2024

Draft Document

DELIVERABLE 3.3

THIS DOCUMENT IS IN DRAFT FORM AND PENDING OFFICIAL APPROVAL. IT IS SUBJECT TO REVIEW AND MAY BE UPDATED.

D3.3 Environmental Impact Optimizer Module I

This project has received funding from the European Union's Horizon Research and Innovation Actions under Grant Agreement № 101093216.

Title:	Document version:
D3.3: Environmental Module I	1.0

Project number:	Project Acronym	Project Tittle
101093216	UPCAST Project	UPCAST Project

Contractual Delivery Date:	Actual Delivery Date:	Deliverable Type*-Security*:
M18 (June 2024)	M18 (June 2024)	OTHER - PU
Responsible:	**Organization:**	**Contributing WP:**
Aditya Grover	CDR	WP3

Authors (organization):

Abstract:

The UPCAST project is a pioneering endeavour that enhances data sharing across diverse sectors providing plugins to increase efficiency and effectiveness of data marketplaces through a unified platform. Leveraging advanced AI and data management technologies, the plugins address complex challenges in digital marketing, healthcare, public administration, and genomics research. The project integrates cutting-edge tools to facilitate seamless data discovery, processing, privacy enforcement, pricing, and environmental impact assessment. UPCAST promotes open science, gender neutrality, and adheres to ethical AI principles. With a comprehensive data governance structure, it optimizes data utilization while ensuring privacy and compliance. By fostering cross-sector collaboration, UPCAST accelerates innovation and empowers decision-making for a data-driven future.

This technical specification document supplements the demonstration of the first version of the environmental impact optimizer plugin, which is the main focus of this deliverable. This document delves into the background, experiments, API specification and insights gained during the design and development of the plugin.

Keywords:

Environment, power, carbon footprint, profile, processing, API, operation, storage

REVISION HISTORY

Revision:	Date:	Description:	Author (Organization)
V0.1	01.05.2024	Outline and Table of Contents	Aditya Grover (CDR) Yibrah Gebreyesus (CDR)
V0.2	11.06.2024	Content added to sections	Yibrah Gebreyesus (CDR) Hanene Jemoui (CDR) Aditya Grover (CDR)

V0.3	18.06.2024	Draft document sent for peer review	Aditya Grover (CDR)
V0.4	27.06.2024	Peer Review comments integrated	Yibrah Gebreyesus (CDR) Hanene Jemoui (CDR) Aditya Grover (CDR)
V1.0	29.06.2024	Final Document Ready for Submission	Aditya Grover (CDR)

This project has received funding from the European Union’s Horizon Research and Innovation Actions under Grant Agreement Nº 101093216.

More information available at https://upcastproject.eu/

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Purpose of the Deliverable

This deliverable presents the first version of the Environmental Impact Optimizer plugin (D3.3), which is linked to task T3.3 of Work Package 3 in the UPCAST project. It serves as a public record of the progress achieved in this task, showcasing the plugin development through practical demonstration.

In Work Package 1, Deliverable D1.2 defines the UPCAST MVP and its core functionalities. Figure 1 shows the relevant plugin highlighted in this deliverable.

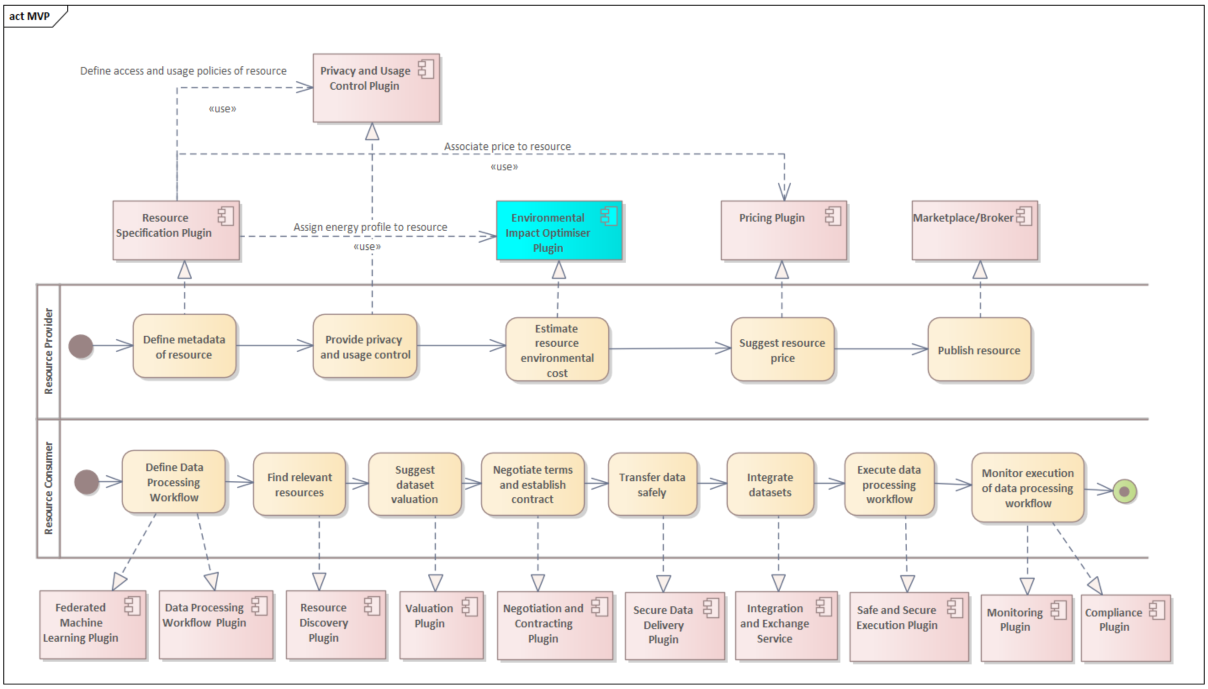

Figure 1: Core functionality included in the UPCAST MVP. Environmental Impact Optimizer Plugin highlighted for this deliverable.

This document is accompanied by a video demonstration showcasing the use of the first version of the plugin that is presented here. https://youtu.be/hI5gDodvHZcon

1.2 Scope of the Document

This document serves as the technical specification for D3.3 of the UPCAST project. It is a public document that describes the progress of development and the practical demonstration of the environmental impact optimizer plugin. This document is structured into the following key chapters:

- Chapter 2 explains the motivations behind the development of the plugin. It examines some of the pressing environmental challenges being faced in high performance computing (HPC), and the key regulatory requirements being introduced by the European Union (EU). Section 2.3 analyses the results of a survey sent to the UPCAST Pilot partners to understand their hardware infrastructure, key processes and capabilities.

- Chapter 3 describes the first version of the plugin, showcasing the progress carried out in Task T3.3. It includes the architecture, API and steps on how to run the demonstrator. It also describes the various tests and experiments carried out towards the development of the plugin.
- Chapter 4 concludes the document by summarizing the key activities performed in this deliverable and outlines the next steps for task T3.3.

2 Background

2.1 Environmental Challenges in HPC

AI and Machine Learning (ML), big data services, analytics, blockchain, 5G, modelling and simulations, virtual reality, research applications, and other cross-cutting technologies are exploding the demand for HPC infrastructure that supports heavy operations. HPCs are energy-intensive infrastructures due to their intensive computational power, such as fast processors (e.g., multi-core CPU, GPU, and TPU), large memory configurations, and parallel processing for simultaneous tasks. This significantly impacts the energy used for storing, processing, and transferring information, with even greater effects expected in the coming years.

The development and training of AI and ML models come with a significant energy and carbon footprint. Studies indicate that training can account for 20-40%, while inference contributes 60-70% (Meta[1], Google[2]). As AI models, particularly large language models (LLMs), increase in size and applications proliferate, the energy demand for AI operations continues to escalate. Therefore, developing environmentally efficient AI operations is essential for reducing the carbon footprint. The environmental impact optimizer plugin addresses this need by promoting energy-efficient practices, from basic atomic operations to more complex AI and ML models.

2.2 EU Regulation and Compliance

With data centers emerging as major energy consumers, the evolving landscape of the EU and international environmental directives directly influences how these facilities manage and monitor their environmental impact. This includes energy consumption, carbon footprint, and the overall environmental burden associated with storing, processing, and transferring data.

The European Union (EU) Code of Conduct on Data Centre Energy Efficiency is a voluntary program launched in 2008 to help data centers reduce their energy consumption. It was most

[1] Wu, C.-J., Raghavendra, R., Gupta, U., Acun, B., Ardalani, N., Maeng, K., Chang, G., Aga, F., Huang, J., Bai, C., Gschwind, M., Gupta, A., Ott, M., Melnikov, A., Candido, S., Brooks, D., Chauhan, G., Lee, B., Lee, H.-H., ... Hazelwood, K. (2022). Sustainable AI: Environmental Implications, Challenges and Opportunities. *Proceedings of Machine Learning and Systems*, *4*, 795–813

[2] Patterson, D., Gonzalez, J., Hölzle, U., Le, Q., Liang, C., Munguia, L.-M., Rothchild, D., So, D., Texier, M., & Dean, J. (2022). *The Carbon Footprint of Machine Learning Training Will Plateau, Then Shrink* (arXiv:2204.05149). arXiv. https://doi.org/10.48550/arXiv.2204.05149

recently revised in 2022.[3] The Code provides guidelines for data center operators to improve energy efficiency, covering areas such as cooling systems, server utilization and power management.

The European Green Deal[4] is the European Union's ambitious plan to transform the bloc into a sustainable and climate-neutral economy by 2050. The main goal is to achieve net-zero greenhouse gas emissions by 2050.

The European Climate Neutral Data Centre Pact[5] is an industry-driven initiative launched in 2020 with the ambitious goal of achieving climate-neutral data centers across Europe by 2030. The key areas of focus include improving energy efficiency, utilizing renewable energy sources, and minimizing the carbon footprint throughout the lifecycle of a data center. The Pact serves as a collective effort by the European data center industry to become more sustainable and contribute to the EU's Green Deal objectives.

The recent updates to the EU's Energy Efficiency Directive[6] (EED) emphasize transparency and efficient resource utilization. This highlights the crucial role of data centers in promoting environmental, economic, and societal sustainability. Data centers exceeding 500 kW in rated power must report energy efficiency metrics starting from May 15, 2024. To comply with these regulations, adopting environmentally efficient practices for data storage, processing, and transfer is crucial.

2.3 Survey of Hardware Infrastructure of Pilots

To gather requirements and to inform the design of experiments for the environmental impact optimizer plugin, a survey was distributed to the UPCAST pilot partners gathering their requirements and understanding their current practices. A Google Form[7] was sent to the partners to gather information about the hardware infrastructure, type of data possessed and their typical workflows.

