Indoor Positioning Using the OpenHPS Framework

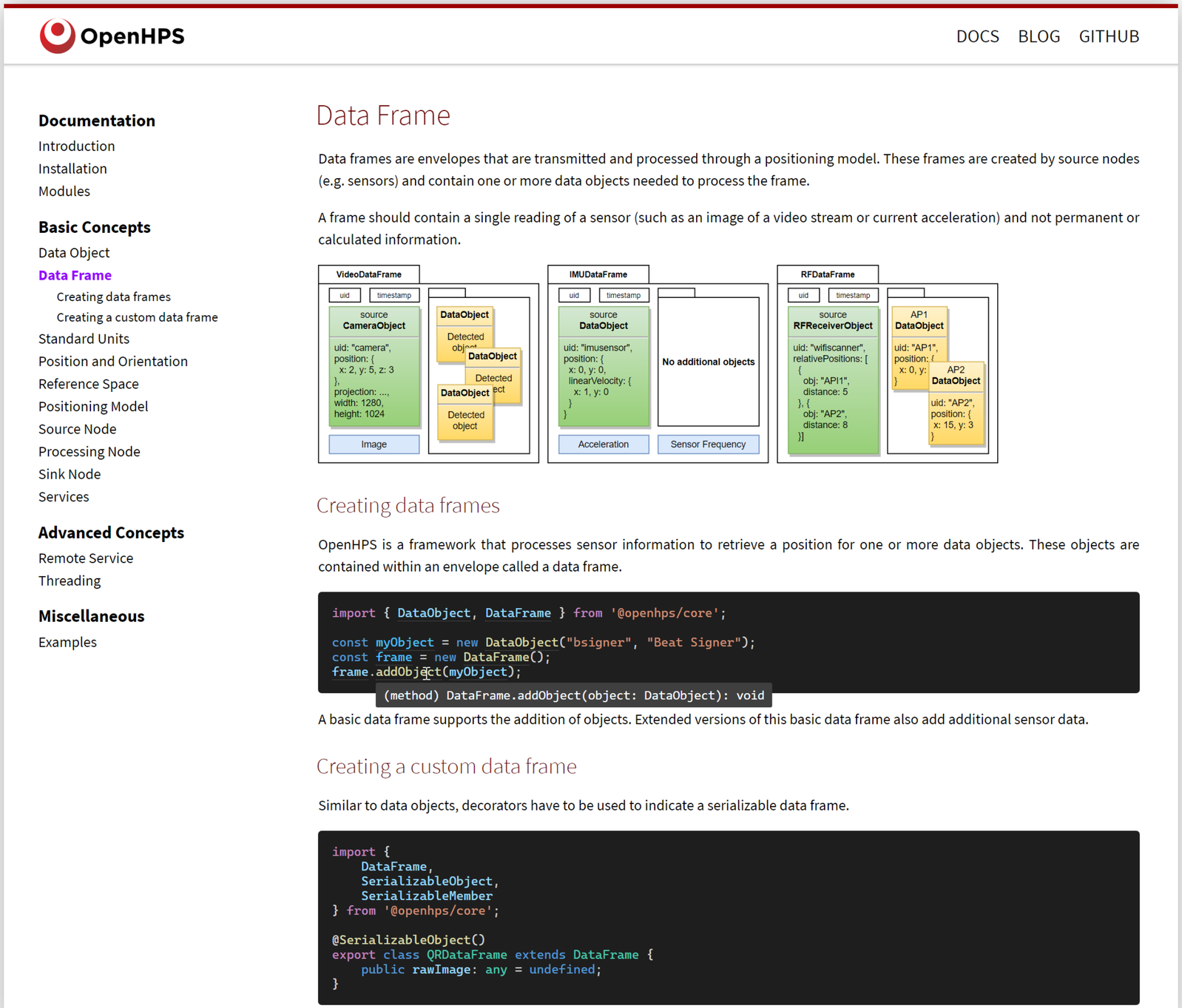
Maxim Van de Wynckel, Beat Signer

Web & Information Systems Engineering Lab
Vrije Universiteit Brussel

What is OpenHPS?

OpenHPS

An Open Source Hybrid Positioning System



What is OpenHPS?

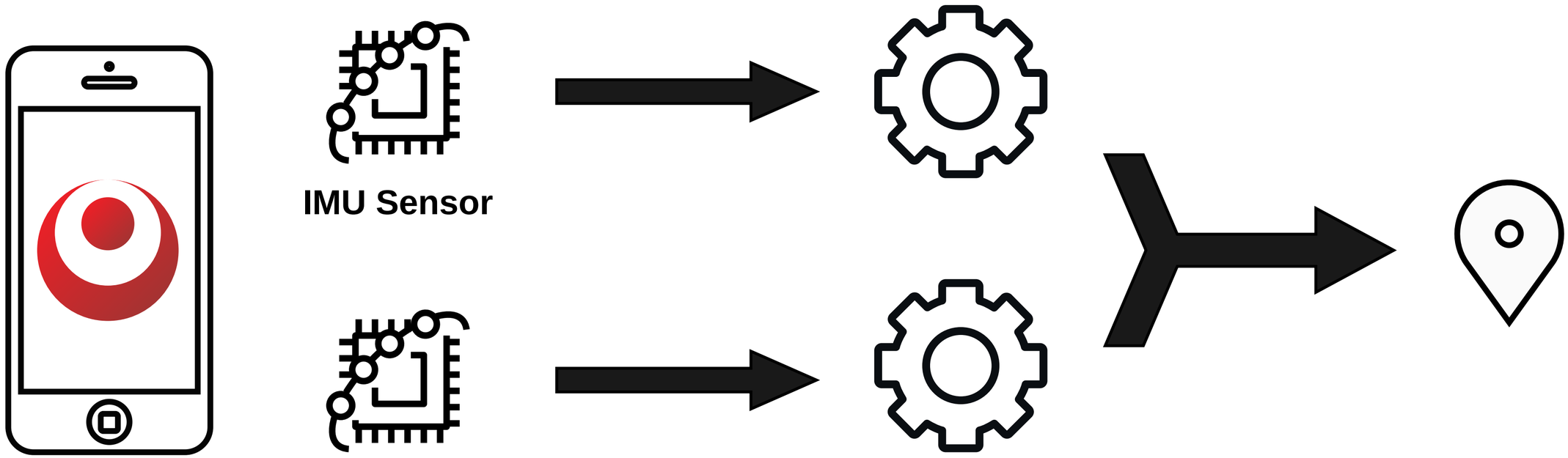
OpenHPS

An Open Source Hybrid Positioning System

- Any technology
- Any algorithm
- Various use cases
- Flexible processing and output
  - Accuracy over battery consumption, reliability, ...
- Aimed towards
  - Developers
  - Researchers

