Funded by the European Union under the
HORIZON-EUROHPC-JU-2023-INCO-06 programme.
Grant Agreement No. 101196247

Deliverable D4.1 - Requirements and geohazard projects description

WP4 – Geophysical Hazards

Funded by
the European Union

Document Information

Deliverable Number	D4.1
Deliverable Title	Requirements and geohazard projects description
Due Date	2025-11-30 (PM10)
Deliverable Lead	GUF
Authors	Nishtha Srivastava, Natalia Zamora, Leonardo Mingari, Carlos Sánchez-Linares, Jorge Macías, Marisol Monterrubio-Velasco
Keywords	UCIS4EQ, SAIPy, Tsunami-HySEA, FALL3D
WP	WP4
Nature	Report
Dissemination Level	Public
Final Version Date	2025-11-24
Reviewed by	Dmitry Morozov (JYU), Paola Mercogliano (CMCC), Francesca Mele (CMCC), Cathrine Bergh (KTH)
MGT Board Approval	2025-11-27

Document History

Partner	Date	Comments	Version
BSC, CSIC, GUF, UMA	2025-11-05	First draft	0.1
BSC, CSIC, GUF, UMA	2025-11-24	Final version	0.2

List of acronyms

Acronym	Description
ACC	Accelerated H100
AI	Artificial Intelligence
COG	Cloud Optimized GeoTIFF
CPU	Central Processing Unit
DMP	Data Management Plan
FAIR	Findable, Accessible, Interoperable, and Reusable
FDSN	The International Federation of Digital Seismograph Networks
GPP	General Purpose Partition
GPU	Graphics Processing Unit
HPC	High Performance Computing
ML	Machine Learning
MN5	MareNostrum5
PTF	Probabilistic Tsunami Forecasting
SaaS	Software as a Service
STAC	SpatioTemporal Asset Catalog
UC	Use case / test case
WP	Work package

Executive Summary

This deliverable, D4.1, establishes the foundational technical and data requirements for Work Package 4 (Geophysical Hazards) of the EU-funded GANANA project. It consolidates the operational frameworks for four advanced geohazard modelling codes: UCIS4EQ for urgent and non-urgent earthquake simulations, SAIPY for near-real-time deep learning-based seismic data processing, Tsunami-HySEA for tsunami propagation, and FALL3D for smoke dispersion. By defining shared standards for data formats, computational workflows, and technical specifications, the document ensures the interoperability and reproducibility of these tools across diverse high-performance computing (HPC) environments, laying the groundwork for efficient cross-border research and rapid response systems. The report outlines precise input and output definitions and addresses the integration of these workflows into automated pipelines. Furthermore, the deliverable sets the stage for future developments, including the deployment of web-based simulation services and real-time monitoring platforms, ensuring that the resulting geohazard solutions are robust, scalable, and accessible to both European and Indian stakeholders for effective risk mitigation.

Table of Contents

1 Introduction

A clear and comprehensive description of the codes and the underlying data used in a project is essential for ensuring transparency, reproducibility, and long-term usability of the developed tools. Detailed documentation enables other researchers and practitioners to understand the assumptions, algorithms, and implementation choices that drive the configuration of each of the projects within WP4 and the processing and interpretation of outcomes. In particular, the explicit characterization of data, its sources, structure, preprocessing steps, and limitations is critical, as data quality and provenance directly influence model behavior and the validity of derived insights. Moreover, well-documented codes and datasets facilitate efficient collaboration across teams, support debugging and maintenance activities, and enable future extensions or integrations into larger digital frameworks. Ultimately, rigorous documentation of both code and data is a cornerstone of robust scientific and engineering workflows, ensuring that the developed solutions can be confidently used in our exchange program, audited, and adapted to the real-world applications selected in WP4 and future studies.

This document presents a unified plan for collecting the technical and data requirements across all four geohazard projects during the first nine months (PM1–PM9) of WP4 activities. The plan is aligned with the objectives of Task 4.1 under the EU–India collaboration, emphasizing common actions such as stakeholder coordination, data curation, IT compatibility, and the assessment of computational resources. It provides a structured and integrated overview of the work carried out, summarizing the general characteristics of each code, the associated requirements, the data formats used, and selected illustrative examples.

Four codes are included within WP4: (i) UCIS4EQ (Barcelona Supercomputing Center, BSC), (ii) SAIPy (Goethe University Frankfurt, GUF), (iii) TsunamiHySEA (Universidad de Málaga), and (iv) FALL3D (*Consejo Superior de Investigaciones Científicas,* CSIC). The corresponding repositories, licence information, and additional technical details for each code are presented in the following sections.

In all four cases, we will ensure compliance with the FAIR (Findable, Accessible, Interoperable, and Reusable) data management principles. Data will be shared openly whenever possible, in alignment with these principles. We will ensure to follow the Data Management plan delivered for this project (D 1.3). Exceptions will apply only in cases where data cannot be made open due to specific restrictions, primarily affecting datasets provided by Indian partners. For example, the seismic data to be obtained from the National Center of Seismology (NCS) in New Delhi is not openly distributed. To enable collaboration, C-DAC India has established a data-sharing agreement with NCS under the umbrella of the GANANA project. In such instances, appropriate measures will be taken to respect confidentiality and data ownership. Most outcomes, however, will be made available via open repositories. For all four cases, the computational resources will be hosted on HPC infrastructures available to the project, potentially including national facilities such as the Barcelona Supercomputing Center (BSC), institutional clusters, or other European HPC platforms accessible through project allocations, such as the Spanish Supercomputing Network (RES) or Euro-HPC. The final hosting configuration will be determined based on resource availability and project needs.

We present a general overview of the steps that have been completed for this deliverable D4.1, which is presented in Table 1. Some elements will be updated in month M14 following the completion of mobility programs.

Table 1. *Actions taken in relation to the code and data requirements.*

Actions	Task	Description	Participants
A1	Stakeholder coordination / discuss the code requirements and data availability with key partners and collaborators.	Kickoff meetings with Indian and EU partners to identify focal points and data/code ownership	BSC, GUF, UMA, CSIC, CDAC, NCS
A2	Data inventory and gap analysis	Collect existing datasets and models for earthquakes, tsunamis, fires; identify missing components	All partners in WP4
A3	Define data standards and formats	Agree on common formats for input/output, metadata, and ML readiness	BSC, GUF, UMA, CSIC
A4	Assess data quality and FAIR principle compliance	Evaluate data quality, curate metadata, and ensure compliance with FAIR principles, in relation to the project's data management plan WP1	All data contributors
A5	Interoperability and IT compatibility first draft - to be updated.	Compare environments (EU vs India) and ensure code/workflows can operate across platforms	BSC, GUF, UMA, CSIC, CDAC
A6	Computational resource survey	Assess HPC/Cloud resource needs per project and compatibility with partners' infrastructure	BSC, CSIC, UMA, CDAC
A7	Repository strategy	Define repository structure for static/shared data and versioned models	BSC,GUF, UMA, CSIC
A8	Preliminary runs to exchange data with Indian partners.	Run basic tests for I/O, small-scale workflows, and sample simulations with real or synthetic data	Each project lead + Indian partners
A9	Requirement documentation description (D4.1)	Compile results of the above tasks into deliverable D4.1 – Requirements and Geohazard Projects Description	GUF and BSC (lead), all partners contribute

2 Project data description in each code's pipelines

2.1 Project 1 - UCIS4EQ

2.1.1 Overview

Urgent Computing Integrated Services for Earthquakes (UCIS4EQ) is a fully automatic high-performance computing (HPC)-based system designed for rapid response during or immediately after seismic emergencies. It provides an automated seismic workflow capable of delivering fast, physics-based assessments of earthquake impacts by running large-scale 3D full-waveform simulations. These simulations yield accurate, relatively short-term ground motion estimates that help evaluate the effects of moderate to large earthquakes on infrastructure and potential secondary risks such as dam failures, fires, or landslides.

