eMOTIONAL Cities

Mapping the cities through the senses of those who make them

BYTEROAD connecting data

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement nº945307. This document reflects only the author's view and the Commission is not responsible for any use that may be made of the information it contains.

DELIVERABLE 3.4

Data Acquisition Methodologies and Description of the SDIs II

FEBRUARY 2025

Project Title	eMOTIONAL Cities: mapping the cities through the senses of those who make them
Deliverable	**D3.4** – Data Acquisition Methodologies and Description of the SDIs II
Work package	**WP3** – Data infrastructure
Task	**T3.4**
Number of pages	32
Dissemination level	Public
Leader	Byte Road
Main authors	Joana Simoes (Byte Road) Antonio Cerciello (Byte Road) Goncalo Lopes (NeuroGears) Andre Almeida (NeuroGears)
Peer review	NeuroGears
File name	**eMC_2024.12_D3.4_Data_Acquisition_Methodologies_and_Description_of_the_SDIs_II**
Versions	V1: 28-02-2025 (Submitted version)
Revision	-

Index

Conteúdo

Executive Summary

This deliverable, D3.4, presents the methodologies used for acquiring experimental data, based on the collaboration of the experiment partners, supported by the NeuroGears and the final version of the spatial data infrastructure developed by ByteRoad. The system has been refined to capture the requirements that emerged during recent iterations of data acquisition. The data acquisition phase encompasses both the interfaces with various hardware devices used in the experiments and the software required for data capture and processing. The subsequent ingestion phase takes the resulting geospatial datasets and loads them into the various tools provided by the SDI.

A fundamental improvement in version 2.0.0 of the Spatial Data Infrastructure (SDI) is the integration of the OGC SensorThings API (FROST), which enables the direct and programmatic ingestion of raw sensor data. This simplifies the sharing of raw datasets among partners and provides a dedicated programmatic interface for this phase before ingestion. Additionally, the adoption of GeoParquet offers an efficient, columnar storage format that reduces both storage and bandwidth costs while meeting the needs of modern data analysts. Together, these enhancements ensure that the data acquisition phase and subsequent management within the SDI are robust, reproducible, and fully aligned with the goals of open science.

1 Introduction

The eMOTIONAL Cities project aims to understand how the natural and built environment influences human emotions and cognitive processes, ultimately shaping urban well-being. It does so by taking a data driven approach, combining controlled laboratory experiments with field ecological research.

Moving neuroscience experiments out of the lab into real-world settings remains challenging. Quantifying human behaviour and physiology in natural settings is fundamental to interpreting non-invasive measures of brain activity, which are subject to all kinds of movement and environmental artifacts, and noise (Giangrande, Botter, Piitulainen, & Cerone, 2024). The traditional approach of dealing with these difficulties is to restrict studies to indoor laboratory settings, under conditions that minimize all irrelevant movements in hopes of keeping all inputs and outputs to the brain under control as much as possible (Mobbs, et al., 2021).

However, modern neuroscience has also developed an appreciation for the importance of studying brain activity under freely moving and naturalistic conditions (Krakauer, Ghazanfar, Gomez-Marin, MacIver, & Poeppel, 2017; Stangl, Maoz, & Suthana, 2023; Cisek & Green, 2024). The study of navigation and hippocampal place fields is the paradigmatic demonstration that certain brain phenomena become observable only when an animal is allowed to move freely in an environment, and its neural activity correlated with temporally precise and synchronised measures of behavior on a moment-by-moment basis (Pfeiffer & Foster, 2013).

In the eMOTIONAL Cities project, we aim to help lay methodological foundations for freely moving human neuroscience in urban settings. We tackled this challenge head-on by prototyping a new wearable experimental platform combining multiple environmental and health sensors, and which can be used for both indoor and outdoor freely moving neuroscience experiments in natural urban environments.

Our goal was to support the collection of a wide range of behavioral, environmental, and physiological signals from human participants walking in simulated and real environments. Here we describe the technology stack enabling acquisition of multiple heterogeneous data streams to ensure that: a) data is logged using a common reference frame for time and space; b) measurements are comparable across both indoor and outdoor datasets. To achieve this, we have integrated all data collection into a common acquisition framework based on the Bonsai visual reactive programming language (Lopes & Monteiro, 2021), using standard interfaces to access and synchronize each data source.

In order to render it accessible to other researchers, the data collected during the neuroscience experiments is ingested and published in a SDI, along with other contextual data produced/collected in this project. In previous deliverables, specifically D3.2 (Simões and Cerciello 2022a) and D3.3 (Simões and Cerciello 2023), we described and released a preliminary version of the SDI to manage and share refined geospatial

data through a hybrid approach that combined modern and legacy systems. Over time, significant advances in OGC API standards have enabled substantial improvements in our infrastructure, which kept pace with these evolving standards.

This deliverable, D3.4, introduces the final version of the SDI: a more refined platform that integrates an extended set of standards. Among these, the OGC SensorThings API is a major new addition, providing a dedicated REST API that allows internal partners to directly ingest and share raw experimental sensor data. This approach complements the traditional process of manually uploading to the data lake and ensures data can be shared even during the initial ingestion phases. In parallel, the inclusion of GeoParquet addresses the evolving requirements of data analysts by offering an efficient, cloud-native storage format that improves both performance and cost efficiency. The final version of the SDI remains, as it always has been, an infrastructure-independent system that can be rapidly deployed on any cloud provider, reinforcing our commitment to open science and data reproducibility while meeting the needs of a research and development project such as eMOTIONAL Cities.

2 Data Acquisition Methodologies

In this section, we describe the data acquisition methodologies developed in the scope of this WP; we discuss mechanisms for temporal synchronization, which were developed and benchmarked to ensure reproducible timing, and how location data cross-referenced with GNSS or georeferenced VR systems can ensure a common spatial component for tracking participants throughout experiments. All developed software has been released as open-source packages to encourage open data collection techniques and facilitate reproduction for future studies.

2.1 Pluma Wearable Acquisition Unit

To support data collection in both indoor and outdoor experiments, NGR and NGR-PT (for the neuroscience data) and CLIMA (for the environmental data) have developed the **Pluma** multi-environment wearable acquisition unit, based on the HP VR Backpack G2. Such development contemplated insights from FMUL (to allow adequate relationship with the experimental protocols) and SLAB (to allow integration with the wearable EEG device). A common clock source is provided by a custom PCB board implementing the Harp synchronization protocol (see harp-tech.org). An outline of the system configuration for both types of experiment is presented in Figure 1.

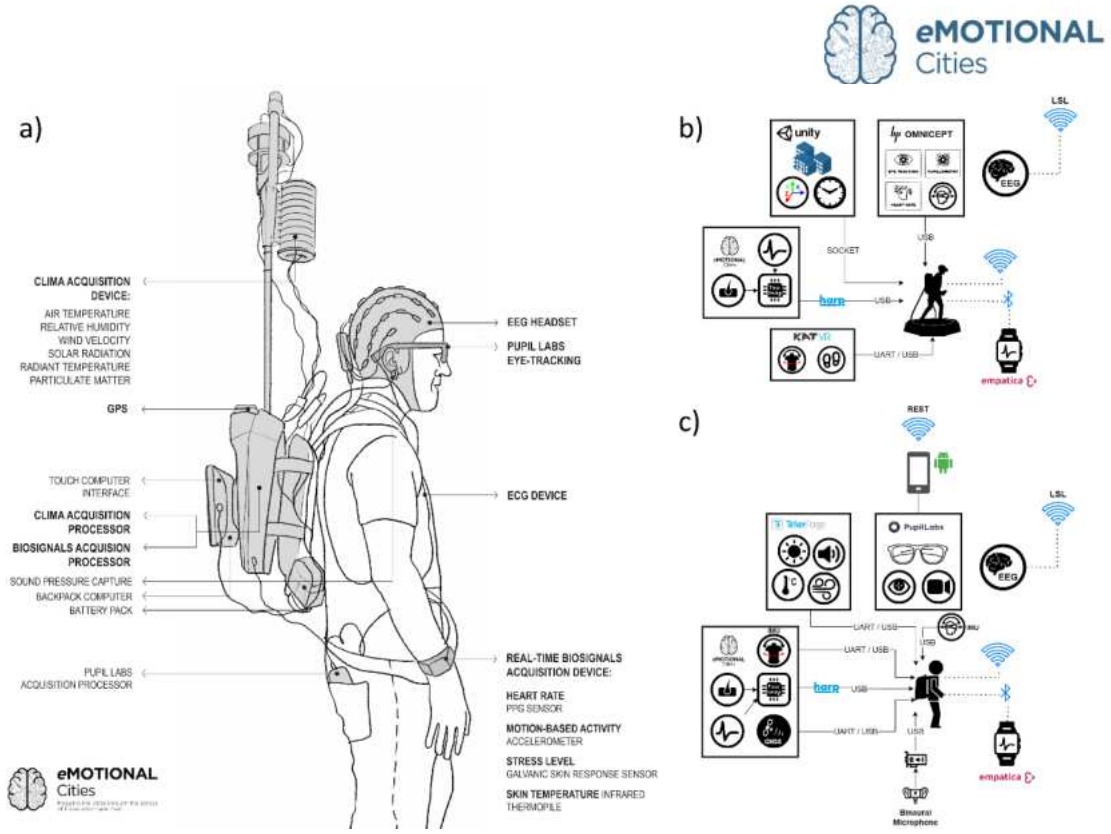

Figure 1. Example configuration of wearable data acquisition unit for both indoor and outdoor experiments. a) the fully assembled wearable unit, including environmental sensors and EEG headset, drawn at scale. b) indoor configuration. Treadmill, head and body tracking together with bio-signals, EEG and streaming of the VR model into the headset. c) outdoor configuration. External climate sensors, eye tracking, wireless EEG and bio-signals.

