Deliverable D2.4

Documentation on BioEngine application development (accessible from the AI4Life website)

Project Title	Artificial Intelligence For Image Data Analysis In The Life Sciences
Project Acronym	AI4Life
Project Number	101057970
Project Start Date	01.09.2022
Project Duration	36 Months

WP N° & Title	WP2: User Services and Computing Infrastructure
WP Leaders	KTH
Deliverable Lead Beneficiary	KTH
Dissemination Level	PU
Contractual Delivery Date	31.08.2024 (M24)
Actual Delivery Date	04.09.2024
Authors	Wei Ouyang
Reviewers	Dorothea Dörr, Nils Mechtel, Joanna Hård, Arrate Muñoz Barrutia

AI4Life has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement number 101057970. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Executive Agency (REA). Neither the European Union nor the granting authority can be held responsible for them.

www.ai4life.eurobioimaging.eu
ai4life@eurobioimaging.eu
@AI4LifeTeam

Change Log

Version	Date	Author	Description of changes
v0.1	16.07.2024	Wei Ouyang	Initial draft
v0.2	29.08.2024	Wei Ouyang	Revision
v0.3	03.09.2024	Dorothea Dörr, Nils Mechtel, Joanna Hård, Arrate Muñoz-Barrutia	Edits and suggestions
v0.4	04.09.2024	Wei Ouyang	Final draft approved for submission

Acronyms and Abbreviations

AI	Artificial Intelligence
Apps	Applications
D	Deliverable
GPU	Graphical Processing Unit
PR	Pull Request
RPC	Remote Procedure Call
SAM	Segment Anything model
UI	User Interface
v	Version

Table of contents

www.ai4life.eurobioimaging.eu
ai4life@eurobioimaging.eu
@AI4LifeTeam

Executive Summary

The AI4Life project aims to transform bioimage analysis by creating the BioImage Model Zoo, an open-access platform for sharing and deploying Artificial Intelligence (AI) models tailored to life sciences. Central to this initiative is the BioEngine platform, a cloud-based infrastructure designed to streamline the deployment, execution, and sharing of AI models. This report provides a comprehensive overview of the BioEngine platform and its applications, focusing on how it addresses the growing complexity of data in life sciences through scalable, efficient, and collaborative tools.

BioEngine overcomes the limitations of traditional desktop software by offering a cloud-native solution that integrates seamlessly with widely used bioimaging tools. It supports a range of AI frameworks and enables researchers to perform high-throughput data processing directly from the web, without the need for complex local installations. The platform's architecture, which separates user interface (UI) applications (Apps) from Compute Apps, allows for flexible, modular development, making it easier for developers to create and deploy custom applications.

This report offers a high-level overview of the BioEngine application development process, with detailed documentation and code examples available at https://github.com/bioimage-io/bioengine/tree/main/docs. It outlines the steps for creating both UI and Compute Apps and provides guidelines for contributing to the BioEngine ecosystem. Positioned as a pivotal tool in modern bioimage analysis, the BioEngine platform delivers the scalability, flexibility, and collaborative capabilities that are essential for advancing life sciences research.

www.ai4life.eurobioimaging.eu
ai4life@eurobioimaging.eu
@AI4LifeTeam

1. Introduction

The AI4Life project aims to revolutionize bioimage analysis by providing an open-access platform, the BioImage Model Zoo, for sharing and discovering AI models tailored to the life sciences. This initiative addresses the growing need for advanced computational tools capable of handling the escalating data complexity in life sciences. At the heart of this endeavor lies the BioEngine platform, a cutting-edge cloud infrastructure designed to simplify the deployment, execution, and sharing of AI models across the scientific community.

Traditional desktop applications for bioimage analysis often fall short in managing the large-scale datasets and sophisticated computational requirements that modern research demands. These limitations are further compounded by the complexities of setting up local environments, managing dependencies, and ensuring hardware compatibility. Moreover, existing AI model zoos frequently require a high level of programming expertise which limits their accessibility to a broader range of users. Taken together, these challenges underscore the need for a more scalable, user-friendly solution for AI model serving.

BioEngine rises to this challenge by offering a state-of-the-art cloud-based platform that powers the BioImage Model Zoo, enabling researchers to test and deploy AI models directly from the web without the need for complex local installations. Its cloud-native design ensures scalability, allowing the platform to handle high-throughput data processing and advanced AI-driven analysis effortlessly. BioEngine integrates seamlessly with widely used bioimaging tools, including Fiji, Icy, and napari, This eliminates the need for multiple software dependencies and provides a streamlined user experience.

