

OpenHPS: A Modular Framework to Facilitate the Development of FAIR Positioning Systems

Maxim Van de Wynckel ¹ and Beat Signer ¹

¹ Vrije Universiteit Brussel, Web & Information Systems Engineering Lab, Brussels, Belgium

DOI: [10.21105/joss.08113](https://doi.org/10.21105/joss.08113)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor: [Daniel S. Katz](#) 

Reviewers:

- [@parakhsid](#)
- [@floe](#)

Submitted: 31 March 2025

Published: 05 June 2025

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

Abstract

Positioning systems determine the location of people and objects using various technologies and algorithms. While GPS dominates outdoor positioning, indoor and smaller-scale systems often require alternative technologies for lower latency, higher accuracy, or greater efficiency. Often developed as single-use prototypes lacking standard data formats, these systems hinder reusability and expansion. OpenHPS addresses these challenges by providing a modular, graph-based framework for creating versatile positioning systems on multiple platforms. It supports a wide range of algorithms and enables extensibility through custom nodes for sensor fusion and algorithm integration.

Keywords

Hybrid positioning systems, interoperable positioning systems, indoor positioning, stream processing, linked data, RDF

Statement of Need

Existing positioning systems or frameworks ([Georgiou et al., 2015](#); [Pustka et al., 2011](#); [Scholl, 2020](#)) often focus on specific use cases, platforms, or a set of algorithms. While this facilitates the deployment of a system, it limits the flexibility to design or reuse custom algorithms and systems. In academic systems ([García et al., 2015](#)), proprietary solutions are often difficult to replicate, extend, or deploy in a production-ready environment due to the use of software such as MATLAB. Furthermore, the data produced by such systems is often unstandardised. OpenHPS was created to address these issues by providing a modular data-centric framework that allows developers and researchers to create positioning systems for a wide range of use cases. The framework and its produced data are designed to be interoperable, enabling developers to share their algorithms and data with others. Using additional modules that add support for Solid ([Van de Wynckel & Signer, 2022a](#)) and DHTs, we have enabled the development of findable, accessible, interoperable, and reusable (FAIR) positioning systems.

Framework Overview

OpenHPS is an open-source hybrid positioning system framework written in TypeScript. It runs on the server (Node.js), in the browser, or within hybrid mobile applications. The general design of a positioning system created using OpenHPS consists of a graph containing a set of nodes that process data. Background services can be added to the graph to persist data or to communicate with other systems. We chose TypeScript over low-level languages based on the goal of achieving cross-platform reusability.

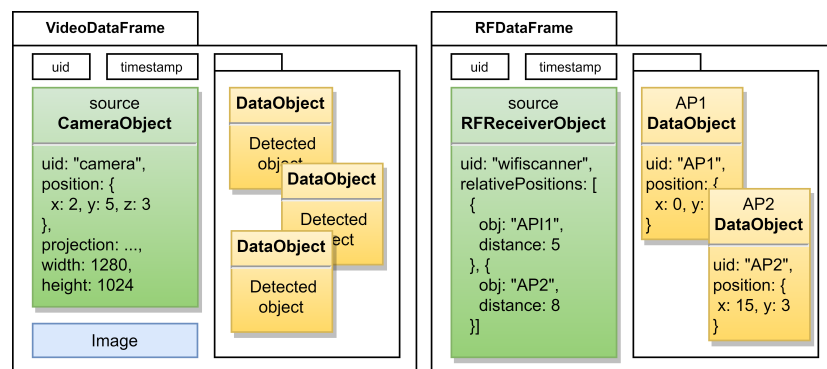


Figure 1: DataFrames and DataObjects

All concepts, ranging from positions to sensor values, can be expressed in various ways with varying units, enabling OpenHPS to support both small-scale use cases such as tracking a pen on paper to larger use cases such as tracking airplanes across the globe. Our framework uses stream-based processing of DataFrames that contain all time-sensitive information. These frames contain one or more DataObjects as illustrated in Figure 1, which indicate the spatial objects relevant for the information within the data frame. This enables the tracking of multiple actors, unlike other frameworks such as ROS (Quigley et al., 2009).

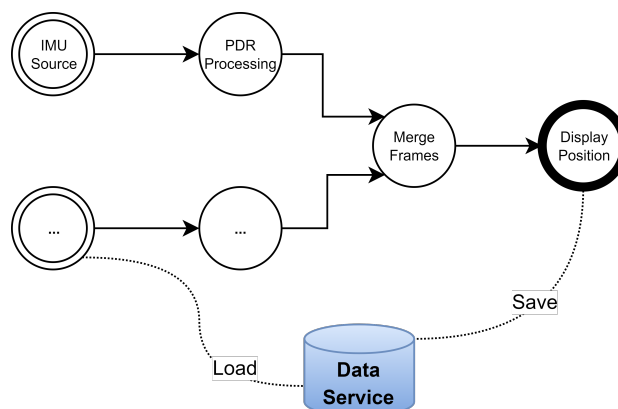


Figure 2: OpenHPS graph of a positioning system

Any data pushed through the graph can be serialised to JSON or RDF data using the POSO ontology (Van de Wynckel & Signer, 2022b), enabling interoperability between systems. Each node in the graph represents a stage in the processing of data frames from source to sink as shown in Figure 2. A SourceNode generates information, a ProcessingNode processes information and a SinkNode consumes information.

OpenHPS is modular by design, mainly due to the ability to extend data frames and objects. These extensions allow for different data objects, including different sensors like cameras, IMU sensors or spatial landmarks such as Bluetooth beacons. Each node in a graph can be extended as well, allowing the creation of custom algorithms that can be added or removed from a positioning system. Researchers can focus on the prototyping of new algorithms without having to worry about the integration of these algorithms into a larger system.

