OVD-SaaS: Online Verifiable Data Science as a Service, an architecture of microservices for industrial artificial-intelligence applications: Architecture and study cases

Jose Armando Hernandez Gonzalez, Thibaut Germain, Billel Yadoughi, Miguel Colom

OVD-SaaS: Online Verifiable Data Science as a Service, an architecture of microservices for industrial artificial-intelligence applications : Architecture and study cases

Jose Armando Hernandez
Centre Borelli ENS-Paris Saclay
Paris Saclay University
Gif-sur-Yvette, France
0000-0002-6692-8640

Thibaut Germain
Centre Borelli
ENS-Paris Saclay
Gif-sur-Yvette, France

Billel Yadoughi
Centre Borelli
ENS-Paris Saclay
Gif-sur-Yvette, France

Miguel Colom
Centre Borelli
ENS-Paris Saclay
Gif-sur-Yvette, France
0000-0003-2636-0656

***Abstract*—There is a growing concern about credibility and trustworthiness in results and claims of research in computational data science, and at the same time, difficulty in taking those results to MVP (Minimum viable product) in technology transfer from the laboratory to the industry. Therefore, We present the OVDSaaS prototype, a new platform to manage Data Science Research products (publication, code, data, MVP) that provides reproducibility certification, traceability, provenance, authenticity, legitimacy, reward, FAIR (Findable, Accessible, Interoperable, Reusable) compliance, and valorization to ML/AI computer scientific research by developing useful online verifiable scientific applications MVP in different industrial domains.**

To bridge the gap among academia, industry, publishers, and ML/AI technology, we propose an architecture design, methodology, policies, and guidelines for the OVD-SaaS project to advance in reproducibility assessment, persistent identification, validation, and badging of research artifacts results.

We present our Scientific Applications case study in the context of reliable scientific research and the validation of results from scientific articles. We describe our experience creating scientific applications and their evolution from simple Demo or non-reproducible artifacts to completely verifiable apps.

We analyze common patterns in the creation of scientific applications, discussing the benefits and difficulties encountered that lead to the proposed OVDSaaS features. In this context, four applications were developed for breathing analysis, clinical tumor segmentation, legal assistance, and smart grids, supported in ML/AI scientific fields such as signal processing, image processing, and NLP. These applications represent end-to-end business solutions cases in the medical, legal, and energy industries. Based on these cases, we performed a reproducibility gap analysis and benchmarking of similar OVDSaaS platforms.

***Index Terms*—PID, Blockchain, IPFS, reproducibility, credibility, trustworthiness, authenticity, legitimacy, FAIR, accountability, transparency, BPML**

I. Introduction

In recent years, the article [1] raised alarms and concerns about an important reproducibility crisis that significantly questioned the credibility and trustworthiness and, in a certain way, the ethics [2] or complex social implications of Data Science research publications. The difficulties in replicating and validating research findings and supporting claims have led to an increasing demand for transparent and reproducible research practices. Journals and conferences have responded by emphasizing reproducibility policies [3] aimed at ensuring that published research can be independently verified and extended by the broader scientific community. However, the complexity lies in the wide variety of technologies, programming languages, hardware, and topics made complicate the reproducibility assessment and the creation of a universal, standardizable, generalizable strategy, methodology, or agnostic tool that effectively optimizes the value of scientific research and rewards all stakeholders, including academia, industry, government, authors, publishers/ editors, and society.

Other works quantitatively highlight the problem of computational reproducibility of the submitted papers in several conferences and journals, demonstrating a low reproducibility percentage that did not exceed 24.9% [4]. Additionally, in the Artifact Evaluation Report of the 56th Annual IEEE/ACM MICRO23, only 26 submissions (26% of all accepted papers) were awarded a reproducibility badge [5]. Therefore, the reproducibility of ML/AI scientific experiments has become a major concern and the subject of multiple initiatives. Several strategies and policies that facilitate the reproducibility of results have been identified in [3].

In most cases, the reproducibility, replication, and validation of results are presented partially through online demos, artifacts, code snippets, and specific workflows to prove research claims. However, the reproducibility of the end-to-end process from the development of the model to its implementation in a real business case is not analyzed heuristically. It has been argued that 60% to 80% of ML/AI models never reach production [6]. In short, it worryingly means that a maximum of 10% (0.249*0.4) of the submitted papers are reproducible, and their models become functional in a real solution.

Traditionally, reaching a commercial solution from Data Science research occurs in a slow serial process since (1) the author(s) submit their scientific research article to a journal or conference, (2) Until industrial domain experts draw their attention to these new AI/ML advances, they must evaluate the credibility and decide on implementing applications for the general public in new business models. In this context, it is common to observe that the industries' needs and requirements are forgotten in research work, and the research topics are disconnected from the real problems.

Recently, to go a step further than simply publishing the code in a Github repository, there is growing movement towards open science, Papers with Code PwC[1], IPOL [7], Runmycode [8], Dagshub [9], CodeOcean [10], Chameleon [11] with KheOps [12], and Whole of Tale [13], and others initiatives, which have facilitated many aspects about the automation of reproducibility and replicability of findings and experiments. However, these concentrate on solving a reproducibility-specific aspect and are limited by design to online demos or the execution of code snippets that describe the main algorithms or methods of the article.

OVD-SaaS (Online Verifiable Data Science as a Service) [9] addresses these complex, open challenges by conceptualizing and prototyping a novel microservices-oriented framework for building reproducible scientific-industrial applications in AI/ML. The platform is designed to facilitate the rapid development of MVPs that can serve as the foundation for creating small and medium-sized enterprises (SMEs) and startups. Each OVDSaaS product will include a fully reproducible codebase, an associated scientific publication, and a corresponding application MVP, thereby bridging the gap between academic research and industrial application.

The primary objective of OVDSaaS is to address critical issues of credibility, result corroboration, and research claim verification within the AI/ML field. By ensuring compliance with the FAIR (Findable, Accessible, Interoperable, and Reusable) [14] principles, OVDSaaS not only enhances the reproducibility of Data Science research but also increases its value through the seamless transfer of technology from the lab to industry. This platform aims to empower researchers to create and share high-quality, reproducible research that can be readily transformed into commercial solutions, fostering innovation and startup creation, contributing to the growth of the AI/ML ecosystem and Social Impact of Data Science Applications.