To isolate both cognitive and emotional processing, NGR, NGR-PT and FMUL have considered hardware/software solutions allowing real-time monitoring of relevant biosignals (heart rate, pupillometry and/or eye movement) during the walks. We have also integrated bidirectional data streaming between our acquisition platform and Unity 3D to enable synchronisation and georeferencing of external biosignals with data acquisition and city simulation in Virtual Reality (VR). Finally, the system allows for ecological momentary assessments, to allow collecting both qualitative and quantitative subjective self-reporting and collection of environmental observations while experiencing the VR environment or outdoor walk.

Crucially, both configurations aim to generate a time-series of accurately georeferenced data in real-time, with temporally precise synchronization of all data streams. If the wearable georeferenced position module is temporally synchronised with the common clock, we can guarantee that all bio-signals, physiology, and behavioural data are themselves georeferenced. Below we provide details on the georeferencing, and temporal synchronisation mechanisms used to ensure consistency and validity of the data throughout the project.

2.2 Georeferencing

For outdoor experiments, we integrated real-time streaming from the ZED-F9R high-precision GNSS positioning module. This allowed us to recover accurate trajectories over long periods of time (about 1h) in a standard geographic coordinate system such as WGS 84 across a widespread range of experimental locations including Lisbon, London, Lansing and Copenhagen (see Figure 2). To further recover the orientation of the subject body we stream data from the BNO055 9-axis absolute orientation sensor mounted in the wearable unit, also temporally synchronised with the GNSS data.

Figure 2. Overview of locations around the world where outdoor data collection experiments were run. The map in the top panel shows the location of each city. For each city, blue dots represent georeferenced city areas in which experiments were run. Multiple data collection sessions were run at each location with different experimental subjects.

For indoor experiments, as part of WP5 we developed an immersive simulation of a virtual environment co-registered with the real city, based on reconstructed 3D data of selected urban areas generated and exported from Google Earth. Custom modelling was required on top of the exported data to ensure simulation quality at the street level (see Figure 3). Crucially, since the model was developed using geospatially registered 3D data from Google Earth, all coordinates in the VR can be mapped into WGS 84, and vice-versa, real trajectories collected in outdoor experiments can be projected onto the simulated environment (see Figure 3d).

We thus used both indoor and outdoor synchronised georeferencing systems to define a standard spatio-temporal reference frame for all freely moving EEG, biosignals, behavioural, and environmental data collected in WP5, ensuring the ingestion of the data is always georeferenced according to OGC API specifications.

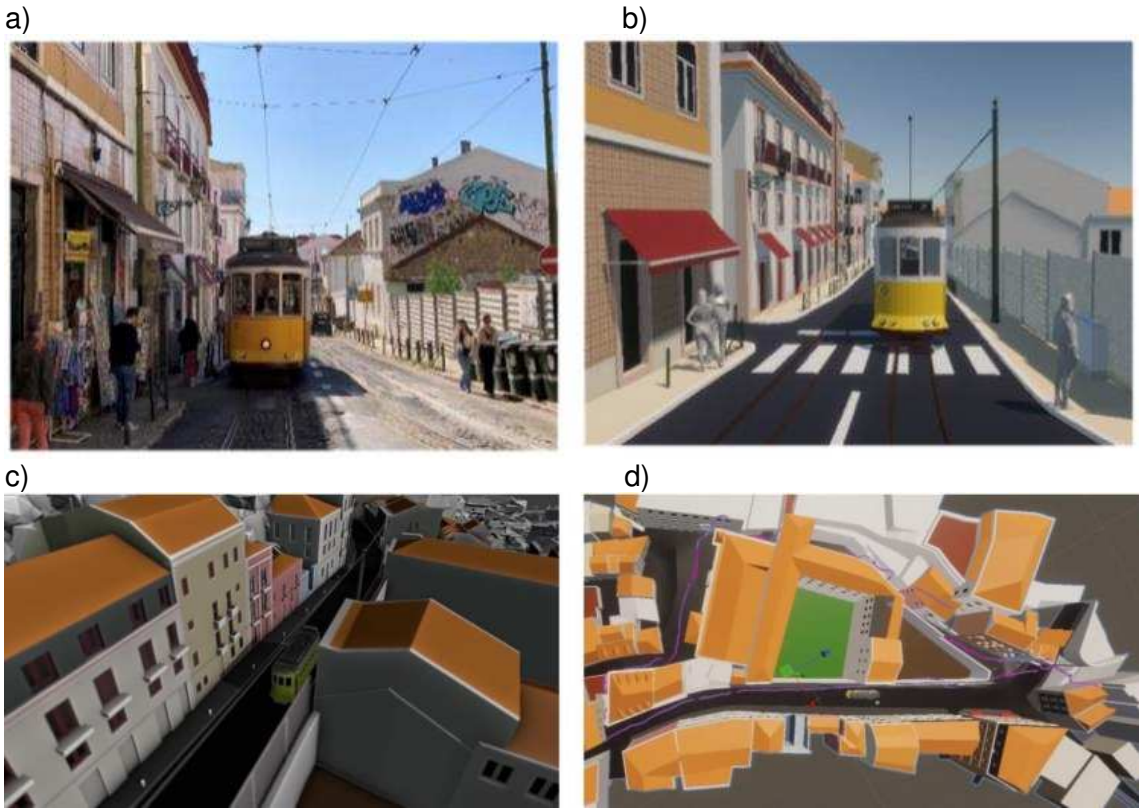

Figure 3. Georeferenced modelling of urban environments. a) capture of 360º video of an urban environment. b) example rendering of the same urban environment modelled in VR. c) contrast of detailed modelling areas (color) compared to non-modelled areas (gray). d) projection of real-world trajectories collected during an experiment onto the modelled georeferenced VR environment.

2.3 Synchronisation

Given the diversity of hardware devices integrated into the wearable acquisition unit, a variety of strategies and benchmarks were required to ensure temporal precision across all the different systems. Our main goals were to ensure that regardless of the synchronisation strategy employed by a specific device, it should be possible to: a) disconnect that device from the system while retaining synchronisation of the remaining devices (modularity); and b) devise a reproducible method to characterise the synchronisation error against the common clock reference (benchmarks).

This was achieved essentially by requiring the presence of the shared Harp clock source, and then proceeding to devise and benchmark each synchronisation strategy always against this reference clock. The sections below discuss how this was achieved for each distinct synchronisation strategy.

2.3.1 GNSS Module

The GNSS positioning module is a core component of the spatio-temporal reference used in outdoor experiments with the system, including EEG and other high-frequency

physiology and behavioural measurements, so we took special care to make sure we could synchronise and benchmark the temporal precision of this system accurately.

To synchronise the two clocks while ensuring resilience against spurious failures, we trigger a TTL (Transistor-Transistor Logic) compatible signal in one of the digital outputs of the Harp Pluma base device pseudo-randomly every 8-16 seconds with a pulse width of 100 ms. This TTL signal is sent to the GNSS positioning module interrupt input/output pin (INT) and generates a hardware timestamped message (TIM_TM2) via the UBX protocol to the system. By detecting and correlating both Harp timestamps and UBX timestamps for each toggle event, we can build a robust linear model to predict Harp timestamps from any UBX timestamp (see Figure 4).