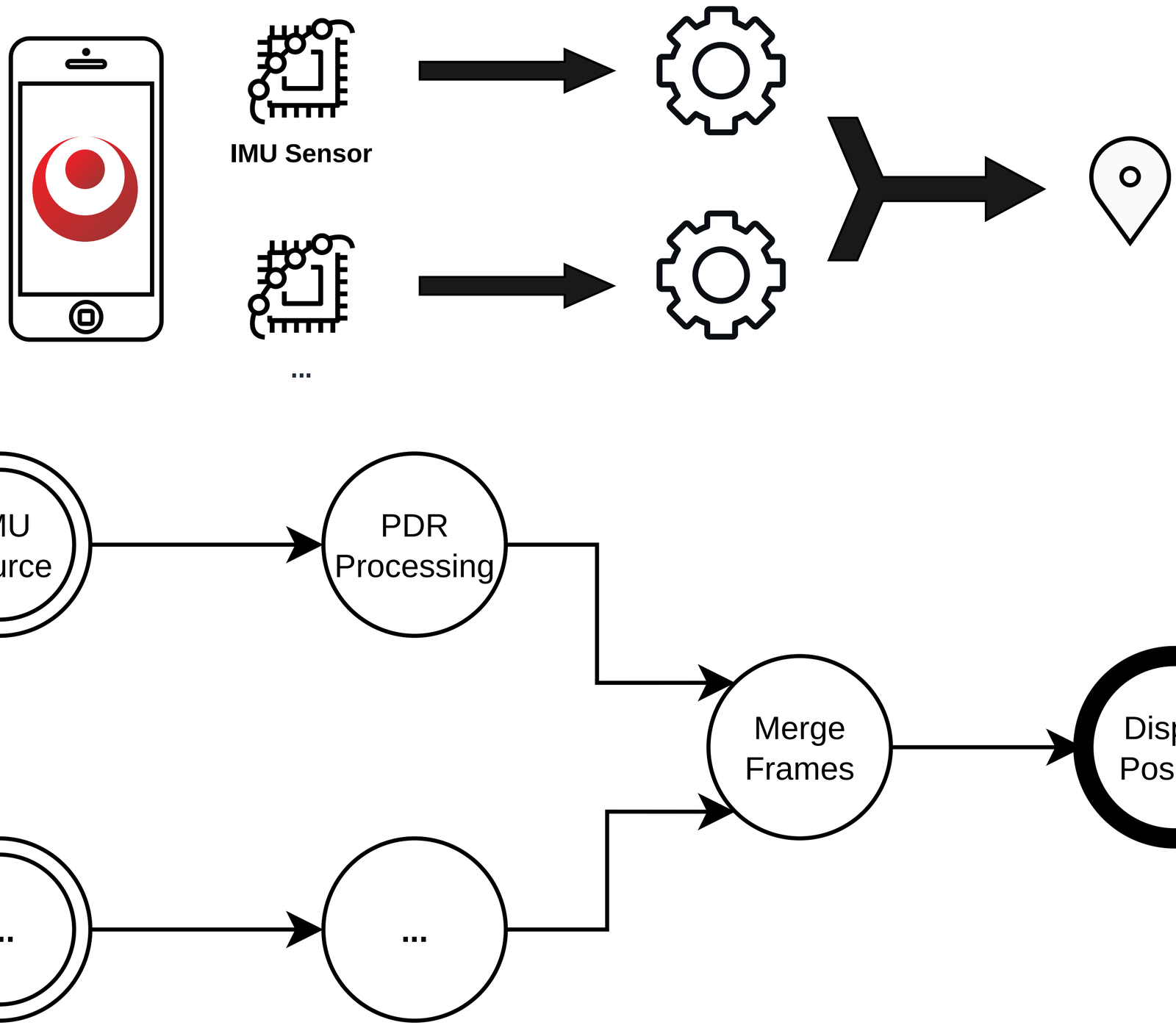
Process Network Design

OpenHPS



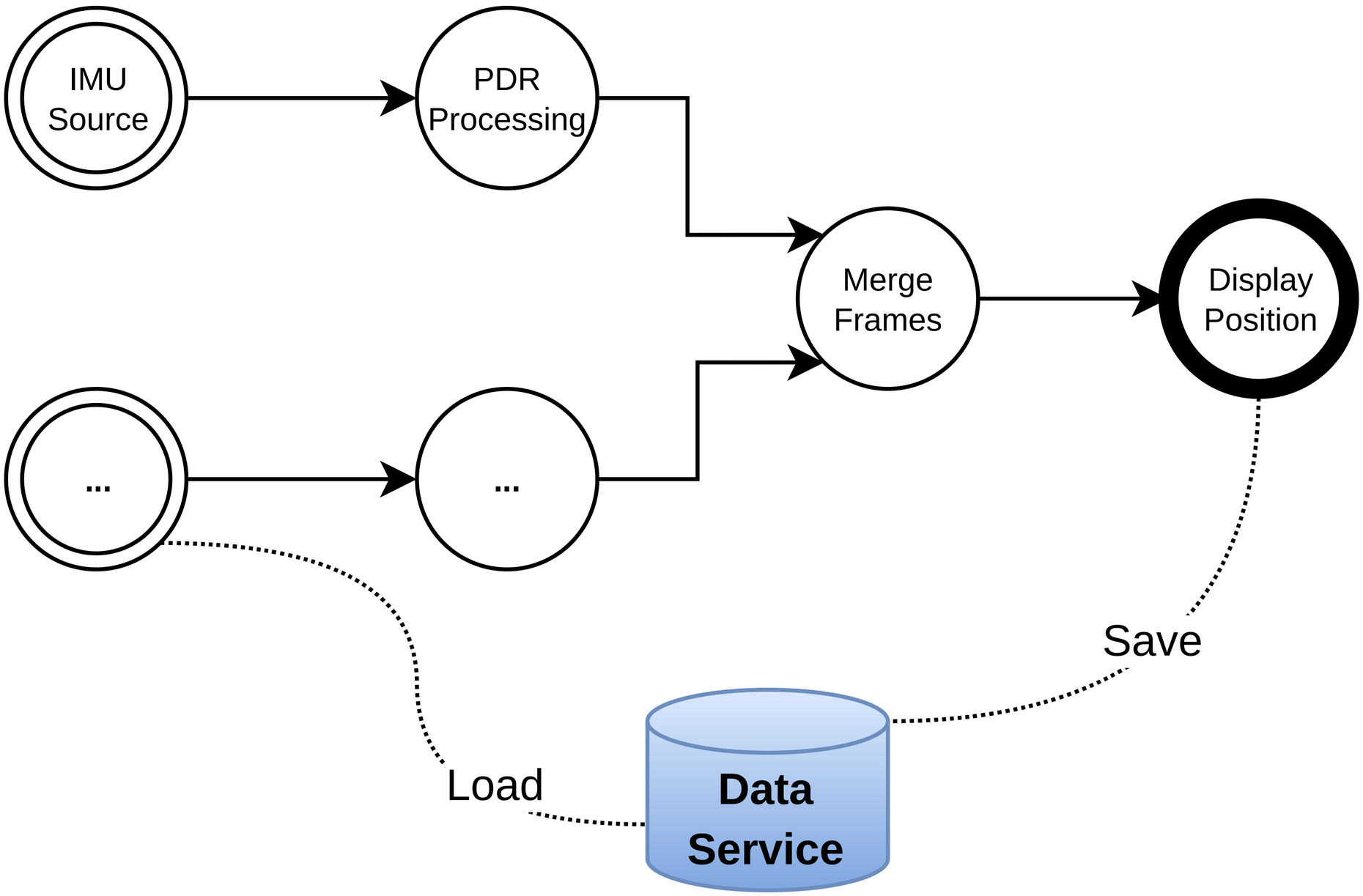
Process Network Design ...

OpenHPS



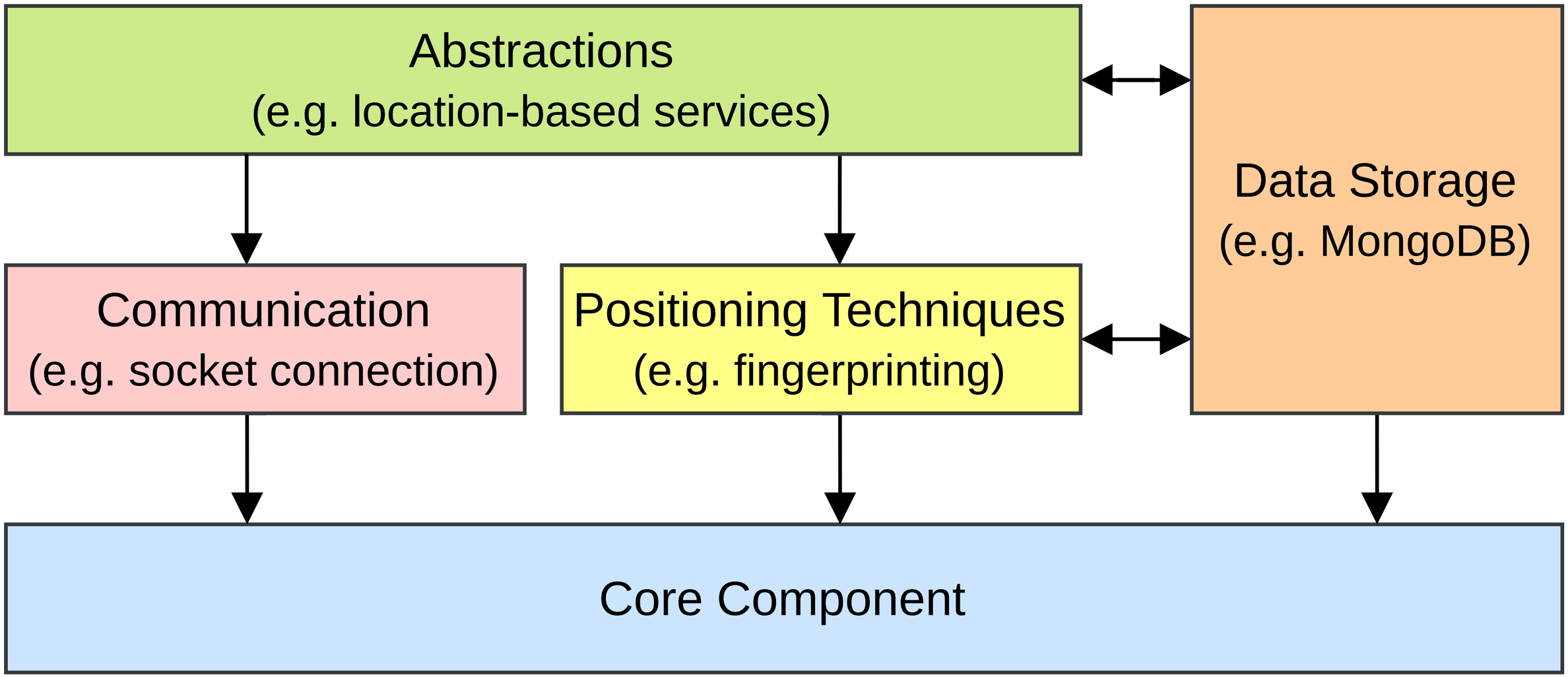
Process Network Design ...

OpenHPS



Modularity

OpenHPS



Modularity ...

OpenHPS

Communication

Socket, MQTT, REST API, ...

Data Storage

MongoDB, LocalStorage, RDF, ...

Positioning Algorithms

IMU, fingerprinting, OpenVSLAM, ...

Abstractions

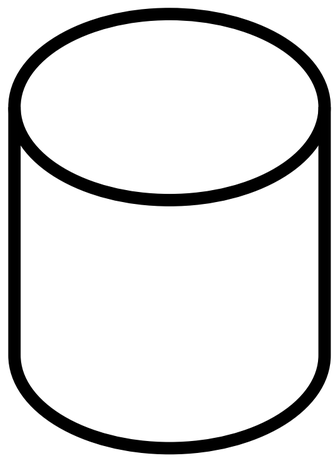
Geospatial, location-based services, geojson, ...

Other

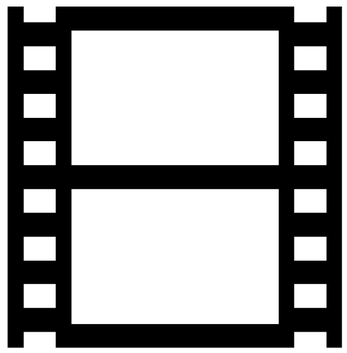
React-Native, NativeScript, Sphero, ...