UCIS4EQ is a flagship initiative under the ChEESE Centre of Excellence and is evolving into an HPC Workflow as a Service (HPCWaaS). It integrates advanced, massively parallel solvers to deliver results within minutes to hours, depending on the availability of the HPC resources. The workflow accounts for key sources of uncertainty, including poorly constrained earthquake source parameters and unresolved local soil effects that can amplify or dampen ground motion. In addition to its functional capabilities, UCIS4EQ is being developed to meet essential non-functional requirements, such as robustness, interoperability, availability, portability, and maintainability. Each component functions as a containerized microservice, ready for cloud deployment. In the broader context of seismic workflows or applications, UCIS4EQ generates synthetic shaking maps for key seismic sources across different time scales. As new events are detected, source parameters are retrieved or computed, and updated shaking maps are added to a growing repository. The system supports both urgent simulations, where speed is critical and uncertainties are high, and non-urgent simulations, where multiple source scenarios can be explored and compared (Ejarque et al. 2024[1]).

UCIS4EQ is deployed on a local server - once activated, the workflow launches in parallel multiple jobs on high-performance computing (HPC) clusters. The current implementation operates on MareNostrum5, using the Salvus code (a suite of software

[1] Ejarque, J, Badia, R.M., Albertin, L., Aloisio, G. et al. (2022).. "Enabling dynamic and intelligent workflows for HPC, data analytics, and AI convergence." Future generation computer systems 134: 414-429.

for performing full waveform modelling and inversion, employing the spectral element method).[2] Job management is handled through PyCOMPSs[3], which provides flexibility and malleability in workflow execution.

To begin with, the study area must belong to the predefined pool of regions that have already been processed and made available for the workflow. Parameters related to time sensitivity, such as maximum update frequency, number of ensemble members, and similar factors, should be determined in advance. The processing workflow (Figure 1) is fully automated and triggered by the detection of a new event, yet the final delivery of results remains a manual step.

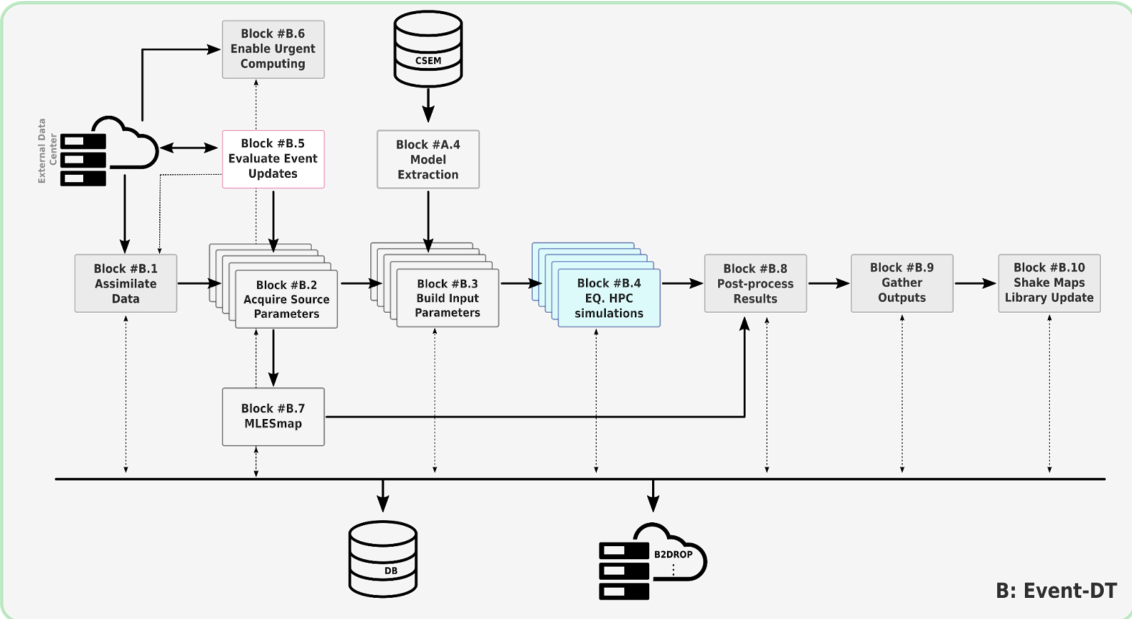


Figure 1. *Building Blocks conforming to the UCIS4EQ pipeline. B. An event that triggers each of the microservices of the urgent computation (UCIS4EQ).*

2.1.2 Code Repository, Input Data and Output

Slipgen and Salvus (Table 2) are the two main codes used to obtain synthetic seismograms and the different intensity measures such as pseudo-spectral acceleration, peak ground acceleration, peak ground velocities. The whole pipeline can be found in eFlows project github, however new features during this project may be moved to other repositories.

---

[2] Afanasiev, M., Boehm, C., van Driel, M., Krischer, L., May, D., Rietmann, M., & Fichtner, A. (2017, April). Salvus: a flexible high-performance and open-source package for waveform modelling and inversion from laboratory to global scales. In EGU General Assembly Conference Abstracts (p. 9456).

[3] Foyer, C., Conejero, J., Ejarque, J., Badia, R. M., Tate, A., & McIntosh-Smith, S. (2020, November). Enabling system wide shared memory for performance improvement in PyCOMPSs applications. In 2020 IEEE/ACM 9th Workshop on Python for High-Performance and Scientific Computing (PyHPC) (pp. 22-31).

Current repository

The project is hosted on GitHub at the following link: https://github.com/eflows4hpc/ucis4eq. This repository is publicly accessible, though additional developments and updates may be available upon direct communication with the team.

Table 2: *Summary of main simulation codes UCIS4EQ and the input data needed.*

Main codes	Inputs / Description
SLIP_GEN	Generates kinematic rupture (slipgen microservice) - Magnitude, hypocenter; - Velocity model (Thickness, Vp, Vs, Rho, Qp, Qs); - GPsetup.json: Graves-Pitarka setup parameters
SALVUS	Generates seismic wave propagation (use slipgen microservice) - Receiver/seismic station locations (real or synthetic); - 3D seismic velocity model; - Topo-bathymetry; - Vs30 layer

Model input and output data

To achieve the complete pipeline of UCIS4EQ, each region must follow a structure defined in DataStructure.json. The data inputs needed are listed in Table 3.

The project provides various output formats to accommodate different types of data and results. These include NETCDF files, which contain tabular data summaries and processed results. PNG and HTML files are used to store generated images and visualizations, such as maps and plots, providing a clear graphical representation of the data. Additionally, HDF5 files are available for high-resolution waveform data and model outputs, particularly for seismograms. Each format serves a specific purpose, making it easier to work with and analyze the results in a variety of ways. For instance, information of real ground motion (waveforms) for comparisons .

Table 3: *Summary of data formats inputs and outputs for UCIS4EQ.*

Data Type	Format(s)	Description / Purpose
Inputs	netCDF (HDF5)	Structured scientific data input used for model initialization and results integration.
	HTML	Input from web-based interfaces or metadata pages.
	Binary raw data	Low-level sensor or instrument recordings.
	JSON	Configuration files, event metadata, and workflow parameters.
Outputs	NETCDF	Tabular data summaries and processed results.
	PNG	Generated images and visualizations (e.g., maps, plots).
	HDF5 (seismograms)	High-resolution waveform data and model outputs.