There are a total of 18 questions and four pilots responded to the survey. The responses are available in a spreadsheet here.

The initial questions focus on how much data each organisation typically collects and what is the frequency of collecting that data. Based on the results of the survey, two pilots generate approximately 1GB of data per day, while the other two declared that the total amount of data they store is 500GB and 150GB. In terms of frequency, there was no fixed time interval for 3 pilots as it depends on each activity or project. One organization collects data once every 24 hours.

Questions 3 to 11 focus on the execution environment or hardware infrastructure that inform this task of the storage and processing capabilities of each pilot partner. All the partners have or access an execution environment where they typically run their workflows. Figure 2, shows

---

[3] https://joint-research-centre.ec.europa.eu/scientific-activities-z/energy-efficiency/energy-efficiency-products/code-conduct-ict/european-code-conduct-energy-efficiency-data-centres_en
[4] https://commission.europa.eu/strategy-and-policy/priorities-2019-2024/european-green-deal_en
[5] https://www.climateneutraldatacentre.net/
[6] https://energy.ec.europa.eu/topics/energy-efficiency/energy-efficiency-targets-directive-and-rules/energy-efficiency-directive_en
[7] https://forms.gle/T1d5YisjLwHCAAiDA

that only one of the pilots solely use their on-premises data center facilities, while the others use a mix of cloud infrastructure (AWS) and on-premises servers.

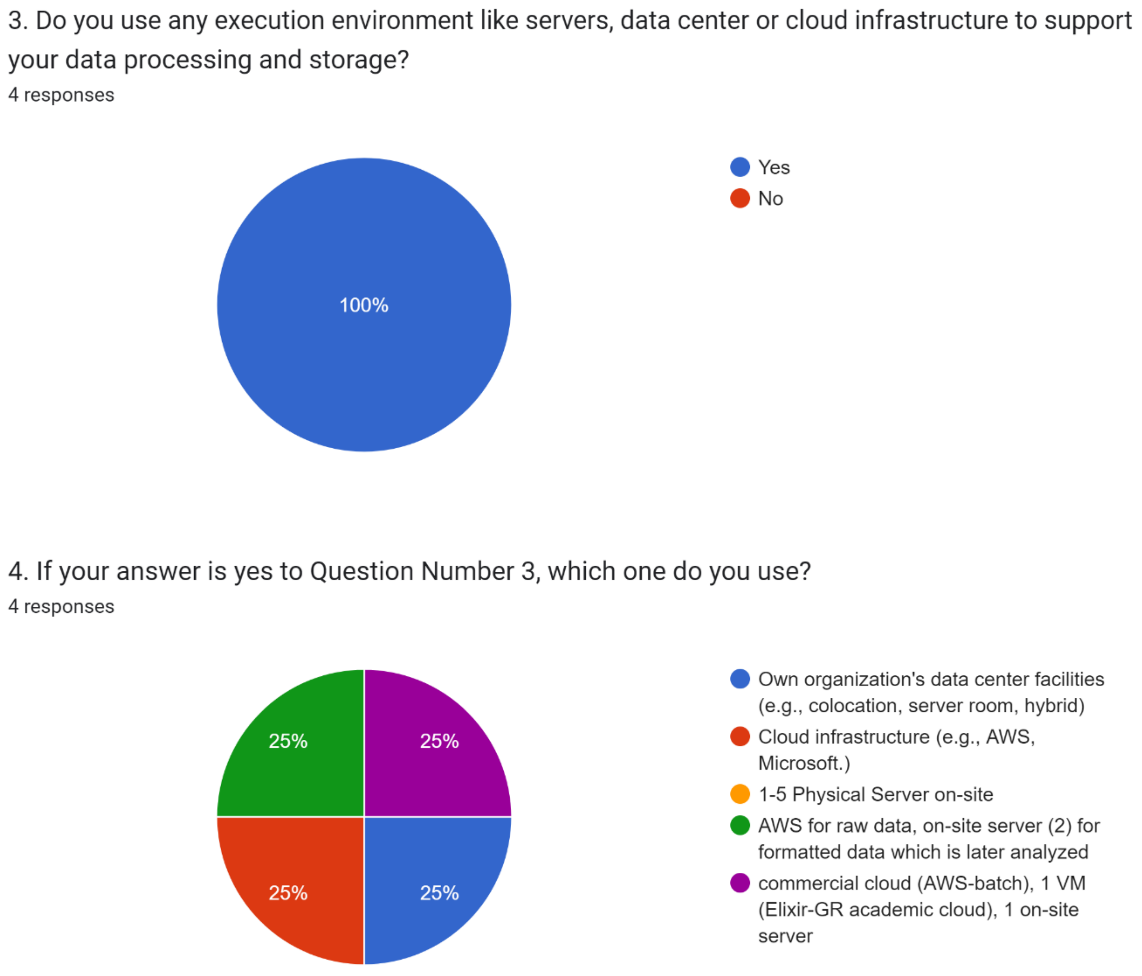

Figure 2: Execution environment infrastructure

The subsequent questions focus on the specifics of the hardware infrastructure, including details about the operating system, processor types, CPU speed, cores, memory (RAM) and storage system. The survey revealed that all partners utilize Intel processors with speeds ranging from 2.2 GHz to 3.6 GHz. The core count varied, with the maximum being 12 cores for a single partner. RAM and storage also showed significant variations among the participants.

Figure 3 illustrates the frequency at which the execution environment is operational.

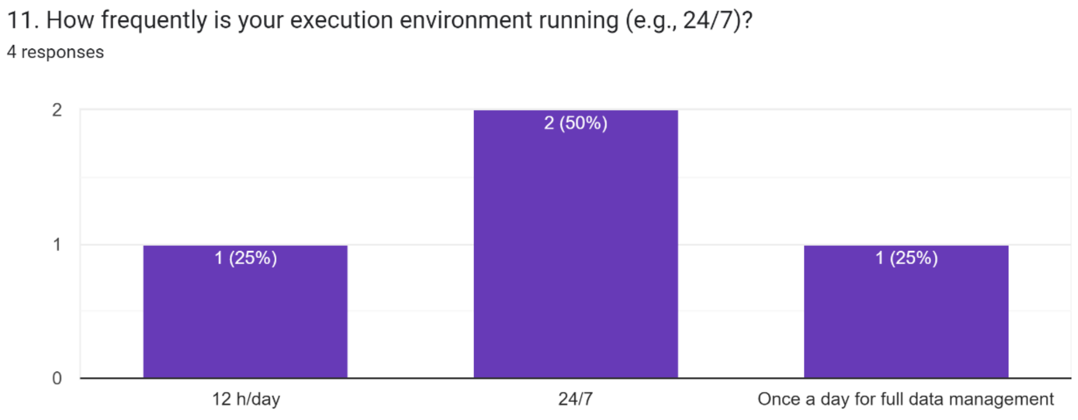

Figure 3: Execution environment frequency

The final section of the survey examined the types of operations performed by each partner for their typical use cases. The results revealed a similar pattern in workflows. Three partners employ workflows consistent with traditional data science methodologies, including data collection, pre-processing, modelling, analytics, and inferencing. One partner utilizes their on-premises server solely for data storage, without engaging in any analytics activities.

The survey also addressed the location of the execution infrastructure, a critical factor since the carbon footprint of computing is significantly influenced by the location of data processing. The geographical distribution of data centers significantly impacts the energy source used for powering them, which in turn affects the overall environmental impact. The pilot partners store and process their data in Greece, Serbia and Western Europe.

3 Environmental Impact Optimizer Plugin

The ultimate aim of the UPCAST environmental impact optimizer plugin is to tackle climate change by fostering more power efficient data workflows. These include:

- **Reduce Carbon Footprint:** Raise awareness and encourage eco-friendly practices within data processing, minimizing environmental impact.
- **Optimize Energy Consumption:** Streamline data processing workflows (DPWs) by suggesting hardware and operations that minimize energy use, leading to cost savings.
- **Simplify Cost Estimation:** Facilitate data exchange with clear energy cost estimations, enabling informed decision-making.
- **Ensure Regulatory Compliance:** Support adherence to environmental regulations for data processing.

3.1 Plugin Architecture & API

The environmental impact optimizer plugin has three core functionalities:

- Dataset storage energy and environmental profiling
- Dataset processing energy and environmental profiling
- Explainable AI (XAI): A mechanism that explains to what degree (either positively or negatively) certain features impact the energy consumption estimation

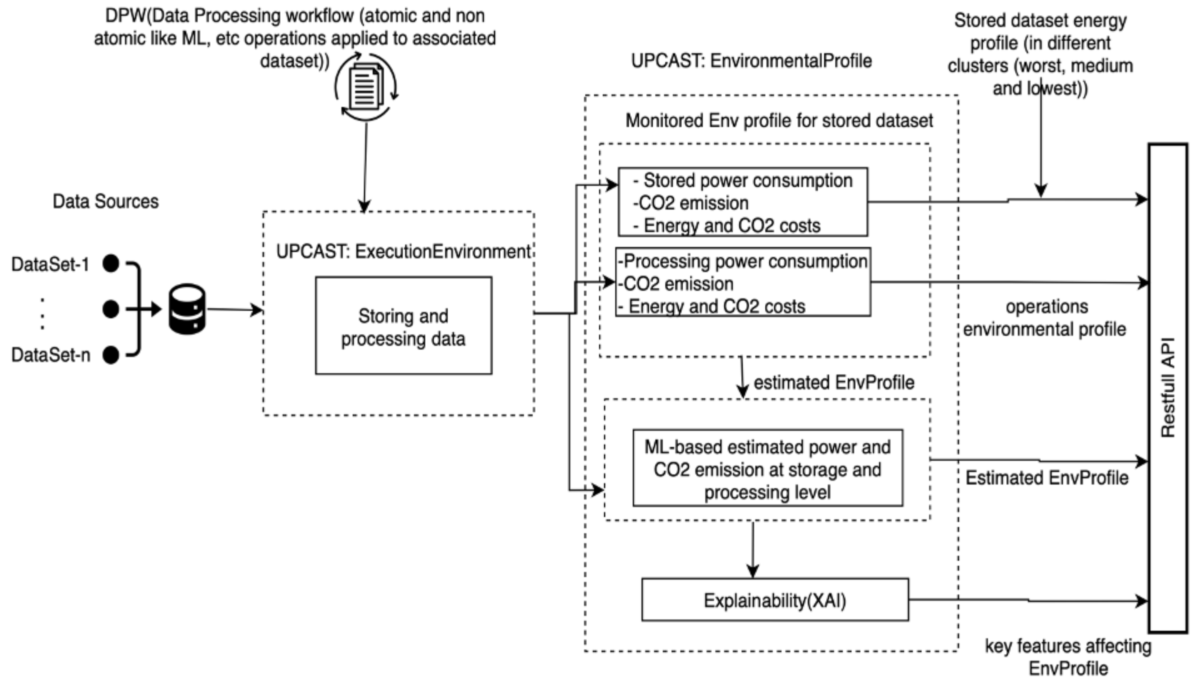

Figure 4: Architectural Overview of the Environmental Impact Optimizer

Figure 4 is the envisioned architecture for the final version of the plugin detailing the system components, all plugin functionalities and their interactions for estimating energy consumption and environmental impact. In this deliverable, the main functionalities described are the dataset storage and dataset processing energy consumption.

