122nd ASEE Annual Conference & Exposition
June 14 - 17, 2015
Seattle, WA
Making Value for Society
Paper ID #12978

Work-in-Progress: High-Frequency Environmental Monitoring Using a Raspberry Pi-Based System

Debarati Basu, Virginia Tech

Debarati Basu is a second year PhD student, advised by Dr. Lohani, in Engineering Education at Virginia Tech. She has B.Tech and M.Tech in Computer Science & Engineering. She has completed several graduate level courses in the Engineering Education Department. She is engaged in developing a system with Raspberry Pi for high frequency real-time environmental monitoring in the LEWAS Lab. She has mentored an undergraduate student who was assisting her is developing the system. She has experience in organizing NSF/REU site for interdisciplinary water sciences and engineering and also in teaching freshman year in VT. She has also helped in developing and implementing a project with LEWAS data into a freshman level course in Virginia Tech.

Mr. John Stanton Goldstein Purviance, Virginia Tech

John S.G. Purviance is a B.S. student in Computer Science at Virginia Tech. He has been working at the Learning Enhanced Watershed Assessment System (LEWAS) Lab for the past two years as an undergraduate research intern. During summer 2014, he worked as an REU fellow at the LEWAS lab. This REU site is hosted at this lab. He has background in python programming.

Darren K Maczka, Virginia Tech

Darren Maczka is a M.S. student in Electrical and Computer Engineering. His background is in control systems engineering and information systems design and he received his B.S. in Computer Systems Engineering from The University of Massachusetts at Amherst. He has several years of experience teaching and developing curricula in the department of Electrical and Computer Engineering at Virginia Tech. He is presently assisting in developing the high frequency real-time environmental monitoring system and upgrading the power distribution system in the LEWAS Lab.

Mr. Daniel S Brogan, VIrginia Tech

Daniel S. Brogan is a PhD student, advised by Dr. Lohani, in Engineering Education with BS and MS degrees in Electrical Engineering. He has completed several graduate courses in engineering education pertinent to this research. He is the key developer of the OWLS and leads the LEWAS lab development and implementation work. He has mentored two NSF/REU Site students in the LEWAS lab. He assisted in the development and implementation of curricula for introducing the LEWAS at VWCC including the development of pre-test and post-test assessment questions. Additionally, he has a background in remote sensing, data analysis and signal processing from the University of New Hampshire.

Dr. Vinod K Lohani, Virginia Tech

Dr. Vinod K. Lohani is a Professor of Engineering Education and an adjunct faculty in Civil & Environmental Engineering at Virginia Tech (VT), Blacksburg. He is director of an interdisciplinary lab called Learning Enhanced Watershed Assessment System (LEWAS) at VT. He received a Ph.D. in civil engineering from VT. Dr. Lohani's research interests are in the areas of computer-supported research and learning systems, hydrology, engineering education, and international collaboration. He has led several interdisciplinary research and curriculum reform projects, funded by the National Science Foundation (NSF), at VT. He has participated in research and curriculum development projects with ˜$4.5 million funding from external sources. He has been directing/co-directing an NSF/Research Experiences for Undergraduates (REU) Site on interdisciplinary water sciences and engineering at VT since 2007. This site has 66 alumni from all over the United States to date. He collaborated with his colleagues to implement a study abroad project (2007-12), funded under the US-Brazil Higher Education Program of the U.S. Department of Education, at VT. He has published over 70 papers in peer-reviewed journals and conferences. He has advised 5 PhD and 10 MS students to completion and is advising 5 PhD and 1 MS students currently. In 2011, he was awarded the American Society for Engineering Education (ASEE) International Division's Global

Engineering & Engineering Technology Educator Award. He has won several awards at VT including the Scholarship of Teaching and Learning Award in 2013. Virginia Tech nominated him for the 2015 Outstanding Faculty Award of the State of Virginia.

Work-in-Progress: High-Frequency Environmental Monitoring Using a Raspberry Pi-Based System

Abstract

The Learning Enhanced Watershed Assessment System (LEWAS) is a unique high-frequency real-time environmental monitoring lab on the campus of Virginia Tech. The LEWAS has the following four stages: 1) data inputs which consist of environmental instruments including an acoustic Doppler current profiler, a water quality Sonde and a weather transmitter taking measurements every 1-3 min., 2) data processing occurring locally on a Raspberry Pi, 3) data storage on a remote server and 4) data visualization through an Online Watershed Learning System (OWLS) (www.lewas.centers.vt.edu/dataviewer/) through which end users access the LEWAS data for research and education. In this paper, we discuss the developmental work that was involved in upgrading the processing (stage 2) and storage components (stage 3) of the LEWAS and its application in various courses at Virginia Tech and in VWCC. The development of this upgraded system started as part of a Research Experience for Undergraduate (REU)/ NSF program during the summer of 2014. A Raspberry Pi computer was adopted as the processing unit primarily because it offers the following advantages: low power consumption, compatibility with multiple programming languages, ability to interface with many sensors and a low purchase price. A python program for each instrument was developed and tested on the Raspberry Pi to collect, parse and store the environmental data locally into a MySQL database. Though the REU work ended at the end of summer, the work is continuing at the time of writing. The aim of this continued development is to make the system more flexible, robust and maintainable in order for the LEWAS to become easily extendable and adaptable to different environments. Redesign of the partially completed upgraded system includes incorporation of the following three new components: 1) refactoring the instrument parsing code, 2) restructuring the database schema that will allow easy integration of new sensors as they are added to the system regardless of the syntactic and semantic structures of the data, and 3) implementing a REST Application Programming Interface (API) to allow for easier user application development. This development will ultimately help to provide smooth operation of user applications like the OWLS. Since 2009, this lab has been utilized in various courses at Virginia Tech and in VWCC, and more than 10,000 students from both these schools have used the LEWAS and/or OWLS to learn about high frequency environmental monitoring and its use continues to grow. The development of the LEWAS into a robust and reliable system will help students/faculty not only from this university but also from various parts of the world use the lab and its data for environmental education and research.