A key advantage of BioEngine is its ability to support a diverse array of AI frameworks, including TensorFlow, PyTorch, and ONNX. This is enabled through a user-friendly API accessible via HTTP or WebSocket. This flexibility allows developers to easily integrate AI model inference into existing workflows, Python scripts, Jupyter notebooks, or web-based applications. BioEngine's scalable infrastructure ensures that even the most demanding computational tasks can be performed efficiently by leveraging cloud resources to optimize GPU utilization.

Despite these advancements, the need for customizable user interfaces and the ability to process heterogeneous data types remains a critical factor in broadening BioEngine's applicability. To address this, BioEngine supports the development of custom applications—both UI Apps and Compute Apps—that can be seamlessly connected to the platform. This flexibility is essential for accommodating the diverse needs of users and expanding the utility of the BioImage Model Zoo, thus enabling FAIR (Findable, Accessible, Interoperable, and Reusable) compliance and ultimately for the benefit of

the scientific community.More broadly, BioEngine represents a paradigm shift in bioimage analysis, offering a scalable, cloud-based alternative to traditional desktop software. It not only provides the computational power needed for large-scale AI model training and inference but also fosters collaboration and AI-human interaction in a cloud environment. By supporting the development and deployment of BioEngine Applications, the platform enables researchers to transcend the limitations of conventional tools, paving the way for a new era of scalable, accessible, and collaborative bioimage analysis.

This report, prepared as deliverable D2.4 of the AI4Life project within the European Union's Horizon Europe research and innovation programme, documents the BioEngine application development process. It serves as a comprehensive guide for developers and researchers, detailing how to create and deploy UI and Compute Apps within the BioEngine platform, thereby contributing to the broader goals of the AI4Life project.

2. Overview of BioEngine Applications

The BioEngine platform, as illustrated in Figure 1, represents a sophisticated cloud-native infrastructure tailored to meet the complex computational demands of bioimage analysis. This section provides a detailed overview of BioEngine and its applications, highlighting how the platform integrates various technologies to facilitate scalable, efficient, and collaborative bioimage analysis workflows.

Here is a list of key design decisions for enabling BioEngine applications:

Hypha Framework as a Communication Hub: At the core of the BioEngine architecture lies the Hypha framework, which serves as the communication hub for the platform. Hypha provides a robust function call interface that enables seamless interaction between different components, including UI Apps and Compute Apps (backend computational services). By leveraging Remote Procedure Call (RPC) mechanisms, Hypha ensures real-time data exchange, service discovery, and efficient management of computational tasks, facilitating a cohesive and responsive user experience.

