D9.2 Data Management Plan

Version 2.0

Documentation Information

Contract Number	9555558
Project Website	www.eFlows4HPC.eu
Contractual Deadline	31.08.2021
Dissemination Level	PU
Nature	ORPD
Author	Rosa M Badia (BSC)
Contributors	Jorge Ejarque (BSC), Riccardo Rossi (CIMNE), Marisol Monterrubio (BSC), Juan Esteban Rodríguez (BSC), Jacopo Selva (INGV), Nikolay Koldunov (AWI), Jorge Macías (UMA)
Reviewer	François Exertier (Atos)
Keywords	Data Management Plan, Datasets, Workflow data

This project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 955558. The JU receives support from the European Union's Horizon 2020 research and innovation programme and Spain, Germany, France, Italy, Poland, Switzerland, Norway.

Change Log

Version	Description Change
V0.1	First version with input collected from D4.1, D5.1 and D6.1 and other internal project sources
V0.2	Input provided from partners
V0.3	Feedback from BSC Data Management Manager added
V0.4	Version with review comments addressed
V1.0	Final version
V2.0	Updates from to initial version (M14). Update on available repositories (p8).

Table of Contents

1. Executive summary

This document presents the data management plan (DMP) of the eFlows4HPC project, which describes the data management life-cycle for all datasets to be collected, processed and/or generated along the lifetime of the project. Concretely, this deliverable describes, among others:

- Which datasets will be used, generated, collected and processed for the development and execution of the eFlows4HPC pillars' workflows and other project research activities.
- Which methodology and standards will be applied to eFlows4HPC datasets.
- How datasets will be stored and handled during the lifetime of the project, and after the end of it.
- How the datasets will be made (openly) accessible.
- The deliverable also describes similar aspects (storage, accessibility, openness) of the source code and software used and developed in the project.

2. Context

The eFlows4HPC project aims to deliver a workflow software stack and an additional set of services to enable the integration of HPC simulations and modelling with big data analytics and machine learning in scientific and industrial applications. The project will integrate existing workflow interfaces, programming models, machine learning, and data analytics libraries to provide a uniform, easy to use platform that enables the exploitation of future large-scale systems. The software stack will allow creating innovative adaptive workflows that efficiently use the computing resources considering novel storage solutions.

The project also aims at proposing and developing the HPC Workflows as a Service (HPCWaaS) concept, as a means for widening the access to HPC from user communities. The goal is to provide methodologies and tools that enable to share and reuse existing workflows and that assist when adapting workflow templates to create new workflow instances.

The project aims to demonstrate the novel technologies through use cases of three application pillars with high industrial and social relevance: manufacturing, climate, and urgent computing for natural hazards, and how the realization of forthcoming efficient HPC and data-centric applications can be developed with new workflow technologies.

3. Data summary

The datasets involved in the project are multiple. One the one hand, there will be the source code of the different components involved in the project, written in different general purpose programming languages (Python, C/C++…). In this sense, this statement considers both the software to be developed as part of the eFlows4HPC project and also the software used in the pillars. On the other hand, the three pillars' will be providing multiple sources of datasets.

Most of the software components of the eFlows4HPC software stack (see Figure 1) already exist, and during the project will be extended and integrated between them. Details about the

components can be found in deliverable D1.1 [1]. The authors of most of the software components used in the project are members of the project partners. More information about how the code will be made available and other aspects is given in the next section.