Data Processing

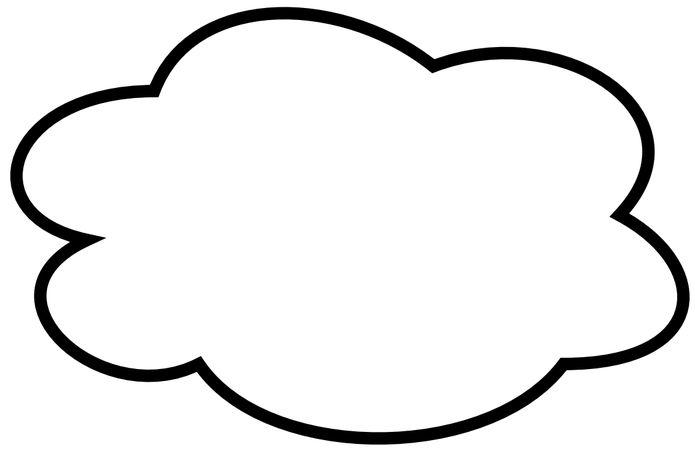
OpenHPS



Knowledge



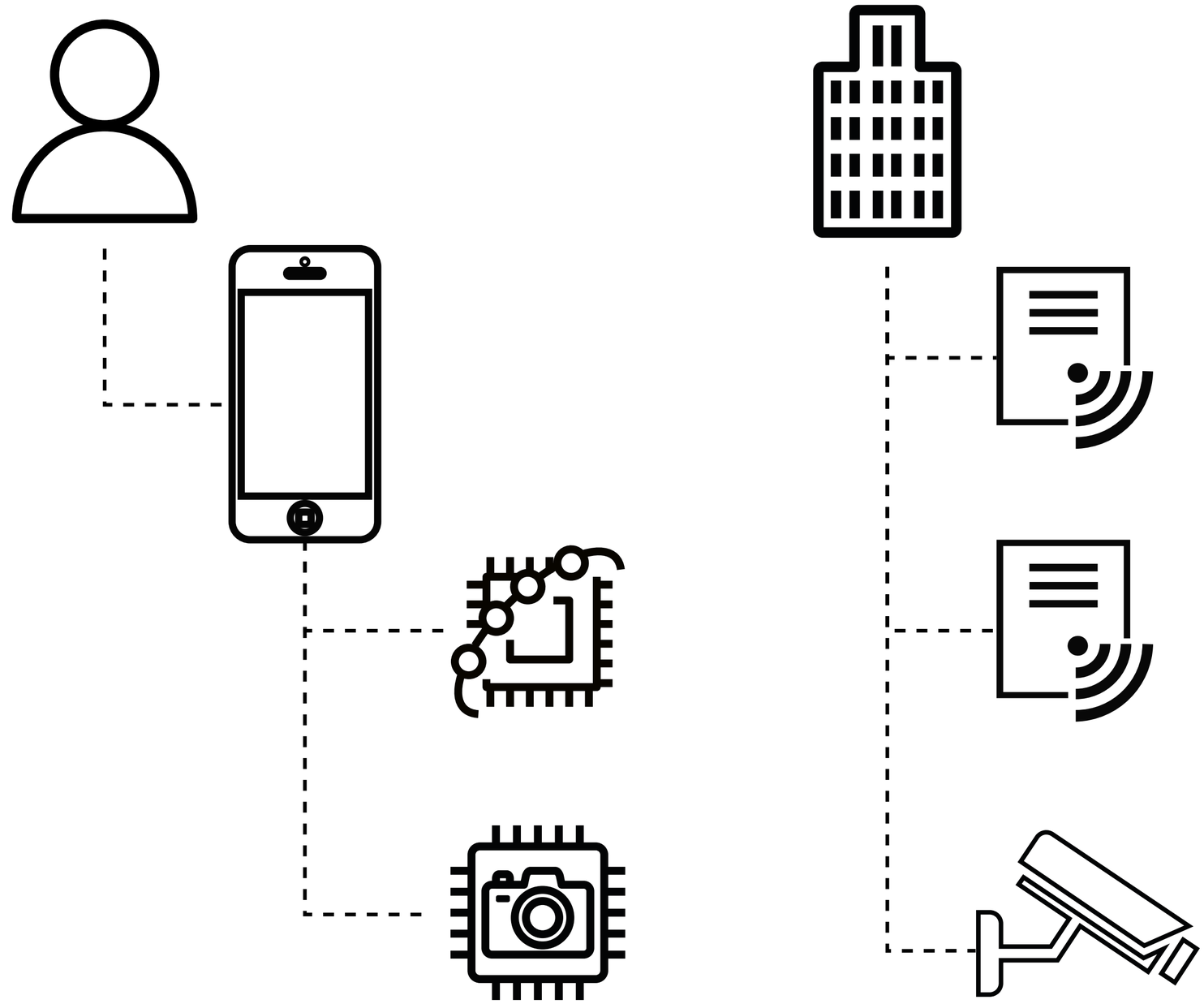
Raw Data



Processed Data

DataObject

OpenHPS



Absolute and Relative Positions

OpenHPS

Absolute

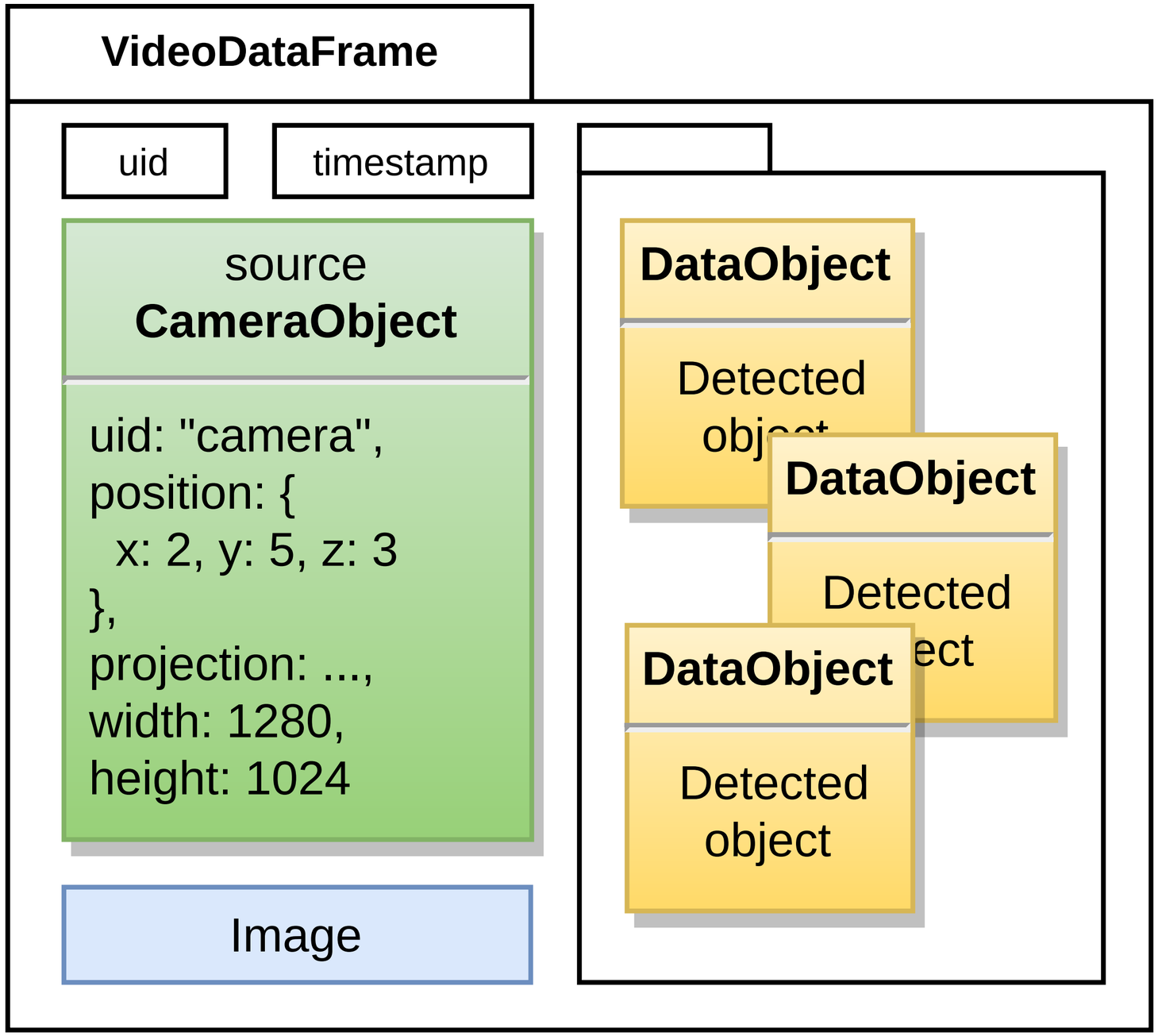
- 2D, 3D, Geographical, ...

Relative

- Distance, angle, velocity, ...
- Relative to another *object*

DataFrame

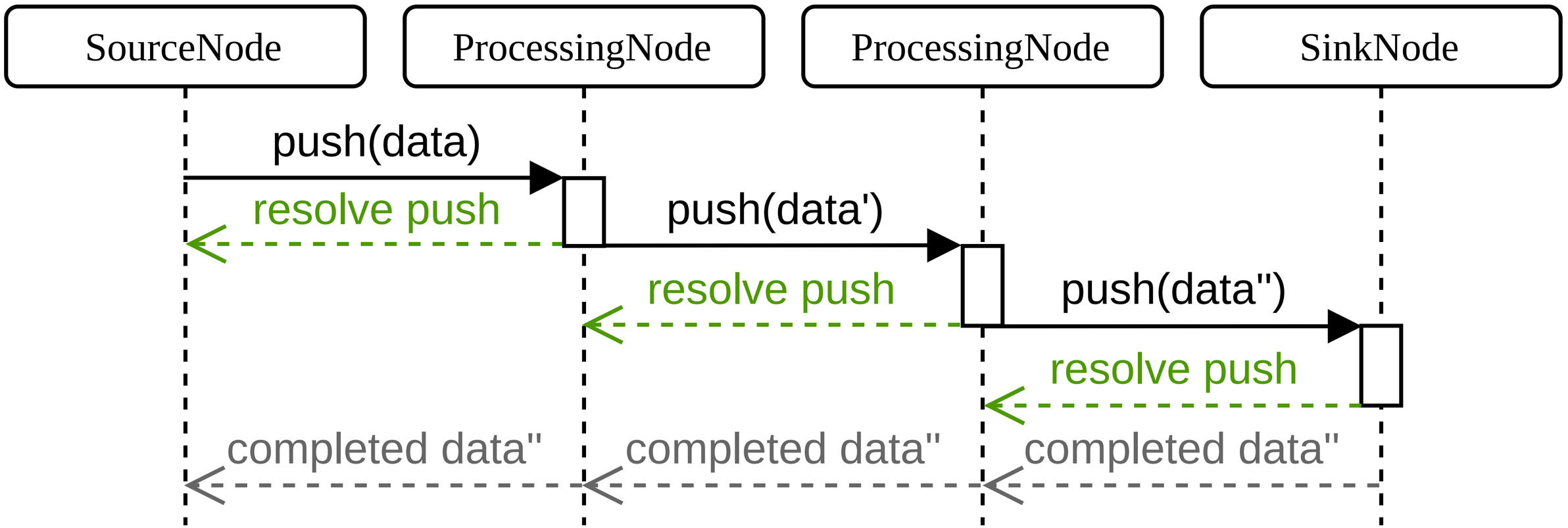
OpenHPS



DataFrame ...

OpenHPS

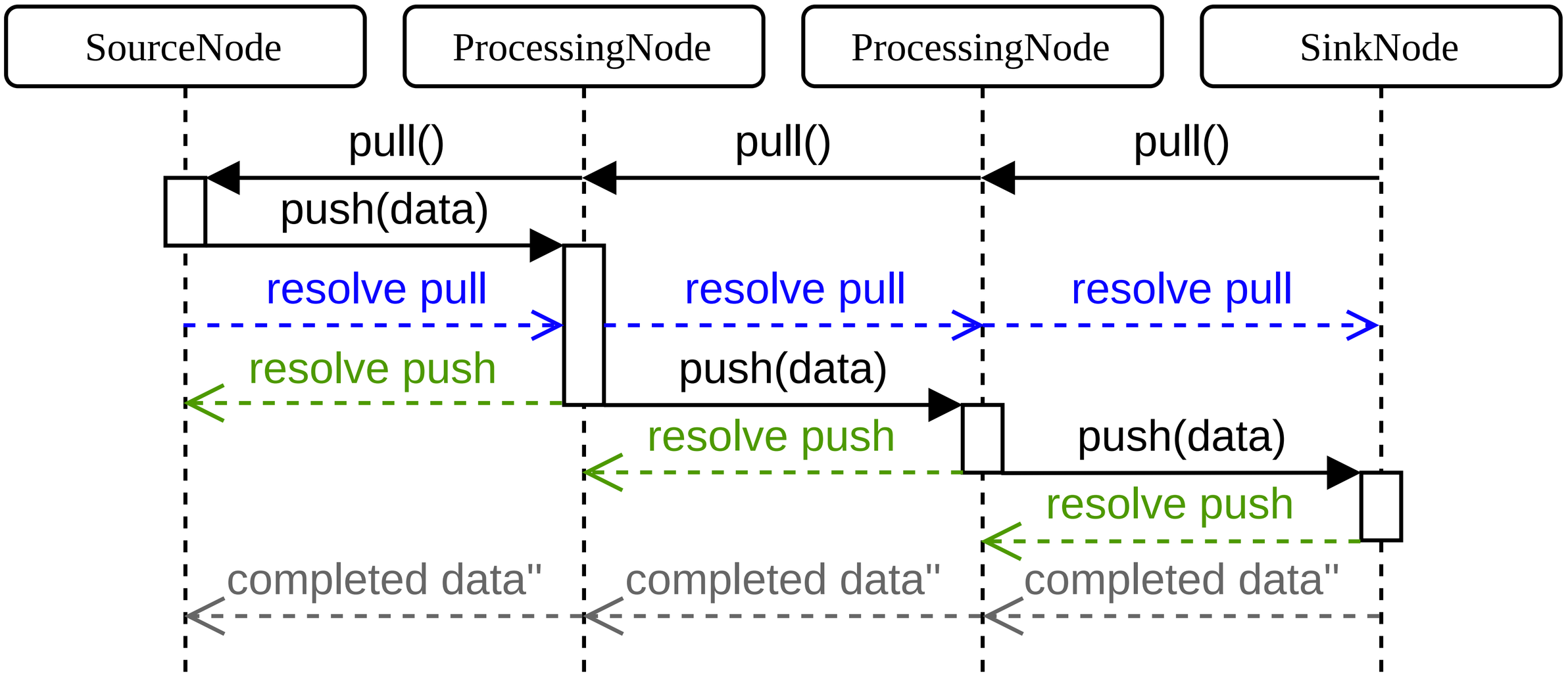
Pushing Data



DataFrame ...

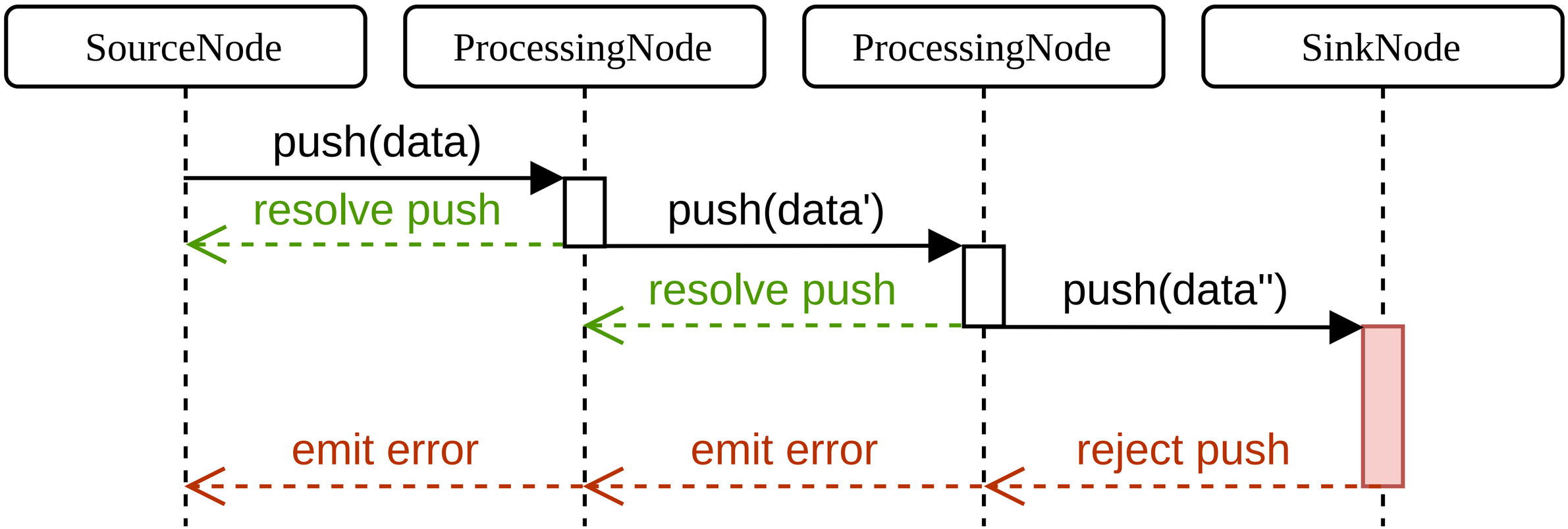
OpenHPS

Pulling Data



DataFrame ...