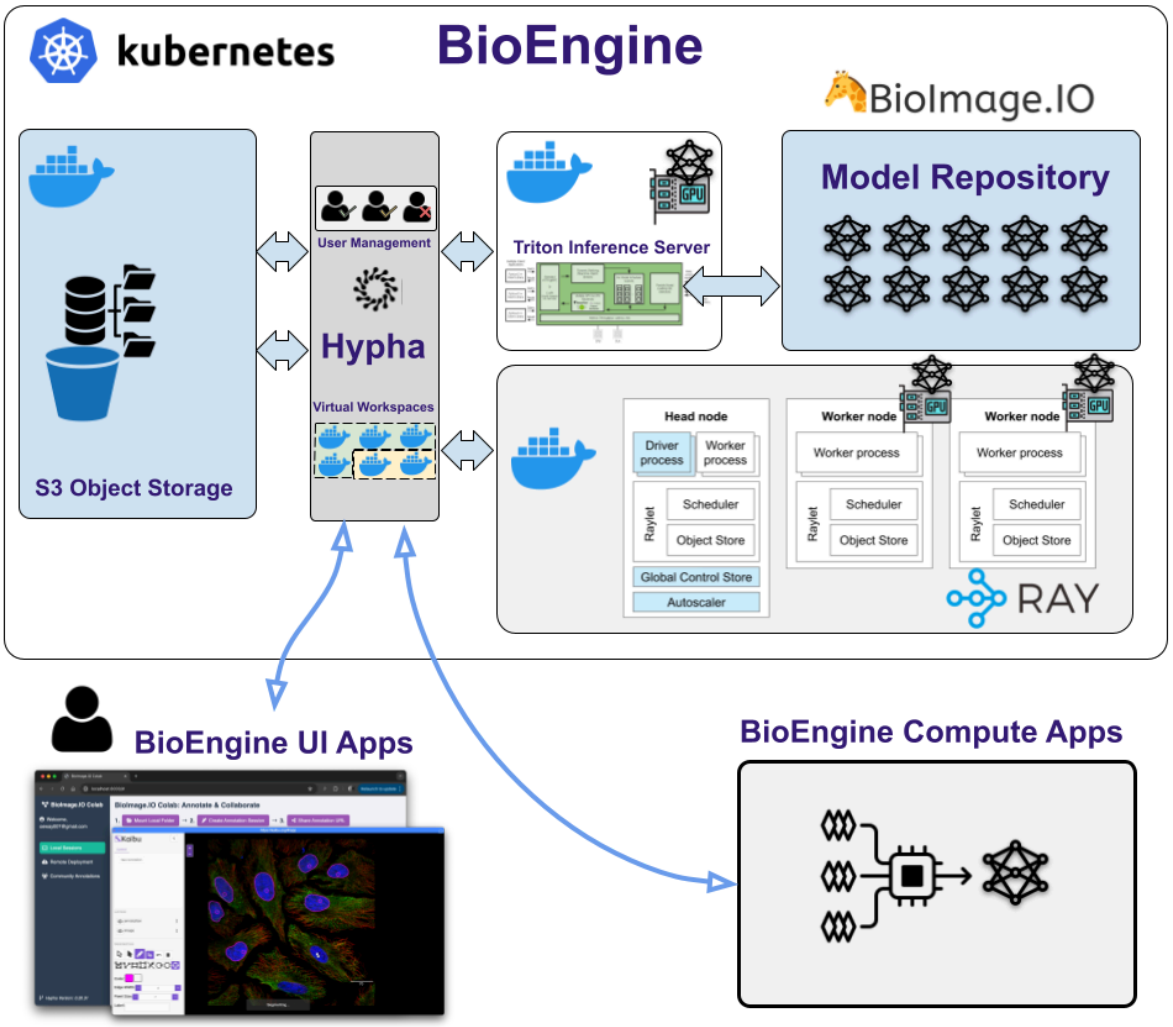

Figure 1. Overview of the BioEngine Architecture and Applications. This diagram illustrates the core components and workflow of the BioEngine platform. At the heart of the system is the Hypha framework, which acts as a communication hub, enabling seamless interaction between various elements, including UI Apps (user interfaces) and Compute Apps (backend computational services). The platform leverages Kubernetes (k8s) for orchestrating containerized components, ensuring scalable and efficient deployment across cloud environments. Virtual Workspaces, supported by Hypha, isolate applications and facilitate secure, multi-user collaboration. The Triton Inference Server efficiently serves AI models from the BioImage Model Zoo, optimizing GPU utilization by allowing multiple models and users to share resources. A Ray server is integrated to provide scalable computational capabilities, supporting custom logic such as pre-processing, post-processing, and AI model training. The architecture enables distributed, scalable bioimage analysis, allowing users to interact with advanced AI tools via lightweight UI Apps while leveraging powerful compute resources remotely.

Kubernetes for Orchestration: The BioEngine platform utilizes Kubernetes to orchestrate its containerized components. Kubernetes ensures the scalable and efficient deployment of all elements, including the Triton Inference Server, Ray clusters, and various storage solutions. This orchestration is crucial for managing resources dynamically, allowing BioEngine to handle complex workflows and large datasets with high reliability and performance.

Workspaces for Application Isolation: To provide a secure and isolated environment for each user, BioEngine employs Hypha's built-in support for Virtual Workspaces. These workspaces isolate BioEngine applications, ensuring that each user's data and processing environment are securely contained. This isolation is essential for maintaining data integrity, particularly when handling sensitive or large-scale bioimaging datasets. Furthermore, Virtual Workspaces facilitate multi-user collaboration, allowing different users to work on shared datasets while maintaining the security and privacy of their individual contributions.

Triton Inference Server for AI Model Serving: The Triton Inference Server is a critical component of BioEngine, designed to efficiently serve AI models hosted within the BioImage Model Zoo. Triton optimizes GPU utilization by allowing multiple users and models to share resources concurrently. This capability is particularly important for supporting high-throughput analysis and ensuring that advanced AI-driven tools are accessible to a wide range of users without the need for extensive local computational resources.