OVDSaaS brings together the experience of IPOL [7] journal in Centre Borelli to go beyond demos through optimization and parallelization of tasks and processes in a collaborative environment. It is a dissemination tool that increases impact, rewards, and values and makes visible Data Science research, positioning itself as a social network that connects the requirements of the productive industrial sector with the advances in research ML/AI.

The document is organized as follows. Section II briefly reviews the state of the art in reproducibility of applications derived from scientific articles, followed by a comparison between *demos* and complete *applications*. We address 3rd-party libraries and initiatives related to reproducibility of applications. In Section III, The OVDSaaS architecture prototype and methodology are presented, which allows for automation, simplification, and low code development of reproducible scientific applications in different industrial domains. Section IV discusses the experiences using an argument strategy by examples. It provides a brief reproducibility gap analysis that identifies patterns and common requirements for reproducible scientific applications in the four use cases we analyze. We highlight the evolution from a no-reproducible application Smartgrid case to a complete OVD-SaaS reproducible application PethAPP case. Finally, a benchmark is made with existing tools OVDSaaS similar. Section V concludes the paper and presents ideas for future work.

[1]https://paperswithcode.com/

II. Reproducibility in Scientific Applications

The main differential, contribution, and value promise of OVDSaaS are to facilitate and provide service creation, assessment, PID assignment, validation, certification, and rewarding (with badges) reproducibility for scientific applications based on scientific research. Therefore, it is crucial to establish the differences between demos/artifacts and applications and show the additional aspects to consider when evaluating the reproducibility of applications compared to the traditional evaluation of artifacts/Demos and other initiatives.

Comparison of online demos and complete applications

Several platforms [7] [8] [9] [10] [11] [12] [13] allow one to quickly obtain results from executing algorithms associated with scientific articles. However, they are not *complete* applications, given that they lack the extra functionality that might be required in end-to-end scenarios with more complex requirements. These do not include the persistence of the data or security aspects, among others. Table I summarizes and compares these characteristics of *demos/ simple artifacts* and complete *applications*.

TABLE I
Comparison between characteristics of *Demos, simple artifacts* and complete *Applications*.

Characteristics	Demo/ Simple Artifact	Complete Applications
Online Accessibility	Limited functionality for demonstration purposes. Restricted access.	Full-featured and accessible to users. Enhanced accessibility.
Functionality Workflow	Basic or partial features. Limited capabilities. Demonstrates core aspects.	Comprehensive functions Domain and Business capabilities. Fully operational Workflow.
UI Interactivity	Minimal or no UI customization options. Pre-defined settings.	High customization Controls UI options and settings. Tailored to user preferences.
Persistence	Limited customization options. Pre-defined settings.	Different Data sources DB options and settings. Tailored to user preferences.
Security	None customization options. Pre-defined settings.	RBAC (Role Base Access Control) AAA (Authentication, Authorization) Tailored to user preferences.

Reproducibility assessment of applications

Reproducibility assessment of applications is not so simple, considering that there is no standardized way to do it. Several works have proposed qualitative and quantitative methods, and generally only for artifacts/demos evaluation. Therefore, we must adapt some of the OVDSaaS assessment criteria to the specific characteristics already mentioned in the applications.

Table II summarizes reference elements and objectives sought to be met in each of the case applications, many of them come from our previous analysis [3] and the proposed by Gundersen for platforms [15], which identifies three main elements to check, Method, Data, and Experiment.

A few initiatives in reproducibility of scientific applications

We present some relevant initiatives related to application reproducibility. These enable rapid prototyping, and we can use some of their elements and approaches in our proposal to provide a general reference architecture for end-to-end reproducible applications. We do not address the numerous initiatives related to reproducibility [3] that could eventually be applied to applications, and we limit ourselves to mentioning those relevant to our OVD-SaaS proposal.

Marcelle [16] allows the creation of machine-learning (ML) interfaces in simple workflows. Although this framework does not allow for building complete and complex applications, it helps with prototyping interactivity.

Cknowledge, CM (Collective Mind) [17] is a complete language that addresses reproducibility of AI/ML models. CM has proven to be very efficient in artifact evaluation, reproducibility, and automation of experiments for state-of-the-art benchmarks for MLperf of MLCommons; and has been used successfully in conferences with reproducibility component [5][2]. The possibility of reproducibility of applications associated with ML models is nevertheless in progress. Similarly, GaNDLF[3] proposes a framework for scalable end-to-end clinical workflows but requires more elements to become a clinical reproducible application.

III. Design of OVD-SaaS architecture and proposed methodology

The key contribution of OVD-SaaS is a framework incorporated in an integrated platform that relies on state-of-the-art technologies (microservices, blockchain, cloud computing) and best practices (MLops, scrum) to manage the overall process (publication, data, code, MVP business model) from the conception of scientific research to the industrial implementation of an MVP reflected in the development of a reproducible scientific application, in an agile end-to-end life cycle, through optimization, parallelization of tasks and processes in a collaborative environment.

The OVD-SaaS project facilitates, guides, and advises the development of applications and their industrialization, from lab research to end-to-end prototypes. Incorporates open science principles, policies, recommendations, and guidelines surveyed in our research [3] that identify common patterns and needs that help design OVD-SaaS. These works highlight a reproducibility gap and challenges that must be addressed; in this sense, OVDSaaS bridges the gap from four approaches or points of view: that of the Academia (authors, Labs), publishers (journals, conference, reviewers), the industrial business, and the reproducibility cloud technology available.