2.1.3 A test case of earthquake simulation

Here, preliminary results using the SALVUS code which is used to validate the whole UCIS4EQ pipeline and prepare the use cases for the exchange program (Figure 3). These are preliminary results; future work will incorporate comparisons with real data and empirical ground motion models for the region, for at least two recorded earthquakes in the region. The outcomes of these simulations will be used by the Indian counterpart to simulate earthquake-induced landslides.

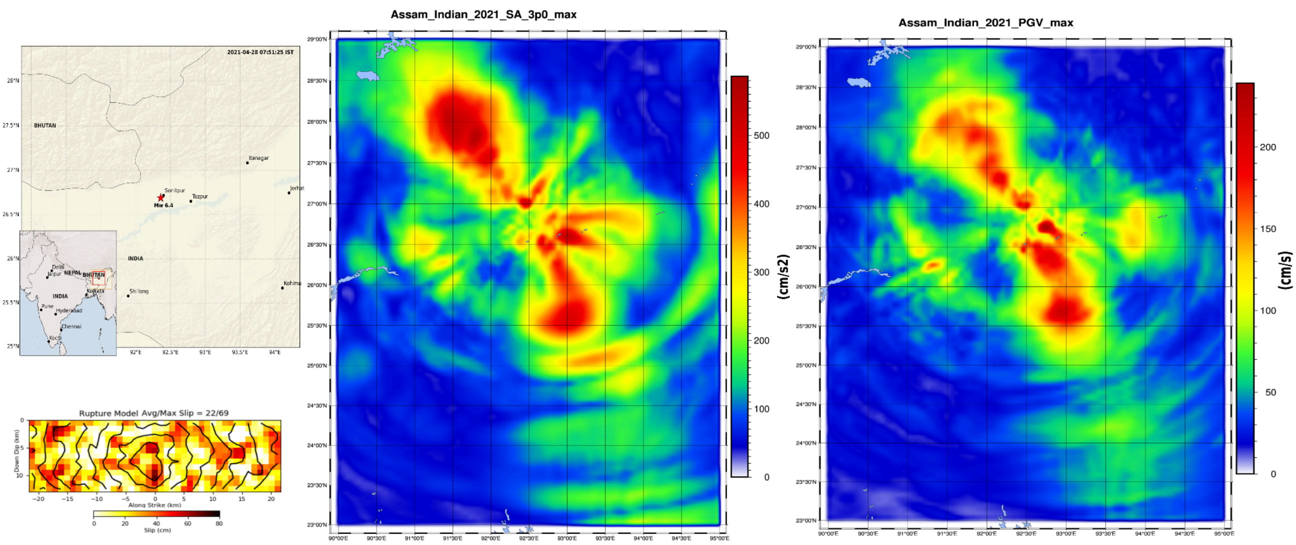


Figure 2. *Example of the Assam 2021 Mw 6.4 earthquake. Upper panel: Hypocenter location. Lower panel: Kinematic rupture (resulted from slipgen) is an input for the wave propagation modeling (Salvus code). Mid panel: example of the pseudo spectral acceleration at 3s. Right panel: example of the peak ground velocity.*

2.1.4 Deployment and Supported Platforms

The deployment architecture of UCIS4EQ consists of a hybrid setup involving both local servers and High-Performance Computing (HPC) infrastructure. The system is designed to be deployed on a local server—such as a laptop, workstation, or virtual machine—which hosts the listening service and orchestrates the execution of processes triggered by earthquake detections from FDSN agencies.

Once activated, the workflow automatically launches multiple jobs in parallel on HPC clusters. The current implementation operates on MareNostrum5 (MN5), utilizing specific partitions for different stages of the workflow:

- Pre- and Post-processing: These tasks are submitted to the GPP (General Purpose) partition.
- Simulations: The computationally intensive 3D full waveform simulations are executed on the ACC (Accelerated H100) partition using the Salvus code.

Job management within the system is handled through PyCOMPSs, which provides flexibility and malleability in workflow execution, along with fault tolerance mechanisms. While the previous deployment on MareNostrum4 efficiently utilized approximately 2,000 nodes, the current setup on MN5 leverages H100 GPUs, allowing a few dozen accelerated nodes to achieve comparable performance. Additionally, each component of UCIS4EQ functions as a containerized micro-service, making the system ready for cloud deployment.

2.1.5 Future Developments

Future developments for UCIS4EQ focus on the validation and practical application of the simulation pipeline. While preliminary results using the SALVUS code have been used to validate the current pipeline, future work will incorporate rigorous comparisons with real data for at least two recorded earthquakes in the region.

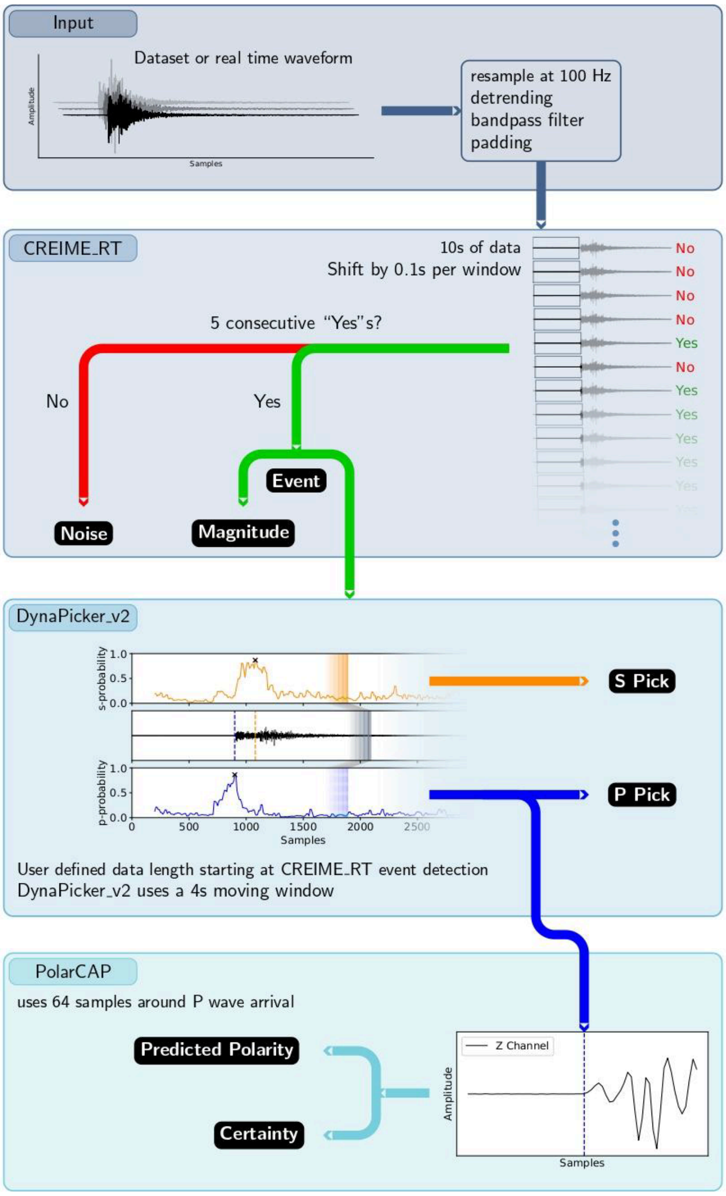
Furthermore, the outcomes of these simulations are intended to be utilized by the Indian counterpart to simulate earthquake-induced landslides. As the project progresses, new features developed may be moved to other repositories to support these evolving capabilities.

