OPEnS Hub: Real-time Data Logging, Connecting Field Sensors to Google Sheets

Thomas DeBell[1]

[1]First Author

November 22, 2022

Abstract

In earth science, we must often move and store tremendous amounts of data from remote locations. Present options are typically limited to costly proprietary devices which are rigid in structure and require numerous ongoing expenses. The Openly Published Environmental Sensing (OPEnS) Lab at Oregon State University developed the OPEnS Hub, a new approach using low-power, open-source hardware and software to achieve near real-time data logging from the field to the web. The Hub is two orders of magnitude less expensive than commercial products, inherently modular and flexible, and aims to reduce technical barriers for users with little programming experience. Data can be collected remotely from nearly anywhere on Earth using a host of transmission protocols to relay data from distributed in-situ monitoring devices. Telemetry options include 900 MHz Long Range Radio (LoRa) with up to 25 km range and Nordic Radio Frequency (nRF) for higher data rates. Internet gateways include the established cell network infrastructure, Wi-Fi for high bandwidth applications, and Ethernet where available. The OPEnS Hub is capable of mesh networking with several nodes and backs up to an onboard microSD card. The Hub engages a dynamic, low-latency portal to Google Sheets via the free Application Programming Interface (API), PushingBox, and an adaptable Google Apps Script. This framework was tested on 12 unique sensor suites at remote sites in Oregon. This manuscript details our methods and evaluates PushingBox, Google Apps Script, Adafruit Industries' open-hardware Feather development boards, the Hypertext Transfer Protocol (HTTP), and the foregoing modes of data transfer.

OPEnS Hub: Real-time Data Logging, Connecting Field Sensors to Google Sheets

Thomas C. DeBell[1, 2, 3*], Luke D. Goertzen[1, 2, 4], Lars T. Larson[1, 2, 3], William J. Selbie[2], John S. Selker[2, 1, 3], Chet Udell[2, 1, 3]

[1]Oregon State University, United States, [2]Openly Published Environmental Sensing (OPEnS) Lab, United States, [3]Department of Ecological and Biological Engineering, Oregon State University, United States, [4]Department of Electrical Engineering and Computer Science, Oregon State University, United States

Submitted to Journal:
Frontiers in Earth Science

Specialty Section:
Hydrosphere

Article type:
Brief Research Report Article

Manuscript ID:
434616

Received on:
01 Nov 2018

Frontiers website link:
www.frontiersin.org

Conflict of interest statement

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest

Author contribution statement

TD designed the framework for the project, wrote the majority of the manuscript and constructed the physical device, LG and WS contributed to the software library development, and acted as a reference for all software considerations, LL served as chief editor of the paper and provided imperative guidance in the construction of the article, CU was the primary mentor on the project, JS served as the head principal investigator on the project.

Keywords

open-source, In-situ, arduino, LORA communication, google-sheets, Data-logging, IoT, low-cost

Abstract

Word count: 250

In earth science, we must often move and store tremendous amounts of data from remote locations. Present options are typically limited to costly proprietary devices which are rigid in structure and require numerous ongoing expenses. The Openly Published Environmental Sensing (OPEnS) Lab at Oregon State University developed the OPEnS Hub, a new approach using low-power, open-source hardware and software to achieve near real-time data logging from the field to the web. The Hub is two orders of magnitude less expensive than commercial products, inherently modular and flexible, and aims to reduce technical barriers for users with little programming experience. Data can be collected remotely from nearly anywhere on Earth using a host of transmission protocols to relay data from distributed in-situ monitoring devices. Telemetry options include 900 MHz Long Range Radio (LoRa) with up to 25 km range and Nordic Radio Frequency (nRF) for higher data rates. Internet gateways include the established cell network infrastructure, Wi-Fi for high bandwidth applications, and Ethernet where available. The OPEnS Hub is capable of mesh networking with several nodes and backs up to an onboard microSD card. The Hub engages a dynamic, low-latency portal to Google Sheets via the free Application Programming Interface (API), PushingBox, and an adaptable Google Apps Script. This framework was tested on 12 unique sensor suites at remote sites in Oregon. This manuscript details our methods and evaluates PushingBox, Google Apps Script, Adafruit Industries' open-hardware Feather development boards, the Hypertext Transfer Protocol (HTTP), and the foregoing modes of data transfer.