Pushing Error

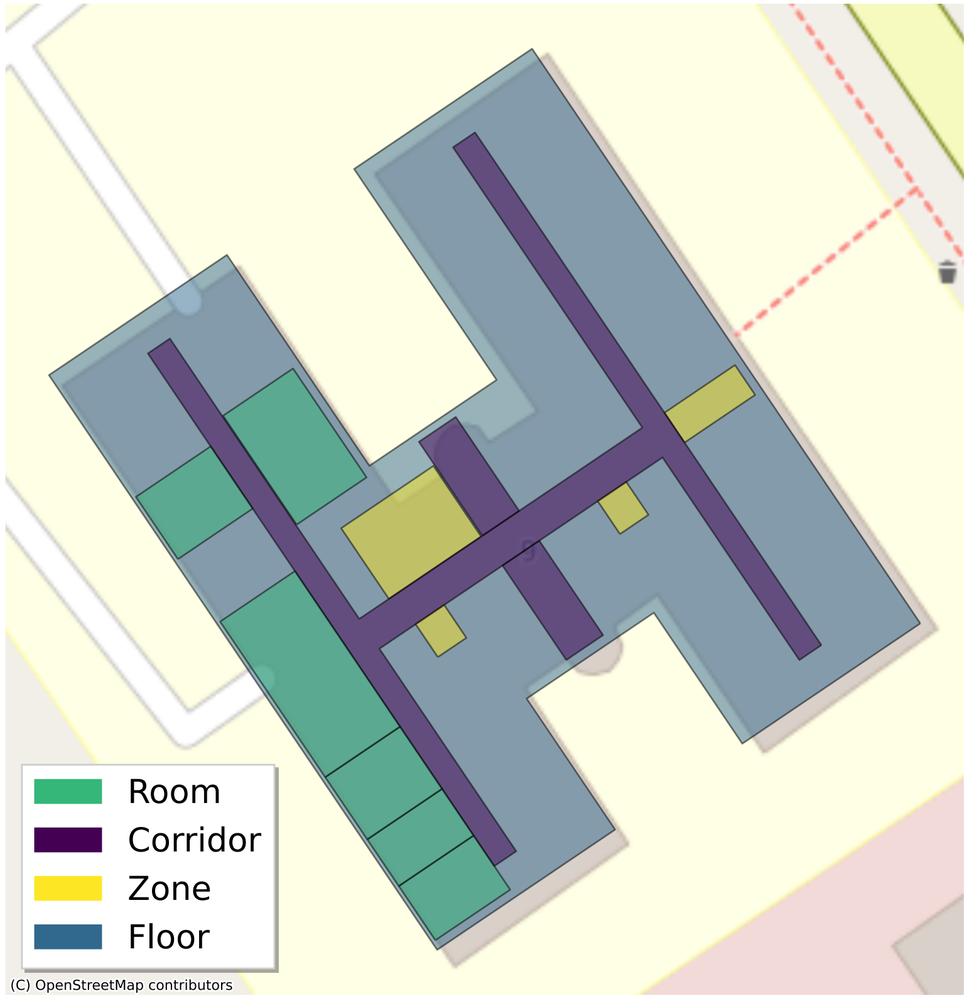


SymbolicSpace

OpenHPS

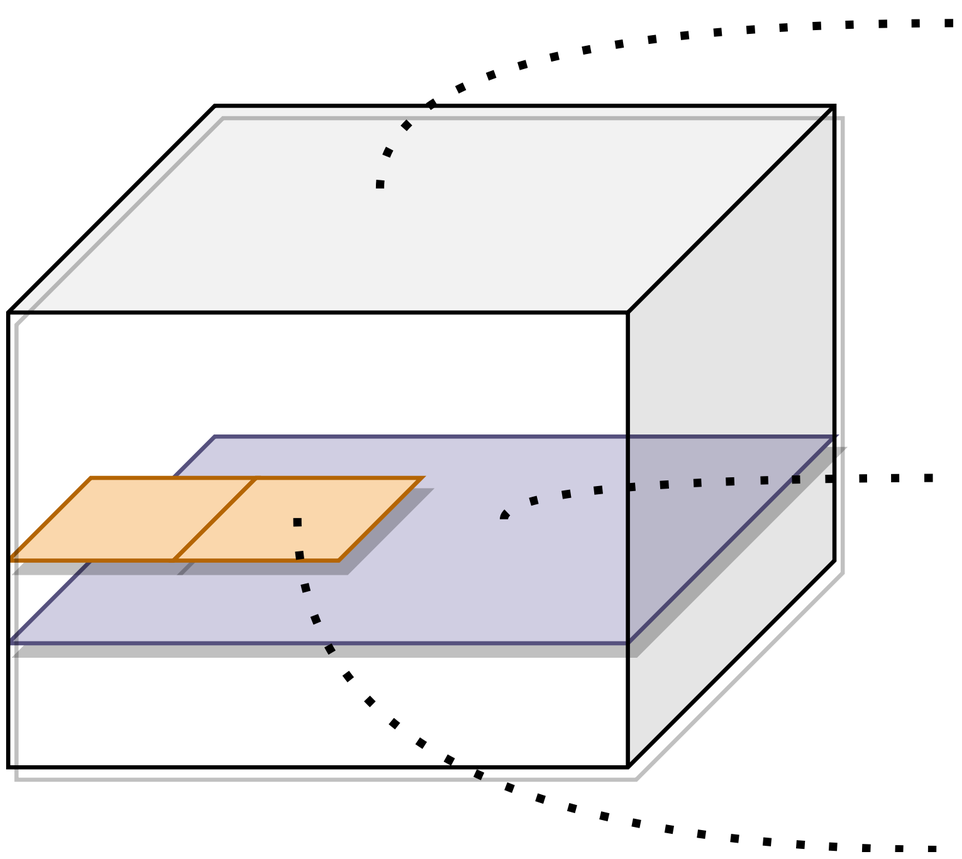
An object that semantically defines a space

- Spatial hierarchy
- Graph connectivity with other spaces
- Geocoding
- GeoJSON compatibility
- Can be used as a location
- Can be extended ...



SymbolicSpace ...

OpenHPS

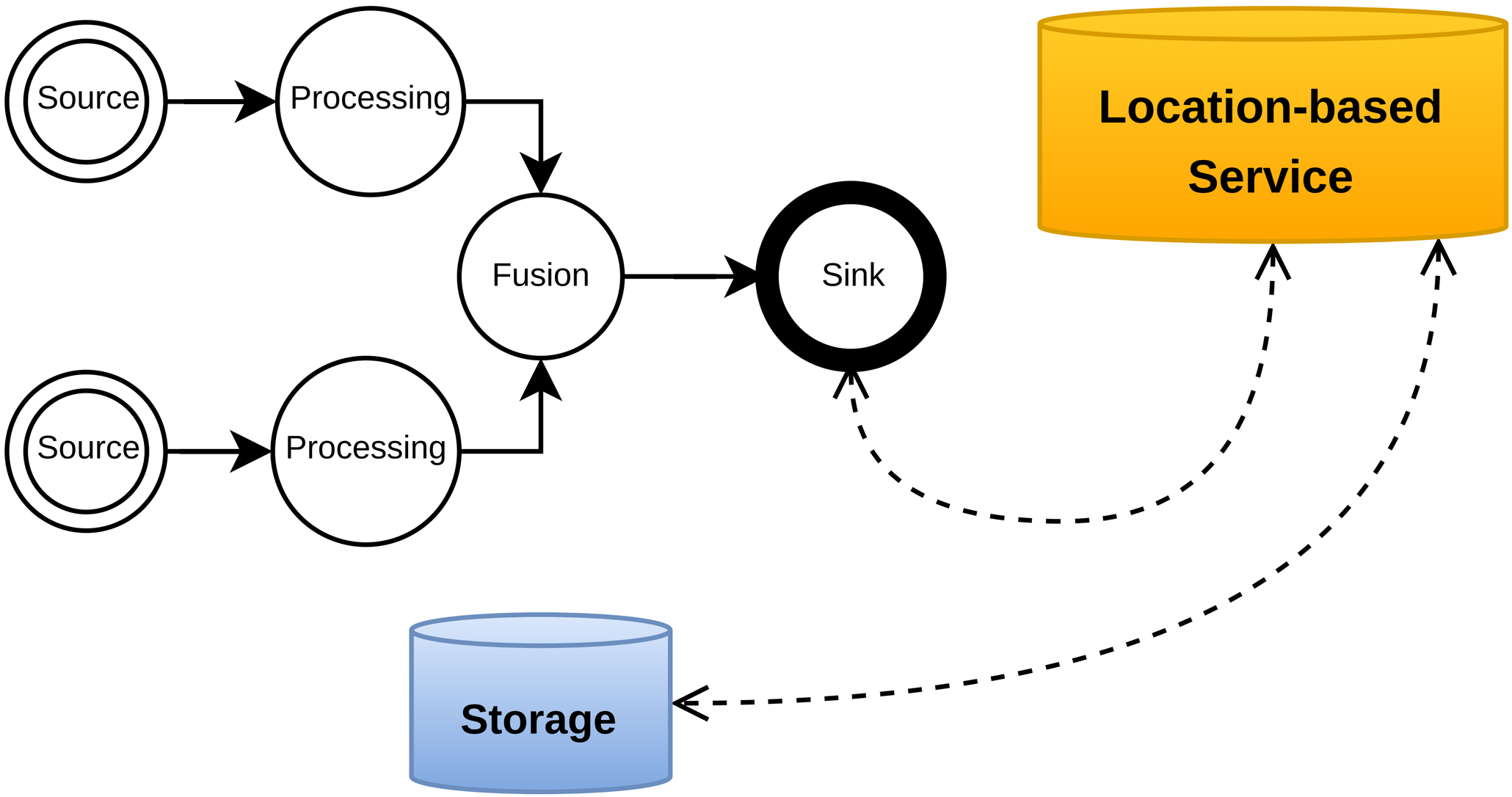


```
const building = new Building("PL9")
    .setBounds({
        topLeft: new GeographicalPosition(
                50.8203,
                4.3922),
        width: 46.275,
        height: 37.27,
        rotation: -34.04
    });

const floor = new Floor("PL9.3")
    .setBuilding(building)
    .setFloorNumber(3);

const office = new Room("PL9.3.58")
    .setFloor(floor)
    .setBounds([
        new Absolute2DPosition(4.75, 31.25),
        new Absolute2DPosition(8.35, 37.02),
    ]);
```