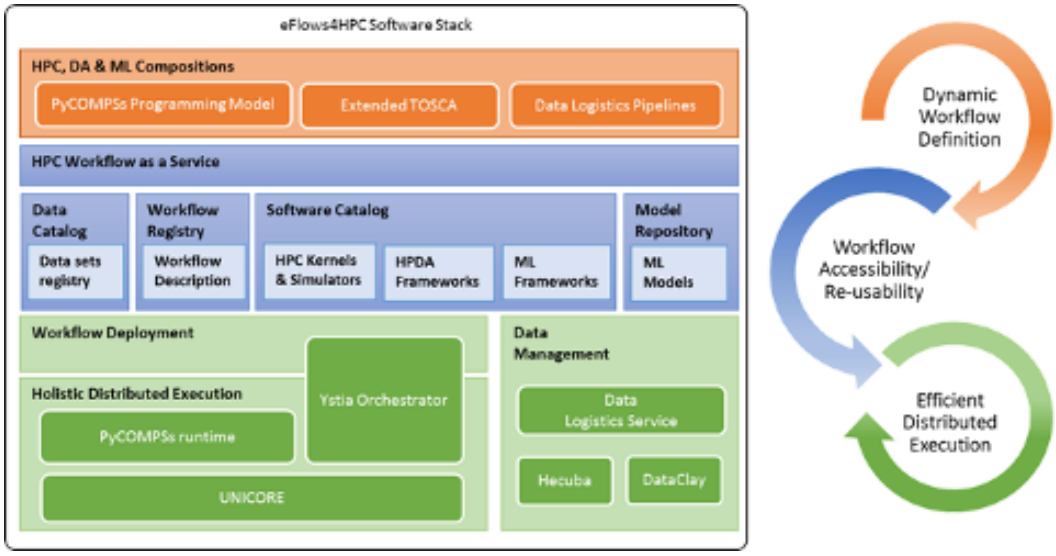

Figure 1. eFlows4HPC Software Stack Overview

The input datasets used in the project are contributed by the project partners, or come from public repositories or from data for which the partners have licenses for their use. More information about how the data will be made accessible and other aspects is described in the next section. The next paragraphs describe the datasets involved in the eFlows4HPC Pillars:

Pillar I: ROM, Digital twin in Manufacturing

The following datasets will be involved in the Pillar I workflows:

1. Input dataset for the simulation, which consist of a model definition completed by several setups describing the simulation scenario to be considered. Some input data is in JSON files, the meshes are solver specific formats (HDF5) - few GB.
2. Output of the simulation, that consists of a snapshot matrix storing the results of the simulation - around 100 GB.
3. Output of the data extraction phase consists of a smaller matrix of doubles (or a ML model) - There are also a set of temporary files that will occupy around 1 TB.
4. The result of the hyper-reduced phase consists of a list of indices and weights which defines the Reduced Order Modelling (ROM) model - order MB.

Pillar II: Dynamic and adaptive workflows for climate modelling

The following datasets will be involved in the Pillar II workflows:

1. Computational meshes: the meshes contain information about the region to consider in the computations. For a geographical domain (the whole globe, ocean, land, particular area) the region is split into small regions that are represented by discrete points that can

be either aligned regularly (quadrilateral grid), or irregularly (unstructured mesh). For many Earth system components there is also vertical discretization. In most cases computational meshes are reused, as every new mesh most probably means a new set of tuning parameters. However, there are cases when mesh generation could be part of the workflow. Mesh files are in ASCII or NetCDF format. The size of these files is typically tens of MB, but can be up to a few GB.

2. Initial state or conditions for the climate model: initial state that is characterized by a distribution of properties, such as temperature and salinity in the ocean, or atmospheric pressure in the atmosphere. There are two main sources, where spatial distributions of those properties can be obtained. One is observational data, usually taken in some gridded form for the area of interest (e.g. global atmosphere, or ocean). Another is the results of simulations performed by other models, often performed on different computational grids. In case of the ensemble runs, perturbations in initial conditions are usually used to make ensemble members slightly different from each other. This data is traditionally in ASCII or NetCDF files. The typical size is from a few MB to a few GB.
3. Forcing data (other initial conditions): periodic updates of information about conditions not directly simulated by the climate model or one of its components. In the case of the full coupled model, examples are solar constants, CO2 concentrations, and aerosol concentrations. This data is traditionally in ASCII or NetCDF files. The size of these files depends on the spatial and temporal resolution of the forcing data and can be from few tens of GB (e.g. COREII forcing) to tens of TB (e.g. ERA5 forcing).
4. Configuration files with the information to be used to run each alternative to be modelled (typically a few KB).
5. Output of the models. They are generated from time to time and written to disk. The size can vary from gigabytes to terabytes per model year. This data is typically in ASCII, NetCDF or GRIB format. In the project, some existing datasets will be used: CMIP6, CMCC-CM3. The size can be from few MB to tens of PB, depending on the spatial and temporal resolution, as well as time of the computation and number of ensemble members. In the project, the AI-assisted ESM member workflow will probably generate up to tens of TB of data. The Statistical analysis and feature workflow would probably generate a larger size.
6. Processed data: the output data can be transformed (change format, data analysis,…). This data can be found in NetCDF, Zarr formats, or images. The size of NetCDF and Zarr files range from a few MB to tens of PBs. The size of the images is on the order of MBs to a few GBs.
7. The final data can be archived to tape. This data can be found in NetCDF or Zarr formats. The size of these files ranges from a few MB to tens of PBs.

