Edge-Enabled Real-Time Data Processing in Power-Efficient Weather Stations using IBIS

William Fowler
Tufts University
Medford, MA, USA
william.fowler@tufts.edu

Kate Keahey (advisor)
Argonne National Laboratory
Lemont, IL, USA
keahey@mcs.anl.gov

Alicia Esquivel Morel (advisor)
University of Missouri
Columbia, MO, USA
ace6qv@mail.missouri.edu

Abstract

There is a growing need to acquire a larger quantity of meteorological data to address climate change. In this paper, we design an improved Automatic Weather Station (AWS) based on a prototype from the National Center for Atmospheric Research (NCAR). We integrate this weather station with IBIS, a platform for adaptable, multi-sensor data collection on edge devices. Our solution utilizes a Raspberry Pi 4 to aggregate sensor data from AWSs over LOng-RAnge (LoRa) radio frequency. A real-time data visualization platform, using Grafana, InfluxDB and hosted on the Chameleon testbed, is presented. We show how the expanded peripherals allow for the implementation of novel weather forecasting techniques and demonstrate the power-efficiency of our solution by comparing the power consumption of our choice of microcontroller to the Raspberry Pi. Lastly, we examine how our implementation can address challenges in big-data weather forecasting.

CCS Concepts: • **Applied computing** → **Environmental sciences**.

Keywords: Automatic Weather Stations, LoRa, Power Efficiency

ACM Reference Format:
William Fowler, Kate Keahey (advisor), and Alicia Esquivel Morel (advisor). 2024. Edge-Enabled Real-Time Data Processing in Power-Efficient Weather Stations using IBIS. In *Proceedings of ACM Student Poster Competition (SC '24).* ACM, New York, NY, USA, 3 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Meteorological data has increased exponentially in recent years [4]. This has led to several attempts to create solutions to collect and analyze these large datasets. Based on the literature [3, 7], one of the solutions is deploying AWSs that automate data collection from weather sensors. Additionally, the analysis of larger-size data, including images or videos, has been shown to enhance weather prediction methods [2]. In this paper, we present an improved version of the OpenIoTwx AWS from NCAR [8]. Our prototype is integrated with IBIS [1], an observational instrument for data collection on a large scale. The solution transmits collected sensor data over LoRa to edge devices. To assess whether we make AWSs more power-efficient, we consider two configurations, one using a microcontroller and the other using a Raspberry Pi.

SC '24, Nov 12-17, 2024, Atlanta, GA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/XXXXXXX.XXXXXXX

2 Approach

Our implementation differs from OpenIoTwx by including a camera as an additional peripheral and utilizing LoRa, a protocol optimized for low-powered IoT devices to transmit the collected data. A Raspberry Pi 4 is used to aggregate data from stations within a range of 5 km as well as operate the camera. The Raspberry Pi is integrated with the IBIS testbed [1] and uploads aggregated data to our cloud service over MQTT protocol using Paho-mqtt [9]. Our cloud service, hosted on the Chameleon testbed [6], uses Mosquitto, an MQTT broker, to receive data from the Raspberry Pi. Our design is outlined in **Figure 1**: Telegraf transfers data from the MQTT broker to InfluxDB which timestamps and stores the data. To visualize the data, we rely on Grafana. Plots of metrics collected from the sensors is described in **Figure 2**. Chu et al. [2] use a random forest classifier to predict weather from images. A novel functionality of our AWS would be to combine sensor data with images to improve upon this weather prediction method. The Chameleon node used for hosting the data visualization and storage services could also be used to train new classifiers on the collected data and run inference models.