Funding statement

This work is supported in part by the USDA National Institute of Food and Agriculture, Hatch project NI18HFPXXXXXG055, and the Agricultural Science Foundation at Oregon State University. All work was done using the resources of the Openly Published Environmental Sensing Lab at Oregon State University. Student funding for the project was thanks to in part the College of Agricultural Science through the beginning and continuing research support program.

Ethics statements

(Authors are required to state the ethical considerations of their study in the manuscript, including for cases where the study was exempt from ethical approval procedures)

Does the study presented in the manuscript involve human or animal subjects: No

Data availability statement

Generated Statement: The datasets generated for this study are available on request to the corresponding author.

OPEnS Hub: Real-time Data Logging, Connecting Field Sensors to Google Sheets

Thomas DeBell[1*], Luke Goertzen[2], Lars Larson[1], William Selbie[2], John Selker[1], Chet Udell[1]

[1]Openly Published Environmental Sensing Lab, Oregon State University, Biological and Ecological Engineering, Corvallis, Oregon, United States.

[2]Openly Published Environmental Sensing Lab, Oregon State University, Computer Science, Corvallis, Oregon, United States.

*** Correspondence:**
Thomas DeBell
debellt@oregonstate.edu

Keywords: open-source$_1$, in-situ-sensing$_2$, arduino$_3$, lora$_4$, google-sheet$_5$, data-logging$_6$, IOT$_7$, low-cost$_8$

Word Count:1917

Number of Figures and Tables: 2

Abstract

In earth science, we must often move and store tremendous amounts of data from remote locations. Present options are typically limited to costly proprietary devices which are rigid in structure and require numerous ongoing expenses. The Openly Published Environmental Sensing (OPEnS) Lab at Oregon State University developed the OPEnS Hub, a new approach using low-power, open-source hardware and software to achieve near real-time data logging from the field to the web. The Hub is two orders of magnitude less expensive than commercial products, inherently modular and flexible, and aims to reduce technical barriers for users with little programming experience. Data can be collected remotely from nearly anywhere on Earth using a host of transmission protocols to relay data from distributed in-situ monitoring devices. Telemetry options include 900 MHz Long Range Radio (LoRa) with up to 25 km range and Nordic Radio Frequency (nRF) for higher data rates. Internet gateways include the established cell network infrastructure, Wi-Fi for high bandwidth applications, and Ethernet where available. The OPEnS Hub is capable of mesh networking with several nodes and backs up to an onboard microSD card. The Hub engages a dynamic, low-latency portal to Google Sheets via the free Application Programming Interface (API), *PushingBox*, and an adaptable Google Apps Script. This framework was tested on 12 unique sensor suites at remote sites in Oregon. This manuscript details our methods and evaluates *PushingBox*, Google Apps Script, Adafruit Industries' open-hardware Feather development boards, the Hypertext Transfer Protocol (HTTP), and the foregoing modes of data transfer.

1 Introduction

Advancements in sensing technology have sparked a new age of data acquisition and transmission that continues to change the way we understand the world around us. However, proprietary data loggers, such as the ubiquitous Campbell Scientific suite of loggers, range from $1,650 for their flagship CR1000 to $3,163 for their more versatile CR3000 (Campbell Scientific, 2018b). Both have seen few fundamental technological developments since the mid-1990’s and require learning a complex programming interface. These systems often store data onboard, demanding intermittent retrieval from the field or paying up to $2,495 for remote access with satellite telemetry (Campbell Scientific, 2018a).

The solution to the problem of logging data from remote locations leverages the “Internet of Things” (IoT) movement: everything can be connected to the internet. Specifically, the OPEnS Lab has established an “Internet of Agriculture” initiative using open-source IoT-enabled devices to collect scientific data on environmental conditions. A significant challenge to the Internet of Agriculture is that systems are deployed in remote areas where Wi-Fi is not accessible. The goal is to create an inherently modular, cost-effective alternative for collecting field data by accommodating a variety of long-range wireless telemetry options and providing open-source documentation at a technical level such that a farmer, scientist or hobbyist would be able to replicate and generate valuable data sets. The OPEnS Hub stands to simultaneously lower the cost of experimentation and data collection and break down traditional technical barriers.