Pillar III: Urgent computing for natural hazards

The following datasets will be involved in the Pillar III workflows:

PTF/FTRT workflow

1. Event data: Initial earthquake source data - key value files (JSON like format), QuakeML, YAML files and plain text.

2. Event data: Seismic and Tsunami data retrieved from international transmission services (i.e SeedLink). The data is traditionally a key-value file (< 10 MB).
3. Ensemble data of earthquake scenarios: For each scenario to be evaluated, related to the ensembles to be simulated. The data is traditionally ASCII and raw files (< 10 MB).
4. Tsunami simulations data: Topo-Bathymetric data: NetCFD, GRD and ASCII files (40-350 files).
5. Results of the tsunami simulations: NetCDF files (<200 MB).
6. Tsunami emulator output data: output TBD.
7. The results of the simulation are aggregated into a single file. The data will be a NetCDF file (<100 MB).
8. Quantification of Probabilistic forecast at target points. The results is traditionally NetCDF and raw files (<10 MB).
9. Visualization of the results: output TBD.
10. Permanent storage and long-term post-process: all previous data (>50GB).

UCIS4EQ workflow

1. Event data or its updated information. The data is traditionally in QuakeML's format (XML like format). The size of this file is typically < 10 MB.
2. Time series raw data. The earthquake raw data is in SEED format. The size per seismic station is below 1MB. Consider a maximum of 100 stations (~ 100 MB).
3. Assimilated data. The event data is assimilated and stored as a key-value set in an XML file. The size of this file is typically < 10 MB.
4. Result of the computation of the fault kinematic source description. This data is stored as a key-value plain text file. The size of this file is typically below 1 GB.
5. Velocity models, regional bathymetry and topography are used to generate the computational meshes. The size of the velocity models is below 10 MB and they are stored in HDF5 files. The size of the regional bathymetry and topography maps is below 1 MB and they are stored in HDF5 files too.
6. Computational meshes used as input for the earthquake simulation code. The file is stored in HDF5 format and the size of these files depend on the frequency used in the simulation. Typically, the size will be below 5 GB (frequency of 5 Hz). For other higher frequencies (e.g. 20 Hz) the size will be below 250 GBs.
7. Results of the simulation of the earthquake in HPC: seismic full-field and waveforms at specific geographic sites. The results are stored in HDF5 file format. The size of these files depends on the simulation frequency. For example, for 5 Hz frequency the size will be below 2 GB, and for 10 Hz is typically below 15 GB.
8. Historical Earthquake catalogs. This data is traditionally on CSV or XML files. The size of these files is typically below 400 MB.
9. Machine learning models trained offline < 10 MB, stored as serialized data in files (pickled).
10. Ground shaking maps computed based on Machine Learning models. These maps are jpg or png images < 10 MB.

11. Maps and documents gathered as a result of all the workflow as reports for the stakeholders in form of png or jpg images or videos < 10 MB.
12. Uncertainty quantification analysis - document, png or jpg images < 10 MB.

4. FAIR Data

4.1 Making data findable

With the goal of making the data used in the project easy to find, the project is delivering a component lists all relevant data sources: the Data Catalogue.

The Data Catalogue summarizes all data sources used by the Pillars, including metadata such as their API schemas, protocols used, and intended usage. The Data Catalogue is used by the Data Logistics Service component, which is responsible for setting up the data movements required by a workflow. It may be data stage-in and stage-out, or periodical transfers to synchronize data produced outside the HPC systems. The Data Catalogue is available in this link and the list of datasets is available here. More information about these processes can be found in deliverable D1.1.

