.1 Scope

This deliverable summarises the status of the development of the 19 thematic modules relevant for implementing DTs across multiple domains related to environmental and climate sciences. Additional thematic modules (i.e., esgpull_rucio, WFLOW.jl, HydroMT-WFLOW, and RA2CE) have been introduced with respect to the previous release [\[R1\]](#) to address the requirements of the DT applications. Moreover the module eddiesGNN has been renamed to eddiesML while CompEvPoEToE has been renamed to emergence.compound.

It is important to mention that, although this document aims to describe the final version of all the thematic components from the environmental domain, more developments and extensions are planned for most of the thematic modules in the remainder of the project to (i) further improve the modules and (ii) adapt the solutions as the integration with the Digital Twins and other components of the interTwin DTE progresses.

Starting from the updated design and requirements of the thematic modules reported in D7.5 “Updated report on requirements and thematic modules functionalities for environment domain” [\[R2\]](#), this document outlines the main development and extensions included in the final version of such modules.

1.2 Document Structure

[Section 2](#) provides an overview of the thematic modules developed as well as the links with the Digital Twins applications. [Section 3](#) reports about the modules functionalities, value, licence and documentation. For each module Release Notes, with information about the current version status, and Future Plans, with developments that need to be completed, are also described. All the software is released as open-source code and available on git/Github. Moreover, most of the modules are described in the interTwin web portal. Finally, [Section 4](#) provides a summary of the integration status with respect to the project DTE and the DT from Work Package 4, as well as the main conclusions.



2 Overview of Thematic Modules for the environmental domain

The thematic modules can be classified according to their main scope as:

- Climate analytics and data processing ([Task 7.4](#))
- Earth Observation Modelling and Processing ([Task 7.5](#))
- Hydrological model data processing ([Task 7.6](#))

2.1 T7.4: Climate analytics and data processing

The goal of task 7.4 is to provide a set of thematic software modules for supporting climate/weather-related data processing including data-driven models for climate-based DTs. The thematic components are developed taking into account the requirements from WP4 DTs applications and in particular concerning extreme events. The thematic modules address capabilities related to climate/weather data gathering and preparation, data-driven approaches for extreme events detection and prediction, as well as data downscaling, algorithms for extreme characterization and compound event analysis, and post-processing of results.

As reported in previous documents (i.e., D7.3 [\[R1\]](#) and D7.5 [\[R2\]](#)), the set of thematic modules has been revised and consolidated throughout the project to better address the requirements of DT applications. Moreover, with respect to previous deliverables a dedicated module has been implemented for addressing climate projection data gathering from the Earth System Grid Federation (ESGF) and ingestion in the interTwin data lake. The thematic modules developed in T7.4 are (more details in subsections from [3.1](#) to [3.7](#)):

- **ML TC detection:** Python modules for tropical cyclones-related data analysis and events detection.
- **ML4Fires:** Python modules for wildfires-related data analysis and events prediction.
- **eddiesML:** Python modules for oceanic mesoscale eddies data analysis.
- **xtclim:** Python module for generic detection and characterization of climate extreme changes and impacts in the future climate projections.
- **downscaleML:** Python package for downscaling climate data.
- **emergence.compound:** Library for detection of time or periods of emergence for compound events.
- **Esgpull_rucio:** toolkit for gathering CMIP6 data from ESGF, through the esgpull [\[R6\]](#) tool, and uploading it to the RUCIO data lake (module added in this release).



2.2 T7.5: Earth Observation Modelling and Processing

This subsection covers the progress, adaptations and final remarks that have been made by the project consortium since the finalisation of D7.5 [R2] for developing thematic modules to run DTs based on EO data, with openEO [R5] as the driving technology. The developed thematic software modules support the Earth Observation data Modelling and Processing blocks, with the interaction between the modules for both flood and drought scenarios described in the C4 models available in deliverable D7.5 [R2], section 1.1.1. The thematic modules are reported on in more detail in [subsections 3.7, 3.8 and 3.9](#).

2.3 T7.6: Hydrological model data processing

Task 7.6 aims to develop thematic software modules to support data processing for hydrological models. The focus is on developing the necessary modules to facilitate the near-automatic setting up of flood hazard and impact models for user-defined regions of interest anywhere on Earth, including complementary Earth Observation data processing pipelines. With respect to the previous version of the environmental thematic modules three new modules have been included in this version (i.e., WFLOWs.ji, HydroMT-WFLOW, RA2CE).

The following thematic modules will be reported on in more detail in [Section 3](#):

- **FloodAdapt**¹: A software package Support System which can be used to assess the benefits and costs of Flood Resilience measures in a community. It uses output from SFINCS (Super-Fast INundation of CoastS), WFLOW, Delft-FIAT and RA2CE as calculation cores.
- **HydroMT**²: An open-source Python package that facilitates the process of building and analysing spatial geoscientific models with a focus on water system models.
 - **HydroMT-SFINCS**³: A plugin that provides an implementation of the model API for the SFINCS model.
 - **HydroMT-FIAT**⁴: A plugin which can be used to build a Delft-FIAT model anywhere in the world.
 - **HydroMT-WFLOW**⁵: A plugin to facilitate building and updating WFLOW models (module added in this release).
- **SFINCS**⁶: A reduced-complexity model designed for super-fast modelling of compound flooding events in a dynamic way.
- **Delft-FIAT**⁷: A fast and flexible flood impact assessment tool which combines hazard, vulnerability, and exposure to calculate damages and risk.

¹ <https://deltares-research.github.io/FloodAdapt/>

² https://deltares.github.io/hydromt/v1.0.0/guides/user_guide/hydromt_intro.html

³ https://deltares.github.io/hydromt_sfincs/latest/

⁴ https://deltares.github.io/hydromt_fiat/latest/

⁵ https://deltares.github.io/hydromt_wflow/latest/

⁶ <https://sfincs.readthedocs.io/en/latest/index.html>

⁷ <https://deltares.github.io/Delft-FIAT/stable/>



D7.7 Final version of the thematic module for the environment domain

- **WFLOW.jl**⁸: A hydrologic model to account for precipitation, interception, snow accumulation and melt, evapotranspiration, soil water, surface water, groundwater recharge, and water demand and allocation at the catchment scale (module added in this release).
- **RA2CE**⁹: A graph-based model that helps to quantify the resilience of critical infrastructure networks disrupted by flood hazards (module added in this release).
- **Hython Wflow_SBM Hydrological Model**: A spatially distributed hydrologic model to estimate hydrological fluxes, including snow accumulation and melt, interception, evapotranspiration, soil moisture, streamflow and actual evapotranspiration.

The application of the above thematic modules is demonstrated in Jupyter Notebooks that describe setting up the necessary models, configuring them and running scenarios and events and analysing their output. The Jupyter Notebooks are published here: <https://github.com/interTwin-eu/DT-flood/>.