The plugin operates independently of data processing workflows (DPWs), data exchange methods and the pilot scenarios. However, the input from the pilots regarding their hardware infrastructure informed some of the decisions while designing the experiments, which have been described in Section 3.2.

Users can access the plugin through three channels:

- **REST API:** The plugin is deployed as a REST API service via the https protocol for programmatic interaction
- **Data marketplaces**: Integration with data marketplaces or platforms is envisioned.
- **UPCAST Monitoring Dashboard**: The plugin will also connect to the monitoring dashboard for visualization and insights.

Authorized users, such as marketplaces, monitoring entities, data providers and consumers, can request energy consumption estimates via the API endpoint.

For this initial version, the plugin estimates energy consumption for a single storage instance and a single data processing workflow (DPW) instance. It does not consider storage redundancy mechanism or multiple data workflows. Users are responsible for calculating overall consumption based on their specific use case. For instance, if a provider broadcasts data to multiple consumers using the same workflow, the plugin will estimate the energy cost for one execution.

For the initial version of the plugin, the API was designed to provide a single endpoint for the following reasons:

- Users (both consumers and providers) require energy consumption evaluation while storing and/or processing the data, regardless of the data exchange methods (API, ETL, file transfer etc.) or operational scenarios (such as workflow execution from consumer or provider side, database storage in consumer side). This unified plugin functionality accommodates all types of users.
- Both the energy consumption estimators for database storage and process/operations utilise the same input data (hardware infrastructure and computational properties). Consolidating these functionalities into a single endpoint streamlines the design and eliminates redundancy, sparing users from having to input the same data twice to obtain their energy consumption metrics.

Note: In the future, a decision may be made to split the endpoint into two endpoints: one for the operations and the other for the storage. This would only be necessary if both functionalities do not share the same input data.

A query parameter was added to the endpoint to separate the storage energy profile from the workflow energy profile.
For example, when run locally, the URL's could be:

- http://localhost:9000/estimates/environmentalProfile?profileType=FULL
- http://localhost:9000/estimates/environmentalProfile?profileType=DPW
- http://localhost:9000/estimates/environmentalProfile?profileType=STORAGE

Where DPW corresponds to retrieving the profile for operations in the data processing workflow, STORAGE pertains to storage energy consumption and FULL encompasses both functionalities.
Furthermore, for this deliverable, the workflow has been simplified to include a sequential list of operations (without iteration or parallel execution).

The primary objective of this deliverable was to conduct experiments and analyze their results that can help in designing a robust plugin. The final version of the plugin will expose REST API endpoints for the functionalities, adhering to the OpenAPI specification. Section 3.3 shows the swagger interfaces of the API.

3.2 Analysis and Results of Experiments

To ensure the effectiveness of the environmental impact optimizer plugin, a series of targeted experiments have been conducted. This section details the experimental approach, tests carried out, the key learnings from the data, and how these insights are helping shape the final plugin design.

The experiments can be classified into two main categories:

- Dataset Storage Energy Consumption
- Dataset Processing Energy Consumption

3.2.1 Experiments for Dataset Storage Energy Consumption

The aim of this section is to measure and analyze the energy consumption of storing datasets in hardware infrastructures.

3.2.1.1 Methodology

Measuring energy consumption and estimating the environmental impact of storing datasets is one of the core functionalities of the UPCAST environmental optimizer plugin. The energy consumption and environmental cost is monitored based on metadata and storage system specification. For each dataset, the aim is to estimate the environmental impact of storing a dataset in a given infrastructure. Figure 5 provides an architectural overview of modelling a dataset based on energy and environmental costs.

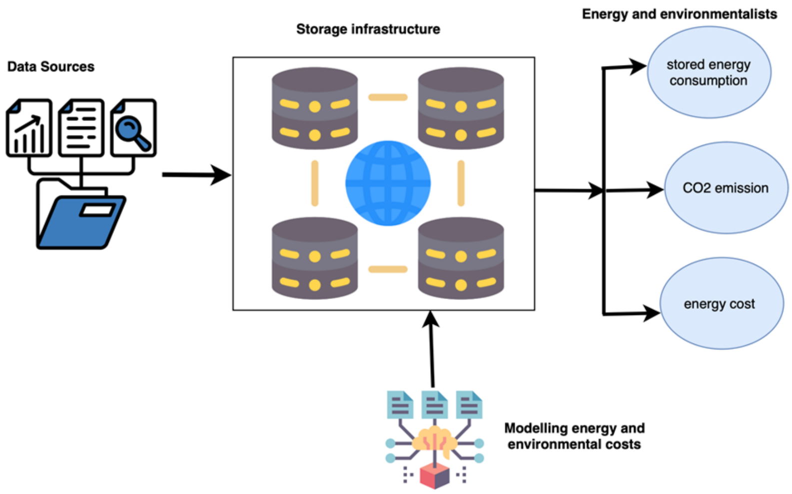

Figure 5: Overview of Dataset Storage Energy Consumption

The energy and environmental costs of a dataset are calculated by multiplying storage system utilisation by static power, excluding traffic or processing actions. The statistical model is utilised for calculating energy and environmental costs, derived from equations (1-3). The power consumption of a given dataset in a specific storage system is calculated using equation (1).

$$\boldsymbol{P} = \boldsymbol{static\ power} * \boldsymbol{Storage\_utilisation}(\%) \quad \textbf{(1)}$$

Where P is the power consumption of a dataset in a storage system, determined by the system's utilisation, with static power representing the system's power when idle. Storage_utilisation is the percentage of the storage system that has been occupied by the dataset. The energy consumption of a dataset in a specific storage infrastructure is determined by its duration. Therefore, the energy rating of the dataset over a period time (T) is given as,

$$\boldsymbol{E} = \boldsymbol{P} * \boldsymbol{Time}(\boldsymbol{T}) \qquad \textbf{(2)}$$

Where E is the energy consumption computed over time T.

We determine the environmental impact, essentially the CO2 emissions of a dataset, by multiplying its energy consumption by its carbon intensity (kg/kWh)

$$\boldsymbol{CO2\_emissions} = \boldsymbol{E} * \boldsymbol{Carbon\_Intensity}(\boldsymbol{kg}/\boldsymbol{kWh}) \qquad \textbf{(3)}$$

These mathematical models have been implemented in python to calculate energy and environmental costs of dataset storage across various infrastructures, allowing data producers to compute and estimate their datasets energy consumption and environmental impact.

3.2.1.2 Experiment Data & Results

To collect observations, several tests were performed by storing a variety of dataset sizes across multiple storage infrastructures to which CeADAR currently has access. This is due to the absence of a dedicated execution environment in the project. These dataset sizes range between 1 GB and 26 GB. The input features used for these experiments have been described in Table 1:

Table 1: Input features for Experiments

Input Feature	Description
storage_system	Storage technology or infrastructure (SDD, HDD, NAS etc.)
storage_system_model	A specific storage system identification.
Storage_system_size_gb	Size of the storage system (in GB).
storage_energy_consumption_w (static power)	A static power being consumed by storage technology when idle (with no traffic)
dataset_size_gb	The size of a given dataset (GB) stored in a given storage technology.
dataset_storage_utilisation (%)	The percentage of a given dataset occupied in the storage technology.
geographic_location_country	A country where the dataset is stored and processed.
stored_dataset_time_span_h	The time length for how long a given dataset is stored.
dataset_update_freq	The update frequency of any given dataset

Using these features, the following outputs are calculated:

- **Stored Dataset Energy Consumption**: This is the total power consumption (in Wh) associated with storing the dataset in the infrastructure.

- **CO2 Emissions**: The carbon emissions (in kg/kWh) can be calculated based on the average emissions of 0.309 kg/kWh. This value is the CO2 emission factor provided in the BEIS report in 2018[8].
- **Energy Costs**: The total energy costs are calculated based on where the computing infrastructure is located. Since these experiments were performed in Ireland, the value is based on the average price of 0.36 Euro per kWh[9].

Table 2 shows the detailed results of the experiments conducted.

Table 2: Results of Dataset Storage Experiments

Storage system	Storage Size (GB)	Dataset Size (GB)	Storage_utilisation (%)	Time Span (hrs)	Energy Consumption (Wh)	CO2 emissions (kg/kWh)	Energy Price (euro)
SDD-NVMExpress-1	**500**	1.399	0.2798	12	5.2043	0.0052	0.0019
		2.637	0.527	15	12.253	0.0037	0.0044
		5.415	1.083	30	50.3595	0.0151	0.0181
		8.485	1.697	12	31.564	0.0095	0.0113
		10.829	2.166	32	107.434	0.0322	0.0387
		11.751	2.35	21	76.49	0.0229	0.0275
		14.963	2.993	8	37.113	0.0111	0.0134
		15.823	3.165	4	19.623	0.0059	0.007
		16.970	3.394	7	36.825	0.011	0.0133
		17.826	3.565	3	16.577	0.005	0.006
		21.098	4.22	9	58.869	0.0177	0.0212
		23.735	4.747	6	44.147	0.0132	0.01589
		26.372	5.274	16	130.795	0.0392	0.047
SDD-NVMExpress-2	**500**	1.399	0.2798	7	3.036	0.00091	0.0011
		2.637	0.527	7	5.718	0.00172	0.0021
		5.415	1.083	8	13.429	0.004	0.0048
		8.485	1.697	8	21.043	0.0063129	0.0075
		10.829	2.166	32	107.434	0.0322	0.0388
		11.751	2.35	6	21.855	0.0066	0.0079
		14.963	2.993	2	9.28	0.0028	0.0033
		15.823	3.165	2	9.811	0.0029	0.0035
		16.970	3.394	7	36.825	0.011	0.0133
		17.826	3.565	4	22.103	0.0066	0.00796
		21.098	4.22	8	52.328	0.0157	0.0188
		23.735	4.747	3	22.074	0.0066	0.0079
		26.372	5.274	4	32.6988	0.0098	0.0118
	4000	1.399	0.035	8	1.12	0.00036	0.00043