Ray for Enhanced Computational Scalability: To further extend its capabilities, BioEngine integrates Ray, a distributed computing framework that provides scalability for various computational tasks. Ray enables the deployment of Compute Apps that include custom logic, such as pre-processing, post-processing, and AI model training. This scalability is crucial for handling large foundational models and complex workflows, making BioEngine a future-proof platform capable of evolving with the advancing needs of bioimage analysis.

ImJoy Web Plugins for Creating UI: BioEngine leverages ImJoy web plugins to develop its UI apps. ImJoy offers a flexible and extensible interface for creating interactive web-based applications that integrate seamlessly with BioEngine's backend services. These plugins allow developers to build and customize user interfaces that facilitate data input, visualization, and interaction with AI models, all within a browser-based environment.

BioEngine Applications: UI and Compute Apps: BioEngine's architecture separates the concepts of UI Apps and Compute Apps, a design choice that significantly enhances the platform's flexibility and scalability.

- UI Apps are lightweight, web-based interfaces that provide users with the tools to upload images, configure analysis parameters, and visualize results. These apps run directly in the user's web browser, requiring minimal local resources while leveraging the powerful compute capabilities of BioEngine's backend services.
- Compute Apps handle heavy computational tasks, such as AI model inference and training. These backend services are deployed in the cloud and managed by BioEngine, ensuring that even the most resource-intensive processes are executed efficiently. The separation of UI and Compute Apps allows BioEngine to scale flexibly according to user demand, supporting everything from small, individual analysis tasks to large, collaborative projects.

This separation also facilitates multi-user collaboration, where multiple users can work on the same dataset simultaneously. For instance, in a collaborative annotation tool, users can annotate images in real-time, with AI models running on the Compute Apps providing immediate feedback and assistance. This setup is ideal for human-in-the-loop workflows, where human expertise and AI capabilities are combined to produce high-quality annotations quickly and efficiently.

Example: Collaborative Annotation Workflow

A prime example of BioEngine's application is its role in the bioimageio-colab tool (available at https://github.com/bioimage-io/bioimageio-colab) for collaborative annotation, where both UI Apps and Compute Apps play critical roles. In this scenario, a BioEngine UI App is developed to manage annotation sessions and interact with users through a browser-based interface. Users can initiate annotation sessions and interact with the annotation tool, which is powered by ImJoy web plugins.

On the backend, a Compute App is deployed to run the microSAM model, a specialized AI model for image segmentation. This Compute App handles tasks such as pre-processing images, performing segmentation, and post-processing the results. By leveraging the scalable infrastructure provided by Ray and Kubernetes, the Compute App can manage multiple simultaneous annotation tasks, ensuring that the system remains responsive even under heavy workloads.

In a typical workflow, users start by uploading images through the UI App, which communicates with the Compute App to process the images. The microSAM model running in the Compute App generates initial segmentation results, which are then

displayed to the user in the UI App. Users can refine these results manually, combining human expertise with AI-generated suggestions to produce high-quality annotations. The entire process is managed within BioEngine's Virtual Workspaces, ensuring that data remains secure and isolated throughout the workflow.

This collaborative approach, supported by BioEngine's robust architecture, exemplifies the platform's ability to integrate advanced AI models into user-friendly applications, enabling efficient and scalable bioimage analysis. The modular design of BioEngine Applications, with distinct roles for UI and Compute Apps, ensures that the platform can adapt to a wide variety of use cases, from individual research projects to large-scale collaborative initiatives.

3. BioEngine Apps Development

BioEngine Apps are key components of the BioEngine platform, designed to facilitate scalable, interactive bioimage analysis. These apps are categorized into UI Apps and Compute Apps, each playing distinct roles in the platform's functionality.

UI Apps are responsible for the user interaction layer, providing the front-end interface through which users can engage with computational resources and data analysis tasks. These apps are typically built using web technologies and frameworks, such as ImJoy, React, or Vue.js. The backend services of BioEngine allow these apps to process and visualize data in real-time.

Compute Apps, on the other hand, handle the backend processing. They are developed to execute complex computational tasks, such as running AI models for image segmentation, feature extraction, or other bioimage analyses. Compute Apps can be deployed in cloud environments or run on local workstations with GPU support, depending on the computational requirements.