2.4 Summary of supported Digital Twins

The thematic modules just presented are used for supporting different Digital Twin applications from WP4. The following table provides an overview of the DTs from WP4 currently supported by the different classes of environmental thematic modules.

Table 1: Summary of supported Digital Twins

Thematic modules classification	Digital Twins applications
T7.4: Climate analytics and data processing	<ul style="list-style-type: none">● T4.5:<ul style="list-style-type: none">○ Wildfire danger prediction on climate data○ Tropical cyclones detection and tracking on climate data○ Eddies prediction on unstructured meshes● T4.6: Drought early warning system for the Alps● T4.7: Climate Change Impacts of Extreme Events
T7.5: Earth Observation Modelling and Processing	<ul style="list-style-type: none">● T4.6:<ul style="list-style-type: none">○ Drought early warning system for the Alps

⁸ <https://deltares.github.io/Wflow.jl/dev/>

⁹ <https://ra2ce.readthedocs.io/en/stable/readme.html>



D7.7 Final version of the thematic module for the environment domain


	<ul style="list-style-type: none">○ Flood early warning in coastal and inland regions
T7.6: Hydrological model data processing	<ul style="list-style-type: none">● T4.6:<ul style="list-style-type: none">○ Drought early warning system for the Alps○ Flood early warning in coastal and inland regions● T4.7: Flood climate impact in coastal and inland regions



3 Thematic Modules

3.1 ML TC detection

Table 2: ML TC detection

Component name and logo	 <p>ML TC Detection Thematic modules for tropical cyclones (TCs)</p>
Page on interTwin website	https://www.intertwin.eu/article/thematic-module-ml-tc-detection
Description	Provides a set of Python modules for supporting processing and analysis of TC-related data and data-driven models
Value proposition	Address tropical cyclones analysis by providing the tools for gathering and pre-processing data, training different ML models, and post-processing the results. Furthermore, it provides functions for deterministic tracking and ML model ensemble. Multiple types of ML models are supported, in particular CNN, Transformers and GNN.
Users of the Component	<ul style="list-style-type: none"> • Developers of DTs • Expert scientists
User Documentation	https://github.com/CMCC-Foundation/ml-tropical-cyclones-detection/blob/main/README.md
Technical Documentation	https://github.com/CMCC-Foundation/ml-tropical-cyclones-detection/blob/main/README.md
Responsible	CMCC (Donatello Elia, Davide Donno) and UNITN (Massimiliano Fronza)
Licence	GPLv3
Source code	https://github.com/CMCC-Foundation/ml-tropical-cyclones-detection
Language	Python, PyTorch

3.1.1 Release notes

The Python package containing different libraries is publicly released on GitHub. Some of the key developments and extensions of this version with respect to the previous one, include:

- support for CNN [R8] and GNN [R9] integrated into a single code base. Moreover, other architectures have been explored (i.e., transformers) and are now supported by the module;
- training hyper-parameters (e.g., loss, network, learning rate, optimizer, etc.) of the experiments can be configured via a configuration file in Toml format. Furthermore, the set of input drivers can also be customized starting from the variables commonly associated with TCs. The module for hyperparameter tuning in GNN training has also been polished;
- updated toolkits for data filtering and downloading data used for the models;
- ensemble of compatible (e.g., same training dataset and input drivers) ML models with different architecture integrated;
- PyTorch Lightning Trainer for fast distributed training of the ML models on multiple GPUs and compute nodes;
- functionalities for building the TC tracks based on a deterministic tracker integrated;
- conversion of inference graph-based (GNN) outputs to 40x40 gridded coordinates;
- functionalities for plotting different types of results related to TC detection and tracking (e.g., charts and maps);
- integration of yProv4ML in the GNN side of the pipeline for tracking provenance information during the training process;
- creation of an inference notebook for GNN models to produce CSV files ready for post-processing, compatible with existing modules.

Documentation has been extended to better describe the different features provided by the module. Notebooks for testing and validating the different features of the thematic module are also included in the package.


3.1.2 Future plans

In the remainder of the project the current release will be further improved to enhance the TC tracking inference pipeline. Moreover, additional validation metrics will be supported by the module for evaluating the results and related visualization functions will be extended. Concerning the integration with climate projection data, extensions will be implemented to address the preparation of these data for the TC detection and tracking pipelines. The multi-model pipelines will be extended to also support ensemble tracking over multiple climate projection models/scenarios. Integration with other DTE components will be strengthened.



3.2 ML4Fires

Table 3: ML4Fires

Component name and logo	 <p>ML4Fires ML4Fires</p>
Page on interTwin website	https://www.intertwin.eu/intertwin-use-case-a-digital-twin-for-projecting-wildfire-danger-due-to-climate-change
Description	Provides a set of Python modules for supporting the processing and analysis of wildfires-related data and data-driven models
Value proposition	Address wildfires analysis and prediction (e.g., producing burned areas maps) providing tools that allow users to pre-process data, choose ML model architecture, train the model, post-process and visualize the results. Furthermore, it integrates functionalities to track ML model metrics and provenance during the training phase.
Users of the Component	<ul style="list-style-type: none"> • Developers of DTs • Expert scientists
User Documentation	https://github.com/CMCC-Foundation/ML4Fires/blob/main/README.md
Technical Documentation	https://github.com/CMCC-Foundation/ML4Fires/blob/main/README.md
Responsible	CMCC (Donatello Elia, Emanuele Donno)
Licence	Apache v2.0
Source code	https://github.com/CMCC-Foundation/ML4Fires
Language	Python, PyTorch

3.2.1 Release notes

The current version of the ML4Fires thematic module is publicly available as open source code on GitHub. This version provides an updated and improved documentation to describe the different capabilities of the module.

In terms of developments and extensions, this version includes:



D7.7 Final version of the thematic module for the environment domain

- improved management of SeasfireCube data [R3] for training, validation, and testing of the ML model;
- refactoring of the thematic module to better organize the codebase in libraries;
- scripts for ML model training data filtering and downloading;
- support for configuring the ML model's hyper-parameters (e.g., loss, layers arguments, learning rate, optimizer, early stopping, scheduler, etc.) and the training process via Toml configuration. Moreover, the set of input drivers and the target can be customized starting from the variables associated with wildfires (i.e., burned areas);
- initial support for multiple ML model architectures;
- extended the inference pipelines for supporting results processing and visualisation;
- support for tracking and monitoring ML model training process on MLFlow, as well as tracking of provenance information of the trained model with yProv4ML, both via the integration of itwinai;
- PyTorch Lightning Trainer to support fast distributed training of the ML models on multiple GPUs and compute nodes;

Demo notebooks for testing and evaluating the capabilities provided by the thematic module are also included in the package. The module includes model training and data visualisation utilities.

