Data Management Plan

July 13, 2023

Atharva Sathe
as6481@columbia.edu
0009-0009-4210-8910

Marco Giometto
mg3929@columbia.edu
0000-0001-9661-0599

1 Introduction

This Data Management Plan is in reference to artifacts surrounding the article "Impact of the numerical domain on turbulent flow statistics: scalings and considerations for canopy flows" to be published later in the Journal of Fluid Mechanics. The Preprint can be seen at https://beta.dpid.org/76/v3/root/publication/main/Preprint.pdf?raw. This Data Management Plan was created post-facto as part of an experiment on the usage of contextualized research objects, dynamic PID systems, and content addressable storage to enhance the data publishing experience in conjunction with the team from DeSci Labs [1], SEAL Storage [2] and Bacalhau [3].

The basis for the data management plan is the template provided by the National Science Foundation's Engineering Directorate [4]. Please note that all links in this file which reference artifacts from the research project are PIDs derived from https://beta.dpid.org/76. dPID is an experimental proving ground for the creation of FAIR PIDs [5].

A large number of links resolve to the underlying metadata associated with a given folder path. If needed, the CIDs provided in the metadata can be used to access the information referenced. The PIDs also act as human navigable folder paths, accessible through the Node Drive. If a human readable format is desired, users should open https://beta.dpid.org/76 and navigate accordingly.

2 Products of Research

What types of data (experimental, computational, or text-based), metadata, samples, physical collections, models, software, curriculum materials, and other materials will be collected and/or generated in the course of the project?The DMP should describe the expected types of data to be retained, managed, and shared, and the plans for doing so. What descriptions of the metadata are needed to make the actual data products useful and reproducible for the general researcher?

Artifacts coming from this project include synthesized data, code, metadata, and a variety of supplementary artifacts surrounding publication, sensemaking and reproducibility. The PID for this research object is https://beta.dpid.org/76. The file structure and high-level artifacts contained within this PID can be seen below in Figure 1 or on this Miro Board