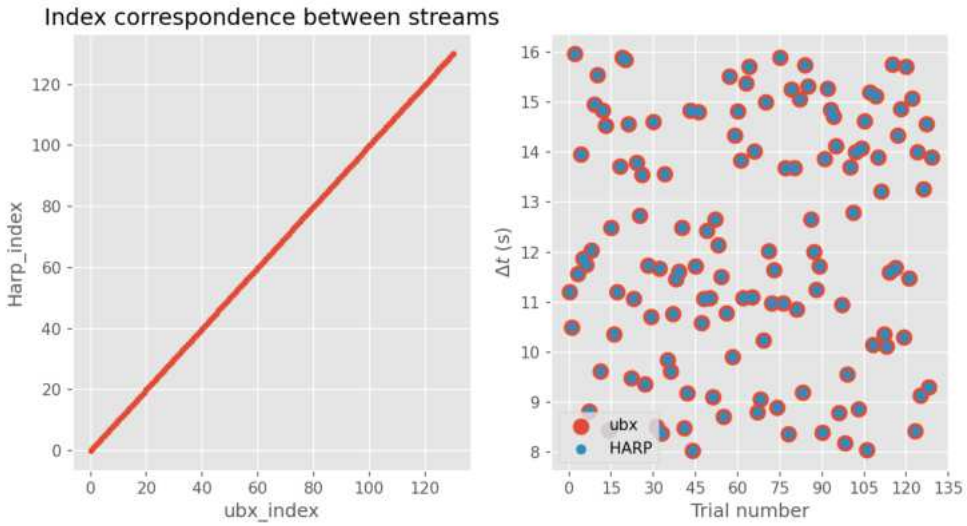

Figure 4. Example clock calibration data used to correlate UBX timestamps to Harp timestamps. For each detected rising edge event we compare the time delta with the previous event as measured on both Harp and UBX clocks (right panel).

In good operating conditions, we expect an exact overlap between both sequences and the linear regression should have a very high coefficient of determination ($R^2 > 0.99$). If a dataset ever falls below this threshold, we signal a warning so further investigation and diagnosis can be made.

This pseudo-random TTL sequence can also be sent to various other subsystems for benchmarking and validation. On devices lacking a direct wire input (e.g. camera streams) we can use the TTL to drive an LED for optical detection of the synchronisation pulse.

2.3.2 Lab Streaming Layer

EEG recordings for both indoor and outdoor experiments are collected via the Neuroelectrics Enobio 32 headset using a proprietary acquisition software which supports streaming data into external systems using the Lab Streaming Layer (LSL) protocol. The LSL protocol is standard in the EEG field and provides a mechanism to measure momentary clock offsets between different receiver and emitter nodes.

Synchronisation of EEG data streams is performed by sending an event marker via LSL from the acquisition system to the EEG device whenever a new pulse is triggered from the pseudo-random TTL sequence described in 1.4.1. Each event marker carries a 32-bit numeric payload, which we set to a monotonically incremented counter, starting at 1, so we can detect and correct for any packet loss. As we also use event marker codes for other applications, we set the counter limit to 35000 which should work for up to 75h continuous recordings, well beyond the limits of any recording session.

Upon reception of the LSL packet, the EEG system logs a new event marker together with the current EEG system timestamp. Like in the synchronisation strategy discussed in Section 1.4.1, we correlate the TTL Harp timestamps and the EEG event marker timestamps to build a linear model allowing us to predict Harp timestamps from any EEG measurement timestamp. The coefficient of determination is expected to be very high ($R^2 > 0.999$) and any datasets below this threshold will automatically trigger a warning for further investigation.

LSL benchmark protocol

All the source code required to reproduce and analyse the benchmark experiments described below has been published at https://github.com/emotional-cities/pluma-benchmarks/ under the Benchmarks/LslJitterTest folder.

To characterise the synchronisation performance of this method, we connected two different computers to separate Harp devices sharing a single digital line. We also connected the two computers over a local WiFi network to benchmark the temporal jitter in LSL timestamps of the same physical trigger using different postprocessing clock correction methods. LSL streams on a local network by default do not perform any clock correction, but receiver nodes can be configured to use NTP to continuously correct the local clocks of wireless LSL streams.

In each experiment, a random timer on the first machine triggers a pulse on the digital output line in the first Harp device. When this line is physically set, this will trigger both a message from the first Harp device to the first computer, and an interrupt on the second Harp device which also immediately triggers a message to the second computer. Therefore, on both computers we have a near-simultaneous response to the change of the digital line. At that time, each computer pushes a sample to its LSL stream outlet. Two LSL stream inlets on each computer read and store the sequence of LSL timestamps from both experimental machines.

Since the two LSL streams are triggered by the exact same physical event, the timestamp difference between messages received from these streams can be used to estimate the error in LSL network timestamp correction (see Figure 5).

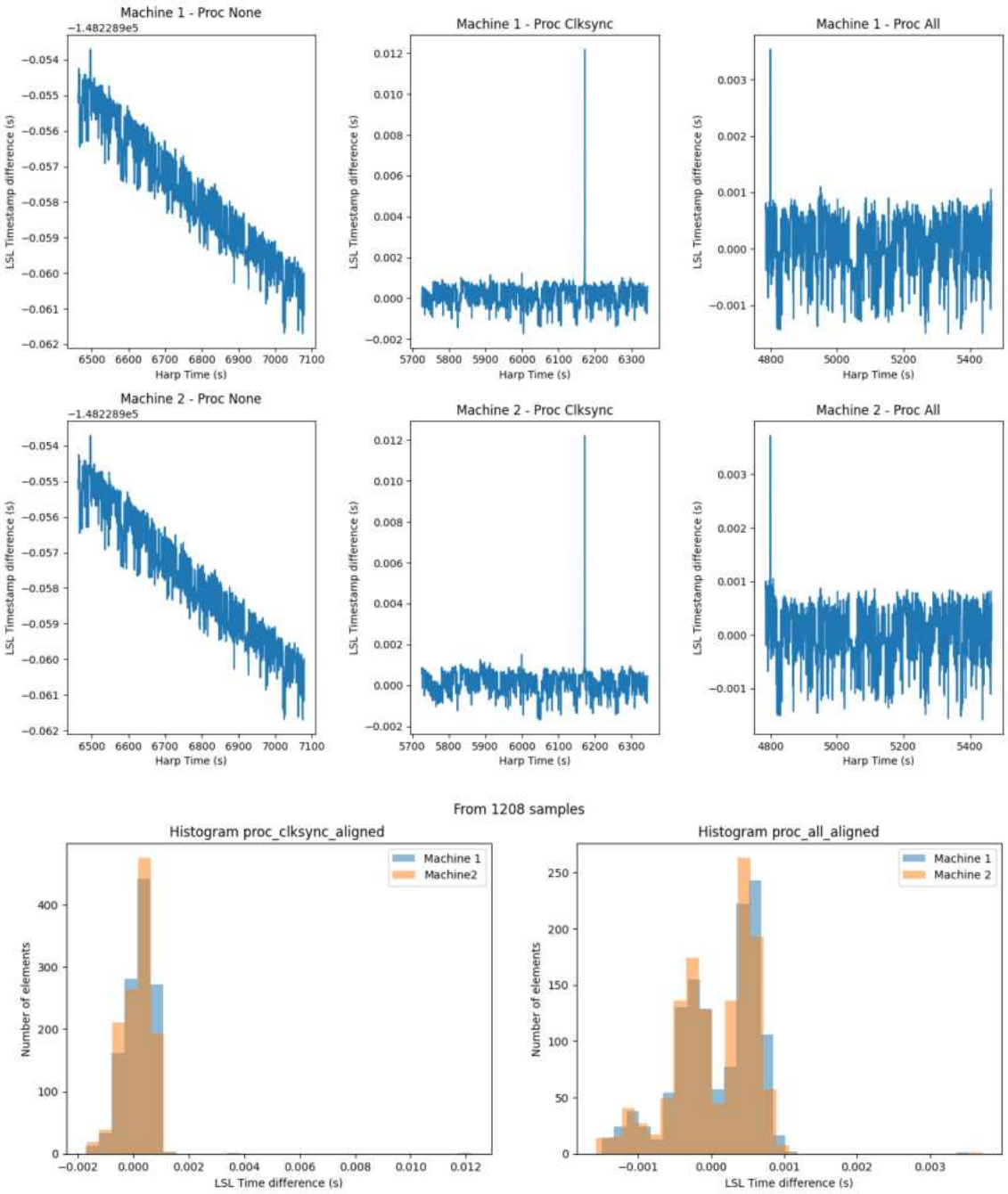

Figure 5. Example LSL timing benchmark. The results using three different clock synchronisation procedures are shown: proc-none) no LSL clock correction is performed; proc-clksync) the LSL network attempts to synchronise the clocks between the machines but performs no other correction; proc-all) the LSL network performs all postprocessing options available, including clock sync, de-jitter, and monotonisation of the timestamps. The distribution of time differences for both the proc-clksync and proc-all procedures are shown in the histograms on the bottom panels.

The observed differences are typically on the order of single-digit milliseconds, which is consistent with the timing precision of the LSL protocol clock correction mechanism.

2.3.3 Hardware Trigger

A few specific systems in the Pluma wearable unit can have their acquisition directly driven by an external TTL signal. Where this is the case, we have used the Pluma Harp device as a periodic TTL source, which automatically guarantees an accurate hardware timestamp below 100 microseconds.

One example of this is the BNO055 inertial measurement unit sensor. This device receives a hardware clock trigger, and all responses are then communicated to the computer via a serial line. Since we know the exact timestamp of each trigger, this information is used to timestamp all measurement data in the raw data log directly.