3.2.2 Future plans

The current version supports two types of CNN, UNet and UNet++ networks [R4], nevertheless, the support for using other ML model architectures will be evaluated in the next steps. Inference pipelines will also be improved with additional visualisation functionalities (e.g., charts and maps). Support for climate projection data will be included by finalizing the data management pipelines. Similarly to the previous thematic module, also in this case multi-model pipelines will be included to support ensemble prediction over multiple climate projection models/scenarios. Integration with the core components from the interTwin DTE will be further strengthened and improved.

3.3 eddiesML

Table 4: eddiesML

Component name and logo	 eddiesGNN eddiesML
Page on interTwin website	https://www.intertwin.eu/article/thematic-module-eddiesgnn



D7.7 Final version of the thematic module for the environment domain

Description	Provides a set of Python modules for supporting processing and analysis of eddy-related data
Value proposition	Address oceanic mesoscale eddies analysis by providing the tools for pre-processing of FESOM2 data and training CNN models
Users of the Component	<ul style="list-style-type: none">• Developers of DTs• Expert scientists
User Documentation	https://github.com/LegoCreation/CNN_eddy_detection/blob/unitn_work/readme.md
Technical Documentation	https://github.com/LegoCreation/CNN_eddy_detection/blob/unitn_work/readme.md
Responsible	UNITN (Contact Point: Massimiliano Fronza)
Licence	GPLv3
Source code	https://github.com/LegoCreation/CNN_eddy_detection/tree/unitn_work
Language	Python, Tensorflow

3.3.1 Release notes

The development of the Python packages is currently underway. In this version, the application from the Alfred Wegener Institut is being ported to the Enes DataSpace. The repository includes the script to perform the main steps of the pipeline, including interpolation of FESOM2 unstructured grids to matrices, production of the segmentation masks with pyEddyTracker, training of CNN models and inference. These last two steps are performed with the help of the Tensorflow framework.


3.3.2 Future plans

Next versions will improve the prediction capabilities of the trained network, including the support for GPU parallel computations and the integration of yProv and possibly with itwinai. Since it's not possible to use future projection data like CMIP6 due to its resolution not being high enough for the detection of oceanic mesoscale eddies, we will work on FESOM2 grids. This new tool can, however, produce segmentation masks on structured climate grids many degrees of magnitude faster than the classic physics-based algorithms that were used until now. A multi-model (both for data and ML models) thematic module could be also explored to support ensemble detection approaches.



3.4 Xtclim

Table 5: Xtclim

Component name and logo	
Page on interTwin website	https://www.intertwin.eu/article/thematic-module-xtclim
Description	xtclim is a Python package implementing an unsupervised Deep Learning method, a CVAE that can characterise generic climate extreme events
Value proposition	Base methods and functions to provide the extraction of generic characteristics of climate extremes. It will enable users to explore the impacts on extreme events on specific users' applications in the context of selected climate simulations.
Users of the Component	<ul style="list-style-type: none"> • Developers of DTs • Expert scientists
User Documentation	https://github.com/cerfacs-globc/xtclim/blob/master/README.md
Technical Documentation	https://github.com/cerfacs-globc/xtclim/blob/master/README.md
Responsible	CERFACS (Christian Pagé)
Licence	Apache 2
Source code	https://github.com/cerfacs-globc/xtclim
Language	Python, PyTorch

3.4.1 Release notes

The current version of the xtclim Python package has evolved to a more stable and integrated version. The level of the software can be considered beta (previous version was alpha), as many aspects are still under development.

Currently, the xtclim package can identify generic climate extremes of temperature for four seasons using a CVAE through pyTorch, on global and regional climate simulations

D7.7 Final version of the thematic module for the environment domain

over a selected geographical region. The software module is generic enough to process any climate variable. The software module can also be adapted to provide the capability to work with up to three concurrent atmospheric variables.

The current version is focused on the implementation of the detection and characterization of temperature extremes, on CMIP global climate datasets.

The main workflow consists of data pre-processing to tailor and prepare data appropriately for the CVAE method, training, and then applying the method. End-user products are then generated at the last step of the workflow such as plots and graphs.

Major changes since the last version:

- Implemented a configuration file
- Extracted the method parameters from the code so they are modifiable by the user
- Made the code more generic
- Refactored part of the code to enhance performance mainly for training


3.4.2 Future plans

Specific developments planned for the next 4 months:

- Create Jupyter Notebook examples that will also be used as user documentation and workflow execution.
- Add more end-user products (already developed but not integrated) at the end of the current workflow.
- Refine and adapt the method for a better integration of precipitation and wind data.
- Finalize the integration with the DTE.

3.5 downscaleML: Downscaling Climate Data

Table 6: downscaleML: Downscaling Climate Data

Component name and logo	 downscaleML downscaleML: Downscaling Climate Data
Page on interTwin website	https://www.intertwin.eu/article/thematic-module-downscaleml
Description	downScaleML is an open-source Python package, designed to streamline the process of climate data downscaling using machine learning techniques. It offers



D7.7 Final version of the thematic module for the environment domain

	an automated workflow tailored for downscaling seasonal forecast climate variables, specifically temperature, precipitation and downward surface solar radiation, with a particular emphasis on addressing climate extremes.
Value proposition	It eases forecast data preprocessing, statistical downscaling, through a selection of machine learning techniques, and result validation. It provides a flexible module for any modelling scheme requiring tailored climate inputs, and enables scalability and applicability to other domains, resolutions, and datasets.
Users of the Component	<ul style="list-style-type: none">• Developers of DTs• Expert scientists
User Documentation	https://github.com/interTwin-eu/downScaleML#readme
Technical Documentation	https://github.com/interTwin-eu/downScaleML#readme
Responsible	EURAC (Alice Crespi, Suriyah Dhinakaran)
Licence	GNU GPL v3
Source code	https://github.com/interTwin-eu/downScaleML
Language	Python, PyTorch

3.5.1 Release notes

In the current developmental stage, the downscaling component enables a two-step downscaling procedure for targeting high scaling factors, the first step using regression methods and the last one using image super resolution ESRGANs under the pretext of statistical downscaling. The package is designed for downscaling modules for reanalysis datasets and for seasonal forecast datasets.

3.5.2 Future plans


For upcoming package updates, our objectives include:

- Optimising the downscaling machine learning models to process seasonal forecast data from ECMWF, with a focus on downscaling climate variables, particularly temperature, precipitation and downward surface solar radiation.
- Containerizing the entire package environment using Docker and establishing a streamlined pipeline for managing input and output data through openEO processes.