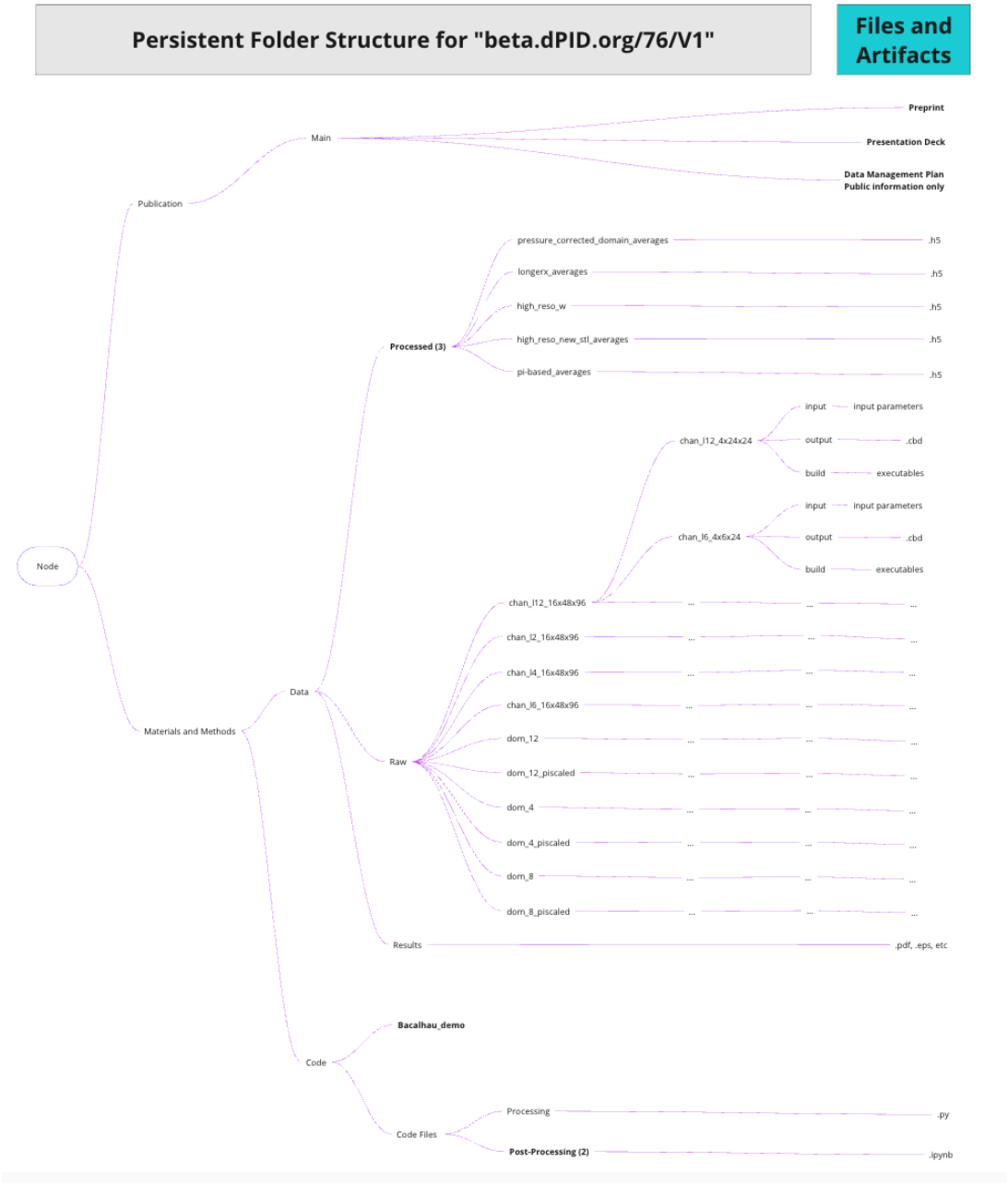

Figure 1: Research Object File Structure

2.1 Overview of Project Workflow

The workflow underpinning this research project was structured into three main steps:

1. **Generate Data:** The first step in this project was generating 135TB of raw Large Eddy Simulation (LES) data across a variety of numerical domains. The raw data from these simulations is stored in 135TB of .cbd files. Generating data was a compute-heavy process that required over 10M compute hours on supercomputer clusters. It is expected that this step will not be reproduced in the future.
2. **Create Time-Averaged Statistics:** Once Raw data was generated, a series of python scripts were used to turn the Raw Data into 200GB of time-averaged statistics which are stored in HDF5 files.

3. **Analyze Time Averaged Statistics:** Finally, post-processing was run on the time-averaged statistics through a series of python notebooks to turn the processed data into final results and visualizations.

More information on this workflow can be found by reading the preprint found at https://beta.dpid.org/76/v3/root/publication/main/Preprint.pdf?raw.

This Research Object is split primarily along two main folders: "Materials and Methods" and "Publication". The "Materials and Methods" folder contains all code and data used to make the experiment and information surrounding reproducibility. The "Publication" folder contains the published manuscript, preprints, presentation decks, peer reviews, and sensemaking data.

2.2 Naming Conventions for Raw and Processed Data Files

Raw Datasets are stored as .cbd files. They can be found at https://beta.dpid.org/76/v3/root/materialsandmethods/data/raw?raw. Averaged statistics are stored as .h5 files. They can be found at https://beta.dpid.org/76/v3/root/materialsandmethods/data/processed?raw. The data is structured in pairs with each .cbd folder having a corresponding .h5 file. Both filetypes conform to two naming conventions:

1. chan_l[number1]_L3xL2xL1.h5
2. chan_hxsx[number2]e-3_L3xL2xL1.h5

In these file names, the prefix "chan" refers to the openchannelflow setup of the simulations. The dimensions L1, L2, and L3 correspond to the streamwise, cross-stream, and vertical extent of the computational domain, respectively.