[8] https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/726911/2018_methodology_paper_FINAL_v01-00.pdf
[9] https://qery.no/consumer-energy-prices-in-europe/

LACiE-HDD		2.637	0.066	15	3.956	0.0012	0.0014
		5.415	0.135	8	4.32	0.0013	0.0016
		8.485	0.212	5	4.243	0.001273	0.0015
		10.829	0.271	15	16.26	0.0049	0.0059
		11.751	0.294	6	7.056	0.0021	0.0025
		14.963	0.374	6	8.976	0.0027	0.0032
		15.823	0.396	5	7.92	0.0024	0.0029
		16.970	0.424	10	16.96	0.005	0.0061
		17.826	0.446	8	14.272	0.0043	0.0051
		21.098	0.527	12	25.296	0.00759	0.0091
		23.735	0.593	17	40.324	0.0121	0.015
		26.372	0.659	17	44.812	0.0134	0.016
SDD-SATA	**250**	1.399	0.5596	24	20.81712	0.006245	0.00749
		2.637	1.055	10	16.3525	0.0049	0.0059
		5.415	2.1659	8	26.85716	0.0081	0.0097
		8.485	3.394	12	63.1284	0.0189	0.0227
		10.829	4.33	9	60.4035	0.0181	0.0217
		11.751	4.700	6	43.71	0.0131	0.0157
		14.963	5.9851	8	74.21524	0.0223	0.0267
		15.823	6.3294	5	49.05285	0.0147	0.0177
		16.970	6.788	12	126.257	0.0379	0.0455
		17.826	7.131	12	132.64	0.0397	0.0478
		21.098	8.439	9	117.72405	0.0353	0.0424
		23.735	9.494	6	88.2942	0.264	0.0318
		26.372	10.549	8	130.8076	0.3924	0.0470
PCLe-SSD	**500**	1.399	0.2798	14	13.7102	0.00411	0.00493
		2.637	0.5274	9	16.6131	0.00498	0.00598
		5.415	1.083	12	45.486	0.0136	0.0164
		8.485	1.697	15	89.0925	0.0267	0.0320
		10.829	2.1659	12	90.9678	0.0273	0.0327
		11.751	2.3502	10	82.257	0.0247	0.0296
		14.963	2.9926	7	73.3187	0.0219	0.0264
		15.823	3.1646	5	55.3805	0.0166	0.0199
		16.970	3.394	13	154.427	0.0463	0.0556
		17.826	3.565	7	87.3425	0.0262	0.03144
		21.098	4.2196	6	88.6116	0.0266	0.0319
		23.735	4.7470	8	132.916	0.0399	0.0478
		26.372	5.274	5	92.295	0.0277	0.0332

3.2.1.3 Insights from Experiments

This subsection describes the data generated from the experiments and some key insights gained that can inform how power consumption varies by storing datasets in different storage systems.
The final cleaned dataset obtained after running these experiments had 164 observations and 11 features. These have been described in Table 3.

Table 3: Storage Dataset Features

Feature Name	Description
Storage_system	Storage technology or infrastructure (SDD, HDD, NAS etc.)
Storage_system_model	A specific storage system identification.
Storage_system_size_gb	Size of the storage system in GB.
Storage_energy_consumption_w	A static power consumption of the storage technology when idle (with no traffic)
Data_source_path	The path or location where the dataset is stored in a given storage technology.
Dataset_size_gb	The size of a given dataset (GB) stored in a given storage technology.
Dataset_storage_utilisation(%)	The percentage of a given dataset occupied in the storage technology.
Stored_dataset_time_span_h	The time length for how long a given dataset is stored.
Stored_dataset_energy_consumption_wh	Computed stored dataset energy consumption (Watt)
Energy_intensity_wh_per_b	The power intensity of a given dataset per bit.
CO2-emission	A footprint of a given dataset that stored for time (t)

Please note that the number of observations is limited; therefore, further experiments will be conducted to obtain more accurate insights. The following analyses are based on the available observations and may vary with more comprehensive experiments.

Figure 6 provides the summary statistics of the key outputs for all observations in the data. The mean energy consumption is 37 Wh, and the mean CO2 emissions are 11 grams per kWh.

	count	mean	std	min	max
stored_dataset_energy_consumption_wh	164.0	3.792200e+01	3.663087e+01	4.040000e-05	1.765892e+02
energy_intensity_wh_per_b	164.0	3.711963e-10	2.925848e-10	4.660000e-11	1.730000e-09
CO2-emission (g CO2/kWh)	164.0	1.137660e+01	1.098926e+01	1.212000e-05	5.297675e+01

Figure 6: Summary Statistics

The histograms presented in Figure 7 show the distribution of the CO2 emissions and total energy consumption for storing the datasets.

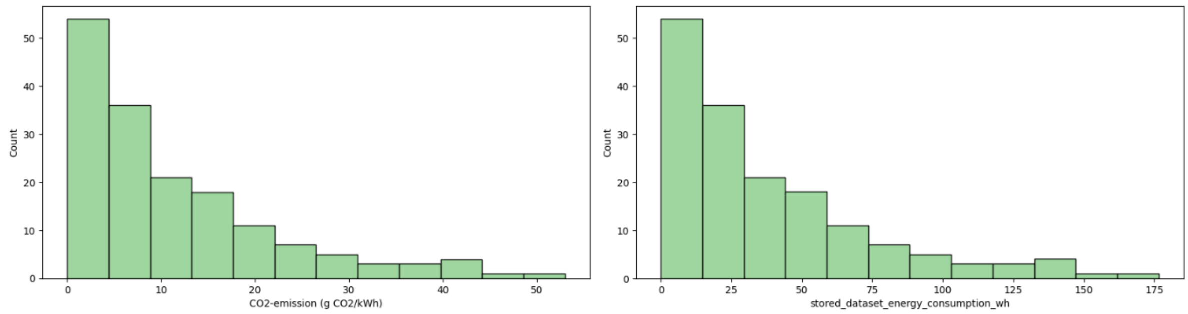

Figure 7: Histograms of CO2 emission & Stored Energy Consumption

The histograms reveal a right skew in the distribution of energy consumption and CO2 emissions. This indicates that most data points cluster towards lower consumption levels, with a few instances exhibiting significantly higher power usage. Logically, the datasets that are bigger in size should have a higher storage energy consumption. However, this also depends on the kind of storage system used and the scatterplot in Figure 8 shows this.

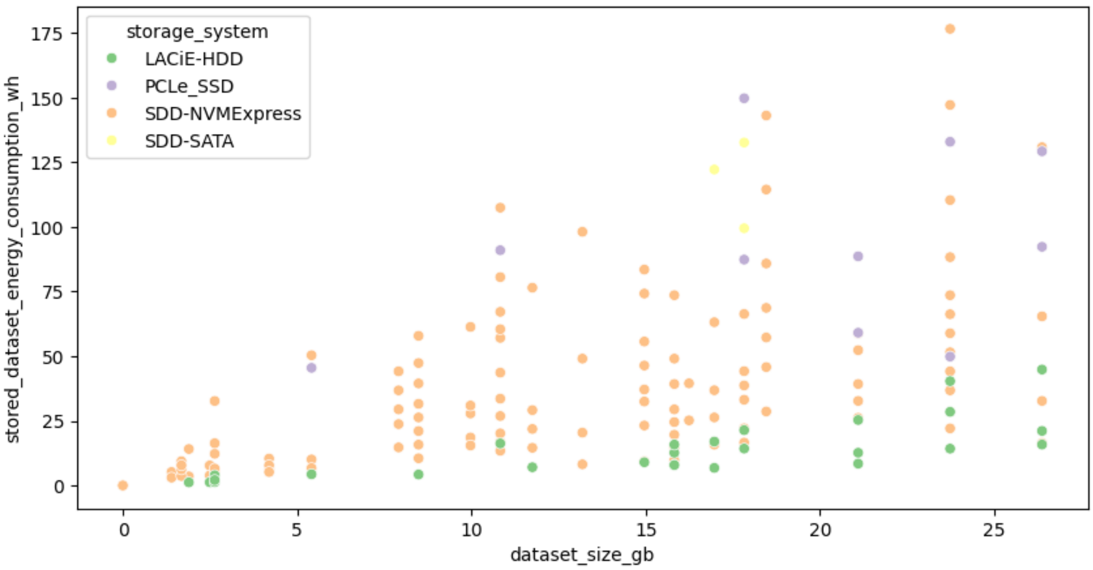

Figure 8: Scatterplot of Dataset Size vs Energy Consumption

From the above, it is evident that the LACiE-HDD storage system (colored green) exhibits lower energy consumption compared to other storage systems as datasets sizes increase.

Further analysis was conducted to assess the impact of storage systems on stored energy consumption and, consequently, on carbon footprint. The boxplot in Figure 9 shows the range of energy consumption across different storage systems.

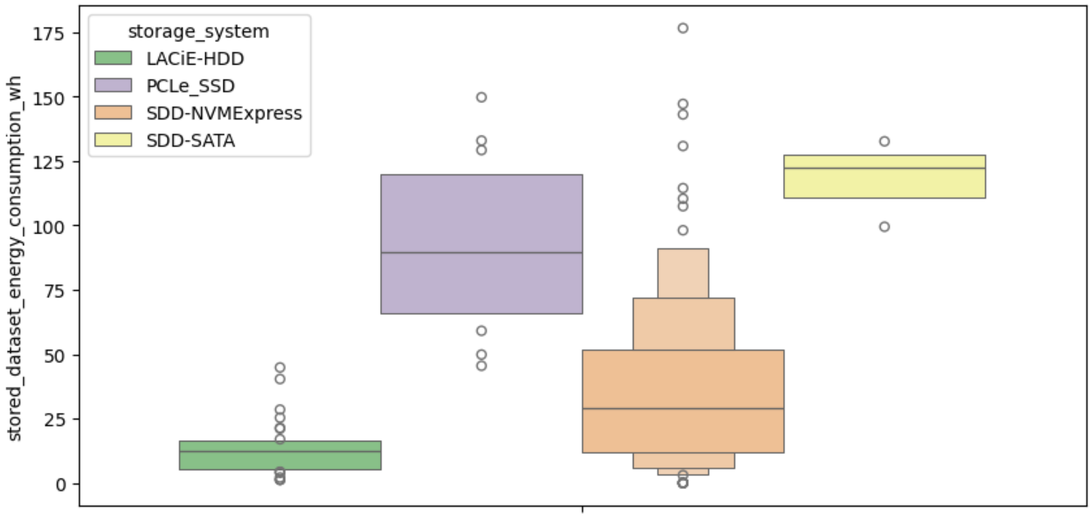

Figure 9: Storage System Energy Consumption Variation

Each storage system varies in actual size. For these experiments, three different storage system sizes were used: 250GB, 500GB and 4000GB. Figure 10 illustrates the variation in energy consumption across different storage system sizes.