Each dataset is identified by its type and object identifier (OID). Each dataset information can be viewed directly with a browser or accessed via the API. Other views can be obtained, as for example a list of all datasets of a specific type. In case the API is used, a JSON document is obtained as a result.

4.2 Making data openly accessible

The data in eFlows4HPC will be made open through the following mechanisms:

- The source code of the eFlows4HPC software stack will be accessible through a github project (https://github.com/eflows4hpc/). For those project components that already have their own github, the github project will link to them. Other software and resources will be directly available there. Other metadata about the source code, such as the documentation, will be public and available from the github and browsable in a friendly format such as readthedocs.
- In addition, the project is developing a set of repositories and registries to make the software components, trained artificial intelligence models, and complete workflows available for the workflow developers and workflow users. These components are: the Software Catalogue, the Model Repository and the Workflow Registry (see Figure 1).
- As mentioned in the previous subsection, information about the datasets are available through the Data Catalogue.
- With regard to intermediate and final datasets, the pillars will be using different repositories. For example, Pillar III is using B2SAFE[1], which is a way to distribute and store

[1] https://www.eudat.eu/b2safe

large volumes of data for a long-term to those sites which are providing powerful data processing, analysis and data access facilities.

4.3 Making data interoperable

The interoperability of the data will be guaranteed through the use of different common formats to exchange the data. Samples of these formats are: NetCFD, HDF5 or JSON.

The actual use of one or another format may change according to what are the best practices in each pillar community.

We are considering the Data catalogue to also include metadata such as access protocol, format and information about how to make data interoperable.

4.4 Increase data reuse

The owners of the data used in the project are responsible for defining the license of the data. Examples of the data license used in each case can be found from the data URL. For example, some of them offer full and open access to the data for scientific use (TraCs), others offer the data under CC-BY 3.0 license (CMCC data in Pangea). Therefore, the license may change from case to case, but most data if offered through open licenses. Another license considered reasonable for some partners is the CC BY-SA[2]

For the new data created in the context of the project, open licenses under Creative Commons will be promoted. For the software code, open-source licenses such as Apache v2 or BSD will be promoted.

At the moment of writing this document we are not aware of any dataset that will not be openly available. Information about how to find the output data and other data involved in the project will be available from the Data catalogue.

4.5 Summary of repositories available

This table summarizes the different repositories available in the project

Source code	https://github.com/eflows4hpc/
Workflow repository	https://github.com/eflows4hpc/workflow-registry
Data catalogue	https://datacatalogue.eflows4hpc.eu/index.html
Software catalogue	https://github.com/eflows4hpc/software-catalog

[2] https://creativecommons.org/licenses/by-sa/4.0/

5. Responsibilities

With regard to the source code, each partner responsible for a given component is responsible for updating the source code in the corresponding github project or in the github linked from there.

With regard to data, each data owner is responsible for selecting the repository to store the generated data and its license, and for updating the information in the data catalogue. Each data owner is also responsible for the metadata and documentation of their datasets.

BSC, as coordinator of eFlows4HPC is responsible of the maintenance of the Data Management Plan.

6. Summary/conclusions

The eFlows4HPC project will handle a large amount of datasets and source code. This deliverable describes the different datasets, the source code and how the data will be made findable, accessible, interoperable and reusable.

7. Acronyms and Abbreviations

- CSV - Comma Separated Values
- D – deliverable
- DMP – Data Management Plan
- DoA – Description of Action (Annex 1 of the Grant Agreement)
- NetCDF - Network Common Data Form
- OID - Object Identifier
- ROM – Reduced Order Model
- TBD - To be defined
- WP – Work Package
- WPL – Work Package Leader
- XML - Extensible Markup Language

8. References

[1] D1.1 Requirements, metrics and architecture design, eFlows4HPC deliverable, https://eflows4hpc.eu/deliverable/d1-2-eflows4hpc-interfaces-and-first-software-stack-release-duplicate-1/