Location-based Service

OpenHPS

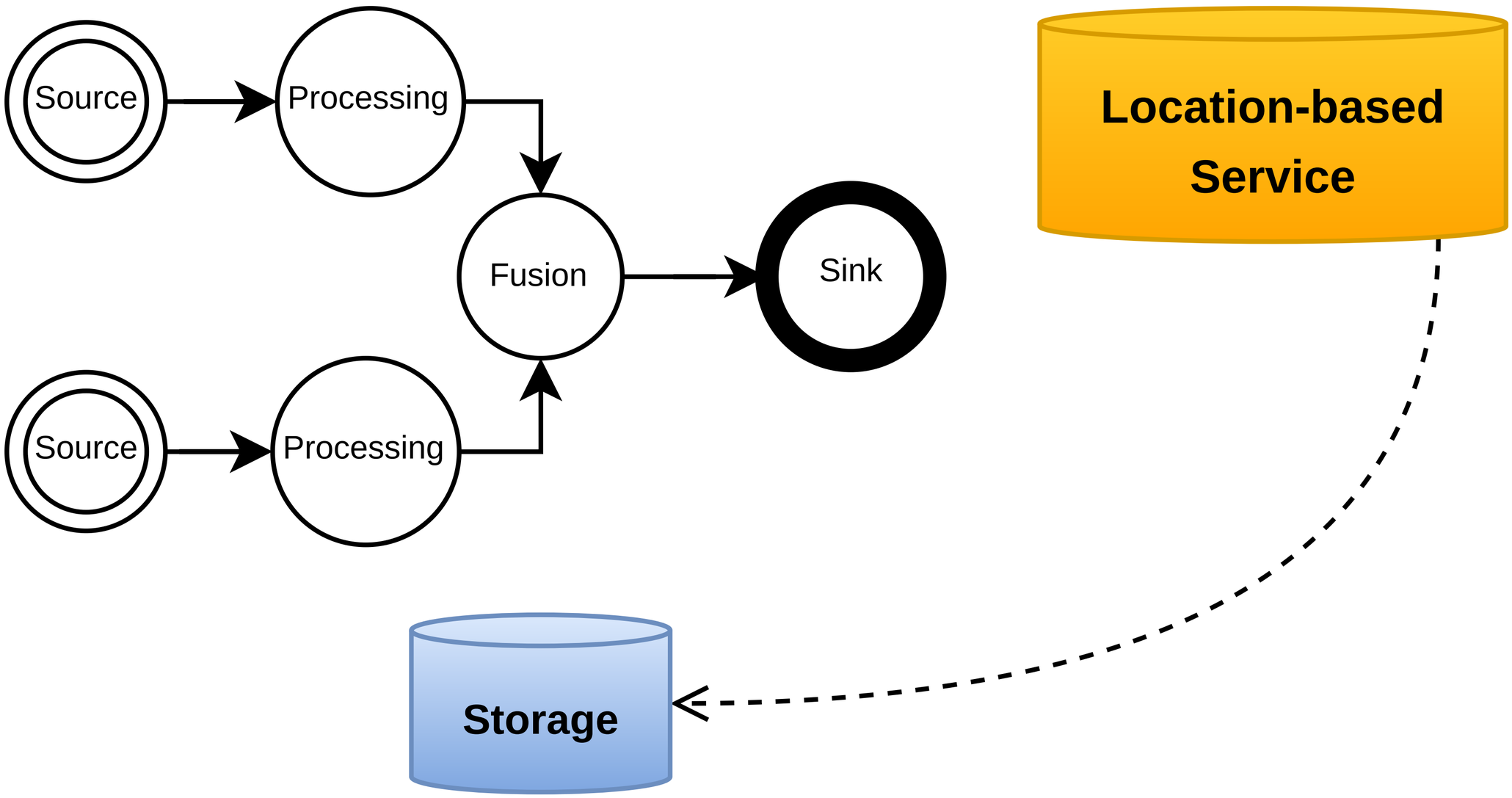
```
getCurrentPosition("me", ...)
```



Location-based Service ...

OpenHPS

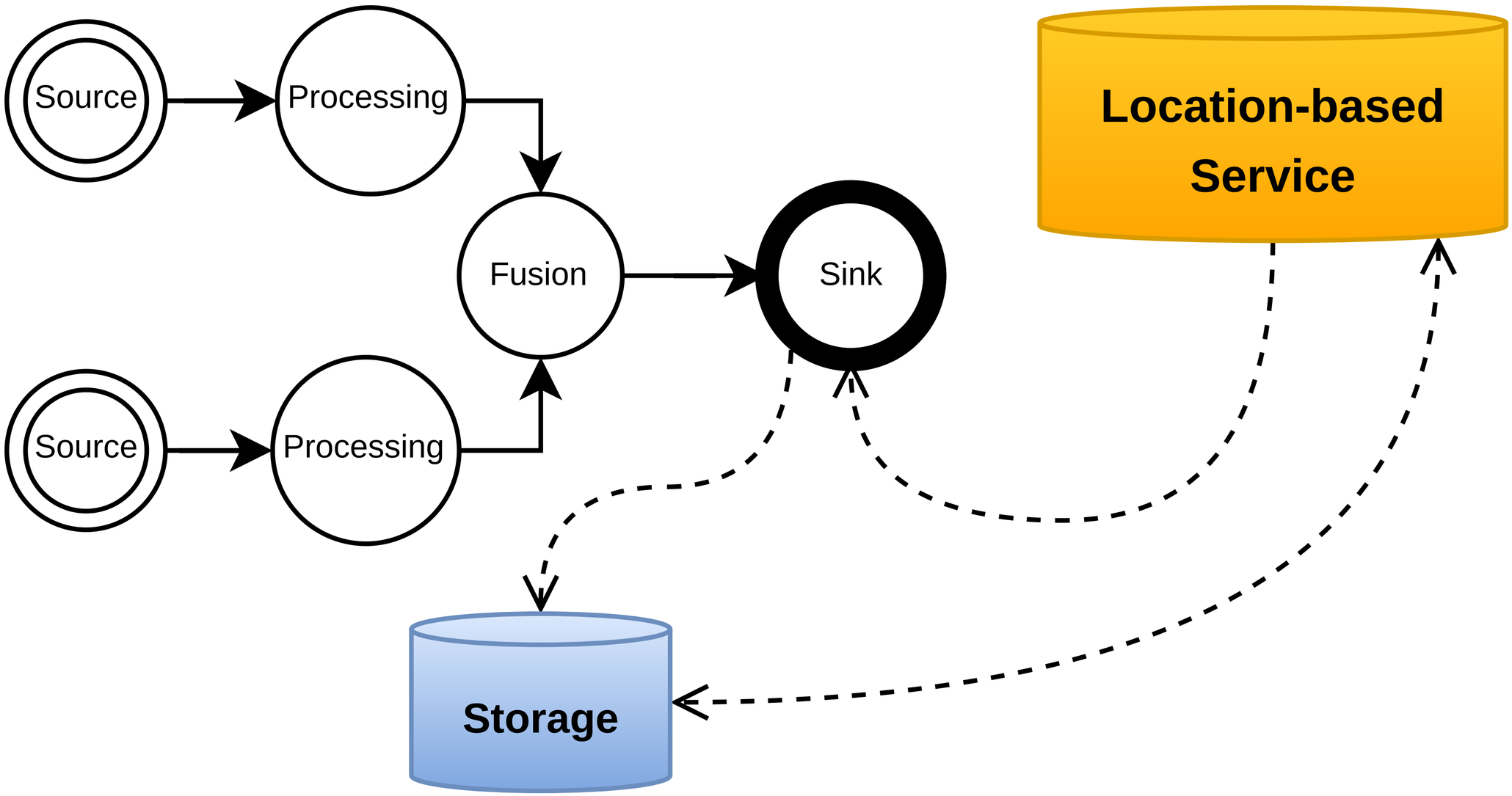
```
setCurrentPosition("me", ...)
```



Location-based Service ...