3.6 emergence.compound

Table 7: emergence.compound

Component name and logo	 <p>CompEVPoEToE emergence.compound</p>
Page on interTwin website	https://www.intertwin.eu/article/thematic-module-compevpoetoe
Description	Provides a set of R functions for determining if periods of emergence (PoE) and/or time of emergence (ToE) of compound events probabilities have emerged in data.
Value proposition	This module allows to statistically model if and how compound events have significantly evolved through time, based on reanalysis or simulated data. The definition of the compound (i.e., involved variables) is made by the user.
Users of the Component	<ul style="list-style-type: none"> Expert scientists
User Documentation	https://github.com/josephine400/emergence.compound/blob/main/README.md
Technical Documentation	https://github.com/josephine400/emergence.compound/blob/main/README.md
Responsible	CNRS (J. Schmutz, M. Vrac, G. Levavasseur)
Licence	CeCill-C
Source code	https://github.com/josephine400/emergence.compound
Language	R

3.6.1 Release notes

The development of the R package is finalized. In this version, the statistical protocol has been well-defined, developed and tested. A set of usual marginal distribution functions and Archimedian copula families are provided. A scientific paper, presenting the methodology, the protocol and an application to Europe hot & dry compound events has been submitted (<https://doi.org/10.5194/egusphere-2025-461>).

3.6.2 Future plans

Next steps include potential extensions to $n (>2)$ variables for more complex compound events, as well as an extension of the methodology to include the possibility to perform "attribution" studies.

3.7 Esgpull_rucio

Table 8: Esgpull_rucio

Component name and logo	esgpull_rucio
Page on interTwin website	<i>To be released with this deliverable</i>
Description	A toolkit that interfaces with esgpull (ESGF download tool) database to update (and keep updated) an intake catalogue as well as create RUCIO datasets, uploading and attaching the files to them. The tool fetches the completed file downloads entries from the database, creates a record of each completed dataset detected and using the RUCIO python API it creates the necessary DIDs (data identifiers) if they do not exist.
Value proposition	The tool is a seamless integration with the esgpull db API, which optimizes the workflow of downloading, cataloguing and appending to RUCIO of ESGF and CMIP data.
Users of the Component	<ul style="list-style-type: none"> • Developers of DTs
User Documentation	https://github.com/AtefBN/esgpullUtilities/blob/main/README.md
Technical Documentation	https://github.com/AtefBN/esgpullUtilities/blob/main/README.md
Responsible	CNRS (A. Ben Nasser)



D7.7 Final version of the thematic module for the environment domain

Licence	CeCill-C
Source code	https://github.com/AtefBN/esgpullUtilities
Language	Python

3.7.1 Release notes

This initial release of the tool allows for synchronization between a local depot of ESGF datasets and RUCIO catalogues. It relies on an intermediary JSON catalogue that represents a targeted subset of data from the esgpull [R6] database.

Another key point about this release is that it does not handle the authentication side of things automatically, an hourly RUCIO login prompt would require using a browser and going through the EGI SSO check-in process.


3.7.2 Future plans

The next steps include:

- bypassing the JSON catalogue intermediary and syncing directly with esgpull's database
- automating the EGI SSO check-in process and generating on-the-fly RUCIO tokens for authentication and authorization
- improving error handling and message logging to future proof the tool.

3.8 openeo-processes-dask

Table 9: openeo-processes-dask

Component name and logo	 openeo-processes -dask openeo-processes-dask
Page on interTwin website	https://www.intertwin.eu/article/thematic-module-openeo-processes-dask
Description	Python implementation of openEO processes.
Value proposition	Base component necessary to run openEO process graphs. All the processes are implemented using Dask [R7], making them easily scalable and parallelizable.
Users of the Component	<ul style="list-style-type: none">• Expert users• Flood and drought modellers



D7.7 Final version of the thematic module for the environment domain

User Documentation	https://open-eo.github.io/openeo-python-client/cookbook/localprocessing.html
Technical Documentation	https://github.com/Open-EO/openeo-processes-dask
Responsible	EODC (Valentina Hutter) and EURAC (Michele Claus)
Licence	Apache 2.0
Source code	https://github.com/Open-EO/openeo-processes-dask
Language	Python

3.8.1 Release notes


The xArray/Dask implementation of the openEO processes is already deployed in the EODC openEO back-end (part of the DTE), available via openEO Platform (<https://openeo.cloud/>) or directly (<https://openeo.eodc.eu/openeo/1.1.0>). The same code is also used for the Client-Side Processing functionality, allowing to prototype an openEO workflow in a local machine with the same Python syntax (<https://open-eo.github.io/openeo-python-client/cookbook/localprocessing.html>), also used for the implementation of flood mapping developed by TU Wien (<https://github.com/interTwin-eu/openeo-flood-mapper-local>). From the latest release many new processes have been added, improving the integration and functionalities related to vector data cubes. Moreover, several improvements and bug fixes have been introduced for the existing processes.

3.8.2 Future plans

The developments in this module have been completed.

3.9 openeo-pg-parser-networkx

Table 10: openeo-pg-parser-networkx

Component name and logo	 openeo-pg-parser networkx openeo-pg-parser-networkx
Page on interTwin website	https://www.intertwin.eu/article/thematic-module-openeo-pg-parser-networkx/
Description	Parse OpenEO process graphs from JSON to traversable

D7.7 Final version of the thematic module for the environment domain

	Python objects.
Value proposition	Base component necessary to parse openEO process graphs, before calling openeo-processes-dask.
Users of the Component	<ul style="list-style-type: none">Expert users
User Documentation	https://github.com/Open-EO/openeo-pg-parser-networkx/blob/main/README.md
Technical Documentation	https://github.com/Open-EO/openeo-pg-parser-networkx/blob/main/README.md
Responsible	EODC (Valentina Hutter) and EURAC (Michele Claus)
Licence	Apache 2.0
Source code	https://github.com/Open-EO/openeo-pg-parser-networkx
Language	Python

3.9.1 Release notes

Similarly, with the openeo-processes-dask component, the openeo-pg-parser-networkx is already in use in the EODC back-end and in the Client-Side Processing. The component received a major refactoring, updating a core library (Pydantic) from version 1.x to version 2.x, increasing the cross-compatibility with other components developed in InterTwin, like HydroMT.

3.9.2 Future plans

The developments in this module have been completed.

3.10 raster-to-stac

Table 11: raster-to-stac

Component name and logo	
Page on interTwin website	https://www.intertwin.eu/article/thematic-module-

D7.7 Final version of the thematic module for the environment domain

	<u>raster-to-stac/</u>
Description	Create STAC metadata for raster datasets.
Value proposition	Makes a resulting dataset easily accessible, interoperable, and shareable.
Users of the Component	<ul style="list-style-type: none">• Expert users
User Documentation	<u>https://raster2stac.readthedocs.io/en/latest/</u>
Technical Documentation	<u>https://raster2stac.readthedocs.io/en/latest/</u>
Responsible	EURAC (Michele Claus)
Licence	MIT
Source code	<u>https://gitlab.inf.unibz.it/earth_observation_public/raster-to-stac/</u>
Language	Python

3.10.1 Release notes

The component has been released officially and integrates the planned features:

- Flexible generation of STAC Collections with matching STAC extension based on the provided parameters.
- Multiple file formats allowed (COGs, netCDFs, Zarr).
- Writing directly to an S3 bucket.
- openEO integration in the save_result process.

3.10.2 Future plans

The developments in this module have been completed.