2 Materials and Methods

2.1 Hardware

The physical components of the Hub rely on an open-hardware suite of development microcontroller boards, or multipoint control units (MCUs), produced by Adafruit Industries. We chose the Adafruit Feather line of microcontrollers for their low power requirements, high data capacity, and versatility. Variants of the Feather include onboard 900 MHz Long Range Radio (LoRa), Wi-Fi, and Ethernet, with extension options including the Global System for Mobile Communication (GSM), nRF, and Bluetooth. Feathers are programmed using C++ (ISO 2013) in the Arduino platform (Arduino Software, *Arduino IDE*). The boards selected for field implementation were the Ethernet FeatherWing to act as the internet gateway between the Hub and the web, the real-time clock FeatherWing to make accurate timestamps of transmissions, and the LoRa variant of the central MCU, which accesses a non-licensed 900 MHz radio band to transmit data from the sensors to the logger. A 3-ft-long, 8-dBI, 50-Ohm impedance, omnidirectional radio antenna was used to improve transmission strength. Custom waterproof enclosures for the hub and devices were designed in Autodesk’s Fusion360 (Autodesk, Inc. *Fusion 360*) and 3D printed.

One of the major advantages of using these microcontroller nodes is their ability to use low power protocols which put the processor to sleep using the DS3231 real-time clock when not taking readings. These battery-optimizing features allow for node devices to be powered by small, 2,000 mAh lithium polymer batteries with battery life ranging from weeks to months on a single charge.

2.2 Software

It was essential to employ a cloud service to process, store, and maximize accessibility of the collected data. Google’s App Script was chosen because it can be easily modified in a language similar to JavaScript. This application also makes the data available in a simple, familiar environment and displays near real-time updates.

The process of getting field data to a Google spreadsheet requires several steps, mainly due to the computational limitations of small, low-power microprocessors. The data must first be packaged into a format that can be sent and parsed, the device must connect to the internet, and finally, an HTTP request containing the data triggers the PushingBox API. The next steps no longer involve the microprocessor; the API can extract and forward the data to a Google Script, which will identify the correct spreadsheet and tab and finally write the data into the corresponding columns.

Much of the complexity of this routine stems from the limited processing capacity of Arduino-like devices for supporting the Secure Sockets Layer (SSL) or Transport Layer Security (TSL) encryption protocol required for HTTPS. This barrier is nontrivial because Google Scripts / Apps can only be accessed via secure connections. As such, the device needs to offload the direct communication with the script to another platform such as the PushingBox API. While PushingBox can trigger a variety of services upon receiving an HTTP request, the OPEnS Hub sends the data to the script URL which effectively converts the original HTTP request from the hub to an HTTPS request to reach the Google script.

The transmission code was modified to send the spreadsheet ID and tab name alongside the data so that the App Script can create any number of Google Sheets from a single Hub. To achieve this, each node sends data in key-value pairs. For every data point sent, the Hub specifies the origin of the data (i.e. the column in the spreadsheet) to be properly organized, coupled with the data value itself. As a result, each data point requires two HTTP *GET* requests. Although sending these key-value pairs adds to the total packet size reducing the amount of sensor data that can be sent, this protocol enables dynamic addition or removal of sensors without needing to change the App Script.

When the Google Script receives a *GET* request, it creates a JavaScript dictionary, relating the keys to the values. Next, it accesses the specified spreadsheet and tab and checks the most recent column headers. The data is then sorted into the correct columns, or a new header is created if the data keys have changed since the last upload. A full visual representation of this process is in Figure 1.

2.3 Lab testing

Since there is nearly an infinite number of devices and sensor combinations, it was necessary to be able to test each device individually and in concert over the internet gateway to know that the data was properly transcribed to the spreadsheet. First, testing was done to confirm that the sensors were accurately measuring the environment and consistently transmitting the correct data at appropriate intervals to the Hub. This also tested the system's scalability: by proving that multiple devices could transmit to the hub simultaneously without losing or corrupting data. The use of a free API presented one of the major constraints of the project because each account is limited to 1,000 HTTP requests a day. For initial testing the sampling frequency was 5 minutes, or 288 readings per day and later scaled the project to support any number of devices as long as the sampling frequency did not exceed 1,000 requests per day. Lab testing included simulating field conditions to ensure the system was resilient to the environment and robust enough to be updated as the number of connected devices changed.