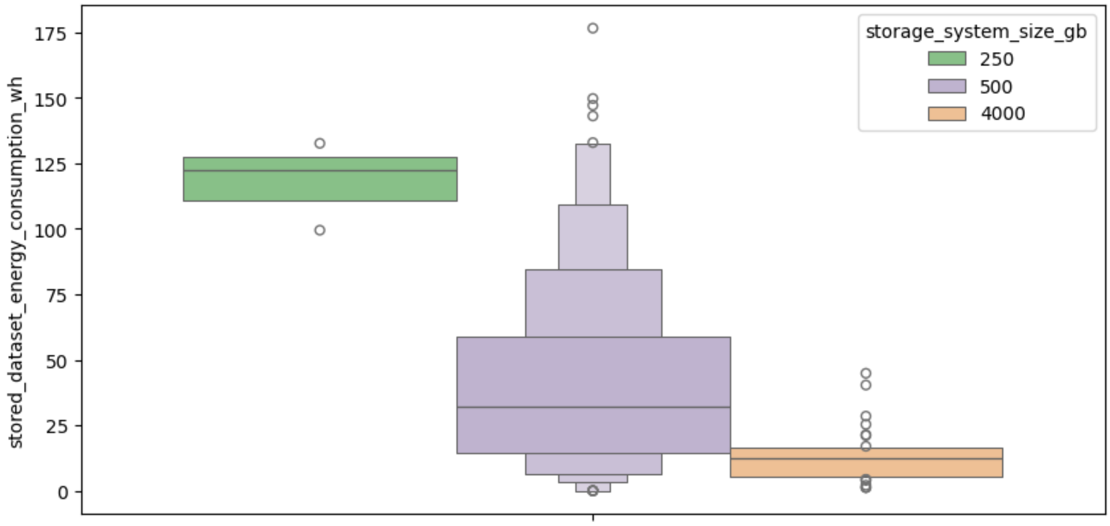

Figure 10: Energy Consumption per Storage System Size

It can be inferred that the size of the storage system majorly influences energy consumption. Figure 11 shows the relation between power consumption and storage time span.

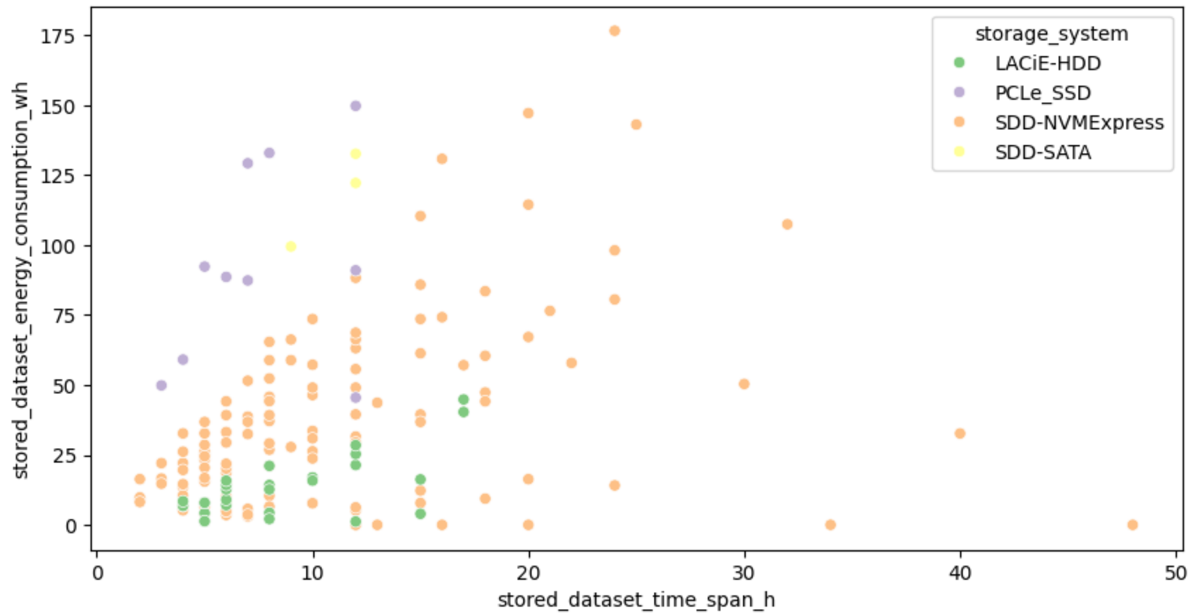

Figure 11: Energy Consumption vs Time Span

Analyses were conducted from multiple perspectives, exploring various features and their impact on the energy consumption of storing datasets. To deepen our understanding of these relationships and predict energy consumption, several regression models were evaluated. Among them, the Random Forest Regressor and Gradient Boosting exhibited promising results, indicating that the features serve as meaningful predictors of energy consumption. R-squared values of approximately 0.81 were obtained as shown in Figure 12

```
regr = RandomForestRegressor(random_state=0)

regr.fit(X_train, y_train)
print(regr.score(X_test, y_test))
```

```
0.8183435566162811
```

```
[67]

reg = GradientBoostingRegressor(random_state=0)
reg.fit(X_train, y_train)

reg.score(X_test, y_test)
```

```
0.8094663552432615
```

Figure 12: Regression Scores (R-Squared)

To visually evaluate the model's performance, refer to Figure 13, where the Energy consumption is plotted against its actual values (Y_true) and the predicted values from the model (Y_pred).

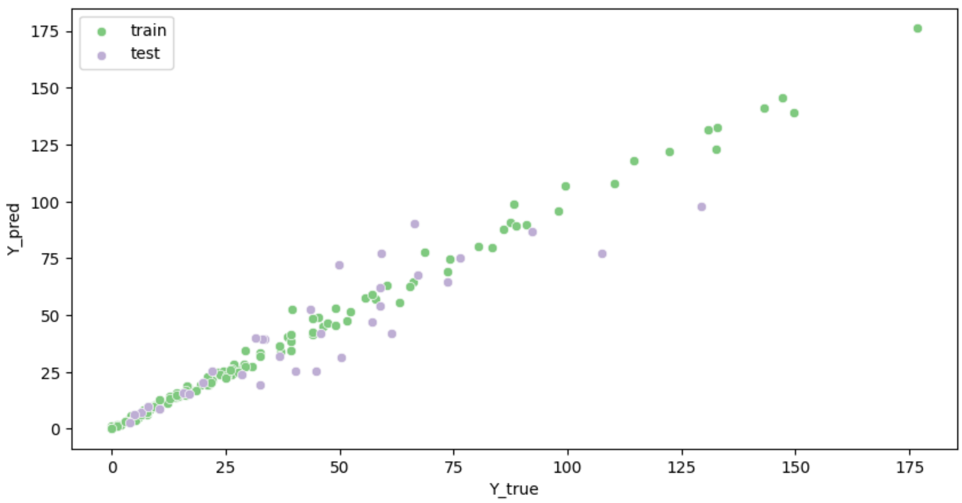

Figure 13: True vs Predicted Values of Energy Consumption

3.2.2 Experiments for Dataset Processing Energy Consumption

The aim of this section is to measure and analyse the energy consumption of processing datasets by applying multiple operations in hardware infrastructures.

3.2.2.1 Methodology

The other core functionality of the UPCAST environmental optimizer plugin is monitoring and quantifying the energy consumption and environmental costs of processing data or applying operations to datasets. The operations can be atomic or AI/machine learning models, and the energy consumption is calculated using a tool provided by CeADAR.

This tool, called Papillon, is a system that monitors processes or workflows that are being executed and calculates its power consumption in real time, every minute. It also collects other system information such as the CPU usage, input/output streams and RAM usage. All this information is stored in its internal database. It employs a client-server architecture, as illustrated in Figure 14.

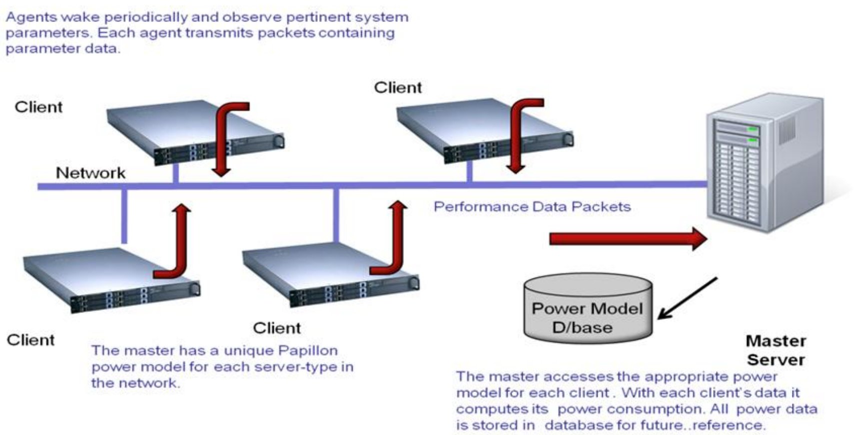

Figure 14: Papillon System

3.2.2.2 Experiment Data & Results

To collect data for analysing data processing workflows and to build a model for estimating their energy consumption, we monitored and calculated the power consumption and environmental costs of processes applied to datasets, using atomic operations and machine learning models across three different computing infrastructures.

- **Datasets:** The experiment utilized open-source time series datasets with three sizes: 400MB, 800MB and 1.4GB.
- **Computing Infrastructure:** The experiment was conducted on three CPU-based hardware infrastructures and three major operating systems: Linux, MacOS, and Windows.
- **Operations:** Python scripts were used to execute atomic operations and machine learning models across the three computing infrastructures, with power consumption calculated and stored in a MySQL database.

Table 4 provides details on the hardware infrastructure and Table 5 lists the set of operations used in the experiments.