3.11 FloodAdapt

Table 12: FloodAdapt

Component name and logo	
-------------------------	--------------------------------------------------------------------------------------

D7.7 Final version of the thematic module for the environment domain

Page on interTwin website	https://www.intertwin.eu/article/thematic-module-floodadapt/
Description	A thematic module which can be used to assess the benefits and costs of Flood Resilience measures in a community. It uses SFINCS, WFLOW, FIAT-Objects, and RA2CE in the background.
Value proposition	FloodAdapt is a decision-support tool and API that seeks to advance and accelerate flooding-related adaptation planning. It brings rapid, physics-based compound flood modelling and detailed impact modelling into an easy-to-use system, allowing non-expert end-users to evaluate a wide variety of compound events, future conditions, and adaptation options in minutes. FloodAdapt serves as a connector between scientific advances and practitioner needs, improving and increasing the uptake and impact of adaptation research and development.
Users of the Component	<ul style="list-style-type: none">• Non-expert end-users• Decision makers• Planners
User Documentation	https://www.deltares.nl/en/software-and-data/products/floodadapt
Technical Documentation	https://github.com/Deltares/FloodAdapt#readme
Responsible	Deltares (Kathryn Roscoe)
Licence	MIT
Source code	https://github.com/Deltares/FloodAdapt
Language	Python

3.11.1 Release notes

The deployment of the FloodAdapt backend is achieved via Jupyter Notebooks developed in interTwin (<https://github.com/interTwin-eu/DT-flood/>). These include basic examples of how to configure SFINCS and Delft-FIAT models for the FloodAdapt backend and a basic example of how to couple to a WFLOW hydrological model and a RA2CE model.

New features include:

- New flood event configuration supporting spatially distributed wind and rainfall through NetCDF files.
- Database builder simplifying the setup of the FloodAdapt backend starting from existing SFINCS and Delft-FIAT models



D7.7 Final version of the thematic module for the environment domain


- Pre-packaged defaults for event metrics.

3.11.2 Future plans

Next releases will include a tighter and more streamlined integration with the WFLOW and RA2CE models.

3.12 HydroMT-SFINCS

Table 13: HydroMT-SFINCS

Component name and logo	 HydroMT-SFINCS HydroMT-SFINCS
Page on interTwin website	https://www.intertwin.eu/article/thematic-module-hydromt-sfincs/
Description	HydroMT (Hydro Model Tools) is an open-source Python package that facilitates the process of building and analysing spatial geoscientific models with a focus on water system models. It does so by automating the workflow to go from raw data to a complete model instance which is ready to run and to analyse model results once the simulation has finished. This plugin provides an implementation of the model API for the SFINCS model.
Value proposition	Easily build and update the SFINCS model with a single line of code.
Users of the Component	<ul style="list-style-type: none">• Expert users• Flood modellers
User Documentation	https://deltares.github.io/hydromt_sfincs/latest/index.html
Technical Documentation	https://deltares.github.io/hydromt_sfincs/latest/getting_started/intro
Responsible	Deltares (Roel de Goede)



D7.7 Final version of the thematic module for the environment domain

Licence	GNU GPL v3
Source code	https://github.com/Deltares/hydromt_sfincs
Language	Python

3.12.1 Release notes

The application of HydroMT-SFINCS is demonstrated in Jupyter Notebooks developed in interTwin (<https://github.com/interTwin-eu/DT-flood/>).

This release offers examples of how to set up and run SFINCS. The HydroMT-SFINCS plugin offers capabilities to easily set up and run SFINCS using globally available data.

Users can select flood walls, levees, pumps, culverts, and green infrastructure as flood mitigation and adaptation measures.


3.12.2 Future plans

In the next release, capabilities will be created so that a user can:

- easily run long hindcast and ensemble simulations and produce probabilistic flood maps using the flood inundation model (SFINCS);
- easily run the flood inundation model (SFINCS) using locally available data.

3.13 HydroMT-FIAT

Table 14: HydroMT-FIAT

Component name and logo	 HydroMT-FIAT HydroMT-FIAT
Page on interTwin website	https://www.intertwin.eu/article/thematic-module-hydromt-fiat/
Description	HydroMT is an open-source Python package, developed by Deltares, to build and analyze hydro models. It provides a generic model API with attributes to access the model schematization, (dynamic) forcing data, results, and states. This plugin provides an implementation for the Delft-FIAT model.



D7.7 Final version of the thematic module for the environment domain

Value proposition	<p>With the HydroMT-FIAT plugin, users can easily benefit from the rich set of tools of the HydroMT package to build and update Delft-FIAT models from available global and local data.</p> <p>This plugin assists the FIAT modeller in:</p> <ul style="list-style-type: none">• quickly setting up a Delft-FIAT model based on existing hazard maps, global and user-input exposure layers, and a global database of vulnerability curves;• adjusting and updating components of a FIAT model and their associated parameters in a consistent way, e.g., to test measures that affect the exposure or vulnerability of a FIAT model or to improve an existing FIAT model with better quality data;• building FIAT models in a reproducible and consistent way.
Users of the Component	<ul style="list-style-type: none">• Expert users• Flood risk specialists
User Documentation	https://deltares.github.io/hydromt_fiat/latest/index.html
Technical Documentation	https://deltares.github.io/hydromt_fiat/latest/index.html
Responsible	Deltares (Frederique de Groen)
Licence	GNU GPL v3
Source code	https://github.com/Deltares/hydromt_fiat/tree/main
Language	Python

3.13.1 Release notes

The application of HydroMT-FIAT is demonstrated in Jupyter Notebooks developed in interTwin (<https://github.com/interTwin-eu/DT-flood/>).

This release offers examples of how to set up and run SFINCS and Delft-FIAT. The HydroMT-FIAT plugin offers capabilities to easily set up and run Delft-FIAT using globally available data.

Users can select raising properties, floodproofing properties and buying out neighbourhoods as flood mitigation and adaptation measures.




3.13.2 Future plans

In the next release, capabilities will be created so that a user can estimate damages to buildings and utilities using locally available data.

3.14 SFINCS

Table 15: SFINCS

Component name and logo	 <p>SFINCS SFINCS</p>
Page on interTwin website	https://www.intertwin.eu/article/thematic-module-sfincs/
Description	SFINCS is a new fast numerical model to simulate 2D compound flooding dynamically for large scale coastal systems, within a fraction of the time required by the Delft3D-1D2D models.
Value proposition	Compound flooding during extreme events can result in tremendous amounts of property damage and loss of life. Early warning systems and multi-hazard risk analysis can reduce these impacts. However, traditional approaches either do not involve relevant physics or are too computationally expensive to do so for large stretches of coastline. The SFINCS model (Super-Fast INundation of CoastS) is a new reduced-complexity engine recently developed by Deltares, that is capable of simulating compound flooding including a high computational efficiency balanced with good accuracy.
Users of the Component	<ul style="list-style-type: none"> • Expert users • Flood modellers
User Documentation	https://www.deltares.nl/en/software-and-data/products/SFINCS
Technical Documentation	https://sfincs.readthedocs.io/en/latest/
Responsible	Deltares (Tim Leijnse, Roel de Goede)
Licence	GNU GPL v3

D7.7 Final version of the thematic module for the environment domain

Source code	https://github.com/Deltares/SFINCS
Language	Fortran

3.14.1 Release notes

The application of SFINCS is demonstrated in Jupyter Notebooks developed in interTwin (<https://github.com/interTwin-eu/DT-flood/>).