OpenHPS

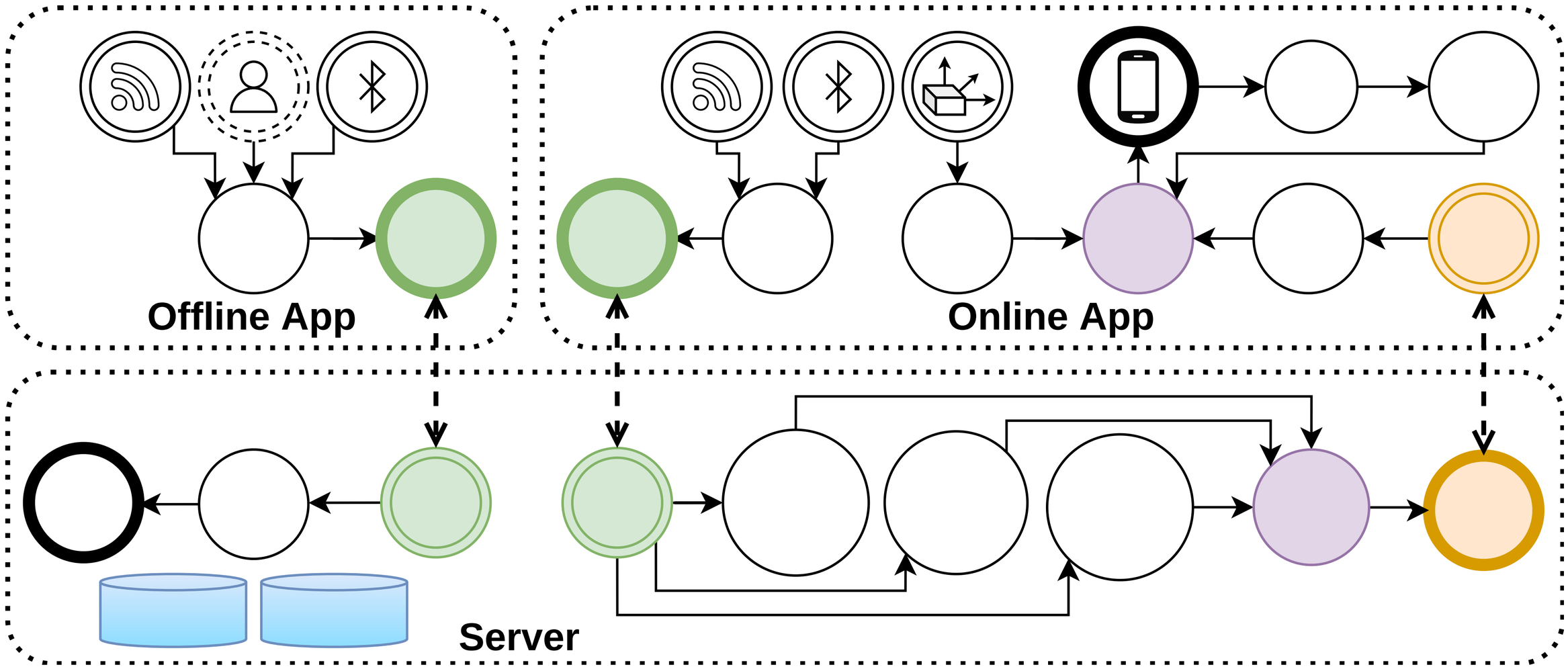
```
watchPosition("me", ...)
```



Demonstration

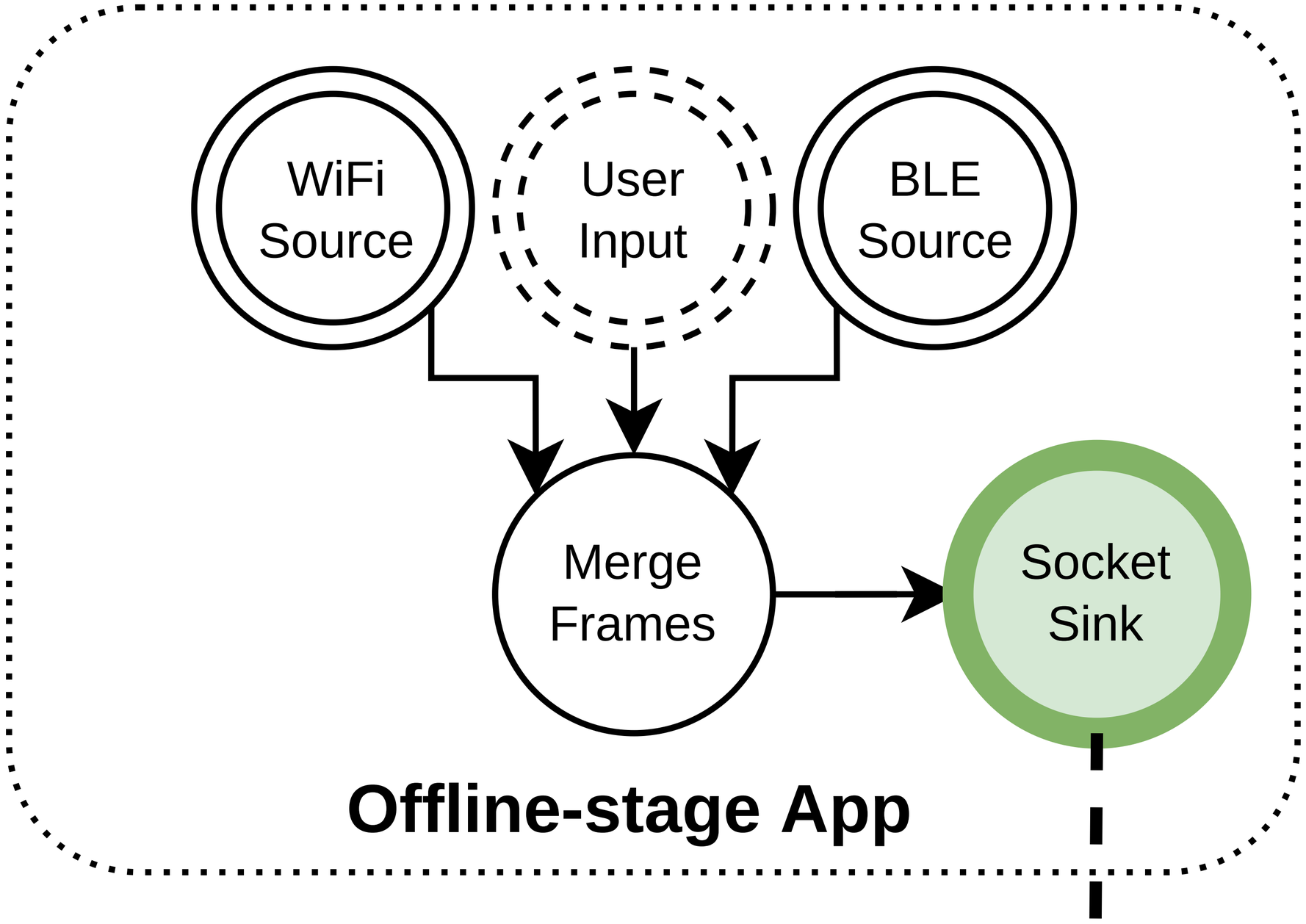
OpenHPS

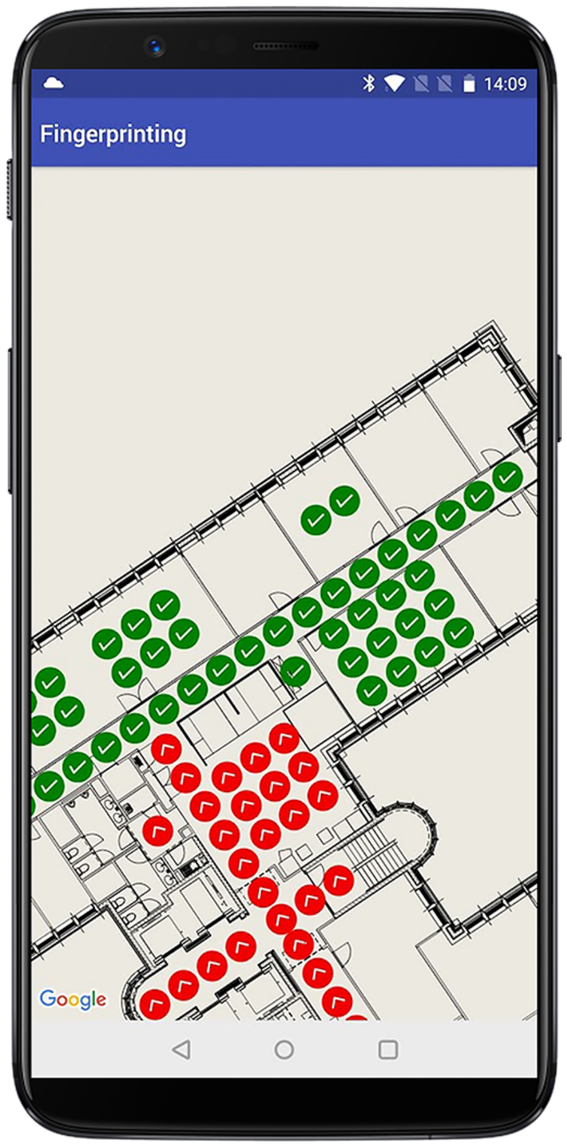
- Indoor positioning **use case**
- Use **existing techniques**
- Validation of **flexibility** and modularity



Positioning Model

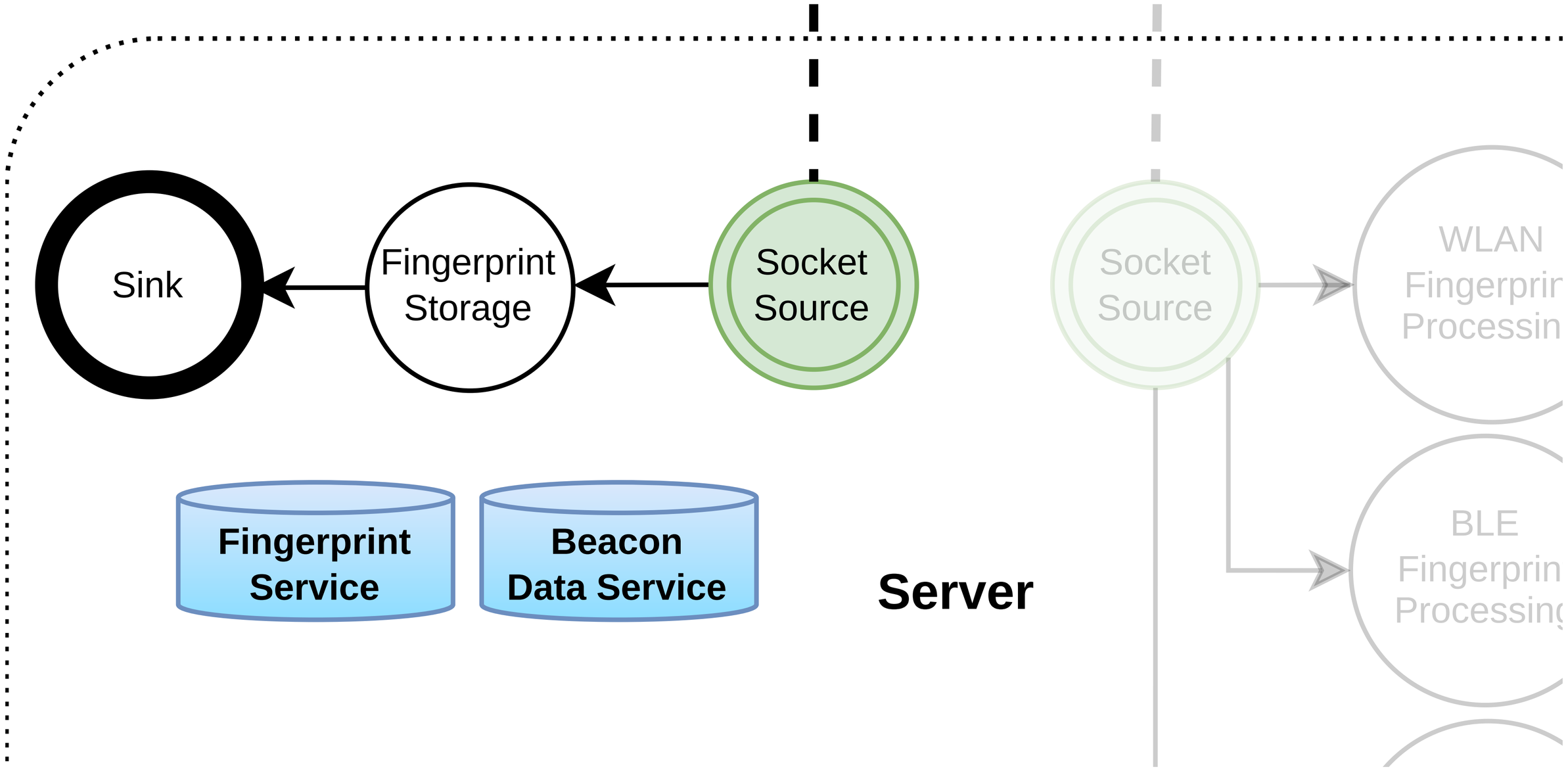
OpenHPS





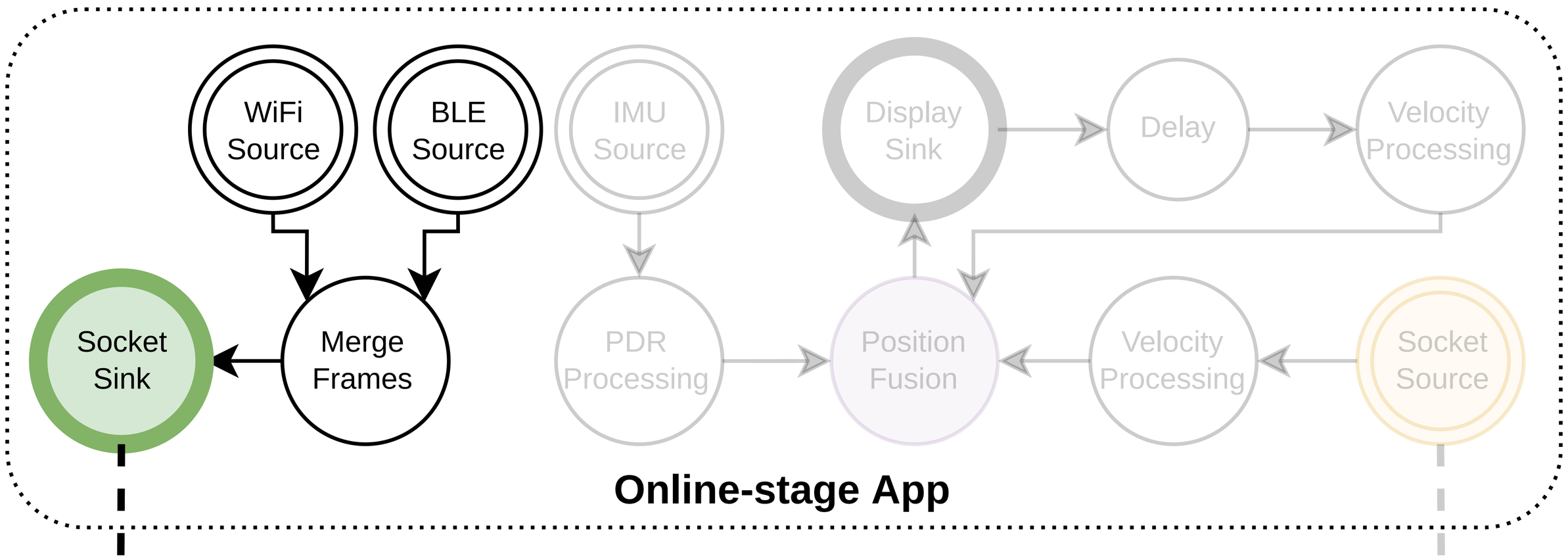
Positioning Model ...