2.3.4 Clock Offset

Several proprietary devices integrated into the system have their own internal low-jitter hardware timestamps, but do not have direct synchronisation support with external systems. Typically, they will be capable of directly streaming data and timestamps into a client computer via a wireless connection (e.g. Empatica E4 over Bluetooth) or using inter-process or network communication protocols (e.g. Pupil Invisible over ZeroMQ).

In such cases, we preserve the timing information in the hardware timestamps by using a series of Harp timestamped data packets to estimate the offset to the device clock.

2.3.5 Software Timestamp

Several environmental and biosignals measurement sensors have relatively low sampling rates (< 10 Hz). If the sampling period of the sensor greatly exceeds the communication latency required to receive each value in the computer, we have chosen for simplicity to synchronise such measurements with a software timestamp.

The software timestamp is always sampled from the most recent Harp clock time arriving at the computer, which has an average latency of 1-3 milliseconds.

2.4 Hardware Acquisition Devices

The Pluma wearable acquisition unit integrates and logs data from more than 25 different data streams collected over at least 10 unique devices all operating simultaneously. In addition, it can drive specific experimental protocols for both outdoor and indoor environments using a modular workflow developed in the Bonsai programming language (bonsai-rx.org) and published as an open-source repository at github.com/emotional-cities/pluma-experiments.

Below we have grouped all the sensors and data sources by device together with their acquisition sampling frequency, to make it easier to understand the full scale of the datasets which are possible to acquire with the Pluma unit. The modularity of the system makes it possible to disconnect or exchange any device, and this has been leveraged throughout the project to run different types of experiments.

Pluma base board:

- Harp clock
- ECG (50 Hz)
- EDA (4 Hz)
- ZED-F9R GNSS Module (1 Hz)
- Tinkerforge Bricklets (environmental sensors):
    - Air Quality (1 Hz)
        - IAQ
        - Temperature
        - Humidity
        - Air pressure
    - Particulate Matter (~1 Hz)
        - PM 1
        - PM 2.5
        - PM 10
    - Sound pressure level (~20 Hz)
    - Current Sensor 0-20mA Solar Light (~10 Hz)
    - Thermocouple Radiant Temperature (~1 Hz)
    - PTC Air Temperature (~1 Hz)
    - Humidity (1 Hz)
- Binaural Microphone (44.1 kHz)
- Pupil Invisible:
    - Front facing camera (30 Hz)
    - Gaze tracking (~250 Hz)
- BNO055 IMU (50 Hz)
- Atmos Wind Sensor (~2 Hz)
- HP Omnicept:
    - Eye tracking (120 Hz)
    - IMU (50 Hz)
- Neuroelectrics EEG Enobio 32 (500 Hz)
- Empatica E4:
    - GSR (4 Hz)
    - Heartrate (1.56 Hz)
    - Skin temperature (4 Hz)

2.5 Acquisition Data Formats

Due to the heterogeneous nature of the data acquired and collected by the Pluma wearable acquisition unit, different acquisition formats were required for different types of data (see Table 1).

Format	Type of data	File extension	List of streams
CSV	Structured heterogeneous data	.csv	EEG event markers; BNO055; Empatica E4;
NEDF	EEG	.nedf	Neuroelectrics Enobio 32
HARP	Homogeneous binary data	N/A	Bio-signals; Environmental sensors; Wind sensor
BIN	Audio data; UBX data; ZeroMQ data	.bin	Microphone; GNSS; Pupil gaze tracking; HP Omnicept; Unity VR
AVI	Video data	.avi	Pupil front facing camera

Table 1: File formats used to log and store data acquired from devices integrated in the Pluma wearable acquisition unit.

Despite collecting data from several dozen of individual heterogeneous sensors and peripherals, using a small set of standard data formats allowed us to increase the scale of the system dramatically while keeping consistent synchronization and data access guarantees.

Furthermore, by ensuring all data is collected following standard specifications, it is now much easier to guarantee the reproducibility of both experimental data collection and processing for data analysis. The same set of specifications has been designed to accommodate both indoor (virtual reality) and outdoor (augmented reality) experiments so that results are more readily comparable.

Data ingestion pipelines for automatically processing, analysing, and exporting the data collected from experiments with the Pluma wearable acquisition unit into the SDI have been developed on an open-source GitHub repository (github.com/emotional-cities/pluma-analysis). This package has also now been published to the Python Package Index (pypi.org/project/pluma-analysis) for ease of reference and use.

2.6 Data Collection Protocols

In the Technical Report for Experiment #4 published in April 2024, a Methodological Protocol was reported for collecting data with the system, including the sequence of steps required to initialize and configure each device correctly for acquisition.

Installation and execution steps have been documented in the README file of the public open-source repository (github.com/emotional-cities/pluma-experiments) for outdoor experiments, and in the VR-Alfama folder of the same repository, for indoor experiments.

Here we would like to highlight a few challenges observed during the data collection runs which have informed our understanding of the design of such systems.

First, given the very large number of devices and sensors integrated into the wearable unit, and the flexibility and heterogeneity involved in their configuration and calibration, it was paramount to provide real-time online data quality visualizers while running experiments (see Figure 6). Without this, even small logistical failures such as the twisting of a synchronization cable, or the loss of coupling in EEG or ECG electrodes, could compromise the validity of an entire experimental session.

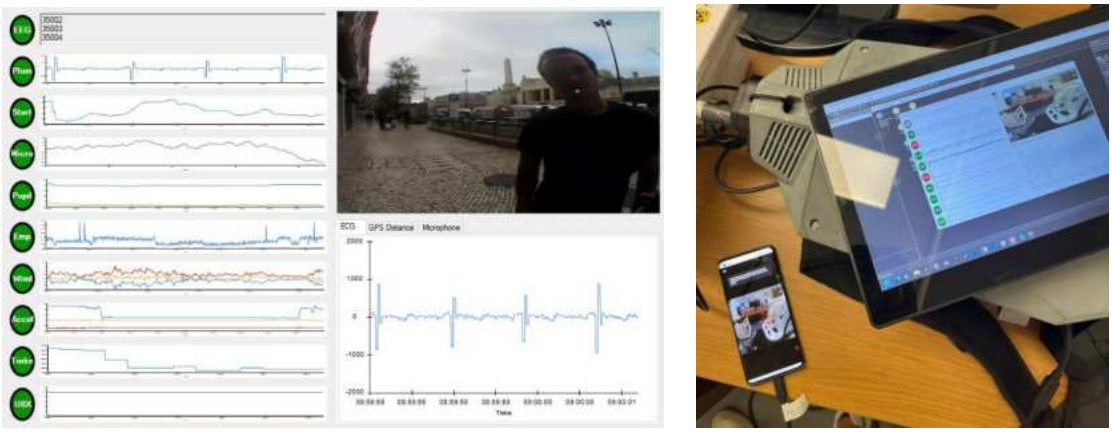

Figure 6. The online experimental dashboard. A summary of the data and status of critical experimental data streams is rendered on a screen placed in the back of the data collection unit or on the desktop computer screen running VR. A green/red semaphore (for each hardware module) indicates if the corresponding data stream is receiving sensor data with valid format. Additionally, raw signal visualization is provided for each of the data streams where the signal data quality can be directly inspected and validated by the experimenter.

Some of the sensors (Empatica E4) have trouble initializing, while others (like the GNSS) take time to converge on a fixed position from which they can start operating. Finally, others require experimenter manipulation to generate a good signal (like the ECG) which depends on cables and sensors being precisely placed in the subject. Given the sheer number of sensors, cables and points of failure it was not always easy to understand if everything was working correctly, and the ability to validate all the signals at the same time using the real-time dashboard was invaluable.

Second, the importance of having an automated data exploration and ingestion pipeline operating in near real-time is crucial to fine tune the experimental design on a location-specific basis (see Figure 7). For example, even if the EEG recording is appearing to proceed smoothly, we found the effect of specific features in the urban environment,

such as exposed electrical cables, can create noise in specific places of a route. Another example is the effect of blocking the GNSS signal due to underground crossings or other obstacles, which can prevent tracking for long periods. Without carefully monitoring and exploring the data soon after it is generated, ideally within a test run, it can be hard to correct and adapt the experimental design to such features.