For configurations where s1 equals s2 (please refer to Figure 1 of the manuscript https://beta.dpid.org/76/v3/root/publication/main/Preprint.pdf?raw for the definitions of s1 and s2), the naming follows the first convention. The symbol "l" represents the length of the repeating unit, and number1 represents the specific length value. In this study, number1 can be 2, 4, 6, or 12.

On the other hand, configurations that utilize boundary layer height-based scaling are named according to the second convention, as s1 is not equal to s2. These configurations satisfy the condition h1/s1 = h2/s2. In the file names, "hxsx" denotes the ratio h1/s1. number2 represents the value of this ratio multiplied by 1000 to avoid decimal numbers. To indicate this operation, number2 is followed by "e-3". In this study, number2 can take the values 500, 250, 167, or 83.

2.2.1 Additional prefixes

The HDF5 files may also have an additional prefix of "ruu," "rvv," or "ruw." The "r" in these prefixes signifies that these files store two-point correlation values. The two variables that follow represent the fields for which the correlation is computed. For instance, a file with the prefix "ruw" would contain the two-point correlation between the streamwise velocity (u) and the vertical velocity (w).

2.2.2 Additional suffixes

In this study, each simulation was initially conducted for 200 large eddy turnover times. However, certain larger simulations were divided into four batches, each consisting of 50 large

eddy turnovers. The averaged values from these simulations are distinguished by the suffixes 'restart[number],' where 'number' ranges from 0 to 3.

Due to insufficient convergence during the initial 200 large eddy turnover times, some simulations required an additional 200 large eddy turnover times. These subsequent runs are indicated by the suffix 'run2'.

2.3 Code

The "Code" folder within "Materials and Methods" is further split into two folders, "Domain_size_analysis_scripts" and "bacalhau_demo".

2.3.1 Domain_size_analysis_scripts Folder

The "Domain_size_analysis_scripts" folder contains the following: A virtual environment, metadata for the project and the code needed to run the experiment. The code needed to run the experiment can be found in the postprocessing folder and can be split into two main categories:

1. The python scripts associated with workflow step 2 were used to create 200GB of time-averaged statistics from 135TB of raw data. They can be found at https://beta.dpid.org/76/v3/root/materialsandmethods/code/Domain_size_analysis_scripts/postprocessing/compute_timeaverage?raw
2. The python notebooks associated with workflow step 3 were used to analyze the 200GB of processed data and create results. They can be found at https://beta.dpid.org/76/v3/root/materialsandmethods/code/Domain_size_analysis_scripts/postprocessing?raw they can be rerun and are more likely to be reused as standalone code

2.3.2 Naming convention in post-processing

Each Jupyter Notebook script typically consists of four dictionaries:

1. **f**: This dictionary is responsible for loading all the variables from HDF5 files.
2. **temp**: This dictionary is used to store the temporal stress values calculated from the variables loaded in f.
3. **disp**: This dictionary is used to store the dispersive stress values computed from the variables loaded in f.
4. **stress**: This dictionary is used to store the total stress values, which are the sum of temp and disp.

The keys of these dictionaries follow the syntax *field _ simulation*, where the field represents a first or second order field (e.g., u, uv, etc.), and simulation corresponds to the HDF5 file name with the prefix 'chan_' removed, as well as the '.h5' extension.

2.3.3 Bacalhau Demo Folder

The "bacalhau_demo" folder contains a demonstration of how dockerized containers running on the Bacalhau open source platform for compute orchestration can be used to the enable reproducibility. This demonstration focuses on a .py version of the grid_resolution_analysis_DeSci.ipynb file.

The Bacalhau job ID for this demonstration is b4e337be-3eae-44a6-9b64-4e106b61e5d7. The Docker container used for this bacalhau job can be found at https://hub.docker.com/r/m0ar/grid_reso_analysis_desci. Additional information and a step by step walkthrouhgh out using or reusing bacalhau can be found at https://beta.dpid.org/76/v3/root/materialsandmethods/code/bacalhau-demo/README.md?raw

3 Data Formats and Standards

In what format and/or media will the data or products be stored (e.g., hardcopy notebook and/or instrument outputs, ASCII, html, jpeg or other formats)? Where data are stored in unusual or not generally accessible formats, how may the data be converted to more accessible formats or otherwise made available to interested parties? When existing standards are absent or deemed inadequate, this should be documented along with any proposed solutions or remedies. In general, solutions and remedies to providing data in an accessible format should be offered with minimal added cost.