2.2 Project 2 - SAIPy

2.2.1 Overview

SAIPy is an open-source Python package designed for fast, automated processing of seismic waveform data using deep learning for single-station earthquake monitoring, offering a robust suite of deep learning solutions for various seismological tasks. It provides comprehensive capabilities for (i) earthquake signal detection, (ii) seismic phase picking, (iii) first motion polarity identification, and (iv) magnitude estimation. SAIPy introduces the first fully automated deep learning-based pipeline for processing continuous waveforms, streamlining the entire monitoring workflow from event detection to the estimation of P- and S-arrival times, magnitudes, and first-motion polarity. This makes SAIPy a powerful tool for real-time earthquake monitoring, enabling timely actions to mitigate seismic impacts. Its user-friendly API simplifies the integration of advanced models and benchmark datasets, and its inherent adaptability allows for continuous retraining with new data, ensuring improved performance and versatility across different regions and geological contexts.

SAIPy is designed with time-critical applications in mind and has the potential for real-time monitoring and prompt decision-making. The package also features a fully automated pipeline for seamless processing of continuous waveform data, reducing the need for manual intervention and ensuring streamlined operation.



s

Figure 3. *Schematic diagram for SAIPy. The input (continuous waveform or sample from a dataset) is resampled and preprocessed. Then CREIME_RT uses 10 s windows shifted by 0.1 to detect earthquake signals. Once five consecutive windows are found to contain an earthquake signal, the magnitude will be determined and an event trigger is placed. As long as there are no 5 consecutive windows found to contain earthquakes, the trace is considered noise. An earthquake signal is processed with DynaPicker_v2 in the next step. There, moving 4 s windows are used to determine the P- and S- wave arrivals. The end of the 4 s moving window is shown by the vertical line in the middle panel of the DynaPicker_v2 box. As a final step, the P-pick is used to select the Z-Channel data of 31 samples before and 32 samples after, which is then fed to PolarCAP for polarity prediction. All model outputs are marked in black boxes.*

Within this project, SAIPy will undergo further development to evolve into a prototype for real-time earthquake monitoring. Additionally, it will be dockerized to ensure compatibility with UCIS4EQ. As part of this project, SAIPy will be tested, validated, and fine-tuned using seismic data from the Indian subcontinent to ensure optimal performance and compatibility with regional geological conditions. The goal is to adapt SAIPy's deep learning models for robust earthquake monitoring tailored to India's diverse seismic zones.

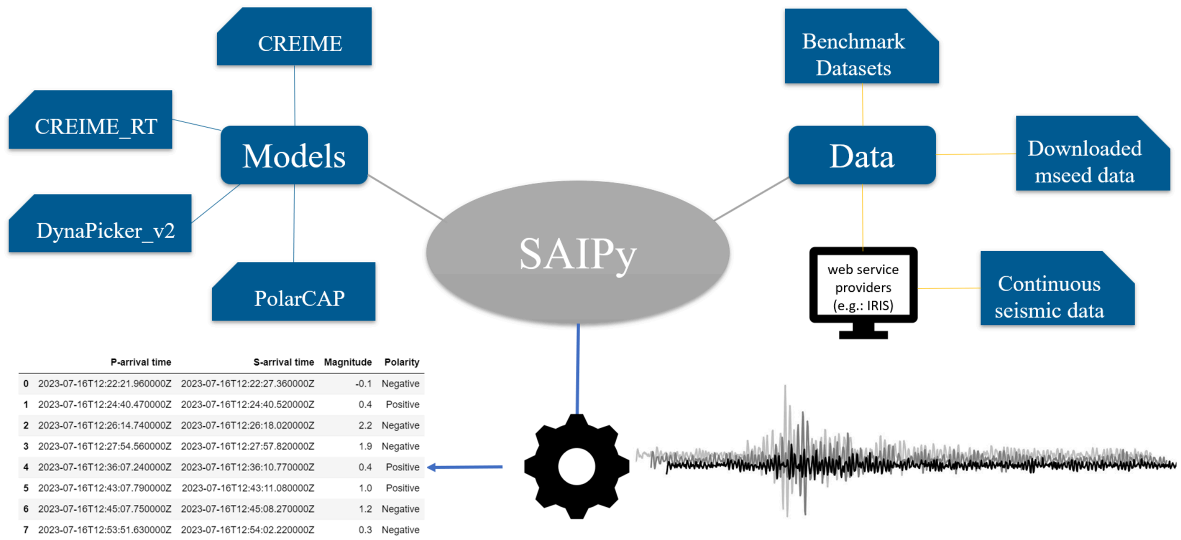
2.2.2 Code Repository, Input Data and Output

A detailed documentation and the code are openly available in Github - https://github.com/srivastavaresearchgroup/SAIPy.

The repository can be cloned using the instructions provided in the documentation, allowing the package to be easily set up and run on a local system.

The SAIPy package supports data in two primary formats: NumPy arrays and formats handled by the ObsPy package, which typically includes standard seismological formats like MiniSEED. The package generates an output catalogue in CSV format, containing detailed information for each earthquake event, including magnitude, P- and S-wave arrival times, and first-motion polarity.

The package generates an output catalogue in CSV format, containing detailed information for each earthquake event, including magnitude, P- and S-wave arrival times, and first-motion polarity.



	P-arrival time	S-arrival time	Magnitude	Polarity
0	2023-07-16T12:22:21.960000Z	2023-07-16T12:22:27.360000Z	-0.1	Negative
1	2023-07-16T12:24:40.470000Z	2023-07-16T12:24:40.520000Z	0.4	Positive
2	2023-07-16T12:26:14.740000Z	2023-07-16T12:26:18.020000Z	2.2	Negative
3	2023-07-16T12:27:54.560000Z	2023-07-16T12:27:57.820000Z	1.9	Negative
4	2023-07-16T12:36:07.240000Z	2023-07-16T12:36:10.770000Z	0.4	Positive
5	2023-07-16T12:43:07.790000Z	2023-07-16T12:43:11.080000Z	1.0	Positive
6	2023-07-16T12:45:07.750000Z	2023-07-16T12:45:08.270000Z	1.2	Negative
7	2023-07-16T12:53:51.630000Z	2023-07-16T12:54:02.220000Z	0.3	Negative

Figure 4. *SAIPy takes seismic data (benchmark datasets, mseed files, or continuous data streams) as input and processes it to automatically generate an output table of earthquake parameters.*

Model input and output data

The following data inputs are essential:

1. Seismic Waveform Data:
    - Waveforms from both single and multiple stations for earthquakes greater than 3 magnitude.
    - Format: Raw counts
    - Sampling rate: 100 Hz.
    - Pre-processing: Bandpass filtered between 1–45 Hz.

2. Event Catalogue:
    - A corresponding catalogue containing information for each earthquake event, including:
        - Magnitude
        - P- and S-wave arrival times
        - First-motion polarity
        - Fault type (if available)

2.2.3 A test case of near real time earthquake data processing

An example of the application of the SAIPy package to the aftershock sequences of the 2023 Turkey earthquake recorded at CEYT (Ceyhan, Adana, Turkey) station belonging to the Kandilli Observatory And Earthquake Research Institute (KOERI) network is shown in Figure 4. SAIPy detects and satisfactorily estimates the magnitude of the events in the catalogue. It is a known challenge to analyze low magnitude earthquakes that follow higher magnitude ones closely in time, and even more so if there is a poor covering of seismic monitoring stations surrounding or the quality of data is not good enough. Thus, these earthquakes are not included in the earthquake catalogues. In that regard, SAIPy has also demonstrated the ability to detect such earthquakes that have not previously been analyzed and included in earthquake catalogues by seismological agencies. The zoomed in plots in Figure 4 show two such examples of detections that appear to be seismic events.