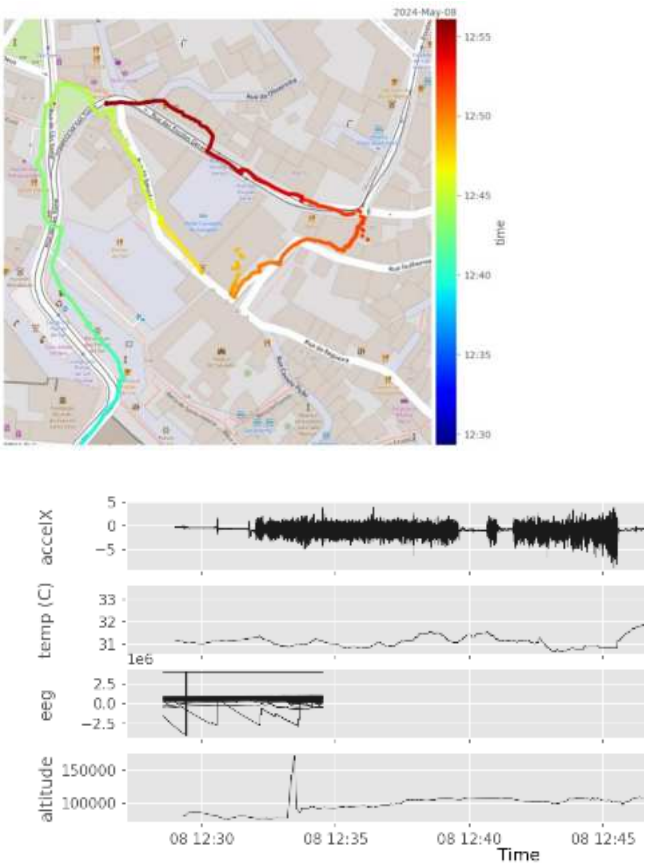

Figure 7. Examples of signal loss. Left panel) Example of GNSS signal loss on turning a street corner. Right panel) Example of EEG signal loss a few minutes into the recordings.

To facilitate rapid data exploration and validation following a data collection run, we devised and published a series of open-source interactive notebooks (github.com/emotional-cities/notebooks) which leverage the standard data infrastructure developed for the project. These notebooks take advantage of the GeoPandas (geopandas.org) package to allow interactive exploration of most Pluma data streams, and also synchronized plotting of data streams time series on a common axis for inspection of correlations and validation of signal quality (see Figure 8).

The notebooks support custom labelling of epochs throughout the recording with the goal of allowing quick identification and validation of problematic hotspots or moments of interest during each session. It also allows development and testing of the unit in new cities, where it can be used by teams operating locally beforehand to assess whether the system needs refinement, and also to test out potential routes for future experiments.

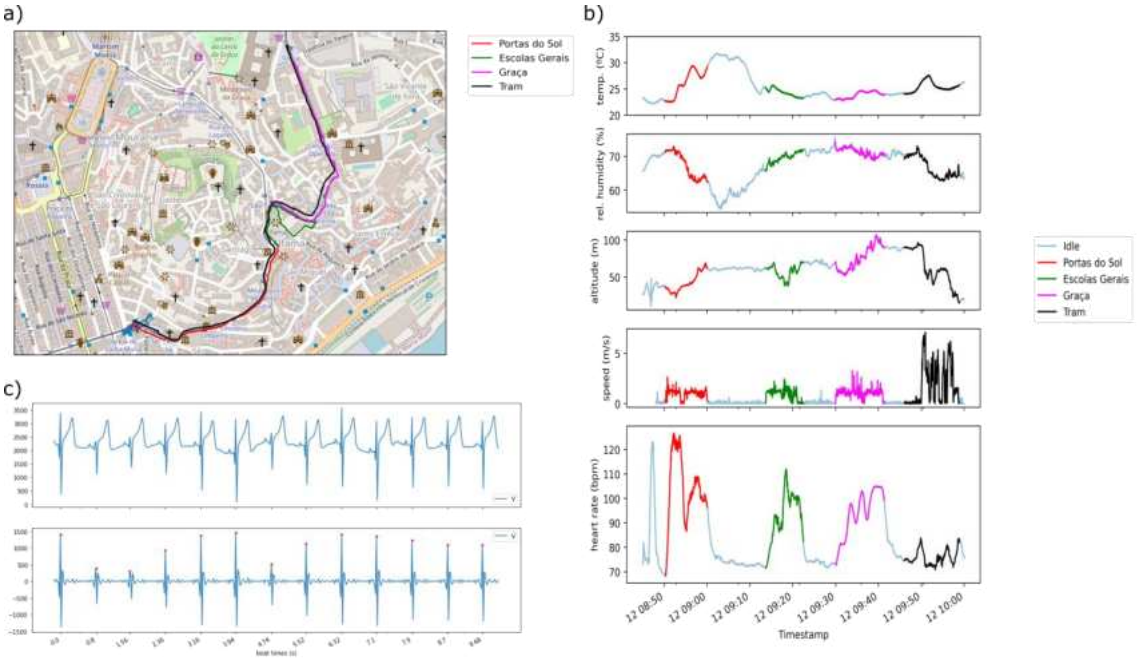

Figure 8. Proof-of-concept walk in Lisbon. GPS, Climate and BioSignal data, colored for each data collection run. a) GNSS positioning data for the entire walk. b) time-series of climate, bio-signals and behaviour data (from top to bottom: temperature, relative humidity, altitude, speed, heart rate). c) heart-rate and beat-detection algorithm for integrated ECG.

Finally, the notebooks also support exporting any data stream into standard data formats, including OGC API Features (for immediate ingestion into the SDI, see Figure 9), or even CSV files for more unorthodox data experimentation. This ability is crucial to enable multi-disciplinary teams to start analysing different aspects of the data as soon as possible. The multi-dimensionality and multi-modality nature of these datasets require multiple experts able to correctly interpret the data, ranging from neurophysiologists to micro-climate experts.

Figure 9. Automated pipeline for synchronizing, georeferencing, and exporting a single or multiple datasets into OGC compliant JSON files to allow direct ingestion to the SDI.

2.7 SDI Data Ingestion

The SDI was designed to ingest, manage, and publish processed geospatial datasets through established OGC standards and legacy services, primarily focusing on refined data outputs ready for analysis.

Given the wide range of tools and geospatial formats we needed to support, we opted for a lightweight and versatile ingestion mechanism. Initially, users upload their data to a centralized data lake. A human triage is then performed to identify any high-level issues with the datasets. After this initial review, custom ingestion pipelines (ByteRoad 2023) are configured for each new dataset type as they emerge, ensuring that the data is gradually refined and integrated into the various components of the SDI.

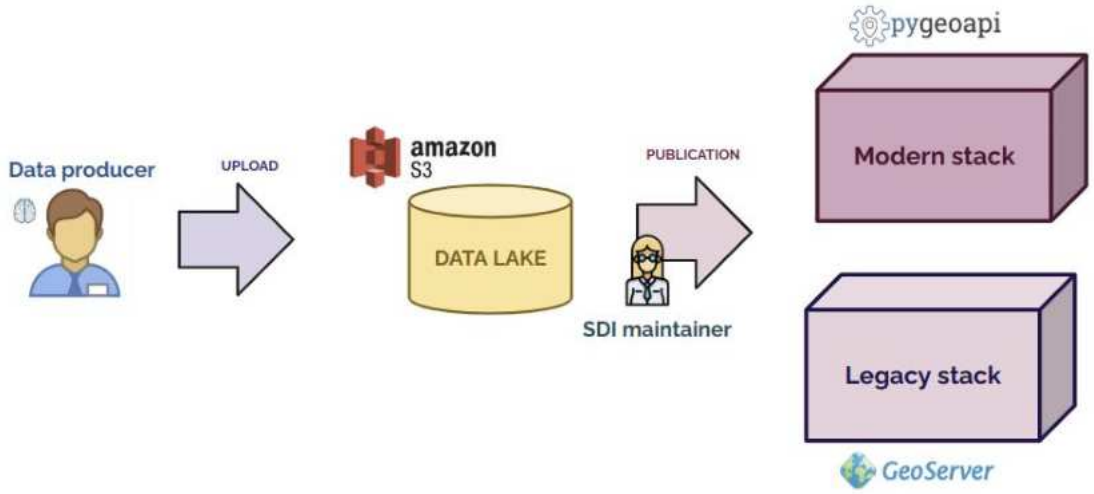

Figure 10 - Data ingestion/publication schema.

However, one key use case was missing: the direct ingestion of raw experimental data. Internal project partners needed a way to share unrefined sensor outputs directly, bypassing the traditional data lake process designed for more refined datasets. With version 2.0.0. of the SDI (Jo and Cerciello 2023), this gap has been addressed by introducing the OGC SensorThings API (implemented through FROST: Fraunhofer Institut IOSB 2024), which provides a dedicated programmatic REST API for data ingestion. Unlike the conventional approach—where users manually upload data to the data lake for human triage and subsequent custom pipeline processing—this new method enables partners to upload data programmatically. Because the raw data must adhere to the single standard defined by Sensor Things API, validation becomes significantly more straightforward and automated. This uniformity not only streamlines the ingestion process but also ensures that the raw experimental data is accurately structured for further refinement and consumption by the other tools.

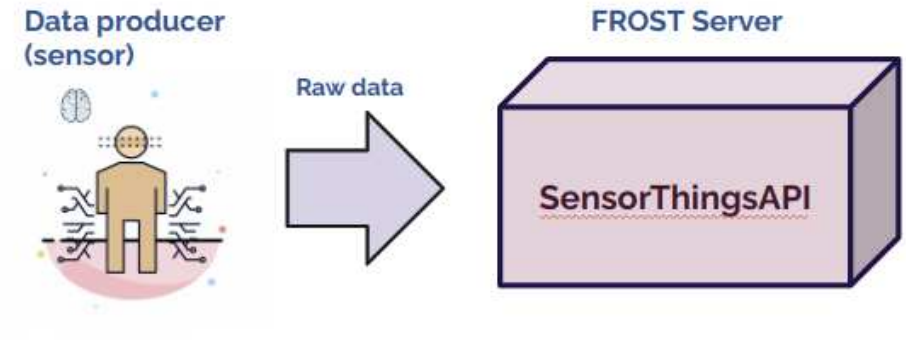

Figure 11 - SensorThingsAPI.