[2]https://ctuning.org/ae/micro2023.html
[3]https://doi.org/10.1038/s44172-023-00066-3

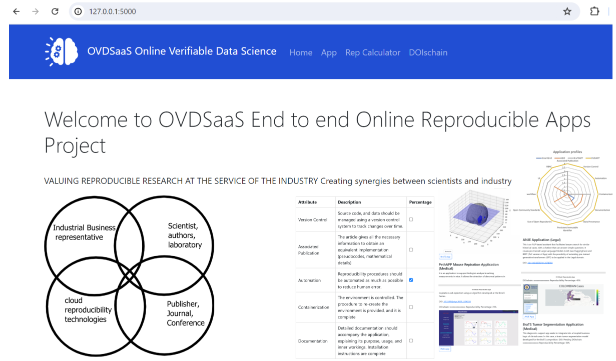

Fig. 1. OVDSaaS User Interface. Four sections (Home, App, RepCalculator, DOIschain). OVDSaaS presentation, Access to apps, Reproducibility assessment, and DOIschain issuance, respectively

Figure 1 shows these four factors on the Home Webpage OVDSaaS to create Reproducible, Replicable, Repeatable, Reusable, Traceable, Corroborate, Verifiable, and Reliable scientific applications that address industrial problems. OVD-SaaS establishes a scientific social network community and connects and promotes interactions among industrial partners, publishers, reviewers, and researchers in these interactions, Figure 2.

OVDSaaS also incorporates blockchain technologies for assigning perennial persistent identifiers PIDs (we call ”DOIs-chain”) to provide provenance, versioning, lineage, and traceability for research products (publication, software, and dataset). **Storage** In addition to the typical github and other open repositories, it proposes IPFS (Interplanetary File System). Reward the reproducible research by badging and use software engineering methods such as user storytelling to determine user and industry requirements in business logic in an iterative process we call ”model ready to business” described by high-level language BPML Figure. 4 3.

The general architecture is described and Figure 5, the **Core** interacts with other editorial information systems like IPOL journal, and open repositories for managing articles and MVP documents. It uses Kubernetes to manage, assign resources, deploy, and execute reproducible scientific applications in MLops [18] cycles (CI/CD). Experimentally, it is compatible with the CM-knowledge collective. The core maintains the database of data from authors, reviewers, industrial partners, developers, reproducibility and other impact metrics, and the web page where scientific products are submitted for review and end up with a reproducible scientific application MVP associated with publication to promote Findable, Accessible, Interoperable, Reusable (FAIR) compliance [14].

From our research [3], we found that a microservices architecture is the most suitable for supporting multi-languages (e.g., Python, Java, Rust, c++, etc.), integrating dissimilar legacy technologies with new advances such as serverless functions to offering RaaS (Reproducibility as a Service) services and language agnostic technologies.

This architecture permits the rapid implementation of some

TABLE II

DESIRABLE ATTRIBUTES FOR ASSESSMENT REPRODUCIBILITY IN SCIENTIFIC APPLICATIONS

Attribute	Description
Associated Publication (10%)	Method - The article gives all the necessary information to obtain an equivalent implementation (pseudocodes, mathematical details). The article discusses and explains both the positive and negative results. The limitations are explained. The experimental setup in the article is adequate to provide evidence and support the claims. The code corresponds to the pseudocode descriptions in the article. The pseudocodes are given with adequate granularity. The fundamental steps are described or referred to in the text
Version Control (10%)	Data - Source code and data should be managed using a version control system to track changes over time for a particular revision (say, commit ID) of the source
Automation (10%)	Experiment -Reproducibility procedures should be automated as much as possible to reduce human error. The code has automatic tests to ensure result artifacts have not changed along refactoring
Containerization /virtualization (10%)	Experiment- The environment is controlled. The procedure to re-create the environment is provided, and it is complete. Packaging the application and its dependencies into containers ensures consistent execution environments.
Documentation (10%)	Method- Detailed documentation should accompany the application, explaining its purpose, usage, and inner workings. Installation instructions are complete. They specify the exact version numbers of the libraries it requires
Data Provenance (10%)	Data - Data sources and transformations should be well-documented to trace the origin of the results.
Validation, Review, Evaluation (10%)	Experiment/Method/Data - Validation of research claims and application results against known datasets or analytical solutions to ensure correctness and reproducibility.
Persistent Immutable Identifier (10%)	Experiment - Assign a persistent immutable identifier to the application to facilitate citation and reference. (e.g., SWHID, DOI)
Use of Open Repositories (10%)	Data - Utilize open repositories for storing source code, data, and documentation to enable easy access and collaboration. The data associated with the project is FAIR and allows authors to replicate the results in the article, including any figures. For example, the data is on Zenodo.
Licensing (10%)	Data - The code is under a FOSS license (GPL, BSD, . . .)
Open Community Standards (10%)	Data - Adhere to community standards and best practices for data formatting, analysis, and reporting. The data and algorithms are encoded with standard, non-proprietary formats

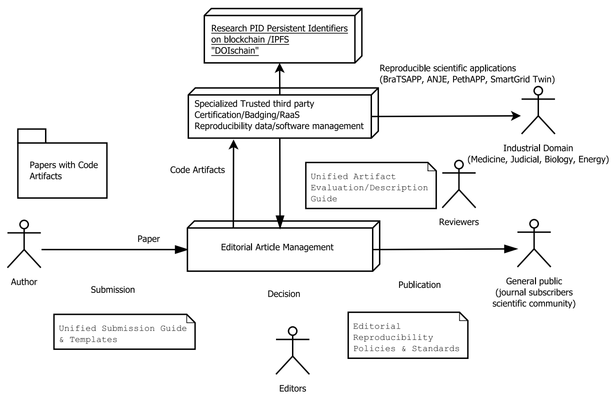

Fig. 2. OVDSaaS multidisciplinary approach: Input- research products, Output-Online reproducible scientific application, Badge, publication and DOIschains. Integrating, articulating, and coordinating synergies, between the actors involved in the process (authors, editors, reviewers, Business Analysts)

of the common software development patterns found in many end-to-end applications with characteristics mentioned in Table I, which include the creation of UI forms and their validation, the definition of roles, authentication, and associated tasks in business logic workflows. The overall process follows agile methodologies.

Researchers should be able to easily, quickly integrate and replicate the results of their research within a business logic where several actors with different roles interact with the application.

To replicate scientific applications, we propose the use of common microservice design patterns [19] as CQRS (Command Query Responsibility Segregation), database independent per app, Event-Driven use of Kafka broker for message exchange and Kubernetes as a docker container orchestrator of microservices and lambda functions. Figure 4 describes the interaction of microservice endpoints with the business logic through standardized JSON REST messages.