Table 4: Hardware Infrastructure Used for Experiments

Operating System	CPU Cores	Processor Speed	Memory (RAM)
Ubuntu	4	2.60 GHz	16 GB
Windows	5	2.00 GHz	16 GB
MacOS	12	2.70 GHz	32 GB

Table 5: List of Operations Used for Experiments

Operation Name	Operation Type	Description
append	Atomic	Add more data to the dataset
count_num_rows	Atomic	Count the total number of rows in the dataset
detecting_missing_values	Atomic	Find how many values are missing
filter_by_spec_cols	Atomic	Retrieve specific feature
groupby_count	Atomic	Get the count of values grouped by feature
groupby_id	Atomic	Group data points by ID
Identify_outlier	Atomic	Detect datapoints with abnormal values
mean	Atomic	Calculate the average value of a feature
remove_null_values	Atomic	Remove all missing values
replace_null_values	Atomic	Replace missing values with another value
select_by_id	Atomic	Retrieve values based on the ID
union_col	Atomic	Combine values of different features
update_atomic_op	Atomic	Update values of all features
update_spe_cols	Atomic	Update values of a specific feature
LSTM_model	ML Model	Applying the LSTM Deep Learning model
RF	ML Model	Applying the Random Forest model

Table 5 shows that several atomic operations have been considered. This is because a data processing workflow can have multiple operations and the possibilities are endless, however, most machine learning or AI workflows will use a combination of these atomic operations, especially when preprocessing or manipulating data to gather insights.

Table 6 shows the detailed results of tests carried out on the 1.4 GB dataset across multiple operations and the three hardware infrastructures.

Table 6: Results of Experiments Applied to 1.4 GB Dataset

Operating System	Operation Name	Energy Consumption (Wh)	CO2 (kg/kWh)	Energy Cost (Euros)
Ubuntu VM	LSTM_model	130.072909	0.03902187	0.04682625
	RF	50.0583945	0.01501752	0.01802102
	Atomic_op_append	1.77976778	0.00053393	0.00064072
	LSTMModel	19.2859297	0.00578578	0.00694293
	Count_num_rows	7.33354578	0.00220006	0.00264008
	Detecting_missing_values	3.67780292	0.00110334	0.00132401
	Filter_by_spec_cols	1.8960032	0.0005688	0.00068256
	groupby_count	3.55127286	0.00106538	0.00127846
	Groupby_id	1.77855303	0.00053357	0.00064028
	Identify_outlier	3.63071122	0.00108921	0.00130706
	mean	27.5538627	0.00826616	0.00991939
	Remove_null_values	3.65483054	0.00109645	0.00131574
	Replace_null_values	20.0985895	0.00602958	0.00723549
	Select_by_id	1.77855303	0.00053357	0.00064028
	Union_col	1.87721062	0.00056316	0.0006758
	Update_atomic_op	1.87346224	0.00056204	0.00067445
	Update_spe_cols	28.0857777	0.00842573	0.01011088
Windows OS	LSTM_model	139.550729	0.04186522	0.05023826
	RF	48.8511355	0.01465534	0.01758641
	Atomic_op_append	3.4759414	0.00104278	0.00125134
	Count_num_rows	7.3835996	0.00221508	0.0026581
	Detecting_missing_values	3.6033893	0.00108102	0.00129722
	Filter_by_spec_cols	1.845362	0.00055361	0.00066433
	groupby_count	5.56337	0.00166901	0.00200281
	Groupby_id	3.674023	0.00110221	0.00132265
	Identify_outlier	3.6441	0.00109323	0.00131188
	mean	22.0942465	0.00662827	0.00795393
	Remove_null_values	3.650979	0.00109529	0.00131435
	Replace_null_values	18.4336174	0.00553009	0.0066361
	Select_by_id	1.826746	0.00054802	0.00065763
	Union_col	1.8895483	0.00056686	0.00068024
	Update_atomic_op	1.93475535	0.00058043	0.00069651
	Update_spe_cols	21.7644609	0.00652934	0.00783521
Mac OS	LSTM_model	125.615685	0.03768471	0.04522165
	RF	46.9733807	0.01409201	0.01691042
	Atomic_op_append	1.85035494	0.00055511	0.00066613

	Count_num_rows	7.44746256	0.00223424	0.00268109
	Detecting_missing_values	3.6678349	0.00110035	0.00132042
	Filter_by_spec_cols	1.86764623	0.00056029	0.00067235
	groupby_count	3.70329923	0.00111099	0.00133319
	Groupby_id	3.9859174	0.00119578	0.00143493
	Identify_outlier	3.68622799	0.00110587	0.00132704
	mean	22.3091912	0.00669276	0.00803131
	Remove_null_values	3.70658204	0.00111197	0.00133437
	Replace_null_values	18.4301561	0.00552905	0.00663486
	Select_by_id	1.83522424	0.00055057	0.00066068
	Union_col	1.8523434	0.0005557	0.00066684
	Update_atomic_op	1.89483734	0.00056845	0.00068214
	Update_spe_cols	22.1358526	0.00664076	0.00796891

The measured results indicate that the ML models consumed more significant energy than atomic operations, causing a substantial environmental impact. The LSTM ML model consumes an average of 131Wh on a 1.4GB dataset across the three computing infrastructures. The experiment demonstrates that the same principle works with various datasets and types across different computing infrastructures that have access to them.

3.2.2.3 Insights from Experiments

This subsection describes the data generated from the experiments and some key insights gained that can inform how power consumption varies with different operations or processes applied to datasets.

Several observations were collected during the execution of the experiments (over 1000 data points), however, after applying preprocessing techniques and removing outliers, the final cleaned operations dataset that has been analyzed contains 137 observations with 9 features that have been described in Table 7.

Table 7: Operations Dataset Features

Feature Name	Description
script	The script name used in the experiment
power	The energy consumption associated with the operation (in Wh)
cpu	Processing unit being used in the experiment
storage	The amount of RAM being consumed (in GB)
IO	The number of input/output streams associated with the operation
co2	The CO2 emissions associated with the operation (in g/kWh)
host	The host system name where the experiment was executed – values are 284, 285 or 286
db	The total size of the dataset (in MB)
powerMode	The power agent generated by the master for every host.

Figure 15 details the summary statistics of the key outputs for all the observations in the data. The mean energy consumption is 4.8Wh, and the mean CO2 emission is 1.4 grams/kWh.

	count	mean	std	min	max
power	137.0	4.886564	5.205796	1.431191	27.553863
co2	137.0	1.465969	1.561739	0.429357	8.266159

Figure 15: Summary Statistics

The histograms presented in Figure 16 show the distribution of the CO2 emissions and Total energy consumption for processing datasets.

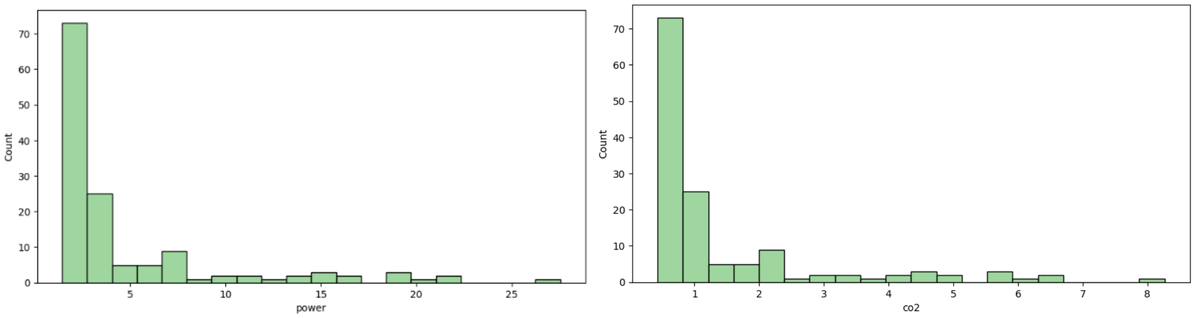

Figure 16: Histograms of Power Consumption and CO2 Emissions

The histograms indicate that the energy consumption and CO2 emission have a right skewed distribution. This is because most observations relate to atomic operations applied to the datasets that consume less power than heavier modelling processes.

Similar to the analyses performed for the storage energy consumption, we visually explored relationships between features in the processed data. A key finding was that dataset size ("db") significantly impacted energy usage.

The following box plot in Figure 17 shows the energy consumption variation compared to dataset size.

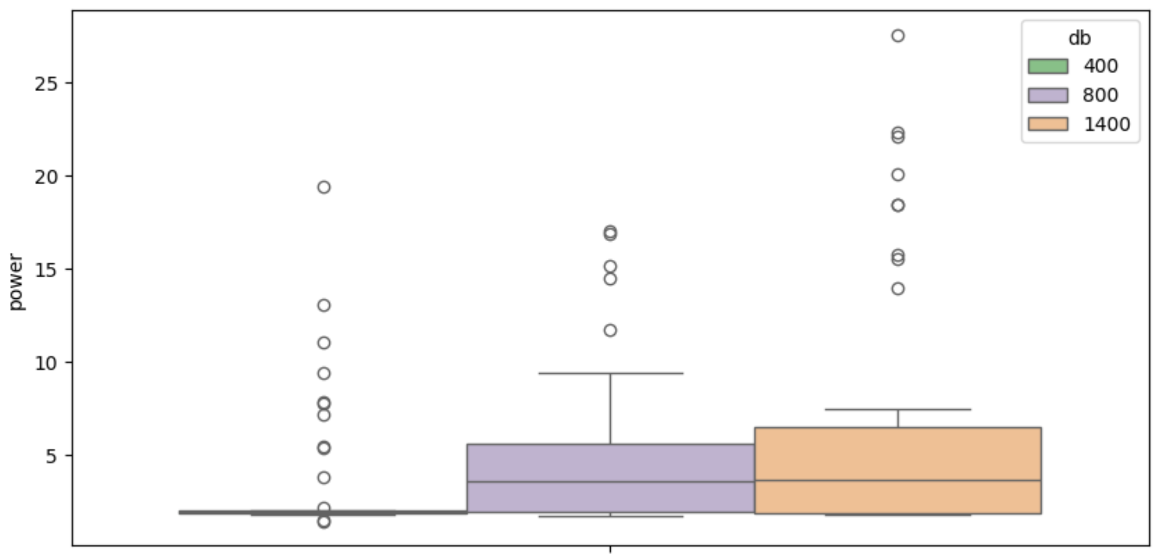

Figure 17: Energy Consumption for Different Dataset Sizes

Logically, it can be inferred that the power consumption for operations applied to the largest dataset (1400 MB) has a higher range than the other datasets. The median value, however, for the 800MB dataset and 1400 MB dataset are similar, whereas for the 400MB dataset, it is much lower.

Another noteworthy observation in the following box plot in Figure 18 that shows the energy consumption variation for each of the three host systems is that there are similar energy consumption patterns for each system.