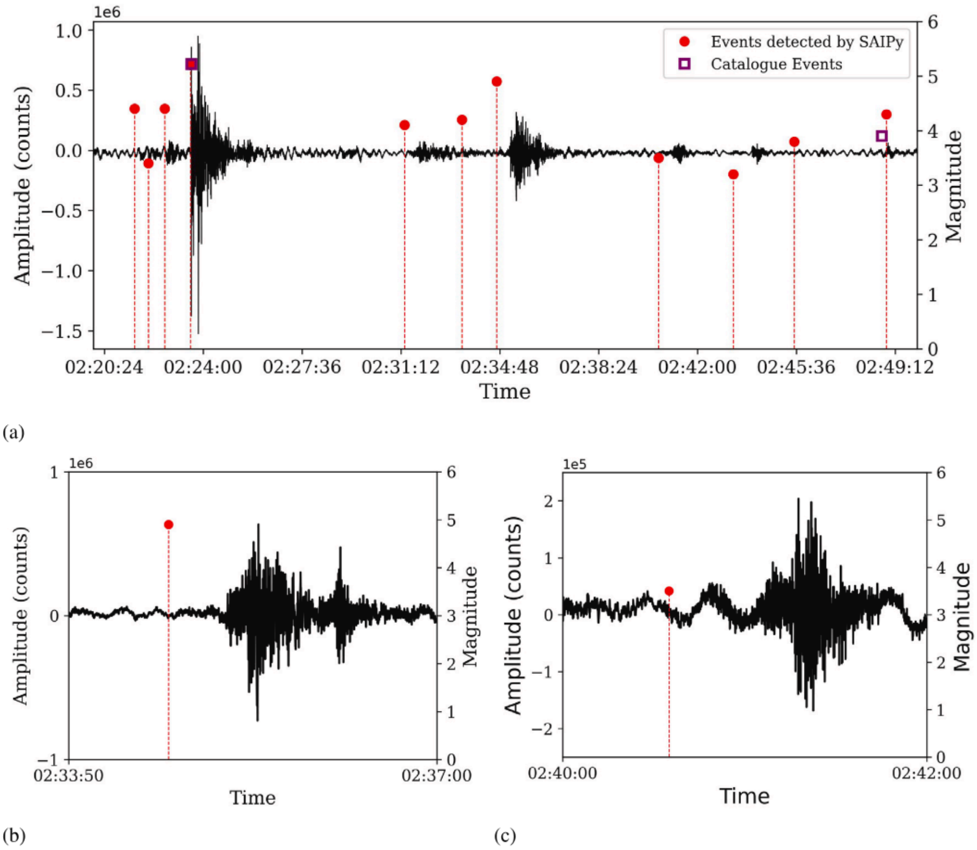


Figure 5. *Predictions of the SAIPy package on one of the aftershock sequences of the 2023 Turkey earthquake downloaded from the station CEYT (Ceyhan, Adana, Turkey) belonging to the Kandilli Observatory And Earthquake Research Institute (KOERI) network. As one can see from the figure SAIPy detects and provides a satisfactory magnitude estimate for the events present in the catalogue. Upon zooming in we notice that SAIPy also seems to detect uncatalogued events.*

2.2.4 Deployment and Supported Platforms

SAIPy is designed to support time-critical applications with a focus on efficiency and portability. The system runs efficiently on standard CPU-based systems, making it accessible for various computing environments. Additionally, it possesses the capability to leverage GPUs for faster processing, which is beneficial for high-throughput tasks.

To facilitate the handling of massive datasets, such as continuous waveforms from large-scale regional networks, SAIPy will be designed to be containerized (e.g., via Docker). This will ensure that the software can function as a portable microservice, allowing for seamless integration and streamlined operations across different platforms.

2.2.5 Future Developments

Future developments for SAIPy within the context of this project aim to evolve the software into a fully operational prototype for real-time earthquake monitoring. A key step in this process involves dockerization to ensure compatibility with the UCIS4EQ pipeline.

Furthermore, the project will focus on the adaptation and validation of SAIPy for specific regional needs. The system will be tested, validated, and fine-tuned using seismic data from the Indian subcontinent. A key future step involves coupling SAIPy outputs with UCIS4EQ framework and Landslide modelling developed by the Indian counterpart. The ultimate goal is to adapt SAIPy's deep learning models to ensure robust earthquake monitoring that is specifically tailored to India's diverse seismic zones.

2.3 Project 3 - Tsunami-HySEA

2.3.1 Overview

Tsunami-HySEA is a high-performance computing (HPC) system developed for the rapid and accurate simulation of tsunami generation, propagation, and coastal inundation. It provides an integrated, automated workflow designed to support both real-time early warning operations and high-resolution hazard assessment studies. Based on advanced numerical solvers for the nonlinear shallow water equations, Tsunami-HySEA efficiently models large-scale tsunami events using massively parallel GPU architectures, delivering results within minutes after an event is triggered.

As a core component of the ChEESE Centre of Excellence, Tsunami-HySEA plays a central role in the development of Urgent Computing services for tsunami forecasting. The system enables the Probabilistic Tsunami Forecasting (PTF) framework, in which thousands of simulations are executed in parallel to account for uncertainties in earthquake source parameters—such as fault geometry, slip distribution, or rupture depth—providing ensemble-based impact predictions. These probabilistic outputs are key to estimating tsunami arrival times, maximum wave heights, and coastal flooding extents, assisting civil protection authorities in making informed decisions during emergencies.

The Tsunami-HySEA workflow integrates all stages of tsunami simulation, from scenario definition to hazard analysis. It begins with the scenario parametrization, where the user or an automated system defines the seismic source parameters (e.g., magnitude, location, and fault mechanism) and computational grids. These inputs are used to generate the initial conditions, typically a co-seismic seabed deformation field. The model then performs the numerical simulation of tsunami propagation and inundation on multiple GPUs, efficiently distributing the computational load. The outputs include both 1D time series (at virtual buoy locations) and 2D spatial fields such as maximum water height, velocities, arrival times, and momentum flux. Finally, the tsunami analysis stage derives key hazard indicators—like maximum inundation height, run-up, and inundation distance—based on the simulated results.