This release offers examples of how to set up and run SFINCS. It offers capabilities to simulate floodings using globally available data.

Users can specify a geographic area of interest and a minimal number of model-specific parameters, and a SFINCS model to simulate flood inundation is automatically set up using globally available data.

Globally available static and dynamic input data can be used to create deterministic flood maps based on SFINCS.

A user can specify a simulation period, and the flood inundation model (SFINCS) can be run on heterogeneous computing infrastructures using Docker or Singularity containers.

A user can select scenarios to simulate, including mitigation and adaptation measures and the flood inundation model (SFINCS) and the necessary static and dynamic input data are automatically prepared to run the scenario.

3.14.2 Future plans

In the next release capabilities will be created:

- Users can specify a geographic area of interest and a minimal number of model-specific parameters, and SFINCS to simulate flood inundation is automatically set up using locally available data.
- Users can easily run the flood inundation model (SFINCS) using locally available data.
- Examples of how to generate probabilistic flood maps using SFINCS will be provided.

3.15 Delft-FIAT

Component name and logo	
Page on interTwin website	https://www.intertwin.eu/article/thematic-module-



D7.7 Final version of the thematic module for the environment domain

	<u>delft-fiat/</u>
Description	Delft-FIAT is a fast, free, Python-based tool developed and continuously improved by Deltares to rapidly assess direct economic impacts on buildings, utilities, and roads for user-input flood maps.
Value proposition	<p>Fast impact modelling removes traditional bottlenecks in climate adaptation planning, making it possible to (1) understand the effectiveness of adaptation options and (2) quantify changes in damage and risk as climate and socio-economic conditions change.</p> <p>Fast and automated Delft-FIAT is fast and can be automated. This makes it possible to evaluate future risks caused by changing drivers like growing populations and economies. It also makes it possible to evaluate the effectiveness of interventions by assessing flood damages - now and under changing conditions (and combinations of) interventions, like home elevations, buy-outs, or floodproofing.</p> <p>Flexible Delft-FIAT has a flexible architecture and is data-agnostic. Exposure data can easily be modified, and hazard data - the flood maps - can come from any source. For example, a user may want to try out different depth-damage functions or include a different class of damage than the traditional structure and content damages.</p> <p>Furthermore, any damage type that can be described with a depth-damage function can be analysed in Delft-FIAT.</p> <p>Customisable Delft-FIAT is also customisable. It can be connected to a tailored user-interface to make a custom damage modelling tool for less-technical users.</p>
Users of the Component	<ul style="list-style-type: none"> • Expert users • Flood risk specialists
User Documentation	<u>https://www.deltares.nl/en/software-and-data/products/delft-fiat-flood-impact-assessment-tool</u>



D7.7 Final version of the thematic module for the environment domain

Technical Documentation	https://github.com/Deltares/Delft-FIAT#readme
Responsible	Deltares (Kathryn Roscoe)
Licence	GNU GPL v3
Source code	https://github.com/Deltares/Delft-FIAT
Language	Python

3.15.1 Release notes

The application of Delft-FIAT is demonstrated in Jupyter Notebooks developed in interTwin (<https://github.com/interTwin-eu/DT-flood/>).

This release offers examples of how to set up and run Delft-FIAT. It offers capabilities to quantify the direct damages resulting from a flood.

Users can specify a geographic area of interest and a minimal number of model-specific parameters and use the output flood maps from SFINCS to estimate damages to buildings and utilities (Delft-FIAT) are automatically set up using globally available data.

Users can configure the input, i.e., the flood map, needed for the flood impact assessment tool (Delft-FIAT) and execute it from a command line interface.

Users can select scenarios to simulate, including mitigation and adaptation measures and the flood inundation model (SFINCS), the flood impact assessment tool (Delft-FIAT) and the necessary static and dynamic input data are automatically prepared to run the scenario.

A docker container has been released on dockerhub (<https://hub.docker.com/r/deltares/flat>) to support running Delft-FIAT on a wider variety of computing environments.

3.15.2 Future plans

In the next release, capabilities will be created so that a user can estimate damages to buildings and utilities using locally available data.

3.16 WFLOW.jl

Table 16: WFLOW.jl

Component name and logo	WFLOW
Page on interTwin website	<i>To be released with this deliverable</i>



D7.7 Final version of the thematic module for the environment domain

Description	<p>Wflow is Deltares' solution for modelling hydrological processes, allowing users to account for precipitation, interception, snow accumulation and melt, evapotranspiration, soil water, surface water, groundwater recharge, and water demand and allocation in a fully distributed environment. Successfully applied worldwide for analyzing flood hazards, drought, climate change impacts and land use changes, wflow is growing to be a leader in hydrology solutions. Wflow is conceived as a framework, within which multiple distributed model concepts are available, which maximizes the use of open earth observation data, making it the hydrological model of choice for data scarce environments. Based on gridded topography, soil, land use and climate data, wflow calculates all hydrological fluxes at any given grid cell in the model at a given time step.</p>
Value proposition	<p>Wflow was born out of the creation of Deltares in 2008, when a strategic review identified the need for a distributed hydrological model to allow the simulation of flows at the catchment scale. With the intention being to encourage greater scientific collaboration. For this reason:</p> <ul style="list-style-type: none"> ● Wflow is free and open source software. ● Wflow is easily coupled with other models and software applications. ● Contribution to the wflow code development is encouraged. <p>From 2021 the wflow code is distributed under the MIT License. Wflow is also available as a compiled executable under the Deltares terms and conditions. The wflow computational engine is built in the Julia language, a high-performance computing language. Wflow does not include a graphical user interface and is designed for maximum user flexibility. Prior to 2021, wflow was developed in Python on top of the PCRaster Python extension. The Python version is still available, but not actively developed.</p>
Users of the Component	<ul style="list-style-type: none"> ● Expert users ● Flood risk specialists
User Documentation	<p>https://deltares.github.io/Wflow.jl/stable/</p>



D7.7 Final version of the thematic module for the environment domain

Technical Documentation	https://deltares.github.io/Wflow.jl/stable/
Responsible	Ali Meshgi
Licence	MIT
Source code	https://github.com/Deltares/Wflow.jl
Language	Julia

3.16.1 Release notes

The application of WFLOW is demonstrated in Jupyter Notebooks developed in interTwin (<https://github.com/interTwin-eu/DT-flood/>).