2.4 **Field testing**

Although there are a variety of telemetry options supported by the OPEnS Hub such as GSM, LoRa, and NRF, LoRa radio proved to be the most applicable for field testing at long ranges. Field testing consisted of three deployments among two different sites. The first experiments were conducted at the H.J. Andrews Experimental Forest near Blue River, Oregon in July 2017 and July 2018, and the other was at Lewis Brown Farms near Corvallis, Oregon in April 2018.

The first field experiment consisted of a Hub equipped with LoRa radio with a wired Ethernet connection and one LoRa-enabled weather station located approximately a quarter mile away through densely wooded forest. The following test at Lewis Brown Farms consisted of a variety of sensor types all equipped with LoRa radio which transmitting at intervals of 10 minutes for two weather stations and 15 minutes for three soil moisture sensors. This data was broadcast at a maximum distance of 0.28 miles to the Hub which was connected to the internet via an Ethernet connection. The final field deployment was conducted, again at the H.J. Andrews Experimental Forest, with five weather stations transmitting a variety of environmental conditions at varying distances from the Hub. The longest transmission reached 0.36 miles.

3 Results

Considering the duration, the number of data points collected, and spatial distribution of data, the cost of the sensor deployment is between one hundredth and one-thousandth the cost of comparable proprietary dataloggers over the life of the project. The results are summarized in Figure 2. The first deployment (represented by the purple pin in Figure 2) yielded almost two months of reliable data transmission approximately a quarter mile through dense forest. Weather data was reported at 5-minute intervals to Google Sheets with less than 10 seconds of latency. The second deployment yielded promising results with expanded capability to receive data transmissions from multiple node devices and a variety of sensor data specific to that node for nearly four months. The App Script proved sufficiently dynamic to generate separate tabs for each device and correctly place their respective dataset into the correct columns, producing a spreadsheet populated with over 300,000 data points. The third and final deployment of this study resulted in receiving weather station data from 5 devices dispersed across the H. J. Andrews Experimental Forest with transmission distances up to 0.36 miles. This third experiment was cut short due to battery issues at the nodes.

4 Discussion

When working with open-source components, the quality control processes are likely not as intensive as a widely-used propriety system. For example, in the testing phase, it was difficult to distinguish a software issue from a hardware issue. This uncertainty resulted in protracted diagnostic testing to isolate the problem, which revealed a defect on the MCU. Another challenge was that the data transmission and the spreadsheet were inherently coupled, which resulted in an end product that lacked flexibility. The spreadsheet assumed the incoming data's order and placed it accordingly, which meant that if the nodes ever changed the data transmitted, or the way the hub started processing the data, then the spreadsheet would organize data incorrectly. This problem was resolved by altering the functionality of the nodes to send key-value pairs so that the data could be order-agnostic. This strategy resulted in a spreadsheet that accurately displays data in the correct columns, regardless of the order of data received, making the system truly dynamic in the event of dropped radio data packets. However, the transmissions were restricted to only 13 different sensor variables as a result.

The Hub's different transmission types, LoRa, GSM, Wi-Fi, and Ethernet, allow for customization depending on the application of use, ranging from transmission of several miles at low bandwidths (LoRa and GSM) or shorter distance transmissions at much higher bandwidths (Wi-Fi, Ethernet). It is also notable that LoRa technology is still developing and has been expanded to transmit to an ever-growing constellation of satellites, making this technology truly global in its applicability (Telkamp, 2018).

5 Conclusion

The OPEnS Hub data upload gateway currently uses Wi-Fi, Ethernet, and the 2G GSM cellular network in a modular fashion which means that modes of transmission are easily interchangeable. The OPEnS Lab is currently investigating additional means of uploading sensor data into a cloud platform with 4G cellular as one of the primary connection methods targeted for expansion. The speed and coverage of 4G in contrast to the limited support for 2G makes 4G an enticing addition for remote logging away from any Ethernet or Wi-Fi networks. Findings of this project will result in the implementation of this technology for use in a collaborative workshop hosted by Kwame Nkrumah University of Science and Technology in Ghana in late November of this year. Although the focus of this article shows results for LoRa testing, the OPEnS Hub's modular adaptability allows the user to use the telemetry solution that is optimal for their unique deployment context.