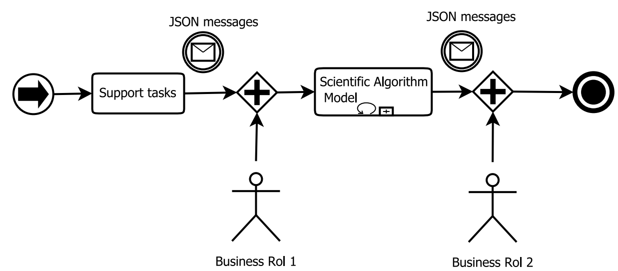

Fig. 3. A scientific ML/AI model integrated into a high-level BPML business logic. "Model ready to Business"

Bonitasoft [4] is used to implement the BPML language to facilitate the agile and high-level development of business logic and allow the orchestration or choreographing of the APIREST services. This approach has benefits orchestration tools [5] and complements Kubernetes, tool because it allows the additional coordination of people (roles in the application) and even robots (RPA).

All this process leads to the "model ready to business" OVDSaaS concept described in Figure 3; this shows a scientific ML/AI model integrated into a high-level BPML business logic.

A. Measure Computational research Reproducibility and FAIR assessment

A special module is required to review and assess the reproducibility of computational scientific research that reflects the degree to which the claims and results of the research are (reproducible, repeatable, replicable, reusable, verifiable, corroborated, credible, and trustworthy) according to the factors that were discussed in section II, as well as the elements to consider it to be a FAIR, scientific application. The reviewer follows a checklist that produces a quantitative and qualitative

[4]https://www.bonitasoft.com/

[5]https://www.forbes.com/sites/forbestechcouncil/2019/10/28/orchestration-of-microservices-monoliths-people-and-robots/#4e9d9df66102

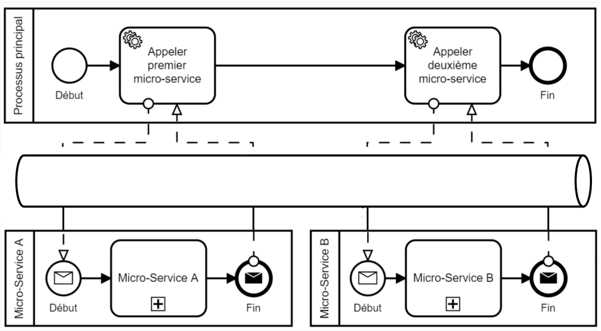

Fig. 4. OVD-SaaS architectural integration of BPML, message Broker, and Microservices for reproducible scientific applications.

scientific application profile with RepCompliance(%) that can be verified in the OVDSaas execution environment to certify and reward the desired attributes. Figure11.

B. API REST Blockchain ”DOIschain”

As part of the OVDSaaS infrastructure, a special microservice was implemented for the assignment of a permanent, persistent identifier ”DOIschain” [20] in Blockchain that encompasses and records in a blockchain transaction all the finalized artifacts necessary for reproducibility of the scientific application, claims, and results of the associated scientific research, as well such as documentation, scientific article, datasets. It allows them to be uploaded to IPFS storage where, through a CID identifier, they become FAIR [14] compliant.

All operations of DOIschain assignment or granting of assessment reproducibility badges are recorded in the blockchain Proof of Work (PoA) so that authenticity, legitimacy, traceability, and versioning will remain persistent, immutable perennially in the long term, without intervention or decisions of a private or central government nature.This module microservice handles a JSON request with the essential data from a front-end client authorized to issue DOIschain. The User Interface for DOIschain issuance is an improved version of the CodeMeta [6] generator tool provided by heritage software; It is an adapted form to fill **supradescriptor (DOIschain)** information for the publication (DOI), code (SwH-ID), and dataset (DataMed-ID). Likewise, the reproducible artifact files are saved in the InterPlanetary File System (IPFS) and accessible with persistent intrinsic hash-based Content Identifiers (CIDs).

1) Badging: This blockchain DOIschain infrastructure allows us also to implement a Reward system badge-based similar to [21]. We define three reproducibility badges according to the RepCompliance(%) percentage. Gold (81%-100%), Silver (61%-80%), and Bronze (41%-60%).

[6]https://codemeta.github.io/codemeta-generator/

IV. OVDSaaS Scientific Applications reproducibility case uses analysis

we shall now analyze our experience in creating scientific applications based on OVDSaaS policies, guidelines, and methodology; we present each use case with a short user story in one sentence. The four cases with associated scientific publications demonstrate that behind each application, there is present a rigorous scientific research process and peer-reviewed contribution to the scientific community.

The four cases are presented to show our process of evolution, continuous improvement, research, and learning from the experience to approach completely OVDSaaS reproducible end-to-end applications. A major attention and detailed analysis will be provided to BraTSApp.

This brief analysis reviews the application’s reproducible qualitative and quantitative criteria, which were explained in the previous section II. From Table I, we analyze characteristics to be considered as an OVDSaaS Scientific application, and from Table II, we will estimate our own reproducibility attributes percentage of compliance with RepCompliance(%).

A. SmartGrid simulation case

This is an energy sector example case where the proposed concept of reproducibility of OVD-SaaS applications cannot be applied, given that this simulation can hardly be reproducible in business logic as it uses proprietary software such as OPNET and MATLAB.

In this context, **Digital twins** reproducibility concept is used, which seeks to replicate the operation of the entire real system of distributed generation of electrical energy in terms of the physical system, the electrical network, and the telecommunications network.Figure 6.

User story: *As a renewable energy producer, I wish to simulate the dispatch and cost of energy. So, make demand response forecasts for distributed generation optimally.*

- **Description:** This application[7] provides an integrated simulation of distributed electrical energy generation resources (DR) in response to demand in a SmartGrid system through quota allocation with an ML replicator dynamics optimization algorithm.
- **FAIR, Reproducible:** No, it integrates the proprietary MATLAB, OPNET telecommunications simulator software. Poor documentation. Inadequate Open Repository. Permanent identifiers: associated publication (DOI) [22]. RepCompliance = 10%

B. ANJE Legal assistant case

User story: *As a lawyer, I wish to be assisted in the history of the results of judicial rulings of previous cases. So I can make an efficient legal defense without making the same mistakes in unfavorable cases.*

- **Description:** This NLP-based assistant[8] facilitates lawyers’ search for similar historical cases with a chatbot