UI Apps Development

UI Apps in BioEngine are designed to be flexible and interactive, enabling users to engage with complex datasets and computational resources through a browser-based interface. The process of developing a UI App involves several key steps:

- Setting Up the Development Environment: Start by installing the necessary development tools and familiarizing yourself with web development technologies, particularly HTML, CSS, and JavaScript. For more complex interfaces, you may choose to use frameworks like React or Vue.js.

www.ai4life.eurobioimaging.eu
ai4life@eurobioimaging.eu
@AI4LifeTeam

- Creating the User Interface: Design the UI using basic web technologies. If using ImJoy, create plugins that can integrate seamlessly with BioEngine services, allowing for rapid prototyping and deployment of UI components.
- Connecting to BioEngine Services: Use the hypha-rpc library to connect your UI components with BioEngine's backend services. This enables real-time data processing, where the UI interacts with Compute Apps running on the platform.
- Testing and Deployment: Test the UI App locally to ensure it interacts correctly with BioEngine services. Once satisfied, deploy the app on platforms like Kubernetes or package it as an ImJoy plugin for distribution.

For detailed instructions and examples on how to develop UI Apps, please refer to our GitHub documentation: https://github.com/bioimage-io/bioengine/tree/main/docs.

Compute Apps Development

Compute Apps represent the computational backbone of BioEngine, performing the heavy lifting for data processing tasks. These apps can either be developed as services running in independent containers or executed directly on local workstations, particularly when GPU support is required.

1. Developing Compute Apps as Hypha Services:
    - Begin by developing your Compute App using Python and integrate it with the necessary libraries for bioimage analysis.
    - Register your Compute App as a service with a Hypha server using the hypha-rpc library. This allows the app to be accessed by UI Apps or other Compute Apps within the BioEngine ecosystem.
    - For portability and scalability, package your Compute App as a Docker container.
2. Running Compute Apps on Local Workstations:
    - For tasks requiring high-performance computing resources, such as deep learning model training, you can run Compute Apps directly on your local workstation. This approach is ideal for leveraging GPU acceleration.
    - Even when running locally, these apps can connect to BioEngine's UI Apps using hypha-rpc, ensuring seamless interaction between the front end and the computational backend.
3. Submitting Compute Apps to BioEngine:
    - To contribute your Compute App to the BioEngine platform, you will need to wrap it as a Ray app and provide necessary configuration files, such as an initialization script and a manifest file.

www.ai4life.eurobioimaging.eu
ai4life@eurobioimaging.eu
@AI4LifeTeam

- Submit your Compute App by creating a pull request (PR) on the BioEngine GitHub repository. The PR should include your app's configuration in a new folder named after your app.
- The BioEngine team will review your submission and, upon approval, include it in the platform, making it accessible to other users.

For comprehensive guidelines on developing and submitting Compute Apps, please visit our GitHub documentation.

4. Conclusion

The AI4Life project, through the BioEngine platform, marks a significant advancement in bioimage analysis, offering a scalable, cloud-native solution that overcomes the limitations of traditional desktop tools. By integrating cutting-edge technologies like Kubernetes, Hypha, and the Triton Inference Server, BioEngine provides researchers with the computational power and flexibility necessary to manage the increasingly complex data challenges in life sciences.

BioEngine's architecture, which separates UI Apps from Compute Apps, offers a modular and scalable approach to bioimage analysis. UI Apps provide an interactive user interface, while Compute Apps handle intensive computational tasks such as AI model inference and training. This design not only enhances scalability but also allows for the customization of applications to meet specific research needs.

The development of BioEngine Apps, as outlined in this report, provides a clear pathway for researchers and developers to contribute to the platform. By following structured guidelines, users can create and submit their applications, enriching the BioEngine ecosystem and expanding its range of tools and capabilities.

Looking ahead, BioEngine is well-positioned to become a cornerstone of future bioimage analysis. Key areas of focus include:

- **Enhanced AI Integration:** Incorporating support for emerging AI frameworks and technologies to keep pace with advancements in the field.
- **Community Engagement:** Encouraging broader participation from developers and researchers to drive innovation and expand the platform's capabilities.
- **User Experience:** Continuously improving the interface and simplifying app development processes to make BioEngine accessible to a wider audience.
- **Expanded Integrations:** Strengthen compatibility with existing bioimaging tools and extend integrations to other platforms.

www.ai4life.eurobioimaging.eu
ai4life@eurobioimaging.eu
@AI4LifeTeam

In conclusion, BioEngine is paving the way for a new era in bioimage analysis, offering a robust, scalable, and collaborative platform that is essential for modern research. The AI4Life project, through BioEngine, is not only addressing today's research challenges but also setting the stage for future innovations in life sciences.

www.ai4life.eurobioimaging.eu
ai4life@eurobioimaging.eu
@AI4LifeTeam