6 Conflict of Interest Statement

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

7 Author Contributions

TD designed the framework for the project, wrote the majority of the manuscript and constructed the physical device, LG and WS contributed to the software library development, and acted as a reference for all software considerations, LL served as chief editor of the paper and provided imperative guidance in the construction of the article, CU was the primary mentor on the project, JS served as the head principal investigator on the project.

8 Funding

This work is supported in part by the USDA National Institute of Food and Agriculture, Hatch project NI18HFPXXXXXG055, and the Agricultural Science Foundation at Oregon State University. All work was done using the resources of the Openly Published Environmental Sensing Lab at Oregon State University. Student funding for the project was thanks to in part the College of Agricultural Science through the beginning and continuing research support program.

9 Acknowledgments

We are especially indebted to all those who have taken time to provide feedback on this project including Gordon Godshalk, Katherine Darr and Carolyn Gombert. Additionally, we would like to thank the members of the OPEnS lab, especially Cara Walter, who has provided aid to this project since the very beginning. The ability to approach technical writing critically and iteratively is largely a result of the mentorship provided by Dr. Chad Higgins. Lastly, we would like to thank all the

faculty, staff and researchers of the Biological and Ecological Engineering department at Oregon State University, for creating a professional and truly enjoyable place to work.

10 References

Autodesk (2018). A360 | Project Collaboration in the Cloud. Available at: https://a360.autodesk.com/

Adafruit Industries. DS3231 Precision RTC FeatherWing - RTC Add-on for Feather Boards. Available at: https://www.adafruit.com/product/3028

Adafruit Industries (2018). Ethernet FeatherWing. Adafruit Industries, Unique & fun DIY electronics and kits. Available at: https://www.adafruit.com/product/3201

Adafruit Industries (2018). Adafruit LoRa Radio FeatherWing - RFM95W 900 MHz. Available at: https://www.adafruit.com/product/3231

Arduino Software *Arduino IDE* (Version 1.8.7). Windows. Arduino, 2018. Available at: https://www.arduino.cc/en/Main/Software

Autodesk Inc *Fusion 360*. Cloud Powered 3D CAD/CAM Software for Product Design. Autodesk, Inc. 2018. https://www.autodesk.com/products/fusion-360/overview

Campbell Scientific (2018a). Communications - Wireless, remote, hard-wired, direct, or two-way communication. Available at: http://www.campbellsci.com/communications

Campbell Scientific (2018b). CR1000 - Measurement and Control Datalogger. Available at: https://www.campbellsci.com/cr1000

DeBell, T. (2017). Design and Deployment of a General Purpose, Open Source LoRa to Wi-Fi Hub and Data Logger. *2017 AGU Fall Meeting Poster*. Available at: https://agu.confex.com/agu/fm17/meetingapp.cgi/Paper/221030

International Organization for Standardization C Working Group (2013). ISO/IEC 9899 Programming Languages – C. Available at: http://www.open-std.org/jtc1/sc22/wg14/www/standards

LoRa HUB (2018). *OPEnS Lab*. Available at: http://www.open-sensing.org/lora-hub/

PushingBox (2018). Notifications for your Internet of Things devices. Available at: https://www.pushingbox.com/

Telkamp, T. “LoRa transmission from low orbit satellite.” *The Things Network*. Available at: https://www.thethingsnetwork.org/article/lora-transmission-from-low-orbit-satellite

11 Supplementary Material

Device CAD renderings, Data Spreadsheets, Additional Tutorials.

12 Data Availability Statement

The datasets generated by this study with all relevant design files can be found in the OPEnS-Hub_Frontiers GitHub Repository [https://github.com/OPEnSLab-OSU/OPEnS-Hub_Frontiers].

Figure 1. The depiction above represents the data pipeline from the point of acquisition in the field to observation on personal devices.

Figure 2. Summary of field tests (C) at the HJ Andrews Forest (A), and Lewis Brown Farms (B).