3 Description of the SDI II

Initially, when we developed the SDI as described in the previous deliverables D3.2 and D3.3 (Simões and Cerciello 2022a; Simões and Cerciello 2023), we took a hybrid approach regarding Standards adoption. At that time, concerns about the maturity of the OGC API and the limited availability of fully implemented components led us to adopt a solution that combined both modern (OGC API) and legacy (OGC W*S) stacks (Simões and Cerciello 2024b). Since then, significant progress has been made in the Standards, and many are now fully approved and supported (Simões and Cerciello 2024).

In parallel with the development of the specifications, pygeoapi has also progressed. Over the past two years, it has evolved into a leading implementation by actively participating in OGC Code Sprints and serving as both an Early Implementer and a Reference Implementation. This progress reflects a new paradigm in which standards development and implementation go hand in hand.

Based on high-level requirements collected from our project partners, we have now integrated a comprehensive set of standards. These include OGC API – Features for publishing feature data, OGC API – Tiles for distributing vector tiles, SensorThings API for sensor data, and OGC API – Records for metadata. The SensorThings API was introduced last year and is a significant improvement, as it provides a dedicated programmatic REST API for ingesting raw sensor data. This addition allows internal project partners to directly upload unrefined sensor outputs—bypassing the traditional data lake approach—and ensures that the data is automatically validated against a single, well-defined standard.

Additionally, the updated SDI is built on a free and open source software stack, with pygeoapi at its core. By virtualizing the services in Docker containers and orchestrating

them with docker-compose, we have created an infrastructure-agnostic system that can be rapidly deployed on any cloud provider. The source code is available on GitHub under an MIT license (Cerciello and Simões 2024).

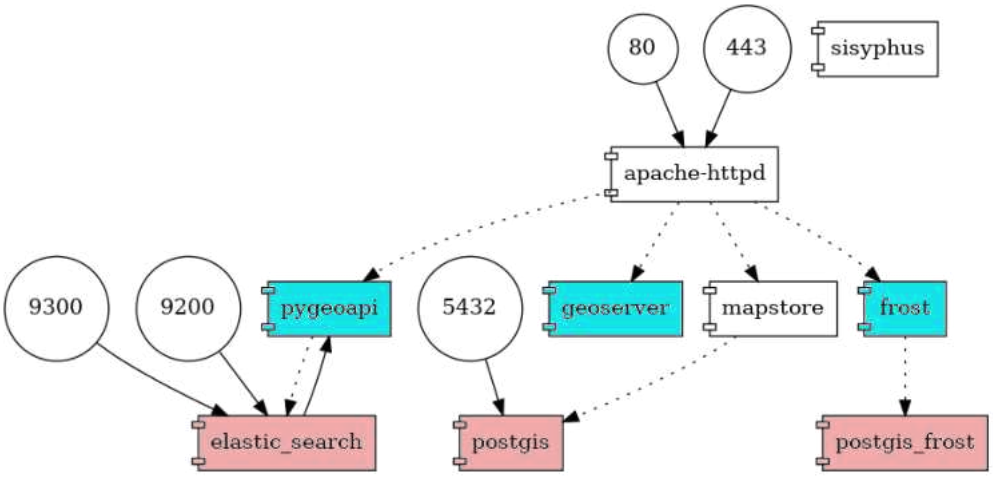

Figure 12 - Architecture of the eMOTIONAL Cities SDI. OGC Standards server implementations are highlighted in blue, while storage backends are highlighted in pink.

We have also set up pipelines to enable both humans and machines to ingest data and metadata into the SDI (see Figure 13) and extensive documentation about how to access the SDI, using clients such as QGIS, MapStore or Jupyter notebooks (Simões and Cerciello 2023).

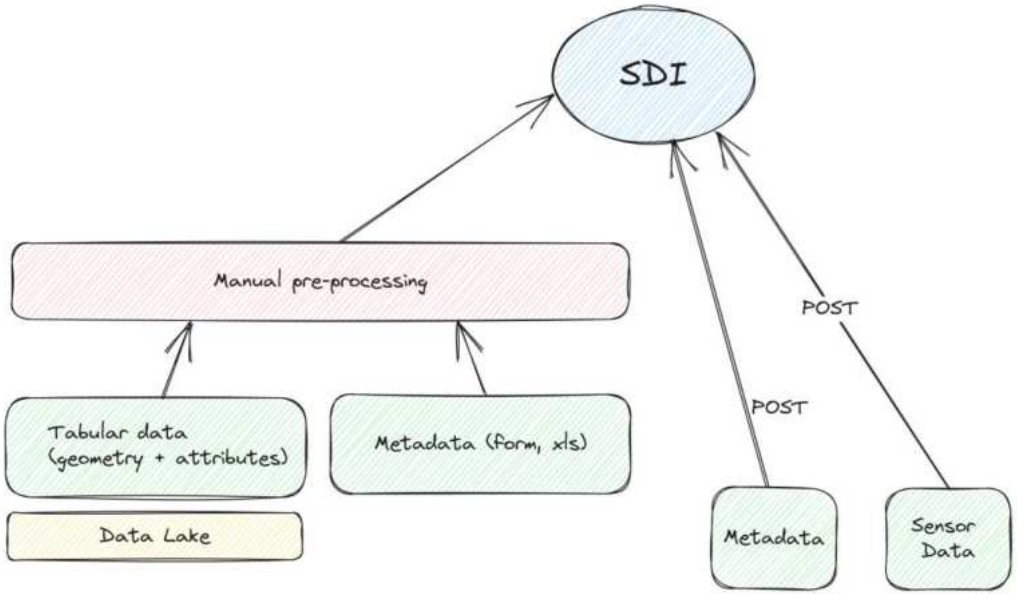

Figure 13 - Pipeline for ingesting data and metadata into the eMOTIONAL Cities SDI.

The SDI is live at http://emotional.byteroad.net/ and includes over two hundred collections from five different cities: Lisbon, London, Copenhagen, Tartu, and Lansing. It includes collections characterizing the physical environment (e.g., Normalized Difference Vegetation Index (NDVI), annual mean NO_2 concentration), the built environment (e.g., buildings with repair needs ratio, average age of buildings), socio-economic aspects (e.g., Area Deprivation Index, number of people who travel by bicycle to work), health data (e.g., crude percentage of adults with depression, mortality rate, prevalence rates of mental health issues in London) (see Figure 14), and results of experiments (e.g., London outdoor walk test data: air quality temperature, sound pressure levels). The data can be discovered and queried in the OGC API - Records searchable catalogue: https://emotional.byteroad.net/catalogue

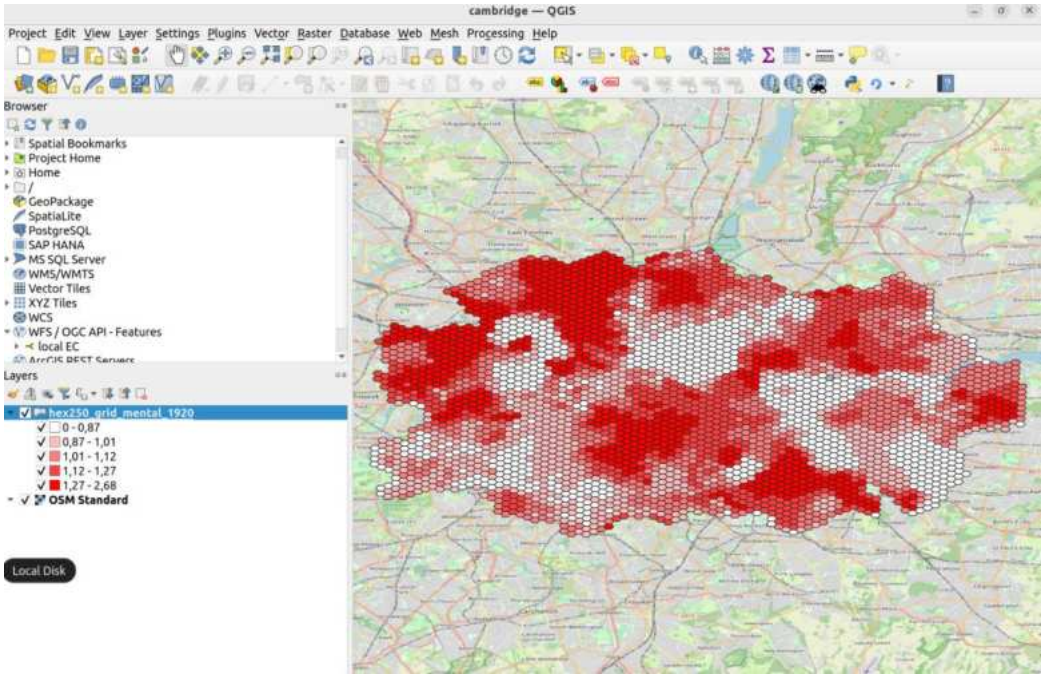

Figure 14 - "Prevalence rates of mental health issues in London" dataset, accessed on QGIS using OGC API - Features.