3 Evaluation

Our initial deployment is done inside a lab where we deploy two weather stations. One reading actual sensor data and one simulating data, and a Raspberry Pi IBIS device. The deployment ran for 40 hours with readings captured from the stations once a minute. The plots in **Figure 2** show a continuous stream of the collected data with a near-zero error rate on transmitted samples. We compare the power-efficiency

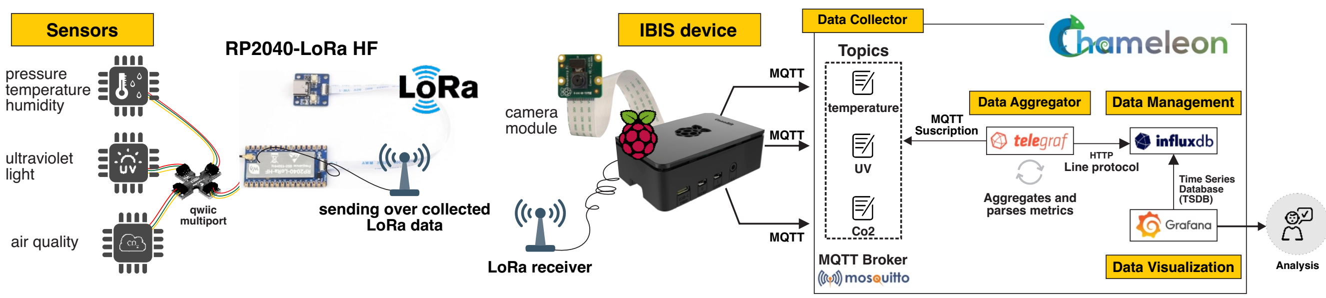
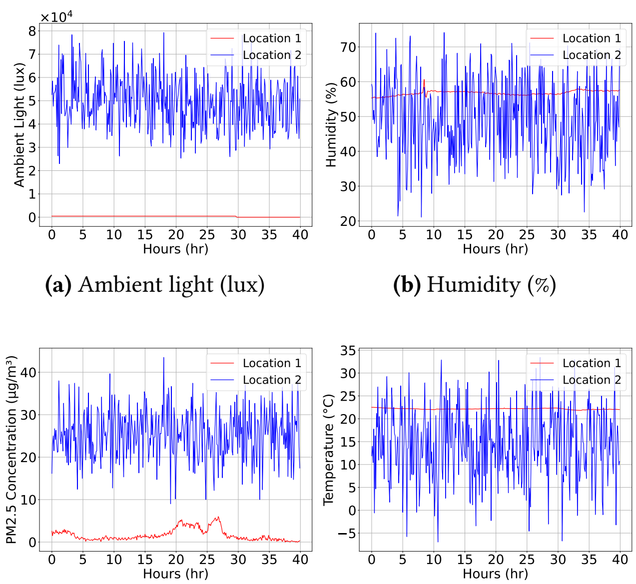
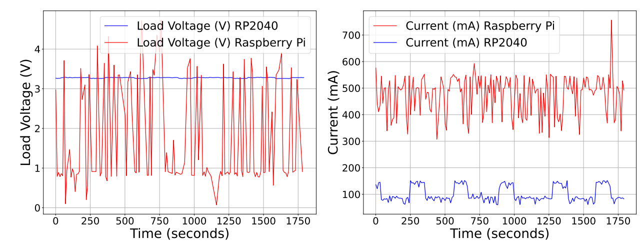


Figure 1. Schematic illustrating the architecture of our weather station. It is integrated with IBIS, an observational instrument for data collection on a large scale.



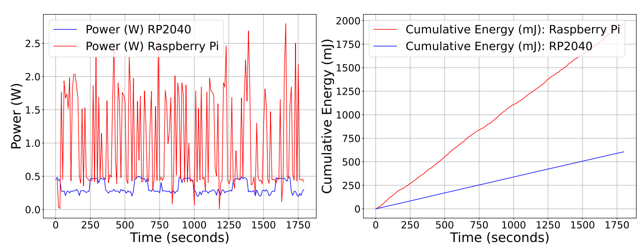
(a) Ambient light (lux) **(b)** Humidity (%) **(c)** PM 2.5 (µg/m³) **(d)** Temperature (℃)

Figure 2. Metrics collected from sensors. Actual sensor readings collected from station 1 are shown in red while simulated sensor readings collected from station 2 are shown in blue. A total of 19 different metrics are collected in total.



(a) Load voltage over time. **(b)** Current draw over time.