Figure 1.JPEG

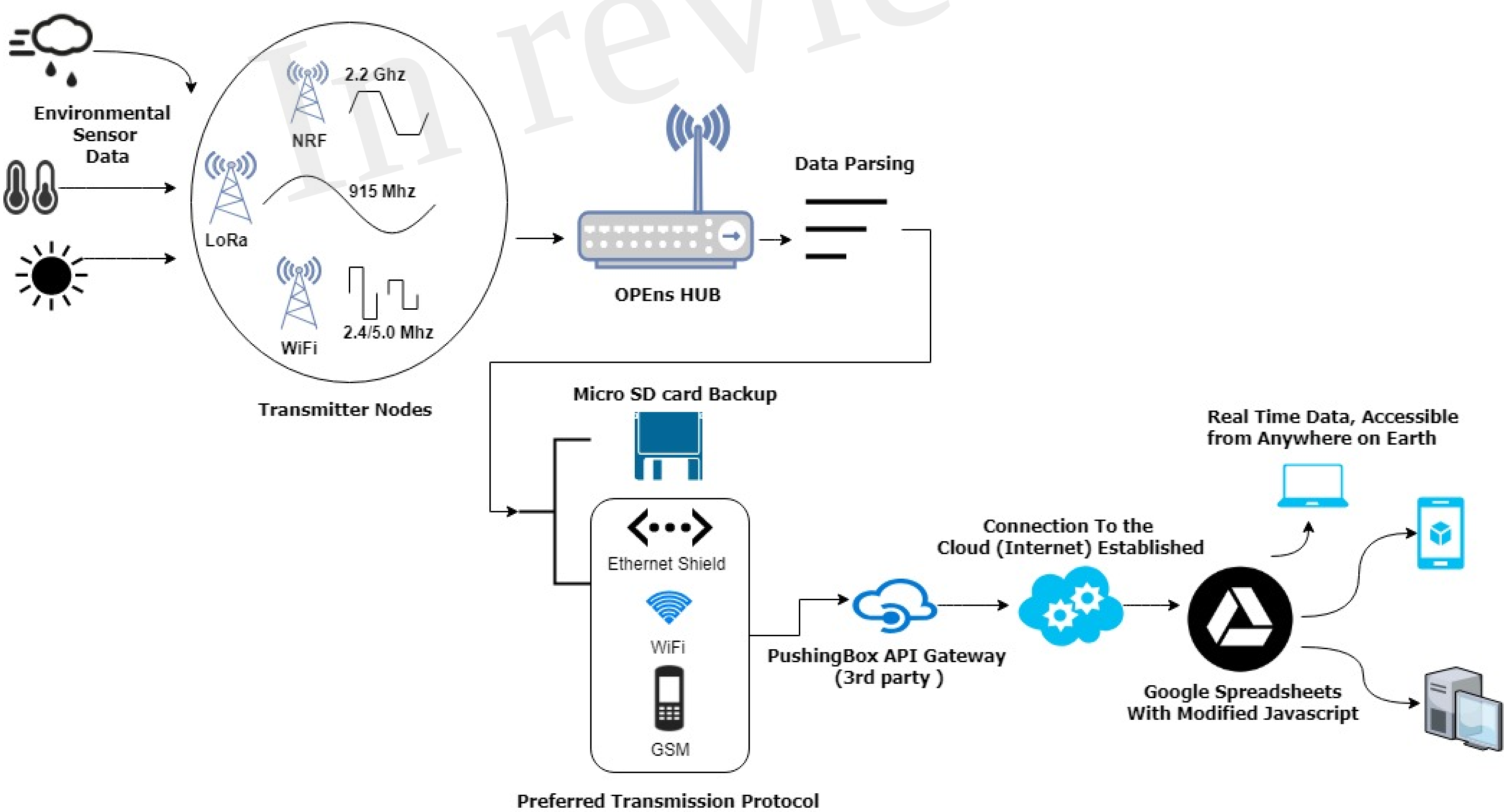
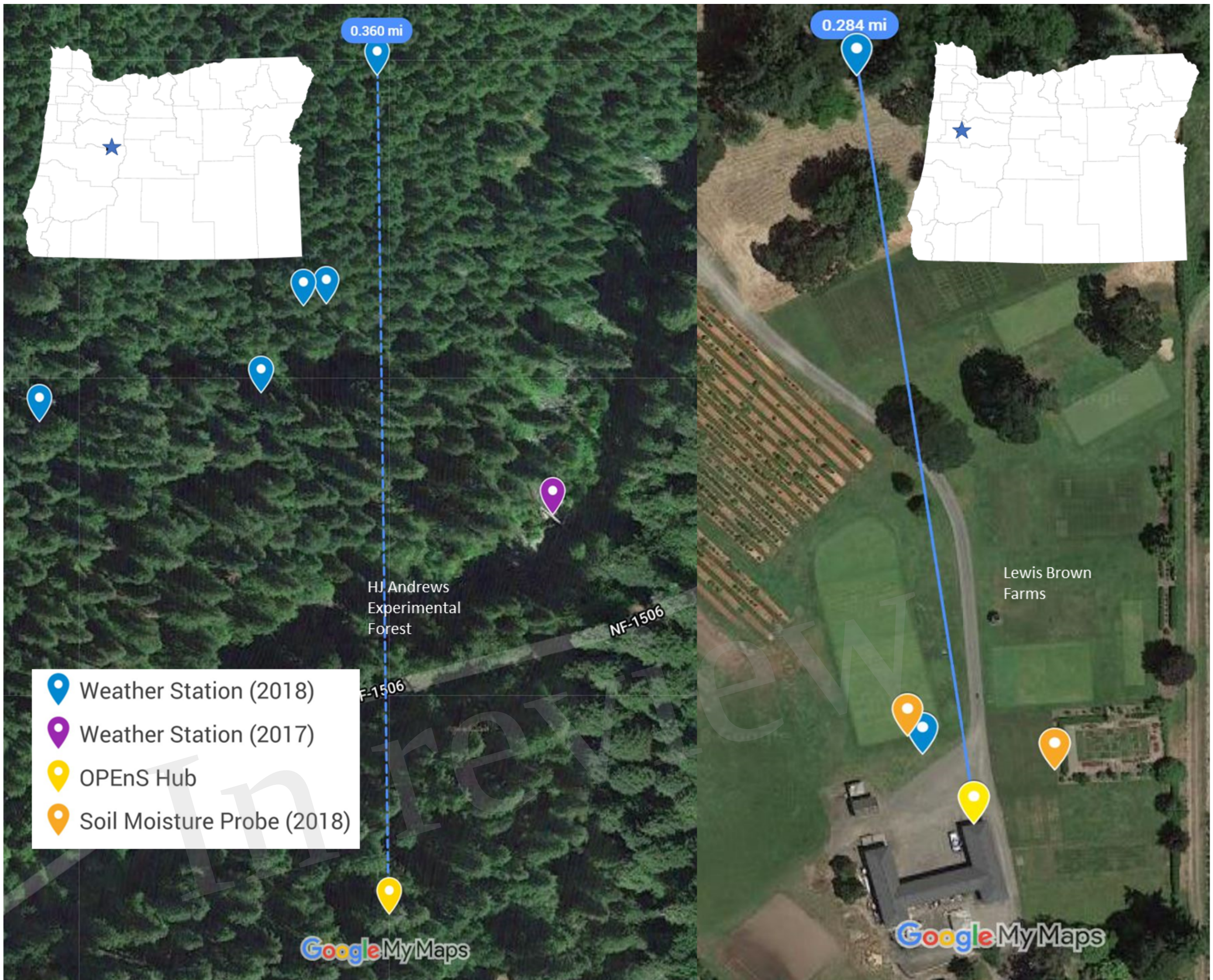


Figure 2.TIF

(A)

(B)



(C)

Design Phase	Location	Device(s)	Time of Deployment	Variables Collected	Collection Interval	Data Points
Field Test 1	HJ Andrews Experimental Forest, Blue River OR	1 Open Source Weather Station	07/24/2017-09/21/2017	Temperature, Humidity, Light Intensity, Change in Mass (Rain)	5 minutes	106,274
Field Test 2	Lewis Brown Farms, Corvallis OR	3 Soil Moisture Devices & 2 Open Source Weather Stations	04/17/2018-08/14/2018	Soil Conductivity, Soil Temperature, Temperature, Humidity, Change in Mass (Rain)	15 minutes	334,961
Field Test 3	HJ Andrews Experimental Forest, Blue River OR	5 Open Source Weather Stations	08/28/2018-9/15/2018	Temperature, Humidity, Albedo, Light Intensity (Full spectrum and IR), Change in Mass (Rain).	8 minutes	41,615
				Total Data Points Collected:		482,850