The SDI has had a couple of iterations before reaching its current format (Cerciello and Simões 2024). First and foremost, not all the OGC API parts we selected were published and implemented initially, which led us to introduce another component, publishing the same data using the first generation of OGC Standards (e.g., WFS, WMS, WMTS) (Cerciello 2022). To a certain extent, that component could be removed now, although certain client applications may not have implemented OGC APIs yet and some users may be more familiar with the previous generation of OGC Standards. In the second release (Jo and Cerciello 2023) we added an implementation of SensorThings API (FROST), which addressed some needs of neuroscientists, in terms of storing and publishing information about sensors and measurements (See Figure 15).

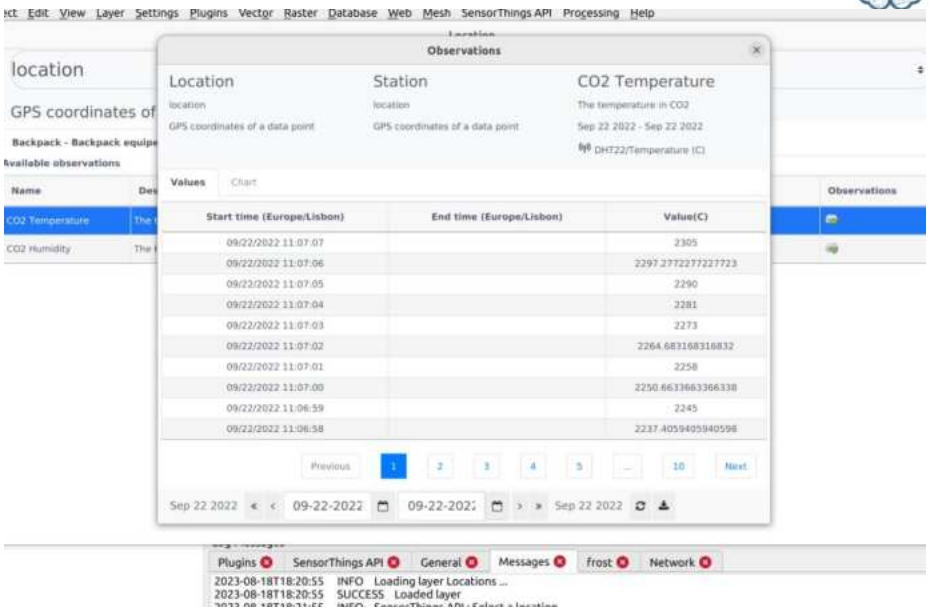

Figure 15 - Details of a STA collection, being accessed in QGIS.

SensorThings API data is proxied through pygeoapi and transformed into a feature collection that GIS scientists can consume without the need for specific STA tools. Figure 15 summarizes the different stakeholders of the SDI, whose needs we tried to address using a variety of OGC Standards.

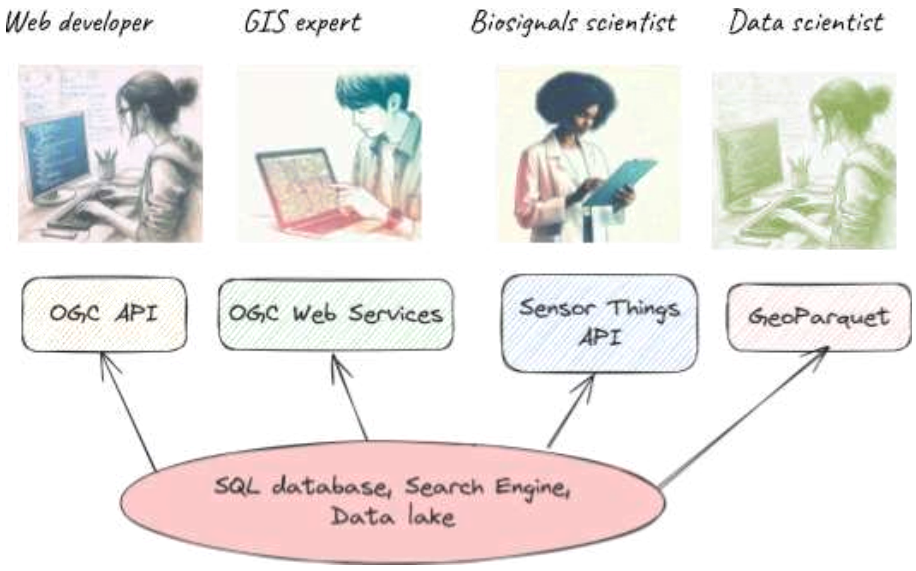

Figure 16 - Main stakeholders of the eMOTIONAL Cites SDI, mapped to OGC Standards.

One of the improvements of this SDI was the reuse of a single backend technology, Elasticsearch, to support OGC API - Features, OGC API - Tiles and OGC API - Records. Elasticsearch (Elastic 2024) is a distributed, RESTful search and analytics engine,

scalable data store, and vector database with geospatial capabilities. The “mvt-elastic” pygeoapi plugin leverages the Elasticsearch capability of generating vector tiles on the fly (Ignacio and Neirynck 2022) to publish vector data as OGC API - Tiles (see Figure 6). This plugin was added during the course of this project (Simões 2022), enabling the eMOTIONAL Cities SDI to publish both OGC API - Features and OGC API - Tiles from a single ElasticSearch index.

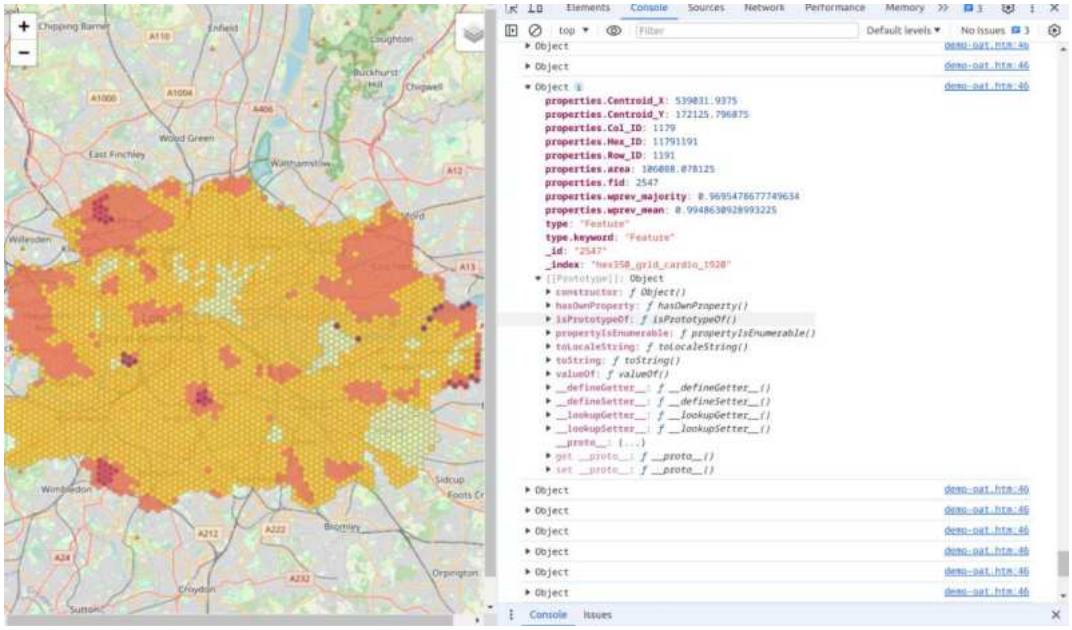

Figure 17 - Vector tiles are very suitable for creating web maps, as they combine fast data delivery along with design flexibility. This screenshot shows a LeafLet-based application that consumes one of the eMOTIONAL Cities collections.

It should also be mentioned that ElasticSearch was the only component of the SDI stack that did not have an Open Source license. As of August 2024, Elastic has added an AGPL license, which is an Open Source Initiative (OSI) approved license (Banon 2024).

Considering the SDI was going to be used by scientists for the purpose of analysis, we also wanted to experiment with an emerging OGC Candidate Standard in the field of cloud-native geospatial. GeoParquet (GeoParquet 2024) is an open-source column-oriented data storage format, which provides efficient data compression and encoding schemes with enhanced performance to handle complex data in bulk. Although GeoParquet started as a community effort, it is now on the path to become a full-fledged OGC Standard (OGC 2022) and it already features a vibrant ecosystem of implementations. We have converted more than two hundred eMOTIONAL Cities datasets to geoparquet using a semi-manual pipeline, which is documented in a blog post (doublebyte 2024). These files were then validated using a tool called gpq (Planet Labs 2024). The validation includes scanning the data to ensure that values in geometry columns conform with the specification (making assertions about the encoding, ring orientation, bounding box, and alignment with other metadata).