Code and data within the research object are all stored in standard file formats. The specific filetypes can be seen in Table 1.

Artifact Name	**File Type**
Raw Data	.cbd
Processed Data	.hdf5
Results Data	.eps and .pdf
Code Files	.ipynb, .py and executable binaries
Publication Artifacts	.pdf

Table 1: Research Object File Types

4 Dissemination, Access and Sharing of Data

What specific dissemination approaches will be used to make data available and accessible to others, including any pertinent metadata needed to interpret the data? In this case, "available and accessible" refers to data that can be found and obtained without a personal request to the PI, for example by download from a public repository. What plans, if any, are in place for providing access to data, including websites maintained by the research group and contributions to public databases/repositories? For software or code developed as part of the project, include a description of how users can access the code (e.g., licensing, open source) and specific details of the hosting, distribution and dissemination plans. If maintenance of a website or database is the direct responsibility of the research group, what is the period of time the website or database is expected to be maintained? What are the practices or policies regarding the release of data – for example, are they available before or after formal publication? What is the approximate duration of time that the data will be kept private? "Data sharing" refers to the release of data in response to a specific request from an interested party. What are the policies for data sharing, including, where applicable, provisions for protection of privacy, confidentiality, intellectual property, national security, or other rights

or requirements? Research centers and major partnerships with industry or other user communities should also address how data are to be shared and managed with partners, center members, and other major stakeholders; publication delay policies (if applicable) should be clearly stated.

All data associated with the research object is being made open on a CC-BY license at the time of publication. Information relevant to the chosen dissemination approach, metadata, and compute over data is detailed below.

4.1 Dissemination Approach

1. **API calls through PIDs:** All artifacts within https://beta.dpid.org/76 can be called using the DeSci.fetch library. See an example of these function calls at https://replit.com/@Erikvan20/dpidfetchdemo. Given that information can be called directly from its PID, the project's "Materials and Methods" (i.e. Code and Data) have been separated from its publication and sensemaking artifacts. Information can be queried as as seen below:

    - **Materials and Methods:** https://beta.dpid.org/76/v3/root/materialsandmethods?raw
    - **All Data:** https://beta.dpid.org/76/v3/root/materialsandmethods/data?raw
        - **Raw Data:**
            - https://beta.dpid.org/76/v3/root/materialsandmethods/data/raw?raw
        - **Processed Data:**
            - https://beta.dpid.org/76/v3/root/materialsandmethods/data/processed?raw
        - **Results Data:**
            - https://beta.dpid.org/76/materialsandmethods/data/results
    - **All Code:** https://beta.dpid.org/76/v3/root/materialsandmethod/code?raw
        - **Data Generation Executables:**
            - Can be found in the "build" files within various raw data folders at https://beta.dpid.org/76/v3/root/materialsandmethods/data/raw?raw
        - **Data Procesing Python Scripts:**
            - https://beta.dpid.org/76/v3/root/materialsandmethods/code/Domain_size_analysis_scripts/postprocessing/compute_timeaverage?raw
        - **Post-Processing Notebooks:**
            - https://beta.dpid.org/76/v3/root/materialsandmethods/code/Domain_size_analysis_scripts/postprocessing?raw

4.2 Metadata

The metadata created for individual artifacts can be seen through the blue and green boxes on Figure 1. In order to minimize the complexity of the metadata schema, Metadata was created for all data, code, and main publication artifacts at the folder level.

The schemas used for specific component types is as follows in Table 2. All metadata is provided in JSONLD.