This release offers a basic example of how to set up a WFLOW model for applications in flood modelling and a basic demonstration of the coupling between WFLOW and the FloodAdapt backend.

Users can specify a geographic area and simulation period of interest and a minimal number of model-specific parameters to set up a WFLOW model to simulate river discharges and couple the outputs to a SFINCS model covering the same area of interest. The WFLOW model is configured using globally available static and dynamic input data. WFLOW can be run using the published docker container (<https://hub.docker.com/r/deltares/wflow>) to support heterogeneous computing infrastructure

This release offers improved capabilities for re-infiltration and coupling between leaf area index and land classes to assess the effects of changes in land use on flooding.

3.16.2 Future plans

The primary focus for future developments is to add capabilities to run WFLOW on GPUs.

3.17 HydroMT-WFLOW

Table 17: HydroMT-WFLOW

Component name and logo	HydroMT-WFLOW
Page on interTwin website	<i>To be released with this deliverable</i>



D7.7 Final version of the thematic module for the environment domain

Description	<p>HydroMT (Hydro Model Tools) is an open-source Python package that facilitates the process of building and analyzing spatial geoscientific models with a focus on water system models. It does so by automating the workflow to go from raw data to a complete model instance which is ready to run and to analyze model results once the simulation has finished. This plugin provides an implementation of the model API for the Wflow model.</p>
Value proposition	<p>Setting up distributed hydrological models typically requires many (manual) steps to process input data and might therefore be time consuming and hard to reproduce. Especially improving models based on global-local geospatial datasets, which are rapidly becoming available at increasingly high resolutions, might be challenging. HydroMT-Wflow aims to make the Wflow model building and updating processes fast, modular and reproducible and to facilitate the analysis of the model results.</p> <p>The HydroMT-Wflow plugin can be used as a command line application, which provides commands to build, update and clip a Wflow model with a single line, or from Python to exploit its rich interface. You can learn more about how to use HydroMT-Wflow in its online documentation. For a smooth installation experience, we recommend installing HydroMT-Wflow and its dependencies from conda-forge in a clean environment, see the installation guide.</p>
Users of the Component	<ul style="list-style-type: none"> • Expert users • Flood risk specialists
User Documentation	<p>https://deltares.github.io/hydromt_wflow/latest/</p>
Technical Documentation	<p>https://deltares.github.io/hydromt_wflow/latest/</p>
Responsible	Deltares (Ali Meshgi)
Licence	GNU GPL v3
Source code	<p>https://github.com/Deltares/hydromt_wflow</p>
Language	Python



3.17.1 Release notes

The application of HydroMT-WFLOW is demonstrated in Jupyter Notebooks developed in interTwin (<https://github.com/interTwin-eu/DT-flood/>).

This release offers examples of how to set up and run WFLOW. The HydroMT-WFLOW plugin offers capabilities to easily set up and run WFLOW using globally available data.

The release also demonstrates how to configure a WFLOW model using HydroMT-WFLOW in order to couple river discharges calculated using WFLOW to an SFINCS inundation model.

3.17.2 Future plans

Future releases will focus on improving the visualizations provided in the Jupyter Notebooks developed for interTwin.

3.18 RA2CE

Table 18: RA2CE

Component name and logo	RA2CE
Page on interTwin website	<i>To be released with this deliverable</i>
Description	RA2CE helps to quantify the resilience of critical infrastructure networks, prioritize interventions and adaptation measures and select the most appropriate action perspective to increase resilience considering future conditions.
Value proposition	<p>The RA2CE - Resilience Assessment and Action perspective for Critical infrastru^E - model has been developed to support infrastructure owners and operators in resilience assessment and adaptation decision-making and has been applied in several settings such as the Netherlands, Philippines, Myanmar, Dominican Republic and Albania.</p> <p>The current capabilities focus on mapping the exposure, criticality, and vulnerability as well as the forthcoming prioritisation of locations to take actions based on cost</p>

D7.7 Final version of the thematic module for the environment domain

	benefit assessment. For further assessment of indirect impacts, inclusiveness and equity principles can be applied. In adaptation and planning studies the platform enables to perform cost-benefit assessments including an uncertain future
Users of the Component	<ul style="list-style-type: none">• Expert users• Flood risk specialists
User Documentation	https://deltares.github.io/ra2ce/index.html
Technical Documentation	https://deltares.github.io/ra2ce/index.html
Responsible	Deltares (Thomas Bles)
Licence	GNU GPL v3
Source code	https://github.com/Deltares/ra2ce
Language	Python

3.18.1 Release notes

The application of RA2CE is demonstrated in Jupyter Notebooks developed in interTwin (<https://github.com/interTwin-eu/DT-flood/>).

This release offers a basic example of how to set up a RA2CE model for applications in flood impact modelling and a basic demonstration of the coupling between RA2CE and the FloodAdapt backend.

Users can specify a geographic area of interest to set up a RA2CE model to assess the impact of floods on the road network in that area, including travel times to important locations. Using the provided notebook example with this release the RA2CE model is automatically configured to couple to the FloodAdapt backend by reading in flood hazard maps from FloodAdapt. RA2CE can be run using a Docker container to support running on heterogeneous computing infrastructure.

3.18.2 Future plans


Future releases will focus on improving the visualizations provided in the Jupyter Notebooks developed for InterTwin.

3.19 Hython Wflow_SBM Hydrological Model

Table 19: Hython Wflow_SBM Hydrological Model



D7.7 Final version of the thematic module for the environment domain

Component name and logo	 <p style="text-align: center;">Hyton Wflow_sbm Hydrological Model Hython_sbm</p>
Page on interTwin website	https://www.intertwin.eu/article/thematic-module-hyhton-wflow_sbm-hydrological-model/
Description	<p>The Hython package enables the development of deep learning based surrogates of grid-based semi-distributed and distributed hydrological models, and it enables the calibration of the model's parameters exploiting satellite-based products. In particular, Hython_sbm is customized to emulate Wflow_sbm's vertical fluxes and states (soil moisture, evapotranspiration, snow water equivalent, etc.), and to calibrate the parameters by leveraging satellite-based products.</p>
Value proposition	<p>Traditional distributed hydrological models are complicated to set up, computationally expensive, and challenging to calibrate. One of the negative consequences is that they often lack an estimation of the output uncertainty. Hython_sbm, as a faster and reliable surrogate model, enables the pixel-by-pixel calibration of the hydrological model parameters, by leveraging satellite products. In addition, thanks to its increased performance and flexibility, it provides an estimation of the output uncertainty.</p> <p>The publication of the Hython_sbm module as an application package and openEO process, exposes its functionalities to the openEO user interface, where it can be consequently integrated in custom data and modelling workflows. This reduces dramatically the costs for setting up, training, evaluating the model's outputs and facilitating the experimentation with different data inputs and model parameters.</p> <p>The module can be useful to researchers investigating drought prediction and forecasting, and to public authorities in the field of agriculture and river basin management to identify areas potentially affected by hydrological or agricultural drought.</p>
Users of the Component	<ol style="list-style-type: none"> 1. Researchers,

D7.7 Final version of the thematic module for the environment domain

	<ol style="list-style-type: none">2. local/Regional public authorities in the field of agriculture,3. hydrology and river basin management authorities,4. journalist for environmental topics with little expertise about technical data.
User Documentation	https://github.com/interTwin-eu/hython/blob/main/README.md
Technical Documentation	https://github.com/interTwin-eu/hython/blob/main/README.md
Responsible	EURAC (Iacopo Federico Ferrario)
Licence	CC-BY-4.0 Licence
Source code	https://github.com/interTwin-eu/hython
Language	Python, PyTorch

3.19.1 Release notes

The package is currently under development.