[7]https://youtu.be/tnZ4zONaqIk
[8]https://youtu.be/JrEu_X4Pgtc

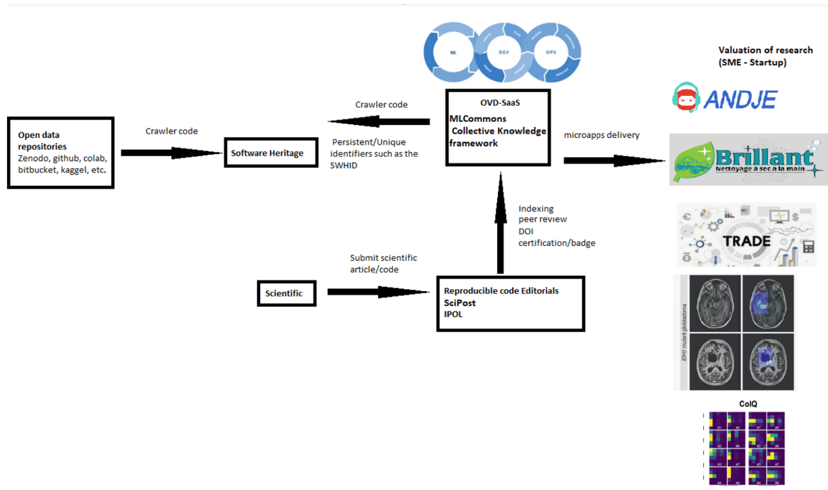

Fig. 5. OVD-SaaS architectural modules. Shows the final scientific MVP case study applications (ANJE, BraTSAapp, PethApp), Relationship with IPOL journal, Repositories and Storages, DOIs

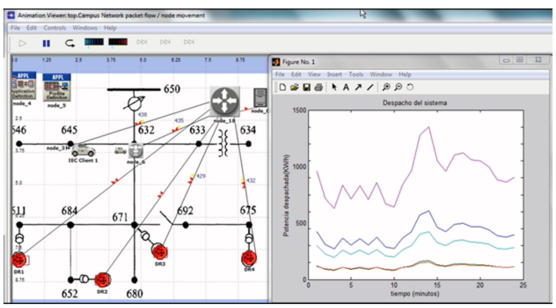

Fig. 6. Smart Grid

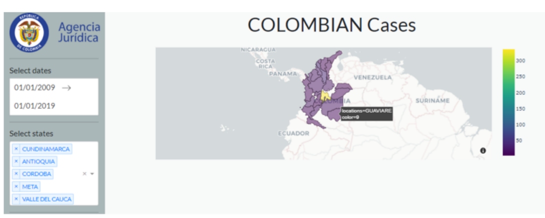

Fig. 7. ANJE

that can answer simple questions. It reuses pre-trained Large Language Models (LLM) (see HuggingFace) and BERT (PoC version of App) with the possibility of extending pre-trained generative transformers (GPT) to be applied in the Legal domain. Figure 7

- **FAIR, Reproducible:** Yes, despite the fact it handles confidential cases and sensitive information. Poor documentation, Inadequate Open repository. Permanent identifiers - Associated publication (DOI) [23]. RepCompliance = 40%

C. BraTS clinical application Case

User story: *As a radiologist, I wish to have updated and increasingly accurate neural network models available in emergency rooms to assist in interpreting brain magnetic resonance images. So, a diagnosis of the patient's severity is required very quickly*

This is a medical domain sector example of a complete end-to-end business application called BraTSAPP, Figure 8, created through MLops development cycles and following the complete OVDSaaS microservices architecture and guidelines. The dataset is BraTS' 2023 Glioma, and the neural network architecture is 3D-Unet. The results of the research were presented and published in IPOL demo [9], BrainLes2023[10] and LAWCN [24][11] we have a complete scientific application with a graphical interface, which can run a basic 3D U-Net model. To guarantee reusability, reproducibility, and benchmarking of the model with other competitors, MLCubes[12] allows the MedPerf[13] evaluation of the scientific findings obtained from the BraTS challenge and the integration with business processes. We used MLcubes[14] to containerize the model in the microservice backend to interoperability with OVDSaaS architecture. Through the orchestration of business processes with BPML (business process language) integration with BonitaSoft, the business logic is implemented.

- **Description:** This diagnostic-support app seeks to integrate into a hospital business logic of clinical cases. In this case, a brain tumor segmentation model was developed for the BraTS 2023 competition.
- **FAIR:** Findable and Persistent Immutable Identifier: DOI(publication), DOIschain (Computer-based Research Artifacts) Accessible and Use of Open Repositories: IPFS repository, IPOL demo, Docker API frontend. Interoperable: MLcube standard, Docker API backend - Reusable:

[9]https://ipolcore.ipol.im/demo/clientApp/demo.html?id=77777000461
[10]https://link.springer.com/conference/iwb
[11]https://link.springer.com/conference/lawcn
[12]https://www.synapse.org/#!Synapse:syn51156910/wiki/622674
[13]https://www.medperf.org/
[14]https://mlcommons.org/working-groups/data/mlcube/