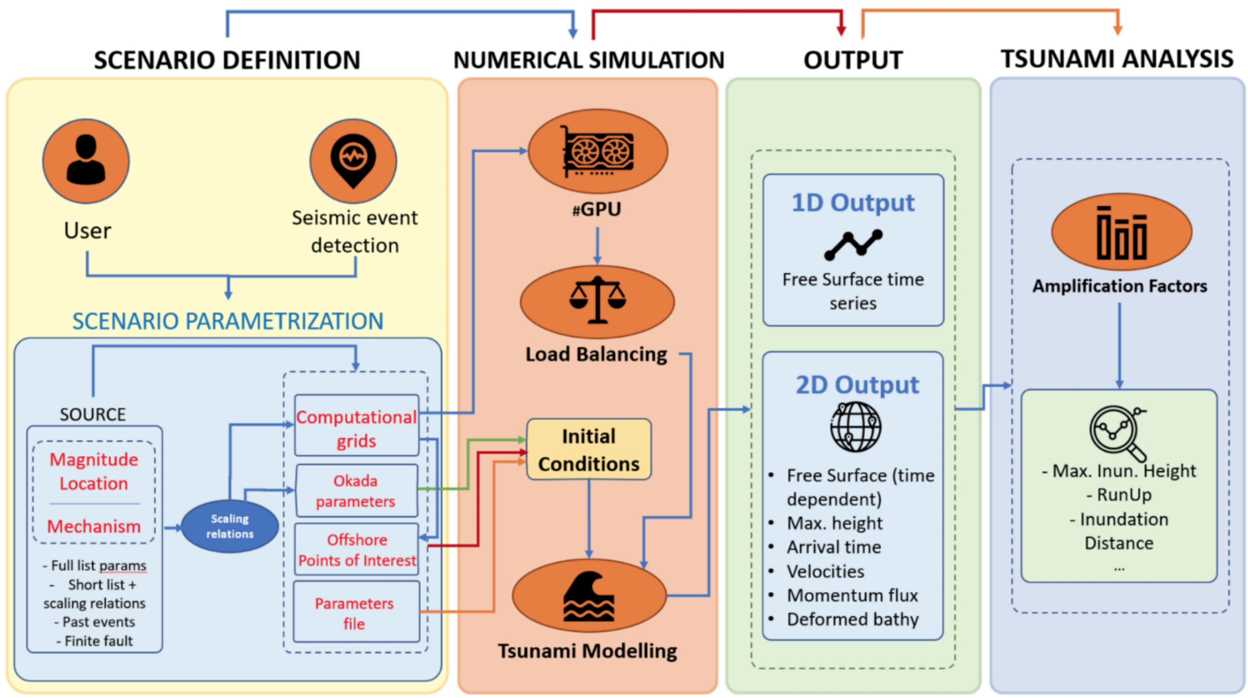


Figure 6. *General workflow of the Tsunami-HySEA modelling system.*

2.3.2 A test case of tsunami simulation

The objective of this test case is to prepare the first stage of a full workflow that integrates tsunami simulations with aquifer-impact modelling, the latter being developed by our Indian counterpart. We employ data from the historical 1755 tsunami affecting the southwestern coast of Spain (Huelva and Cádiz along the Atlantic façade of Andalusia) not due to the availability of observations, but because high-resolution bathymetry and topography are available for this region, enabling robust and reliable simulations. The outputs derived from these simulations will serve as boundary and forcing conditions for the aquifer model developed by our Indian partner. This reference case in the Spanish coast allows us to accelerate workflow development, since obtaining equivalent datasets for a selected location in India would significantly delay the process. Our goal is to deliver a fully operational and validated workflow using accurate topo-bathymetric information in a shorter period of time, that can be then straightforward apply to scenarios in India .

Model input and output data

The test case setup for Tsunami-HySEA requires a set of well-defined input datasets describing both the tsunami source and the computational domain. These inputs include: (i) the bathymetry and topography model defining the seabed and coastal relief; (ii) the initial sea surface deformation, derived from earthquake fault parameters such as location, strike, dip, slip, and rupture area; and (iii) numerical configuration files specifying spatial resolution, output frequency, and boundary conditions. For this experiment, all input data were preprocessed to ensure grid consistency and numerical stability. The simulation was executed on an HPC environment, using GPU-accelerated solvers to reproduce tsunami generation, propagation, and coastal inundation processes with high spatial accuracy.

The present test focuses on the 1755 Lisbon earthquake and the resulting tsunami impact on the coast of Chipiona (Cádiz, west-south of Spain). The simulation was performed using Tsunami-HySEA on a uniform grid with a 5-meter spatial resolution. The objective of this test case is to evaluate the local inundation patterns and wave dynamics near the coastline, using both gridded and point-based outputs derived from the numerical simulation.
The output data files are two main NetCDF files: [file].nc (containing spatial grid data for maximum heights and bathymetry) and [file]_ts.nc (containing time series data at virtual buoy locations). Additionally, buoy positions (longitude, latitude) are defined in a text file [pois_file].txt.

Table 4. *NetCDF variables of the gridded tsunami output.*

Variable	Dimensions	Units	Description
max_height	[lat, lon]	m	Maximum water surface elevation reached during the simulation
deformed_bathy	[lat, lon]	m	Seabed depth after seismic deformation (negative values indicate below MSL)
lat	[lat]	°	Latitude values of the computational grid
lon	[lon]	°	Longitude values of the computational grid

Table 5. *NetCDF variables of the time series output.*

Variable	Dimensions	Units	Description
time	[time]	s	Time steps since the beginning of the simulation
eta	[time, point]	m	Water surface elevation over time at each buoy
max_height	[point]	m	Maximum water surface elevation at each buoy during the event

In addition to the maximum water height (max_height), Tsunami-HySEA can generate a range of additional output variables depending on the simulation configuration. These may include the free surface elevation (eta) and the horizontal velocity components (u, v) of the flow field, when enabled in the model setup. Furthermore, derived quantities such as flow discharge, momentum flux, or energy flux can be computed if required for further analysis.

To ensure consistency between the simulation outputs and the downstream models or analyses, it is important to define in advance the preferred set of variables, along with their temporal and spatial resolutions, and any specific formatting or preprocessing requirements. Once these parameters are established, the corresponding datasets can be prepared and shared for verification, allowing assessment of their suitability for the intended modelling applications.

In general, the input data required by Tsunami-HySEA—such as bathymetry, topography, and seismic source parameters—are obtained from well-established international databases. Typical sources include GEBCO or EMODnet for bathymetric grids, SRTM or Copernicus DEM for topography, and global earthquake catalogues (e.g., USGS, GCMT, or regional seismic agencies) for fault mechanisms and event metadata. These datasets are then preprocessed to match the expected model resolution, coordinate system, nesting structure, and metadata conventions required by the simulation workflow.