Interoperability

Our framework is designed to offer FAIR positioning systems, leveraging its open architecture and the data it produces. Modules such as @openhps/rdf and @openhps/solid enable developers

to serialise their data to RDF and store it in Solid Pods. This facilitates the creation of positioning systems that are transparent and privacy-preserving, while also ensuring that other positioning systems or consumer applications can access the data, regardless of whether they were created with OpenHPS. With various extensions such as semantic beacons (Van de Wijnckel & Signer, 2023), we enable the discovery of these systems and the data they produce.

Performance

OpenHPS uses JavaScript at runtime. To overcome the challenges associated with creating real-time dataflows, our graph can be executed using Web Workers (Van de Wijnckel & Signer, 2020). All data transmitted through our graph is serialisable, eliminating the need for developers to handle this serialisation or communication between workers. In addition to the ability to run graphs on multiple workers, communication nodes, such as MQTT, enable developers to offload the processing of complex tasks to other servers or dedicated processors. For more computationally intensive algorithms, modules like @openhps/opencv and @openhps/openvslam provide C++ bindings to low-level libraries. Although these bindings may not achieve native performance, reusing established libraries remains a key design goal.

Examples of Research Work

OpenHPS has been a building block for various research that has been used in indoor positioning systems (Van de Wijnckel & Signer, 2021a), and its ability to serialise location data to RDF has been demonstrated in an application that aims to preserve privacy and transparency using Solid Pods (Van de Wijnckel & Signer, 2022a). Further, OpenHPS was used in the SemBeacon demonstrator application (Van de Wijnckel & Signer, 2023) that is written in CapacitorJS and uses the framework to deserialise positioning data and positioning systems. Modules were also developed for other academic projects, such as the FidMark ontology (Van de Wijnckel et al., 2024), which enables fiducial marker classification within the framework. OpenHPS's modular node design and domain focus facilitate the sharing of algorithms and findings, as well as the rapid creation of prototypes and demonstrators that make use of location data. Finally, OpenHPS and the related modules have been used to collect several datasets that can be used by researchers to evaluate their algorithms (Van de Wijnckel, 2025; Van de Wijnckel & Signer, 2021b, 2025).

References

- García, E., Poudereux, P., Hernández, Á., Ureña, J., & Gualda, D. (2015). A robust UWB indoor positioning system for highly complex environments. *2015 IEEE International Conference on Industrial Technology (ICIT)*, 3386–3391. <https://doi.org/10.1109/ICIT.2015.7125601>
- Georgiou, K., Constambeys, T., Laoudias, C., Petrou, L., Chatzimilioudis, G., & Zeinalipour-Yazti, D. (2015). Anyplace: A crowdsourced indoor information service. *Proceedings of the International Conference on Mobile Data Management: Systems, Services and Middleware (MDM 2015)*, 291–294. <https://doi.org/10.1109/MDM.2015.80>
- Pustka, D., Huber, M., Waechter, C., Echtler, F., Keitler, P., Newman, J., Schmalstieg, D., & Klinker, G. (2011). Automatic configuration of pervasive sensor networks for augmented reality. *IEEE Pervasive Computing*, 10(3), 68–79. <https://doi.org/10.1109/MPRV.2010.50>
- Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., & Ng, A. Y. (2009, May). ROS: An open-source robot operating system. *Proceedings of the International Workshop on Open Source Software*. <https://api.semanticscholar.org/CorpusID:6324125>
- Scholl, Z. (2020). *The framework for internal navigation and discovery (FIND)*. <https://github.com/schollz/find3>

- Van de Wynckel, M. (2025). *Garage positioning dataset*. Kaggle. <https://doi.org/10.34740/KAGGLE/DS/6654647>
- Van de Wynckel, M., & Signer, B. (2022a). A Solid-based architecture for decentralised interoperable location data. *Proceedings of the 12th International Conference on Indoor Positioning and Indoor Navigation (IPIN 2022)*, CEUR Workshop Proceedings, 3248. <https://ceur-ws.org/Vol-3248/paper11.pdf>
- Van de Wynckel, M., & Signer, B. (2021a). Indoor positioning using the OpenHPS framework. *Proceedings of the 11th International Conference on Indoor Positioning and Indoor Navigation (IPIN 2021)*, 1–8. <https://doi.org/10.1109/IPIN51156.2021.9662569>
- Van de Wynckel, M., & Signer, B. (2020). *OpenHPS: An open source hybrid positioning system*. Vrije Universiteit Brussel. <https://doi.org/10.48550/ARXIV.2101.05198>
- Van de Wynckel, M., & Signer, B. (2021b). *OpenHPS: Single floor fingerprinting and trajectory dataset*. <https://doi.org/10.5281/zenodo.4744380>
- Van de Wynckel, M., & Signer, B. (2022b). POSO: A generic positioning system ontology. *Proceedings of the 21st International Semantic Web Conference (ISWC 2022)*. https://doi.org/10.1007/978-3-031-19433-7_14
- Van de Wynckel, M., & Signer, B. (2023). SemBeacon: A semantic proximity beacon solution for discovering and detecting the position of physical things. *Proceedings of the 13th International Conference on the Internet of Things (IoT 2023)*. <https://doi.org/10.1145/3627050.3627060>
- Van de Wynckel, M., & Signer, B. (2025). *Sphero dead reckoning and CV tracking dataset*. Kaggle. <https://doi.org/10.34740/KAGGLE/DS/6760212>
- Van de Wynckel, M., Valadez, I., & Signer, B. (2024). FidMark: A fiducial marker ontology for semantically describing visual markers. *Proceedings of the Semantic Web (ESWC 2024)*, 235–250. https://doi.org/10.1007/978-3-031-60635-9_14