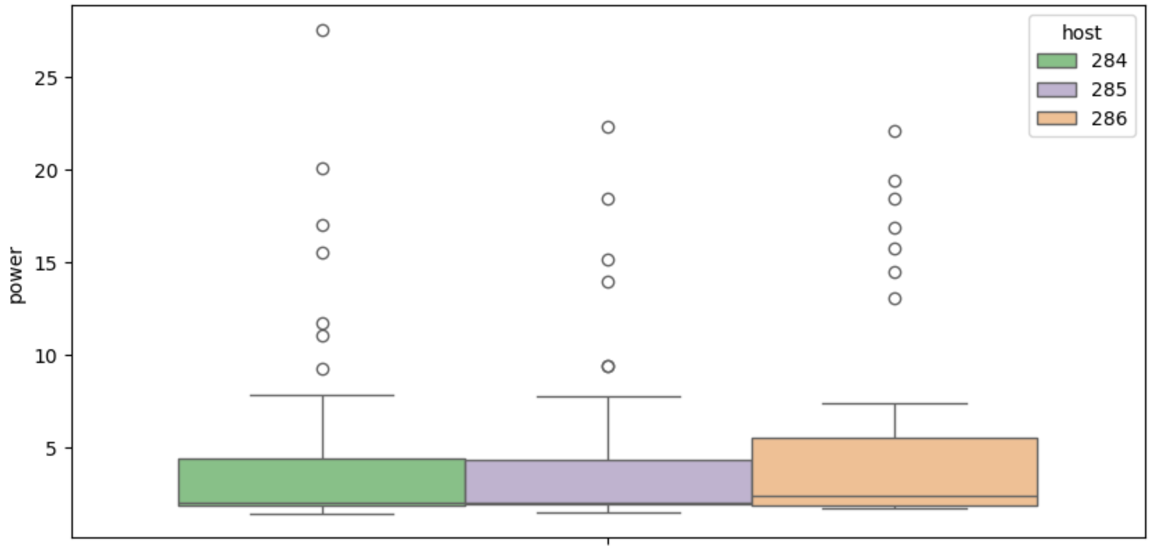

Figure 18: Energy Consumption for Different Host Systems

The following scatter plot in Figure 19 shows the relationship between the power consumption and CPU utilization.

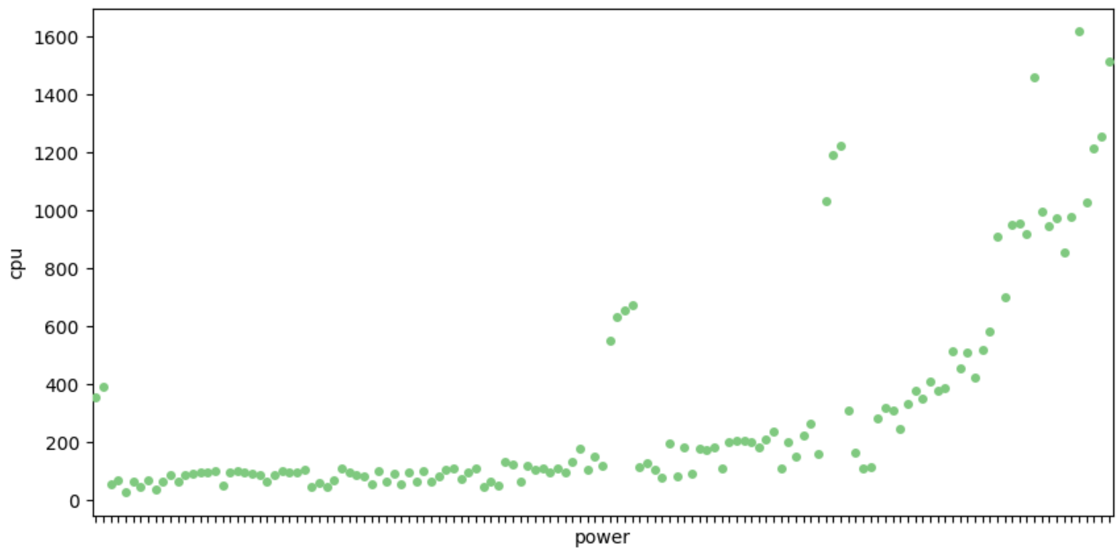

Figure 19: Power Consumption vs CPU Utilization

Experimenting with more complex operations with larger datasets will allow us to gather data with higher power consumption. With the data that has been generated for the first version of the plugin, regression models were applied to it that illustrate the relationships between the feature that needs to be predicted (power consumption) and the independent observations. For this analysis, Decision Tree and Random Forest Regressors were applied and the model performed well with R-square values of 0.84 and 0.87

```
[224]
rg = DecisionTreeRegressor(random_state = 1, max_depth=3)
rg.fit(X_train, y_train)
rg.score(X_test, y_test)

0.845336338018772

[225]
rg = RandomForestRegressor(random_state = 1)
rg.fit(X_train, y_train)
rg.score(X_test, y_test)

0.8770959225680914
```

Figure 20: Regression Scores (R-Squared)

Model performance can be inspected visually in Figure 21 by plotting the Energy consumption by comparing its actual value (Y_true) with the value predicted by the model (Y_pred).

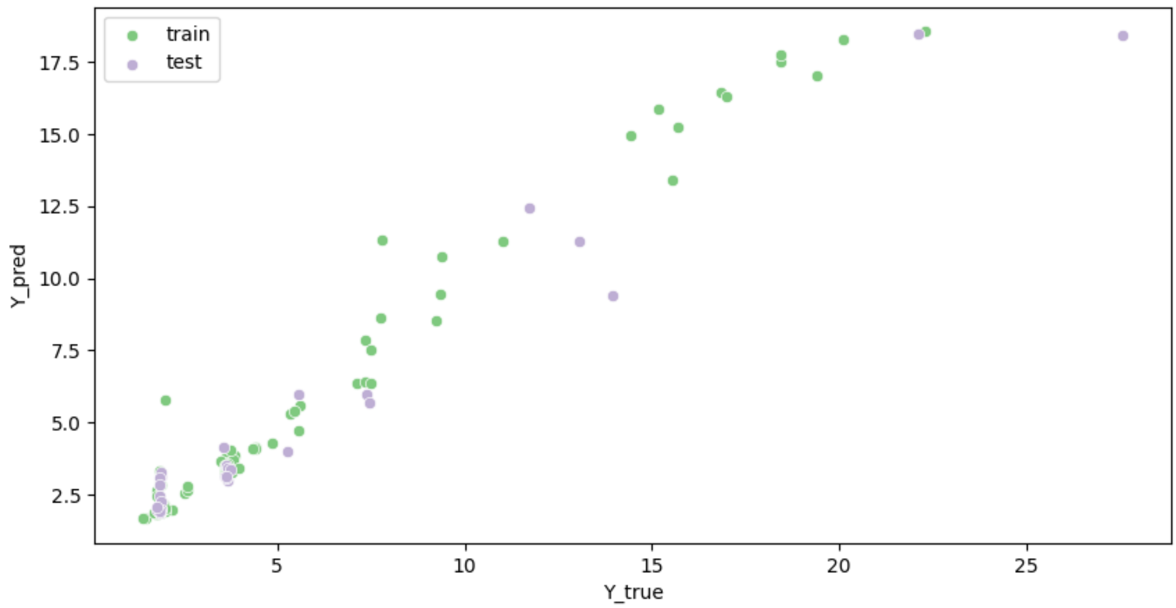

Figure 21: True vs Predicted Values of Energy Consumption

The total energy consumption comprises the dataset storage energy consumption and the workflow energy consumption. To calculate the total cost, we need to consider the carbon cost and the energy cost associated with the total energy consumption. This total energy consumption is the sum of the energy consumption associated with storing a dataset and the energy consumption associated with processing the dataset.

3.3 Swagger API Interface

Figure 22 shows the REST API Swagger specification. There are three endpoints: one to compute the energy consumption, one to estimate it and the third endpoint will be used to explain the energy consumption estimation. The current OpenAPI specification is available in the project's Github repository here:
https://github.com/EU-UPCAST/OpenAPISpecification/tree/main/environmental_plugin

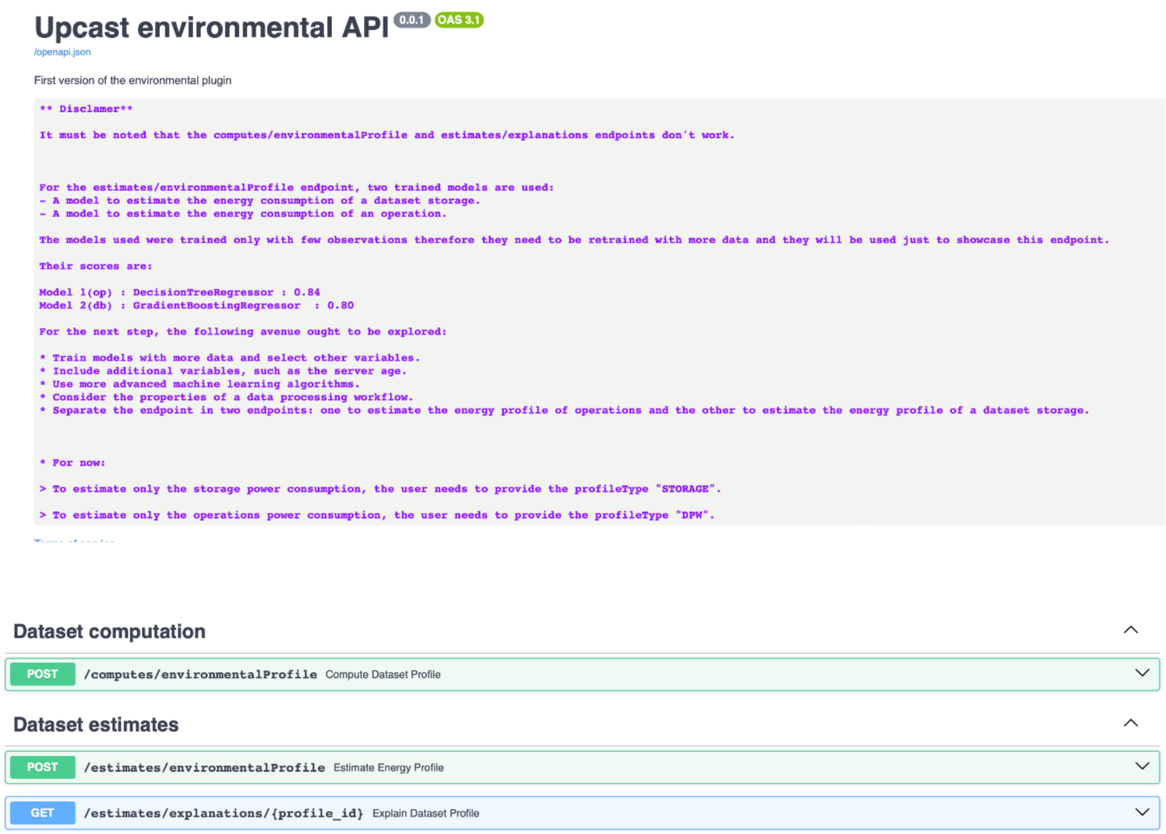

Figure 22: REST API Swagger Specification

Figures 23 to 26 show the Swagger interfaces of the request and response for the dataset storage power consumption estimation, for a sample resource specification with relevant hardware and computing infrastructure properties. The final version of the plugin will access the energy profiling models when called via the endpoint.