A comprehensive refactoring allows Hython to emulate any grid-based distributed hydrological models provided that a related Dataset class is implemented. Hython_sbm is, therefore, the specific Hython surrogate of the Wflow_sbm hydrological model, but others could be implemented.

The package offers two deep learning architectures, a Long Short-Term Memory (LSTM) model and a Convolutional LSTM.

In the current release, the uncertainty estimation is provided only for the LSTM surrogate.

This release supports training and calibration of the Hython_sbm surrogate model and provides Jupyter notebooks that showcase both tasks. In addition a notebook showcases three methodologies to estimate uncertainty.

The calibration consists of the optimization of parameters that control soil moisture and it is performed by using the RT0 product developed by TU Wien within the context of the InterTwin project.

The functionalities of Hython are configurable through yaml files that are fully compatible with itwinai configuration files. Hython integrates and relies on itwinai for running on HPCs and on multi GPUs, for hyperparameter optimization and scaling tests.

3.19.2 Future Work

As next steps for the module, the following aspects are envisioned:



D7.7 Final version of the thematic module for the environment domain

- Evaluation and inference (drought forecasting).
- Full integration in openEO.
- Integration into the Datalake.



4 Conclusions

This report provides an overview of the final version of the thematic modules for the environmental domain. It presents the status of the developments for each thematic module in T7.4 Climate analytics and data processing, T7.5 Earth Observation Modelling and Processing, and T7.6 Hydrological model data processing. The development of all thematic modules has advanced, and in some cases, been completed. Basic documentation is publicly available for all the components.

For each of the 19 thematic modules a set of information about the current version is provided through a common template. Additional details of what is included in the version are reported in the release notes. Moreover, future work planned in the remainder of the project is also outlined. It is important to mention that, even though this document presents the final version of the thematic module, additional developments are expected in the remainder of the project, also driven by the integration with the DTs applications and the rest of the DTE. The final developments will be presented in the integration report D7.9.

In terms of integration activities, the environmental thematic modules are already using multiple infrastructure (WP5) and core (WP6) components from the project and stronger integration is planned for the remainder of the project. Integration with the DT applications from WP4 is also progressing. In the following we provide a brief overview of the integration activities that will be fully reported in D7.9. At the current stage, several thematic modules (Hython wflow_sbm, ML TC Detection, ML4Fires, xtclim) are integrating the itwinai framework capabilities (WP6). Additionally, the thematic modules related to the flood early warning and climate impact DTs (SFINCS, WFLOW, Delft-FIAT and RA2CE) are being integrated in OSCAR (WP6). Preliminary integration for provenance has also been performed in a couple of thematic modules (ML TC Detection, ML4Fires). Other components are integrated with workflows composition components such as OpenEO (e.g., open-processes-dask, open-pg-parser-networkxx, raster-to-stac). There is an ongoing integration of OSCAR and OpenEO to allow interacting with OSCAR services directly from an OpenEO Process graph. At the infrastructure level, evaluation of the components on the project testbed (e.g., Vega) started (WP5). Interactions with WP5 are also ongoing as part of the data needed by the DTs has been made available from the project data lake. The esgpull_rucio component has been specifically developed for moving CMIP data on the RUCIO base data lake.

Data-driven models for supporting the DT applications from WP4 have been implemented and supported by several thematic modules (e.g., ML TC Detection, ML4Fires, eddiesML, xtclim, downscaleML, Hython wflow_sbm). Jupyter Notebooks for testing the capabilities of the modules have also been (or will be) developed in several cases. Docker images are available for multiple components (e.g., xtclim, Delft-FIAT, WFLOW.jl, RA2CE) to enhance their portability.

To conclude, the key activities foreseen as next steps will focus on:

- Finalization of the thematic modules capabilities required to fully support environmental DT applications from WP4;



D7.7 Final version of the thematic module for the environment domain

- Strengthen and complete integration with the interTwin DTE core (WP6) and infrastructural (WP5) components.



5 References

Reference	
No	Description / Link
R1	Fiore, S., Fronza, M., Elia, D., Donno, D., Donno, E., Backeberg, B., Jacob, A., Claus, M., & Pagé, C. (2023). interTwin D7.3 First version of the thematic modules for the environment domain (1 Under EC review). Zenodo. https://doi.org/10.5281/zenodo.10224252
R2	Schramm, M., Harrison, C., Schobben, M., Elia, D., Donno, E., Claus, M., Ferrario, I., Dhinakaran, S., Backeberg, B., Tromp, W., Pagé, C., Durif, A., Vrac, M., Fronza, M., Fiore, S., Manzi, A., & Reimer, C. (2024). interTwin 7.5 Updated report on requirements and thematic modules functionalities for the environment domain (1.2). Zenodo. https://doi.org/10.5281/zenodo.13595096
R3	Alonso, L., Gans, F., Karasante, I., Ahuja, A., Prapas, I., Kondylatos, S., Papoutsis, I., Panagiotou, E., Mihail, D., Cremer, F., Weber, U., & Carvalhais, N. (2023). SeasFire Cube: A Global Dataset for Seasonal Fire Modeling in the Earth System (0.3) [Data set]. DOI: https://doi.org/10.5281/zenodo.8055879
R4	Zhou, Z., Rahman Siddiquee, M.M., Tajbakhsh, N., Liang, J. (2018). UNet++: A Nested U-Net Architecture for Medical Image Segmentation. In: Stoyanov, D., et al. Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support. DLMIA ML-CDS 2018 2018. Lecture Notes in Computer Science, vol 11045. Springer, Cham. DOI: https://doi.org/10.1007/978-3-030-00889-5_1
R5	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
R6	esgpull, https://esgf.github.io/esgf-download/
R7	Rocklin, Matthew. Dask: Parallel computation with blocked algorithms and task scheduling. In: SciPy. 2015. p. 126-132. http://conference.scipy.org.s3.amazonaws.com/proceedings/scipy2015/pdfs/matthew_rocklin.pdf
R8	Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014). DOI: https://doi.org/10.48550/arXiv.1409.1556
R9	Gao, H., Ji, S.. (2019). Graph U-Nets. In: Proceedings of the 36th International Conference on Machine Learning. https://arxiv.org/abs/1905.05178