Artifact Name	**Metadata Folder**	**Schema Used**
Data Files	Folder Level	Schema.org Dataset
Code Repositories	Folder Level	CodeMeta
Publication Artifacts	File Level	Schema.org Scholarly Article

Table 2: Schemas used for metadata provisioning

All information published through dPID comes with a built-in machine actionable API for metadata. By adding the suffix "?raw" or the suffix "?jsonld" to any PID within the https://beta.dpid.org/76 folder structure, a reader can access the metadata manifest file underpinning said research object. See https://beta.dpid.org/76?raw and https://beta.dpid.org/76?jsonld for the complete manifest files on the research object in JSON and JSON-LD respectively.

In addition to the ability to surface metadata directly through the PID, end users can see the files used to create JSONLD-based metadata in the Code folder using the link https://beta.dpid.org/76/v3/root/materialsandmethods/code/Domain_size_analysis_scripts/metadata?raw

Artifact Name	**Metadata PID**
Data Files	https://beta.dpid.org/76/v3/root/materialsandmethods/code/Domain_size_analysis_scripts/metadata/2307_code_metadata.json?raw
Code Repositories	https://beta.dpid.org/76/v3/root/materialsandmethods/code/Domain_size_analysis_scripts/metadata/2307_data_metadata.json?raw
Publication Data	https://beta.dpid.org/76/v3/root/materialsandmethods/code/Domain_size_analysis_scripts/metadata/2307_publication_metadata.json?raw

Table 3: PIDs for Individual Metadata Files

5 Re-Use, Re-Distribution and Production of Derivatives

What are your policies regarding the use of data provided via general access or sharing? For data to be deemed "re-usable," it must be accompanied by any metadata needed to reproduce the data, e.g., the means by which it was generated, detailed analytical and procedural information required to reproduce experimental results, and other pertinent metadata. Practices for appropriate protection of privacy, confidentiality, security, intellectual property, and other rights should be communicated. The rights and obligations of those who access, use, and share your data with others should also be clearly articulated. For example, if you plan to provide data and images on your website, will the website contain disclaimers or condition regarding the use of the data in other publications or products?

5.1 Licensing

All artifacts for this research project are published using a CC-BY 4.0 License.

5.2 Reproducibility

5.2.1 Python Notebooks

All python notebooks can be found https://beta.dpid.org/76/v3/root/materialsandmethods/code/Domain_size_analysis_scripts/postprocessing?raw. These notebooks contain the outputs from their results linked to the preprint associated with publication.

5.2.2 Docker Container

A docker container is a portable way to gather all the required files for a process, and allow others to run the same process through their own interface or OS system.

Currently, we have a sample of one set of results fully reproduced in a docker container https://hub.docker.com/r/m0ar/grid_reso_analysis_desci. This container gives researchers the flexibility to evaluate and reproduce the results from the research object, and is compattible with baccalhau (discussed below in Compute over Data).

5.2.3 Python Scripts and Executables Binaries

The data generation phase for this project originally required extensive compute hours making it difficult to reproduce. However, any researchers can check to ensure that the data generation phase was correctly executed by examining the executable binaries, located in source folders within https://beta.dpid.org/76/v3/root/materialsandmethods/data/raw?raw

After data generation and processing, all processed data was transformed to .h5 from .cbd files. These .h5 files served as the basis for the results and were processed using python scripts found in the https://beta.dpid.org/76/v3/root/materialsandmethods/code/Domain_size_analysis_scripts/postprocessing/compute_timeaverage?raw folder. Replicating the results may require a High-Performance Computing (HPC) cluster and extensive storage. While available, the raw data needed to run these processing scripts is 135TB in size and will required 100+ core hours.

5.3 Community Re-use

Reuse of this information is actively encouraged. All code and data are published on permissible licenses, enabling free use by other provided that they credit the original authors.

One potential follow-on proposal as a part of the experiment with DeSci Labs, SEAL Storage and Bacalhau is to provision dedicated compute servers on top of the archival datasets to enable further re-use. This experiment generated 135TB of data which will be difficult for all but the most technologically advanced labs to use natively. To allow for data reuse, we need the resources to provide the data storage and computational resources necessary for anyone to use it.