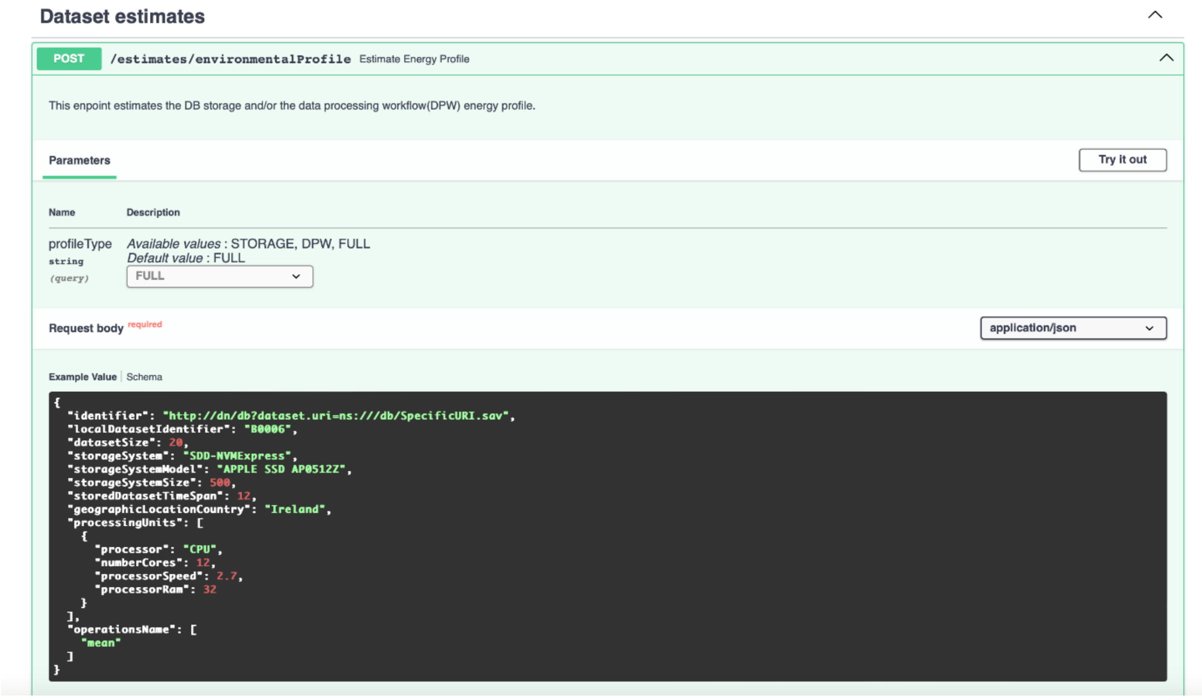

Figure 23: Sample request

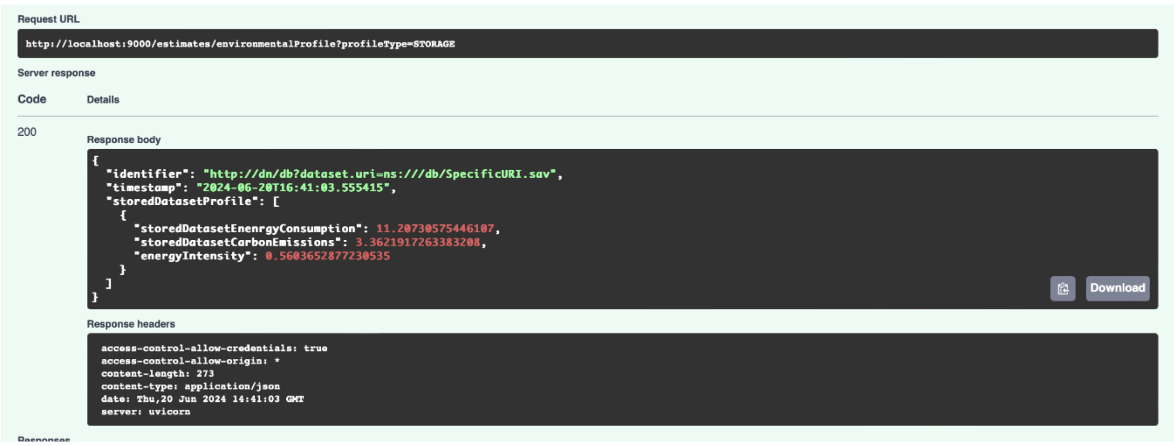

Figure 24: Sample Response profileType=STORAGE

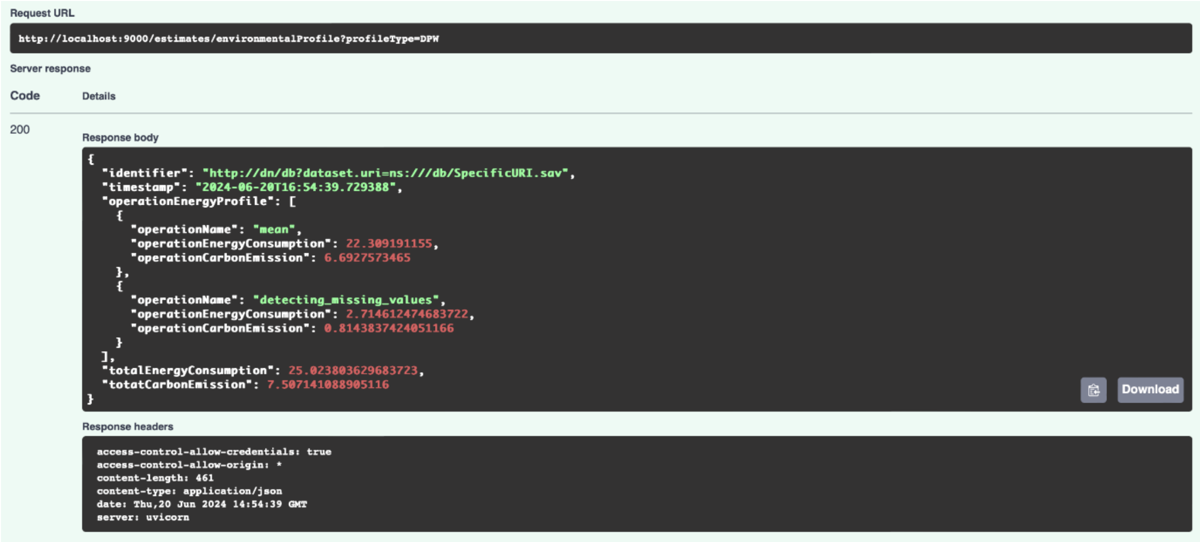

Figure 25: Sample Response profileType=DPW

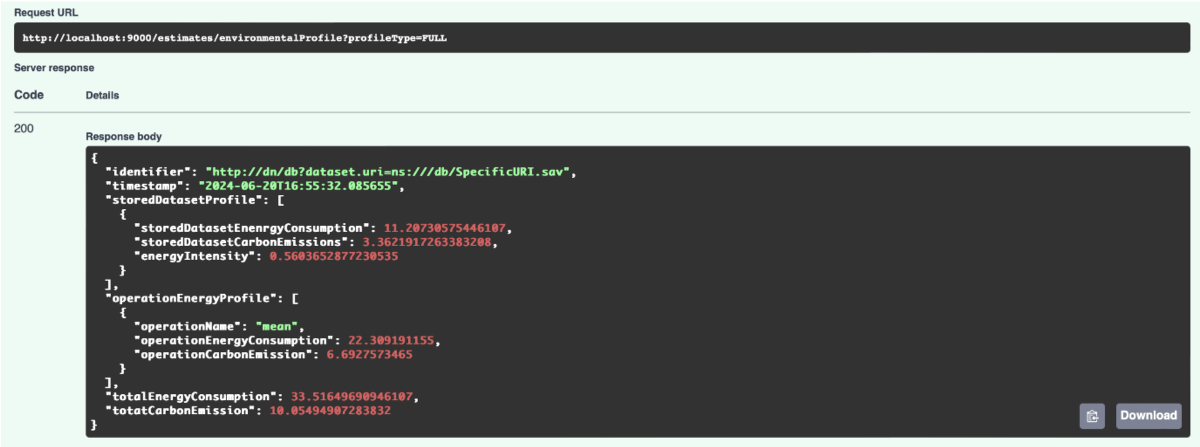

Figure 26: Sample Response profileType=FULL

4 Conclusion and Next Steps

This deliverable is the first version of the environmental optimizer plugin that described the architecture, initial experiments performed and the API specification showing the progress of its design and development.

A major challenge for this task is the lack of a dedicated execution environment. This absence has prevented us from running more standardized monitoring and benchmarking tests. Consequently, we had to adapt and conduct experiments on infrastructure that was available to us.

For the next deliverable, which is the final version of the plugin, the following will be explored:

- Monitoring the power consumption of processes or AI models running in the GPU. Currently, monitoring is done with regards to the CPU power consumption.

- To gather more insights and data, the experiments will diversify in terms of using different types of datasets (such as unstructured image or textual data) and will also be executed on higher performance hardware infrastructures.
- Energy profiles will be generated by applying unsupervised techniques to the experiment data, such as clustering to find meaningful groups. This is contextual and will depend on each data processing workflow or workload.
- Explainable AI techniques (XAI), like those being applied to the pricing plugin will also be used to gain a better interpretation of the energy consumption. A report will include these explanations along with recommendations on how the end user can optimize their workflow or reduce consumption. These can also aid in creating the energy profiles.
- Integration of the plugin within the wider UPCAST framework and data marketplaces/platforms.

ANNEX I ACRONYMS

Acronyms List	
DoA	Description of Action
DPW	Data Processing Workflow
EIO	Environmental Impact Optimizer
XAI	Explainable AI
API	Application Programming Interface
LSTM	Long Short-Term Memory Neural Network
RF	Random Forest
VM	Virtual Machine
OS	Operating System
GPU	Graphics Processing Unit
TPU	Tensor Processing Unit