OpenHPS



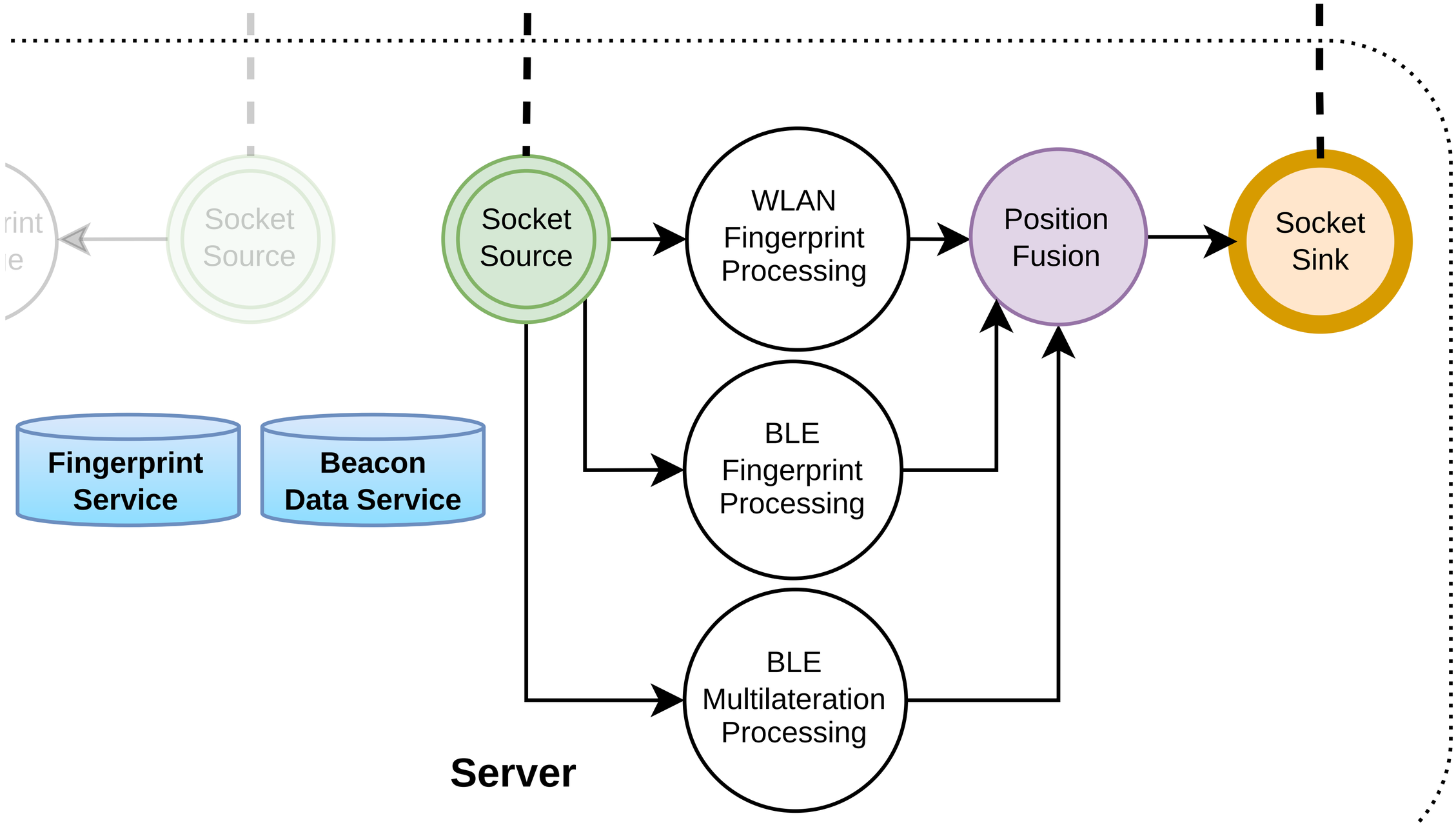
Positioning Model ...

OpenHPS



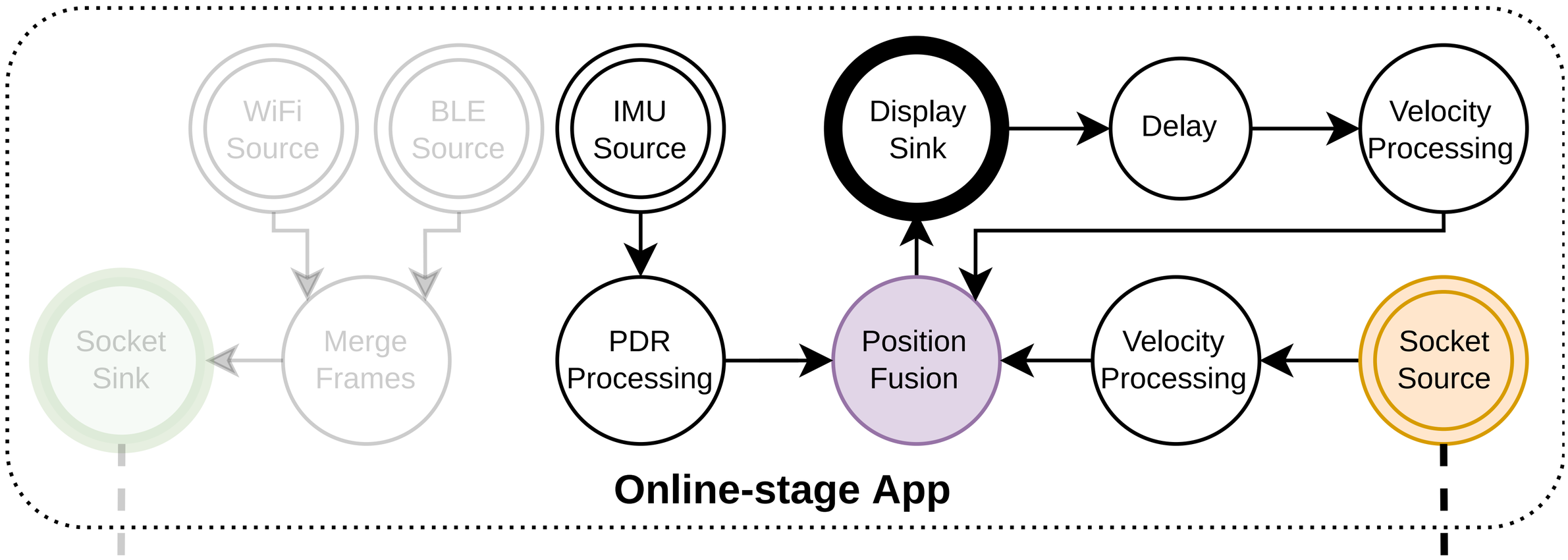
Positioning Model ...

OpenHPS



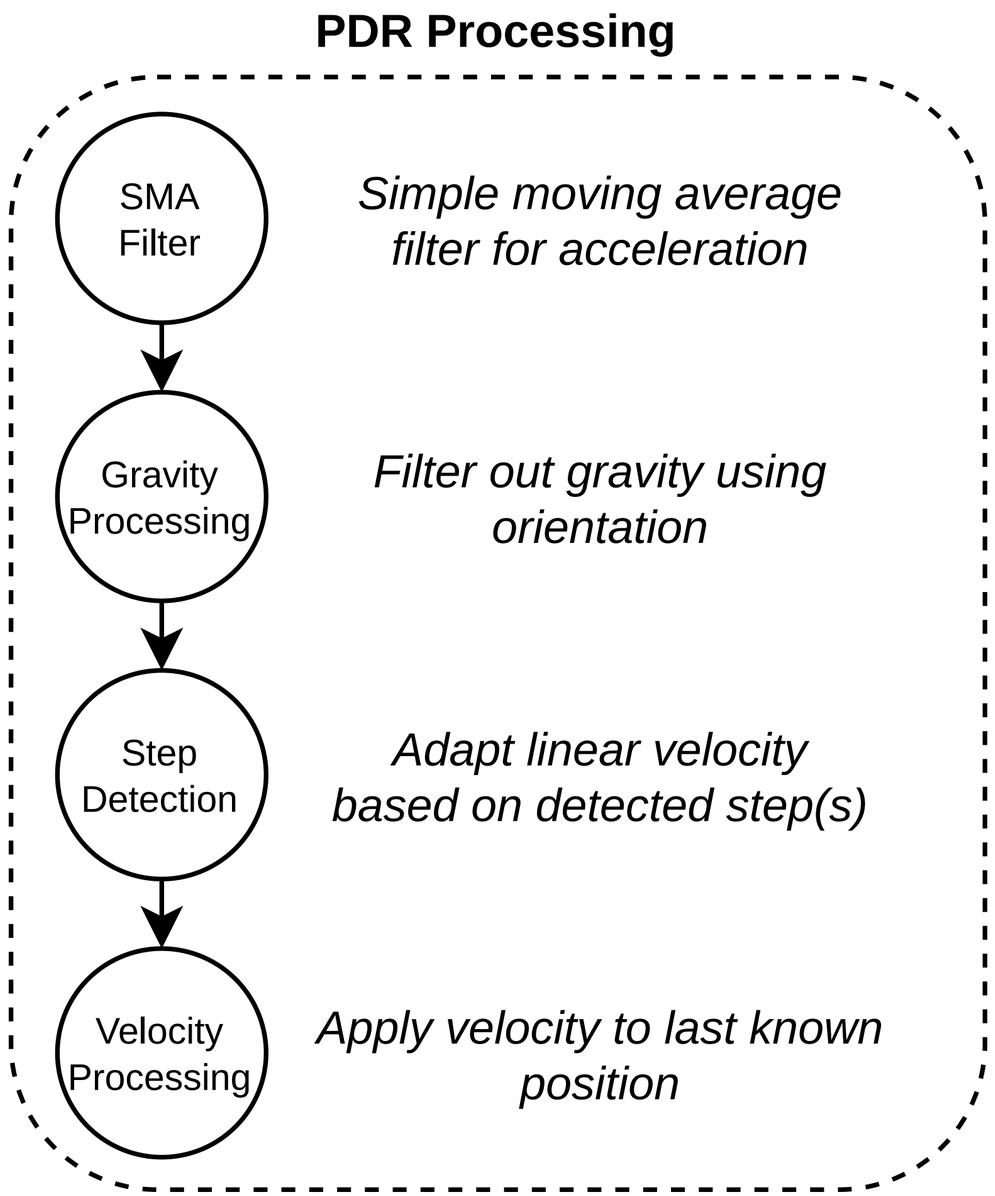
Positioning Model ...