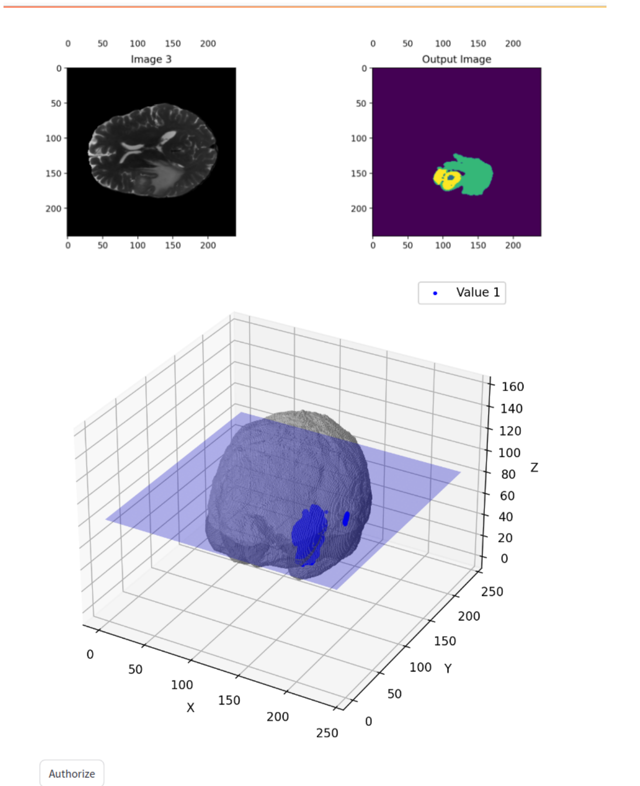

Fig. 8. Final enriched reproducible Scientific application BraTSAPP

Python Model

- **Asosiated publication and Documentation:** Complete Readme file, Publication LAWCN Conference [24]. **Validation, Review, Evaluation:** carried out in BraTS challenge. **Control version:** No version , **data provenance:** Indicated in DOIschain metadata, dataset BraTS2023. **contenerization /virtualization:** Docker, MLcube **Automation:** incomplete integration BonitaSoft BPML, Kubernetes execution. **Licensing:** CC-BY-SA. **Open Community Standards:** JSON, model.keras, mri_images.nii, MedPerf, REST
- **Reproducible:** RepCompliance = 80%

Computer-based Research Artifacts: **Publication**[15] ,**Data**[16] ,**MLcube**[17] , **DOIschain**[18]

D. The OVDSaas-PlethApp case: mouse breath behavior analysis

User story: *As a biologist, I wish to upload long mice's breathing signals recorded with a plethysmograph so that I can analyze and visualize mice breath behavior ("average" behavior, intra/inter-individual variability, evolution through time) depending on the experiment conditions. The analysis includes outputs of machine learning algorithms dedicated to plethysmograph signals, which are also downloadable for external use.*

[15] https://white-historic-chinchilla-599.mypinata.cloud/ipfs/Qmezmd76kUg38qHmfb3xWXJyvJRGhmKJMKeGLfTMFjoPdj

[16] https://white-historic-chinchilla-599.mypinata.cloud/ipfs/QmSgtne642z5FTj71qHCeFoqUJPim3TxxRpBHcLV9JcaWV/

[17] https://white-historic-chinchilla-599.mypinata.cloud/ipfs/QmY8V23yiqABgrhnNFfwmT9WjAvjdnhiQ64HgFreoxiWFf/

[18] https://white-historic-chinchilla-599.mypinata.cloud/ipfs/QmUBP3JvR5ombVPPQcBQZkBBPY9GzRtEgJBJZyQCx1RmuD

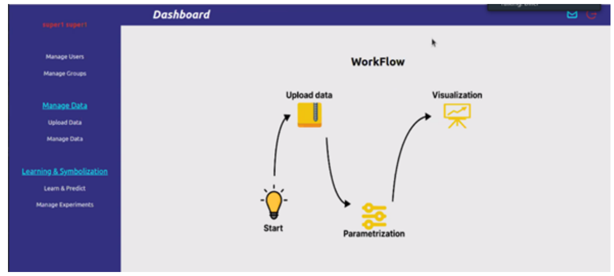

Fig. 9. WorkFlow PethAPP

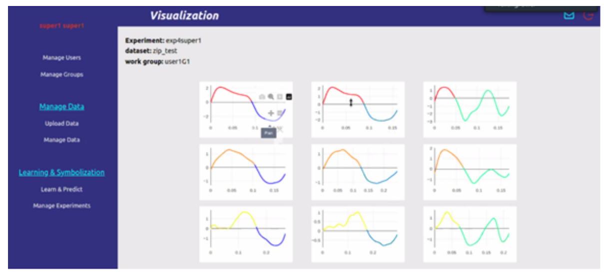

Fig. 10. Reproducible results in PethApp.

This biology domain sector application case allows biologists and scientific researchers to interact easily at Borelli Center. The biologist who analyzes the mouse's breathing does not have to worry about the technical programming aspects, such as the programming language or any other details. The researchers at Borelli Center can implement their algorithms, share them, and interact seamlessly with his medical research team.

Figure. 9 shows the PethApp workflow, and Figure 10 shows the reproducible results of the application.

- **Description:** It is an application that supports biologists in analyzing breathing measurements in mice. It allows the detection of abnormal patterns in inspiration and expiration using an algorithm developed at the Borelli Center. An example of a microservice with an independent Redis database for recording experiments.
- **OVDSaaS Roles:** Author(s) Data Science Research (Thibaut et. al), Industrial partner (Centre Borelli Biologist), Developer (Billel), OVDSaaS AI/ ML Architect (Armando), Scientific Director (Miguel), Article Publisher (Frontiers).
- **FAIR, Reproducible:** Yes. Permanent identifiers [25] associated with a publication (DOI). RepCompliance = 90%

E. Summary Results and Gap Analysis

As has been shown, each application has its specific characteristics and different levels of compliance with desirable

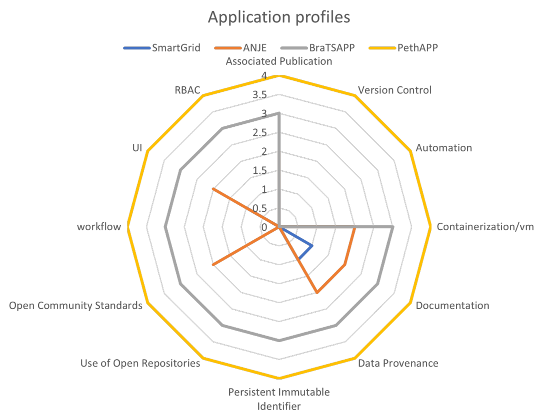

Fig. 11. Scientific applications cases comparision

characteristics for reproducible and OVDSaaS verifiable scientific applications. We provide a gap analysis to check and compare the degree of compliance with the use cases and the evolution towards reproducibility of end-to-end applications. It also indicates an author's effort to make the app completely reproducible.

A quantitative and qualitative analysis was carried out with RepCompliance(%) and verifying the other application criteria, Table II and Table I, respectively. The classification of the cases shows the evolution of our research and continuous improvement toward a fully reproducible scientific application. Our short gap analysis is summarized in Table III.

Figure 11 compares the case studies and shows the evolution in the degree of reproducibility in the 10 pillars chosen to measure and benchmark them in each case, as well as the approach to the concept of complete reproducible scientific application.