Figure 3. (a) Load voltage over time and (b) Current draw over time for the Raspberry Pi and RP2040 micro-controller. The Raspberry Pi uses significantly more current compared to the RP2040.



(a) Power usage comparison. **(b)** Cumulative power use.

Figure 4. (a) Power usage over time and (b) Cumulative power consumption. The figures compare the power usage of the Raspberry Pi and RP2040 micro-controller, showing both time-based and cumulative power consumption metrics.

of using a RP2040 microcontroller to a Raspberry Pi 4, when tasked with collecting and transmitting sensor data. Our experimental setup uses a INA219 power sensor operated by an additional Raspberry Pi, to read voltage, current, and power used by each device during a 30 minute period. We display the results of the experiment in **Figures 3a, 3b, 4a**, and **4b** below, which show the Raspberry Pi using 4× the power of the microcontroller.

4 Conclusions

To address our big-data weather forecasting needs, we must develop infrastructure to collect meteorological data from remote locations at scale. In this paper, we introduce our AWS implementation and outline its improvements over prior work. A demo deployment shows successful visualization of data collected from our AWSs. We demonstrated the RP2040 microcontroller's 4× power-efficiency over a Raspberry Pi for our deployment. Managing large weather datasets is a challenge in modern weather forecasting [5] which our AWS can address solve by concentrating data aggregated from a wide range of locations into a central hub. Access to high-quality weather datasets enhances the efficacy of today's computation-intensive forecasting models. Future work on our AWS will involve utilizing capture images to further improve weather forecasting capabilities.

References

[1] K. Keahey Z. Murry T. Sitzmann J. Zhou A. Esquivel Morel, M. Powers and P. Calyam. 2024. IBIS — An Infrastructure Management Framework for Adaptable, Multi-Sensor Data Collection in Scientific Research. *ISC High Performance 2024 International Workshops* (2024). https://doi.org/10.13140/RG.2.2.11758.22082 Preprint available on ResearchGate.

[2] Wei-Ta Chu, Xiang-You Zheng, and Ding-Shiuan Ding. 2017. Camera as weather sensor. *J. Vis. Comun. Image Represent.* 46, C (jul 2017), 233–249. https://doi.org/10.1016/j.jvcir.2017.04.002

[3] Konstantinos Ioannou, Dimitris Karampatzakis, Petros Amanatidis, Vasileios Aggelopoulos, and Ilias Karmiris. 2021. Low-cost automatic weather stations in the internet of things. *Information* 12, 4 (2021), 146.

[4] Himanshi Jain and Raksha Jain. 2017. Big data in weather forecasting: Applications and challenges. In *2017 International conference on big data analytics and computational intelligence (ICBDAC)*. IEEE, 138–142.

[5] Himanshi Jain and Raksha Jain. 2017. Big data in weather forecasting: Applications and challenges. In *2017 International Conference on Big Data Analytics and Computational Intelligence (ICBDAC)*. 138–142. https://doi.org/10.1109/ICBDACI.2017.8070824

[6] Kate Keahey, Jason Anderson, Zhuo Zhen, Pierre Riteau, Paul Ruth, Dan Stanzione, Mert Cevik, Jacob Colleran, Haryadi S. Gunawi, Cody Hammock, Joe Mambretti, Alexander Barnes, François Halbach, Alex Rocha, and Joe Stubbs. 2020. Lessons learned from the Chameleon testbed. In *Proceedings of the 2020 USENIX Conference on Usenix Annual Technical Conference (USENIX ATC'20)*. USENIX Association, USA, Article 15, 15 pages.

[7] K Lagouvardos, V Kotroni, A Bezes, I Koletsis, T Kopania, S Lykoudis, N Mazarakis, K Papagiannaki, and S Vougioukas. 2017. The automatic weather stations NOANN network of the National Observatory of Athens: Operation and database. *Geoscience Data Journal* 4, 1 (2017), 4–16.

[8] Keith Maull. 2023. OpenIotwx. https://ncar.github.io/openiotwx/.

[9] PierreF and Roger.Light. 2013. Paho-MQTT. https://pypi.org/project/paho-mqtt/.