5.3.1 Compute over Data

Bacalhau is a platform for fast, cost efficient, and secure computation that enables users to run compute jobs where the data is generated and stored. With the open-source Bacalhau project, researchers can streamline their existing workflows without rewriting by running Docker containers and WebAssembly (WASM) images as tasks. This architecture is also referred to as Compute Over Data (or CoD).

For this node, we have chosen to use Bacalhau [3] this system provides the ability for ‘edge compute’. Edge computing indicates the ability to run scripts on data without requiring data transfer. dPID has been formatted to work closely with the Bacalhau platform, enabling efficient community reuse of data

We are still in the process of uploading this dataset to IPFS, which requires funding (see cost evaluation). However, we aspirationally will have a bacalhau node to which researchers can submit jobs.

5.3.2 Thawing Periods

In the case that compute infrastructure can be provided, we propose having a data rotation thorugh which the data remains archival (or cold) for 5 months at a time. During this time any researchers can submit jobs to run using the data and the author can review and approve these jobs. After the 5 month period, we would have a 1 month period during which data is "thawed" or brought out of Archival storage and all approved jobs are run over it, and any results returned to the submitting researchers.

This structure ensures the following characteristics:

- The current author retains control over the generated data, maintaining ownership of how the data is used, and ensuring it is being used correctly.
- The data remains accessible to all academic researchers who might otherwise have needed to spend 1000′s of compute hours and the associated computational costs to generate a similar dataset.
- Costs for storage and compute are minimized by maintaining an archival storage state for most of the year.

Additional work is still needed to refine this process

6 Archiving of Data

When and how will data be archived and how will access be preserved over time? For example, is there a plan to transfer digitized information to new storage media or devices as technological standards or practices change? Will there be an easily accessible index that documents where all archived data are stored and how they can be accessed? If the data will be archived by a third party, please refer to their preservation plans (if available). Where no data or sample repository exists for collected data or samples, metadata should be prepared and made publicly available over the Internet and the PI should employ alternative strategies for complying with the general philosophy of sharing research products and data as described above

As data sizes tend to mandate different archival requirements, different artifacts from this research project are handled accordingly. It should be noted that proper archival for data and compute intensive projects can often be expensive. As this data management plan was written post facto, accordances have been made and will be indicated below. dPID allows for the versioning of a research object so funding related changes will be directly indicated in this document.

6.1 Raw Data

The Raw data for this project consists of 135TB of data. Assuming $3/TB/month over a 10 year period, this dataset will cost $48,600 to store in archival. As a part of this experiment, SEAL Storage [2] has graciously donated the capacity for archival of 27TB (the amount of storage needed for Scale Separation data) of the 135TB to be stored for 18 months, free of

charge. This data will be put on the IPFS distributed network where others can assist in its storage as necessary.

One potential follow-up proposal would be the storage of all 135TB of data (XAR, YAR and SS data) in multiple datacenters across multiple continents to ensure its safety and longevity.

6.2 Processed Data and Other Data

As with the Raw Data, SEAL Storage [2] has graciously donated the capacity for all 200GB of Processed and Other data to be stored hot for 18 months, free of charge. DeSci Labs will also be storing the 200GB of processed data free of charge. This data will be put on the IPFS distributed network where others can assist in its storage as necessary.

In addition to hot storage, all 200GB of processed data will be archived along with the 135TB of raw data.

7. Bibliography

[1] D. L. GmbH, "DeSci Labs," 2023. [Online]. Available: https://www.desci.com/

[2] S. Storage, "SEAL Storage," 2023. [Online]. Available: https://www.sealstorage.io/

[3] Bacalhau, "Bacalhau," 2023. [Online]. Available: https://www.bacalhau.org/

[4] N. S. Foundation, "Directorate for Engineering Data Management Plans Guidance for Principal Investigators," Nov. 2018. [Online]. Available: https://www.nsf.gov/eng/general/ENG_DMP_Policy.pdf

[5] D. Labs, "Decentralized Persistent Identifiers," 2023. [Online]. Available: https://www.dpid.org/