F. Claims Summary and Reproducibility services Benchmark

In this research, we made our Summary claims revision, Table IV, and benchmarked a series of platforms and initiatives identified to provide reproducibility services to highlight the differential elements of OVDSaaS. Table V

V. Conclusion

The main contribution of this work is to generate high social impact and improve the credibility and trustworthiness of scientific research in ML/AI, bridging the gap among academia, industry, publishers, and ML/AI technology through OVDSaaS platform (framework composed of microservices architecture, policies, and methodologies). The platform is oriented towards reproducibility and valorization of scientific research services. This promotes compliance with FAIR principles in ML/AI and adequate rewarding of computer-based scientific research with outstanding and unique characteristics compared to current platforms. It is a step beyond MLops to the "Model ready to the business" concept.

The OVD-SaaS is an ambitious project under intense development, and the prototype achieved is a sound proof of concept. However, several challenges remain, considering how varied and complex scientific research can be. It takes advantage of existing cloud computing technologies, such as blockchain, to produce end-to-end scientific applications with a special focus on reproducibility. In a future version, it is envisaged to include an evaluation of fairness, accountability, transparency and ethics (FATE) in the checklist to stimulate action towards representative and responsible AI

Acknowledgment

The authors thank the financial support to the SESAME's *OVD-SaaS* project from Région Île de France and BPI France. Also, to the Ministry of Science, Technology and Innovation of Colombia (Minciencias), within the framework of Call 885 of 2020, to finance the Doctorates of Excellence Program.

References

[1] M. Baker, "1,500 scientists lift the lid on reproducibility," *Nature*, vol. 533, no. 7604, pp. 452–454, May 2016. [Online]. Available: https://www.nature.com/articles/533452a

[2] A. Jobin, M. Ienca, and E. Vayena, "The global landscape of AI ethics guidelines," *Nature Machine Intelligence*, vol. 1, no. 9, pp. 389–399, Sep. 2019. [Online]. Available: https://www.nature.com/articles/s42256-019-0088-2

[3] J. A. Hernández and M. Colom, "Repeatability, Reproducibility, Replicability, Reusability (4R) in Journals' Policies and Software/Data Management in Scientific Publications: A Survey, Discussion, and Perspectives," Dec. 2023. [Online]. Available: https://arxiv.org/abs/2312.11028v1

[4] C. S. Collberg, "Measuring reproducibility in computer systems research," 2014. [Online]. Available: https://api.semanticscholar.org/CorpusID:62071447

[5] "Micro23," in *56th Annual IEEE/ACM International Symposium on Microarchitecture*. Toronto ON Canada: ACM, Oct. 2023.

[6] A. Paleyes, R.-G. Urma, and N. D. Lawrence, "Challenges in Deploying Machine Learning: A Survey of Case Studies," *ACM Computing Surveys*, vol. 55, no. 6, pp. 1–29, Jul. 2023. [Online]. Available: https://dl.acm.org/doi/10.1145/3533378

[7] e. a. Nicolaï, "The approach to reproducible research of the Image Processing On Line (IPOL) journal," *Informatio. Revista del Instituto de Información de la Facultad de Información y Comunicación*, vol. 27, no. 1, pp. 89–120, Jun. 2022. [Online]. Available: https://informatio.fic.edu.uy/index.php/informatio/article/view/349

[8] V. Stodden, C. Hurlin, and C. Pérignon, "RunMyCode.org: A novel dissemination and collaboration platform for executing published computational results," in *2012 IEEE 8th International Conference on E-Science*, Oct. 2012, pp. 1–8. [Online]. Available: https://ieeexplore.ieee.org/document/6404455

[9] M. Colom, J. A. Hernández, B. Kerautret, and B. Perret, "Development Efforts for Reproducible Research: Platform, Library and Editorial Investment," in *Reproducible Research in Pattern Recognition*, B. Kerautret, M. Colom, A. Krähenbühl, D. Lopresti, P. Monasse, and B. Perret, Eds. Cham: Springer Nature Switzerland, 2023, vol. 14068, pp. 3–21. [Online]. Available: https://link.springer.com/10.1007/978-3-031-40773-4_1

[10] T. Staubitz, H. Klement, R. Teusner, J. Renz, and C. Meinel, "CodeOcean - A versatile platform for practical programming excercises in online environments," in *2016 IEEE Global Engineering Education Conference (EDUCON)*, Apr. 2016, pp. 314–323, iSSN: 2165-9567. [Online]. Available: https://ieeexplore.ieee.org/document/7474573/

[11] K. K. et. al, "Lessons learned from the chameleon testbed," in *2020 USENIX Annual Technical Conference (USENIX ATC 20)*. USENIX Association, Jul. 2020, pp. 219–233. [Online]. Available: https://www.usenix.org/conference/atc20/presentation/keahey

[12] e. a. Rosendo, "Kheops: Cost-effective repeatability, reproducibility, and replicability of edge-to-cloud experiments," in *Proceedings of the 2023 ACM Conference on Reproducibility and Replicability*, ser. ACM REP '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 62–73. [Online]. Available: https://doi.org/10.1145/3589806.3600032

TABLE III

GAP ANALYSIS OF REPRODUCIBILITY COMPLIANCE FOR THE ANALYZED USE CASES. WE TAKE INTO ACCOUNT CHARACTERISTICS IN TABLE II AND TABLE I.