The GeoParquet files are advertised through a link on the metadata records, which uses an item type “application/vnd.apache.parquet”. It is worth noting that GeoParquet files are considerably smaller than GeoJSON (our default format for OGC API) and even than binary formats such as GeoPackage or GeoJSON (See Figure 18).

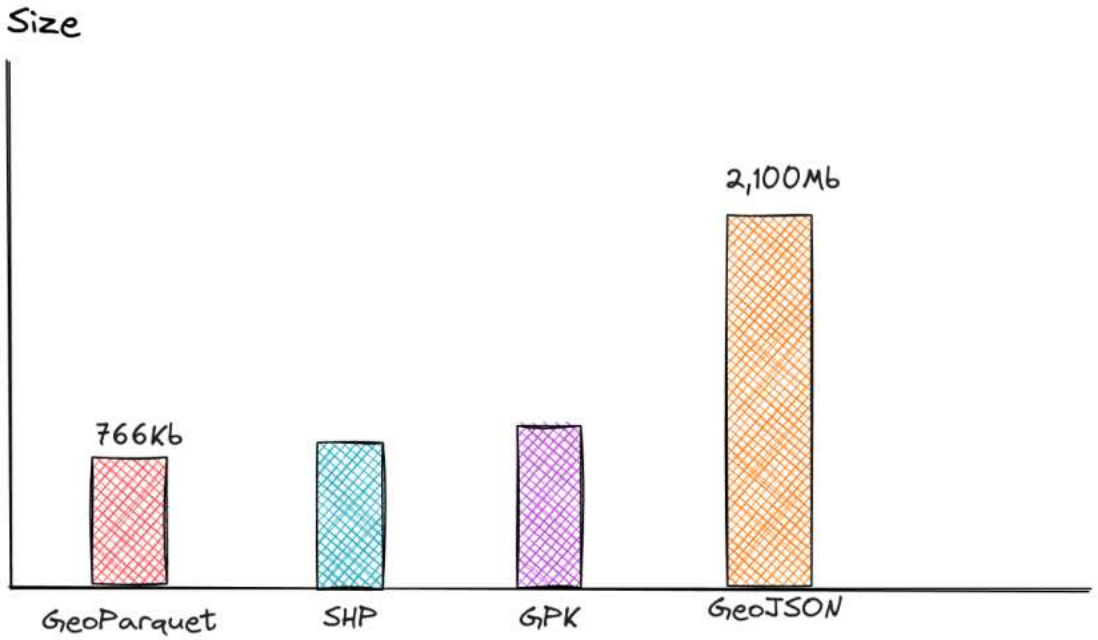

Figure 18 - Size comparison between different formats to download eMOTIONAL Cities datasets, on an average-size dataset

Replacing the existing formats by GeoParquet would result not only in storage savings (and consequently cost savings), but also in bandwidth savings. This is particularly relevant for the use cases where analysis combines multiple datasets.

4 Conclusions

The data acquisition process highlighted the importance of modular and robust design in experimental settings. Our approach demonstrated that integrating different sensor technologies with precise synchronization methods can significantly improve the quality and reproducibility of collected data. Key guidelines for future projects include prioritizing real-time monitoring and automated data pipelines to quickly detect and resolve issues, as well as ensuring that hardware and software components can be independently operated and replaced. Such practices not only optimize data integrity, but also pave the way for adaptable and large-scale experimental deployments in diverse environments.

In parallel, the experimental nature of the project allowed us to explore new approaches to designing and implementing spatial data infrastructures (SDIs). Our experience demonstrates the promising potential of OGC APIs to provide a flexible and scalable foundation for modern SDIs. The adoption of these APIs has proven versatile in addressing a wide range of GIS use cases and emerging fields such as neuroscience, as demonstrated by the successful integration of the SensorThings API. Ongoing efforts by OGC working groups continue to improve the maturity and robustness of these standards. Additionally, leveraging FOSS solutions has allowed us to build much of the infrastructure on a solid open foundation, providing a replicable starting point for future projects.

References

Cisek, P., & Green, A. (2024). Toward a neuroscience of natural behavior. Current Opinion in Neurobiology, 86, 102859.

Giangrande, A., Botter, A., Piitulainen, H., & Cerone, G. L. (2024). Motion Artifacts in Dynamic EEG Recordings: Experimental Observations, Electrical Modelling, and Design Considerations. Sensors, 24(19), 6363.

Krakauer, J. W., Ghazanfar, A., Gomez-Marin, A., MacIver, M., & Poeppel, D. (2017). Neuroscience Needs Behavior: Correcting a Reductionist Bias. Neuron, 93(3), 480-490.

Lopes, G., & Monteiro, P. (2021). New Open-Source Tools: Using Bonsai for Behavioral Tracking and Closed-Loop Experiments. Frontiers in Behavioral Neuroscience, 15.

Simões, Joana, and Antonio Cerciello. 2022a. *D 3.2 - Architecture Definition and Code for the Generic SDI.* Zenodo. https://doi.org/10.5281/zenodo.13365357.

Simões, Joana, and Antonio Cerciello. 2023. *D 3.3 - Description of the SDIs I.* Zenodo. https://doi.org/10.5281/zenodo.13365412.

Mobbs, D., Wise, T., Suthana, N., Guzmán, N., Kriegeskorte, N., & Leibo, J. Z. (2021). Promises and challenges of human computational ethology. Neuron, 109(14), 2224-2238.

Pfeiffer, B. E., & Foster, D. (2013). Hippocampal place-cell sequences depict future paths to remembered goals. Nature, 497, 74-79.

Stangl, M., Maoz, S. L., & Suthana, N. (2023). Mobile cognition: imaging the human brain in the 'real world'. Nature Reviews Neuroscience, 24, 347–362.

ByteRoad. 2023. "ETL Tools." GitHub. https://github.com/emotional-cities/etl-tools.

Jo, and Antonio Cerciello. 2023. *emotional-cities/openapi-sdi: Push Capabilities: FROST and Sisyphus (v2.0.0).* Zenodo. https://doi.org/10.5281/zenodo.8330877.

Fraunhofer Institut IOSB. 2024. *FROST Server.* https://github.com/FraunhoferIOSB/FROST-Server.

Simões, Joana, and Antonio Cerciello. 2022b. "Serving Geospatial Data Using Modern and Legacy Standards: A Case Study from the Urban Health Domain." *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLVIII-4/W1-2022: 419–425. https://doi.org/10.5194/isprs-archives-XLVIII-4-W1-2022-419-2022.

Simões, Joana, and Antonio Cerciello. 2024. "A Spatial Data Infrastructure Using Modern Standards: Lessons Learned from the eMOTIONAL Cities Project." *Free and Open Source Software for Geospatial 2024 (FOSS4G 2024), Belém, PA, Brazil.*, December 2, 2024. https://doi.org/10.5281/zenodo.14223773.

Cerciello, Antonio, and Joana Simões. 2024. *openapi-sdi (v3.0.0).* Zenodo. https://doi.org/10.5281/zenodo.13758189.

Cerciello, Antonio. 2022. *emotional-cities/openapi-sdi: Initial Release (v1.0.0).* Zenodo. https://doi.org/10.5281/zenodo.6591180.

Elastic. 2024. "Elasticsearch: The Official Distributed Search & Analytics Engine." *Elastic*, September 13, 2024. https://www.elastic.co/elasticsearch.

Ignacio, Vera Sequeiros, and Thomas Neirynck. 2022. "Introducing the Elasticsearch Vector Tile Search API." *Elastic Blog*. Elastic, January 24, 2022. https://www.elastic.co/blog/introducing-elasticsearch-vector-tile-search-api-for-geospatial.

Simões, J. 2022. " Added support for elasticsearch tiles in the mvt provider'. https://github.com/geopython/pygeoapi/pull/986

Banon, S. 2024. "Elasticsearch Is Open Source, Again." *Elastic Blog*. Elastic, August 29, 2024. https://www.elastic.co/blog/elasticsearch-is-open-source-again.

GeoParquet. 2024. *GeoParquet: Geospatial Data in Parquet.* https://geoparquet.org/.

OGC. 2022. "GeoParquet." *GitHub.* https://github.com/opengeospatial/geoparquet.

doublebyte. 2024. "Navigating GeoParquet: Lessons Learned from the eMOTIONAL Cities Project." *Heartcode.* Heartcode, August 2, 2024. https://doublebyteblog.wordpress.com/2024/08/02/navigating-geoparquet-lessons-learned-from-the-emotional-cities-project/.

Planet Labs. 2024. gpq. GitHub. https://github.com/planetlabs/gpq