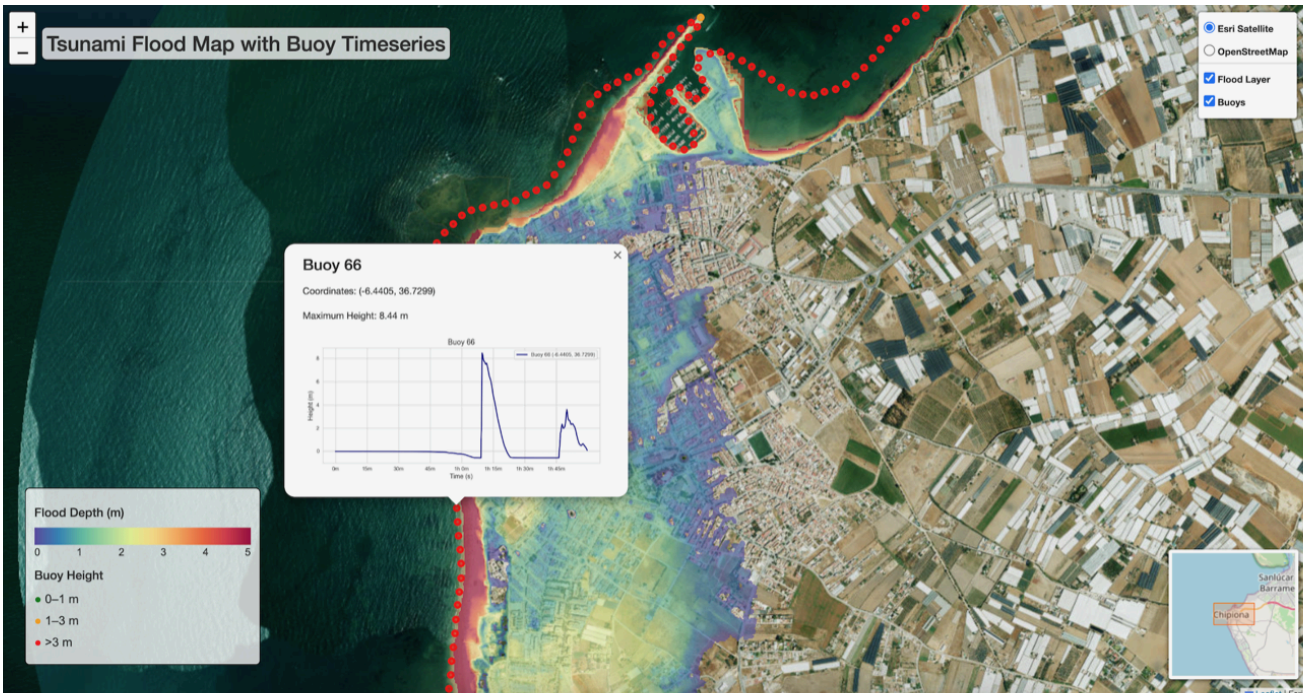


Figure 7. *Tsunami inundation map and virtual buoy time series for the 1755 Lisbon earthquake scenario near Chipiona (Cádiz, Spain). The colored flood layer represents simulated inundation depth derived from the maximum water height field in the gridded NetCDF output. Red markers indicate virtual buoy locations defined along the -1 m isobath, where corresponding time series of water surface elevation are extracted from the second NetCDF file. The example shown for Buoy 66 illustrates the temporal evolution of the tsunami wave height, with a recorded maximum amplitude of 8.44 m.*

All simulation datasets follow FAIR data principles. Metadata and variable descriptions are included in each NetCDF file, ensuring interoperability and reuse. Data is stored in NetCDF format and will be accessed upon request through a data repository and it will be hosted on a long-term, citable research data platform (e.g., Zenodo or an institutional repository at the University of Málaga).

2.3.3 Simulation Platform for Tsunami-HySEA as a Web Service

Within the project context, a future direction involves the deployment of a web-based simulation platform for Tsunami-HySEA, enabling users (both within Europe and the Indian counterpart institutions) to launch, monitor, and retrieve tsunami simulations through an intuitive user interface. The platform is built on the existing HySEA Platform developed by the EDANYA Research Group at the University of Málaga, which provides virtual-access (SaaS) to tsunami models.

The platform will support the definition of scenario parameters (e.g., seismic source geometry, slip, magnitude) and domain configuration (bathymetry/ topography, grid resolution) through a web interface. Users will be able to submit simulation jobs that will run on GPU-accelerated HPC resources, leveraging the multi-GPU, faster-than-real-time (FTRT) implementation of Tsunami-HySEA. These computational resources will be hosted on HPC infrastructures available to the project, potentially including national facilities such as the Barcelona Supercomputing Center (BSC), institutional clusters at the University of Málaga, or other European HPC platforms accessible through project allocations such as RES or Euro-HPC. The final hosting configuration will be determined based on resource availability and project needs.

Output datasets (NetCDF gridded fields, buoy time-series, inundation maps) will be generated and made available for download or visualization directly via the platform.

The service will include user authentication, job monitoring, and data management, thereby facilitating use by non-expert users while maintaining advanced configurability for experienced users.

This platform offers several strategic advantages:

- Interoperability: Europeans and Indian partners can run the same model version via the same service, ensuring consistency of simulation workflows.
- Accessibility: Users without local HPC infrastructure can leverage remote GPU resources, lowering entry barriers.
- Scalability: The service can handle ensemble simulations and nested domain scenarios, aligning with the project's need for scenario exploration and uncertainty quantification.

The following tasks are proposed for this platform:

- User interface specification: Define the input parameter forms, scenario templates, and default grids for the Indian coastline use cases.
- Data ingestion and management: Set up standardised input datasets (bathymetry, topography, source libraries).
- Access and authentication: Implement user roles, project workspaces, and data access policies consistent with data sharing agreements between partners.
- Training and documentation: Prepare user guides and training workshops for Indian and European partners to ensure effective adoption of the platform.

In the future, this web-based simulation platform will play a central role in democratizing access to high-performance tsunami modelling, enhancing collaboration, reproducibility, and rapid decision support within the GANANA geohazard framework.

2.3.4 Deployment and Supported Platforms

Tsunami-HySEA is explicitly designed for High-Performance Computing (HPC) environments, leveraging massively parallel GPU architectures to perform simulations. The system's deployment is flexible and capable of utilizing various infrastructures available to the project, including:

- National Facilities: Such as the Barcelona Supercomputing Center (BSC).
- Institutional Clusters: Local clusters available at the University of Málaga.
- European HPC Platforms: Resources accessible through project allocations like RES or Euro-HPC.

The final hosting configuration will be selected based on resource availability and specific project needs. Additionally, the system currently supports a Virtual-Access (SaaS) model through the existing HySEA Platform developed by the EDANYA Research Group, which allows for remote execution on GPU resources.

2.3.5 Future Developments

The primary future development for Tsunami-HySEA within the project is the deployment of a web-based simulation platform. This service will enable both European and Indian partners to launch, monitor, and retrieve tsunami simulations through an intuitive user interface without needing local HPC infrastructure. Key planned tasks for this platform include:

1. User Interface Specification: Defining input forms, scenario templates, and default grids for Indian coastline use cases.
2. Data Ingestion: Setting up standardized input datasets for bathymetry, topography, and source libraries.
3. Access and Authentication: Implementing user roles and data access policies aligned with project sharing agreements.
4. Training: Preparing guides and workshops to ensure effective adoption by all partners.

In terms of scientific workflow integration, a key future step involves coupling Tsunami-HySEA outputs with aquifer-impact modelling developed by the Indian counterpart. To accelerate this development, the team is currently using a reference case on the Spanish coast (1755 Lisbon tsunami) to validate the workflow before applying it to Indian scenarios.

2.4 Project 4 - FALL3D

2.4.1 Overview

This project aims at developing a forest fire and associated aerosol dispersion modelling framework capable of being deployed as an operational service to forest fire spread and smoke atmospheric dispersion. The framework will be based on the coupling of two scientific models:

- **WRF-SFIRE** is a coupled weather-wildfire spread forecasting model developed on top of WRF model. The rate of fire spread is estimated based on the quasi empirical Rothermel model, whereas, level set method is used for the propagation of wildfire. Considering the model C-DAC has developed an operational SWFMS (Sikkim Wildfire Spread Forecasting and Monitoring System) for the state of Sikkim, India. The SWFMS helps Forest Department, Sikkim to carry out fire spread forecasting up-to next 48 hours, visualise the same in GIS dashboard on google earth and send the SMS alerts in the specified buffer region to the registered stakeholders. The Forest and Environment Department (Govt. of Sikkim) used the system for two live fires in 2023 and 2024 and found the system useful in effective resource deployment beforehand.
  Repository: https://github.com/openwfm/wrf-fire/
- **FALL3D** is an open-source model for atmospheric passive transport and deposition of particles, aerosols, and radionuclides based on the so-called Advection Diffusion Sedimentation (ADS) equation. The model was originally developed for inert volcanic particles (tephra) and has a track record of 100+ publications on different model applications and code validation, as well as an ever-growing community of users worldwide, including academia, private, research, and several institutions tasked with operational forecast of volcanic ash clouds. During the GANANA project, FALL3D will be enhanced by incorporating a new smoke dispersion module capable of being efficiently integrated into an operational service framework by exploiting HPC resources.
  Repository: https://gitlab.com/fall3d-suite/fall3d

2.4.2 Model Requirements and Output Products

Table 5 lists the most important input variables required by the workflow as well end-user products. Both WRF-SFire and FALL3D require meteorological data from external dataset such as ERA5 reanalysis, GFS forecasts or WRF-ARW simulations. In addition, WRF-SFire requires high-resolution (~30 m) static data for topography, landuse, soil type and fuel maps.