OpenHPS



Positioning Model

Online App



Positioning Model

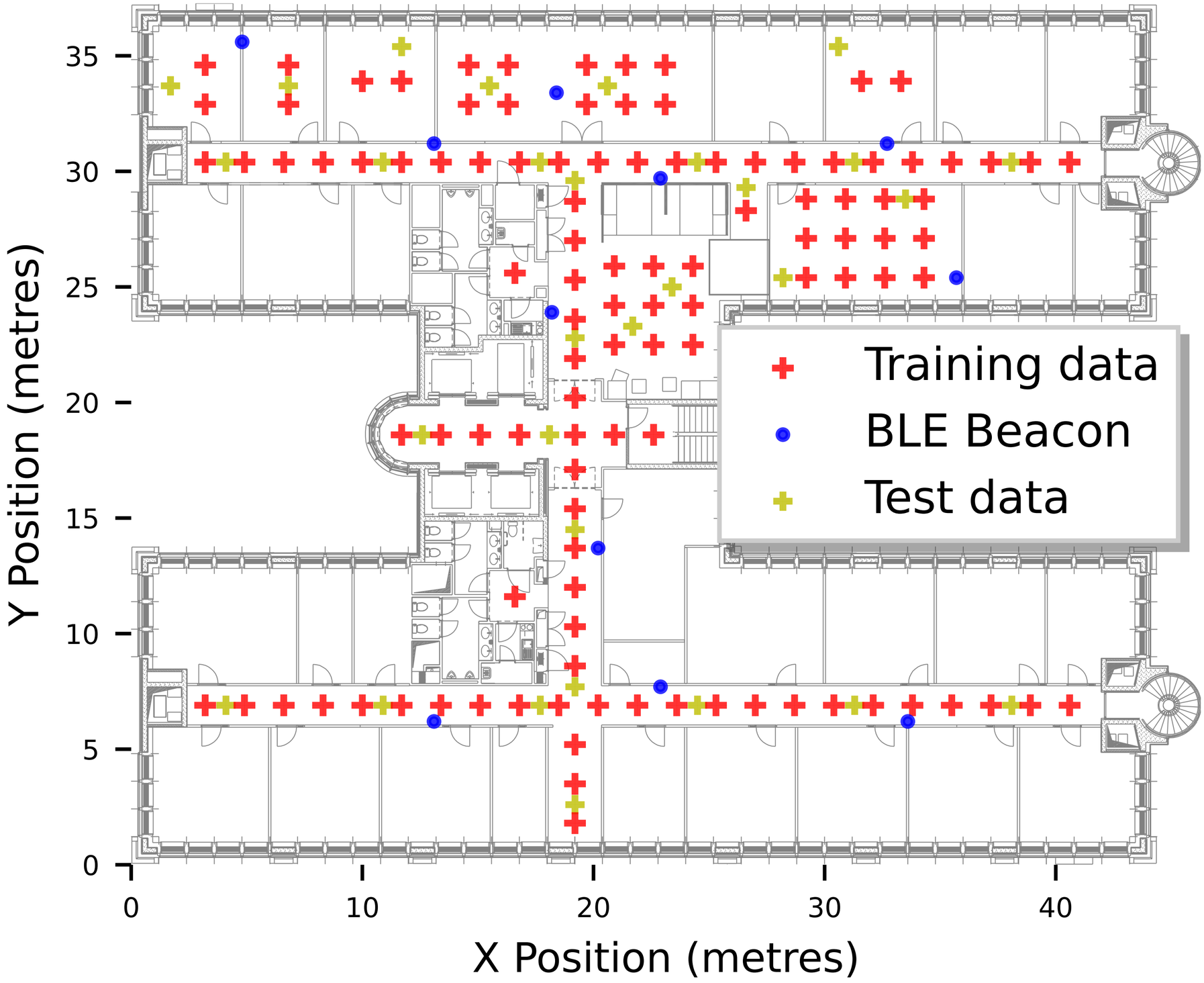
OpenHPS

Online App

```
ModelBuilder.create()
    .addShape(GraphBuilder.create()
        .from(new IMUSourceNode({
            source: new DataObject(phoneUID),
            interval: 20,
            sensors: [
                SensorType.ACCELEROMETER,
                SensorType.ORIENTATION
            ]
        }))
        .via(new SMAFilterNode(
            frame => [frame, "acceleration"],
            { taps: 10 }
        ))
        .via(new GravityProcessingNode({
            method: GravityProcessingMethod.ABSOLUTE_ORIENTATION
        }))
```

Dataset

OpenHPS



Dataset ...

Total BLE Beacons: 11

Total detected WLAN access points: 220

Total stable WLAN access points: 199

	Training	Test
Datapoints	110	30
Total fingerprints	440	120
Duration (per orientation)	20s	20s
Avg. WLAN Scans (per fingerprint)	6	6
Avg. BLE Advertisements (per fingerprint)	16	15

Validation Results

OpenHPS

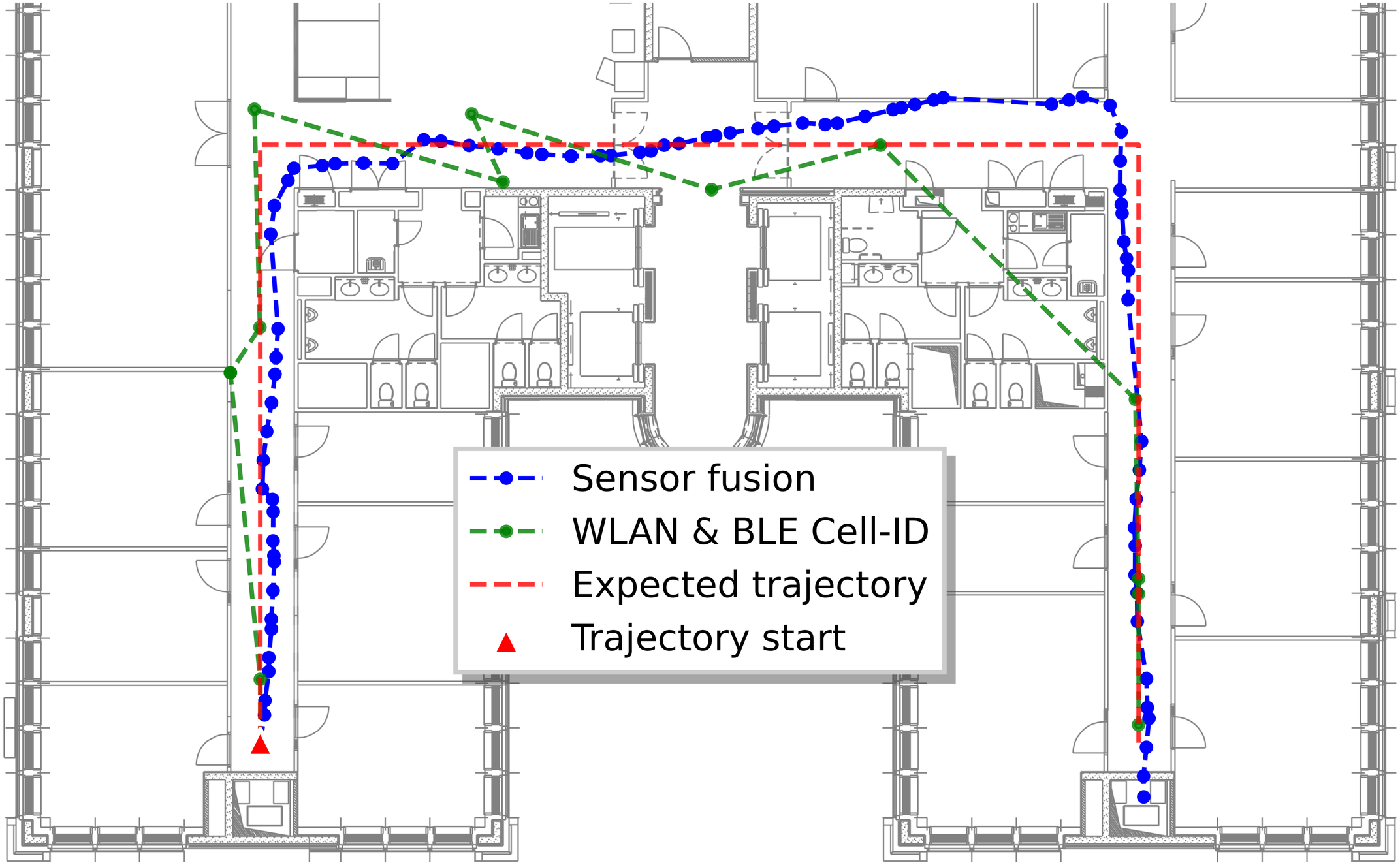
Static Positioning

	WLAN fingerprinting	**BLE** fingerprinting	**BLE** multilateration	**Fusion**
failed points	0	6	12	0
average error	1.23 m	3.23 m	4.92 m	1.37 m
minimum error	0.01 m	0.17 m	0.74 m	0.01 m
maximum error	4.77 m	15.39 m	19.26 m	9.75 m
hit rate	95.82 %	80.83 %	52.50 %	**96.67 %**

Validation Results ...

OpenHPS

Trajectories

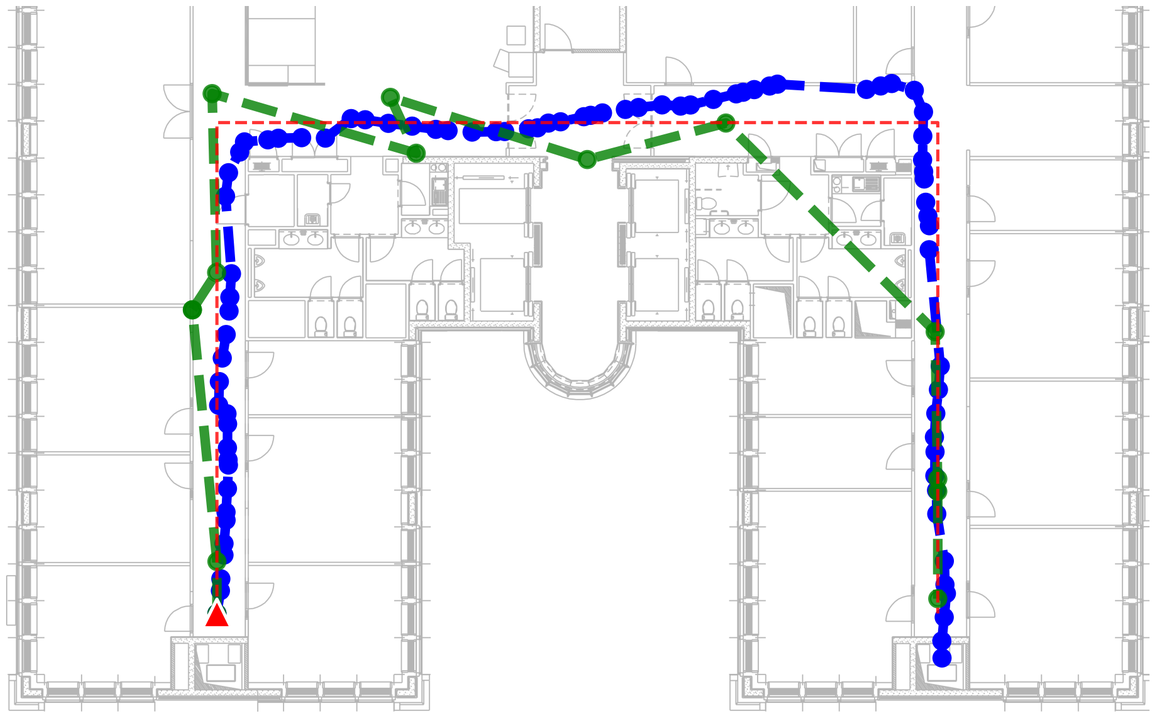


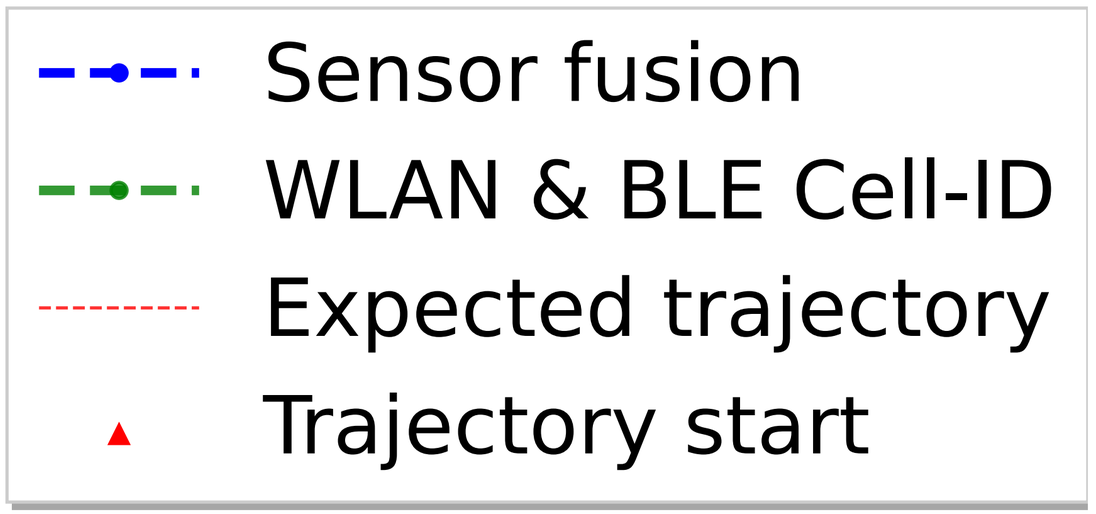
Validation Results ...

OpenHPS

Trajectories

	WLAN + BLE	WLAN + BLE + IMU
average error	3.28 m	1.26 m
maximum error	9.60 m	3.10 m
average update frequency	3.04 s	0.52 s





Contributions and Conclusions

- OpenHPS: **open source** framework for hybrid positioning
  - Aimed towards **developers** and **researchers**
- **Abstractions** such as location-based services and spaces
- Validation of an indoor positioning use case
- Configurable and interchangeable **nodes** and **services**
- **Public dataset** with multiple orientations

Visit https://openhps.org for additional resources, documentation, source code and more!