Domain	**Case**	**Persistence**	**UI**	**Business Logic Workflow BPML**	**AAA RBAC Roles**	**Associated Software**	**Type Data Source**	**Online**	**Associated Publication**	**Gap (%)**	**Badge**
Energy	DR Simulation	DataFile	No	No	No	OPNET, Matlab	open	No	yes [22]	90%	N/A
Legal	ANJE	DataFiles	yes	No	No	Plotly	Private	Online	yes [23]	60%	
Medical	BraTSAPP	DataFiles/ IPFS	yes	yes	yes	MLcubes	Open	Online	yes [24]	20%	
Biology	PethApp	Data Base/ IPFS	yes	yes	yes	Django	Private	Online	yes [25]	10%	

TABLE IV

OVDSAAS SUMMARY CLAIMS SUPPORTED IN THE PRESENTED CASE STUDIES

Main Claims	**OVDSaaS Strategy [3]**
Reproducibility	Reproducibility assessment and verification module, automated workflows, version control for datasets and code, standardized data formats, application execution environment
Credibility	FAIR compliance and Reproducibility verification, peer review integration, transparent metrics and analytics
Trustworthiness	FAIR compliance and Reproducibility verification, secure data storage, audit trails
Authenticity	DOIschain PIDs (Blockchain and IPFS), digital signatures, time-stamping
Legitimacy	DOIschain PIDs (Blockchain and IPFS), Unified Identifiers for (Publication, Code, Datasets), compliance with ethical guidelines, institutional endorsements
Reward	Blockchain Badging, contributor recognition systems, citation tracking
FAIR (Findable-Accessible-Interoperable-Reusable)	Open Repository, DOIschain, IPFS/cloud storage, online reproducibility verification, multilanguage interoperable microservices, Serverless Lambda-Functions, Containerization, metadata standards adherence, persistent identifiers
Accountability	DOIschain PIDs, versioning, provenance, role-based access control, compliance monitoring
Valorisation & Impact	Visibility, dissemination (Open Access Journals, publisher policies), solving industrial needs with scientific applications, establishing network connecting researchers-industrial partners, collaboration tools, impact assessment metrics (Reviews history, downloads measurement and research reputation score)
Reduce Author limitations	Low code tools integration, Use Agile MLOps, BPML methodologies for "Model ready to business" concept, user-friendly interfaces, extensive documentation and tutorials

TABLE V

BENCHMARK FOR VARIOUS OVDSAAS-SIMILAR PLATFORMS BASED ON DIFFERENT CRITERIA.

Tool	Granularity Level	Storage	Workflow	Badging	Persistent Identifiers	Claims Validation	Reproducibility Evaluation	Online Reproducibility	Provenance	Virtualization	Orchestration
RunMyCode	artifact	cloud	No	No	No	No	No	No	No	No	No
Chameleon/ KheOps	experiment	cloud-edge	No	No	No	yes	yes	yes	-	openStack	-
CodeOcean	experiment	cloud	yes	No	yes	No	-	-	-	Docker	-
Whole Tale	experiment	cloud	No	yes	yes	yes	yes	yes	yes	yes	yes
IPOL	Demo	Cloud	No	No	Yes	Yes	yes	No	yes	Docker	No
DAGhub	Workflow	cloud	yes	No	No	-	-	-	-	-	-
OVDSaaS	Application	cloud/ Blockchain/ IPFS	Yes	Yes	(article, software, data)	Yes	Yes	Yes	Yes	Docker	Kubernetes

[13] e. a. Brinckman, "Computing environments for reproducibility: Capturing the "Whole Tale"," *Future Generation Computer Systems*, vol. 94, pp. 854–867, May 2019. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0167739X17310695

[14] M. D. e. a. Wilkinson, "The FAIR Guiding Principles for scientific data management and stewardship," *Scientific Data*, vol. 3, no. 1, p. 160018, Mar. 2016. [Online]. Available: https://www.nature.com/articles/sdata201618

[15] O. E. Gundersen, S. Shamsaliei, and R. J. Isdahl, "Do machine learning platforms provide out-of-the-box reproducibility?" *Future Generation Computer Systems*, vol. 126, pp. 34–47, Jan. 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X21002090

[16] J. Françoise, B. Caramiaux, and T. Sanchez, "Marcelle: Composing Interactive Machine Learning Workflows and Interfaces," in *The 34th Annual ACM Symposium on User Interface Software and Technology*. Virtual Event USA: ACM, Oct. 2021, pp. 39–53. [Online]. Available: https://dl.acm.org/doi/10.1145/3472749.3474734

[17] G. Fursin, "The Collective Knowledge project: making ML models more portable and reproducible with open APIs, reusable best practices and MLOps," *arxiv*, vol. 1, no. 1, p. 3, 2020. [Online]. Available: https://arxiv.org/abs/2006.07161

[18] N. Gift and A. Deza, *Practical MLOps: operationalizing machine learning models*, first edition ed. Sebastopol, CA: O'Reilly Media Inc, 2021, oCLC: on1249501065.

[19] S. Newman, *Monolith to microservices: evolutionary patterns to transform your monolith*, first edition ed. Beijing [China] ; Sebastopol, CA: O'Reilly Media, Inc, 2019.

[20] C. M. Hernandez Jose Armando, "A blockchain-based unified pid with applications in healthcare," in *2024 Sixth International Conference on Blockchain Computing and Applications (BCCA)*. Dubai, AEU: IEEE, Nov. 2024.

[21] S. K. Radha, I. Taylor, J. Nabrzyski, and I. Barclay, "Verifiable Badging System for scientific data reproducibility," *Blockchain: Research and Applications*, vol. 2, no. 2, p. 100015, Jun. 2021. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S2096720921000105

[22] J. Hernández and R. Bustamante, "Simulación integrada de sistemas de Generación Distribuida para el diseño de redes de telecomunicaciones con estándar IEC61850-7-420," in *2023 IEEE Workshop on Power Electronics and Power Quality Applications (PEPQA)*. Cali, Colombia: IEEE, Oct. 2023, pp. 1–13. [Online]. Available: https://ieeexplore.ieee.org/document/10325776/

[23] J. A. Hernandez Gonzalez, "Assistance System for Judicial Awards for The Colombian State Legal Defense Agency Through NLP -ANDJE-," in *Proceedings of the 2022 5th International Conference on Machine Learning and Natural Language Processing*. Sanya China: ACM, Dec. 2022, pp. 258–265. [Online]. Available: https://dl.acm.org/doi/10.1145/3578741.3578793

[24] J. A. Hernández, "Data Augmentation by Adaptative Targeted Zoom for MRI Brain Tumor Segmentation," in *Computational Neuroscience*, e. a. Riascos Salas, Ed. Cham: Springer Nature Switzerland, 2024, vol. 2108, pp. 14–24. [Online]. Available: https://link.springer.com/10.1007/978-3-031-63848-0_2

[25] T. Germain, C. Truong, L. Oudre, and E. Krejci, "Unsupervised classification of plethysmography signals with advanced visual representations," *Frontiers in Physiology*, vol. 14, p. 1154328, May 2023. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fphys.2023.1154328/full