Outputs from the wildfire/smoke dispersion system will be stored in netCDF and geoTIFF formats along with metadata and associated documentation. Output products in netCDF format will include both global and variable attributes holding metadata about scientific data and file properties. GeoTIFF files will be collected as Cloud Optimized GeoTIFF (COG) files within a STAC catalog, short for SpatioTemporal Asset Catalog, allowing a standardized way to describe and organize geospatial data. Documentation about the simulations leading to these products will be reported in multiple log files with different running information, such as computing time, errors and warnings. Similarly, simulation parameters will be gathered in json files which are automatically generated by the models.

WRF-SFire will produce a WRF-like file in netCDF format with the area of the wildfire. This file will be ingested by FALL3D in order to produce a smoke spread forecast. In addition, FALL3D requires some data from the atmospheric model in order to define the smoke source term.

Table 7. *List of output and intermediate products required by the workflow based on the coupling between WRF-SFire and FALL3D models.*

Product	Format	Type	Resolution	Provided by	Required by
Meteorological data (ERA5, GFS…)	GRIB	Wind, temperatur e, etc..	~25 km	NCEP, ECMWF (External datasets)	WRF-SFire FALL3D
Static fuel data	geoTIFF	Gridded data of fuel types	~30 m	Copernicus (estimated from remote sensing data)	WRF-SFire FALL3D
Fire area, fire heat flux, fuel remaining, …	netCDF (WRF-like format)	Categorical (Yes/No)	1h / 40m	WRF-SFire	FALL3D
Wind field	netCDF (WRF-like format)	U- and V- wind components	1h / 40m	WRF-SFire	FALL3D

		at 10 meters			
Particulate Matter (PM) PM2.5 PM10 Black carbon	netCDF / GeoTiff	Concentrati on	1h, ~1-10 km	FALL3D	Final user
Carbon monoxide (CO)	netCDF / GeoTiff	Concentrati on	1h, ~1-10 km	FALL3D	Final user
Nitrogen oxides (NOx) NO2	netCDF / GeoTiff	Concentrati on	1h, ~1-10 km	FALL3D	Final user

The file size for a typical output of WRF-SFire is around 8Gb for every time step. However, FALL3D requires just a small subset of variables including Fire area, fire heat flux, fuel remaining, and vertical profiles of wind and temperature . The following figure shows a summary of the WRF-SFire file structure and some of the most relevant variables:

xarray.Dataset

► Dimensions: (Time: 1, south_north_subgrid: 6160, west_east_subgrid: 7240, south_north: 153, west_east: 180)

▼ Coordinates:

XLAT	(Time, south_north, west_east)	float32	..
XLONG	(Time, south_north, west_east)	float32	..
XTIME	(Time)	datetime64[ns]	..

▼ Data variables:

NFUEL_CAT	(Time, south_north_subgrid, west_east_su...	float32	1
	FieldType : 104 MemoryOrd... XY description : fuel data units : - stagger : Z		
FIRE_AREA	(Time, south_north_subgrid, west_east_su...	float32	0
	FieldType : 104 MemoryOrd... XY description : fraction of cell area on fire units : 1 stagger : Z		
FIRE_HFX	(Time, south_north_subgrid, west_east_su...	float32	0
	FieldType : 104 MemoryOrd... XY description : observed fire heat flux units : W/m^2 stagger : Z		
FUEL_FRAC	(Time, south_north_subgrid, west_east_su...	float32	1
	FieldType : 104 MemoryOrd... XY description : fuel remaining units : 1 stagger : Z		
U10	(Time, south_north, west_east)	float32	4
V10	(Time, south_north, west_east)	float32	-‹

► Indexes: (0)

► Attributes: (132)

Figure 8. *File structure and list of most relevant variables for a WRF-SFire output file. This information is required by the FALL3D model in order to simulate the smoke dispersion in the atmosphere.*

Meteorological input data required by FALL3D must be provided in netCDF format as well. However, most meteorological datasets available out there are typically in GRIB format.

The FALL3D repository (https://gitlab.com/fall3d-suite/fall3d) provides a complete set of tools for format conversion and pre-processing for data from GFS/GEFS global forecasts, ERA5 and CARRA reanalyses, and the WRF-ARW mesoscale model.

2.4.3 Deployment and Supported Platforms

The FALL3D project aims to develop a forest fire and associated aerosol dispersion modelling framework designed to be deployed as an operational service. The framework relies on the coupling between two scientific models: WRF-SFIRE and FALL3D. This system is designed to exploit High-Performance Computing (HPC) resources, ensuring that the service can be used for operational use.

FALL3D runs on modern CPU clusters and on both NVIDIA- and AMD-accelerated GPU partitions. It has been optimized and tested for the major European pre-exascale machines—LUMI (AMD Instinct GPUs), LEONARDO (NVIDIA GPU booster) and MareNostrum 5 (NVIDIA Hopper/H100 GPUs)—and ships with build targets, container images, and job scripts to make deployment on those systems straightforward.

On the other hand, **WRF** model runs efficiently across a wide range of HPC systems, from conventional CPU-only clusters to modern heterogeneous architectures. It supports large-scale parallelism through MPI and optional OpenMP and can be deployed in supercomputer clusters. However, the WRF model does not utilize GPU or any other accelerator technology.

2.4.4 Future Developments

Future developments for FALL3D within the GANANA project focus on enhancing the model by incorporating a new smoke dispersion module. This module is specifically designed to be efficiently integrated into an operational service framework by leveraging HPC resources.

Implementing a smoke-dispersion module within the FALL3D model requires developments across physics, numerics and data handling. At the physical level, the module must represent fire emissions by defining time-dependent source terms based on fire radiative power, fuel load, combustion efficiency, and plume rise. In addition, this module must also define aerosol properties (size distribution, hygroscopicity, optical properties, density), enabling consistent treatment of transport, sedimentation, and deposition.

Finally, an evaluation and tuning of model parameters are needed. This includes validating plume-rise behaviour against lidar or ceilometer observations, comparing modelled concentrations against satellite and ground-based networks, and adjusting emissions and deposition parameters based on event-specific case studies.

The planned tasks and development efforts will produce a complete, physically grounded, and computationally efficient smoke-dispersion module suited for modern atmospheric modelling systems.

3 Final remarks

This deliverable has defined the technical and data requirements for the four geohazard projects within Work Package 4: UCIS4EQ , SAIPy , Tsunami-HySEA , and FALL3D. By establishing common formats and inputs, this document provides a unified technical foundation for the EU-India collaboration. It ensures all partners have a clear and shared understanding of the data structures, code inputs, and computational needs required for the project's success. The requirements detailed herein will serve as the baseline for all subsequent development, testing, and integration tasks within this work package. This document provides the necessary framework to ensure an efficient and interoperable collaboration.