1.0 Introduction

The Learning Enhanced Watershed Assessment System (LEWAS) is a unique real-time high-frequency environmental monitoring lab established to promote environmental monitoring education and research[1,2]. This system was conceived as a part of a PhD dissertation in the Department of Engineering Education at Virginia Tech[1]. The LEWAS field site (Figure 1) is located at the outlet of a small creek (Webb Branch) that flows through the Virginia Tech campus[3]. This stream joins a water quality impaired stream (Stroubles Creek). The Webb branch was chosen as the site of the lab because of its environmental significance as also due to its proximity to the campus[4,5]. The LEWAS site drain a watershed which is approximately 2.78 km^2, and approximately 95% urban. The field site has instruments to measure water quality and quantity parameters including velocity, depth, pH, dissolved oxygen, turbidity, oxidation reduction potential, specific conductance, and temperature. In addition, weather parameters (temperature, barometric pressure, relative humidity, precipitation and wind) are also measured at the site. The LEWAS draws power from both the grid and a photovoltaic panel and connects to the campus wireless network through a high-gain antenna to transmit sampled data to a database every 1-3 minutes. Remote clients can view this sampled data through user applications.

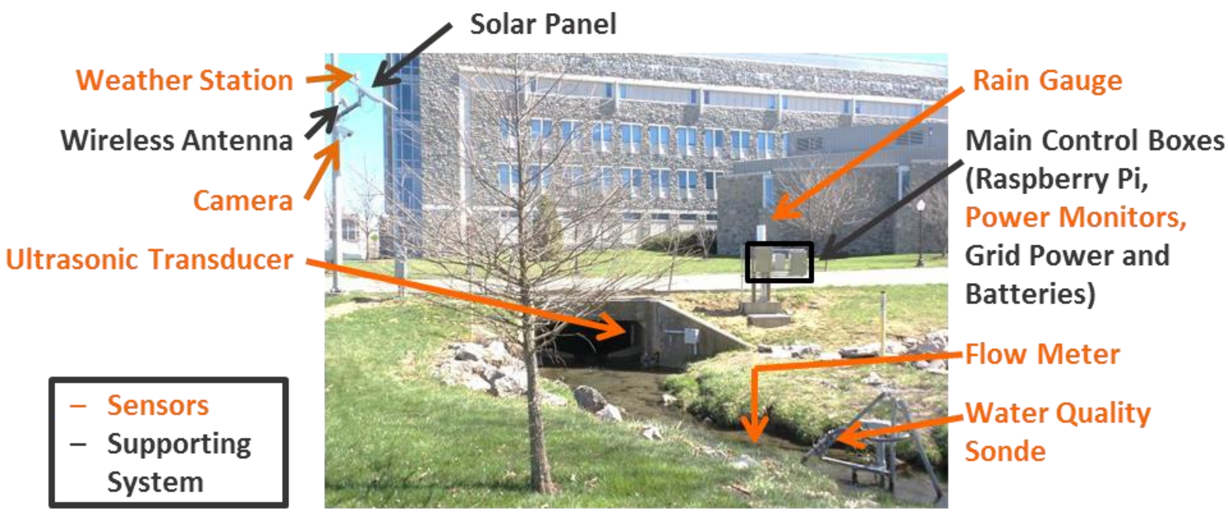

Figure 1. The LEWAS field site.

Diverse expertise and skill sets are required to design, implement and maintain the various components of the LEWAS (described later). In order to meet these requirements, the LEWAS lab has an interdisciplinary team consisting of two faculty members and graduate and undergraduate students from various academic backgrounds including engineering education, electrical engineering, computer engineering, civil engineering, environmental engineering and biology. This team meets on a weekly basis and is responsible for the continued development and expansion of the LEWAS and maintenance of its outdoor field site and indoor lab space. A number of in-class and out-of-class exercises and modules using the LEWAS and its data have been developed and implemented, as part of an ongoing NSF project[6], in various courses at Virginia Tech and Virginia Western Community College (VWCC)[1,2,6-12]. Because one of the goals of the LEWAS Lab is to expand the usage of the system to different institutions around the globe including both educational and data measuring applications and research, the LEWAS is being designed as a flexible and expandable environmental monitoring system that can easily be adapted and deployed in a wide variety of environments.

Preliminary work related to the acquisition and processing of the LEWAS data was done using the National Instruments' CompactRIO (cRIO)[13] and LabVIEW programming based processing unit[1]. However, due to several limitations of this system[14] and for economic and scalability reasons it was decided to make a transition to a Raspberry Pi[15] and Python-based [16] processing unit in the LEWAS. The development of this new system started as part of a Research Experience for Undergraduate (REU)/NSF program during the summer of 2014. Much of the developmental work planned in this 10-week REU project was implemented during summer 2014. However during the fall of 2014, it was determined that the summer 2014 implementation of the Raspberry Pi-based system was not easily maintainable or extendable. In order to address these concerns a redesign was proposed consisting of three components: 1) refactoring the instrument parsing code, 2) restructuring the database schema and 3) implementing a REST API explained later) to provide a consistent way to access the data.

This paper describes the REU project and its continuation. Section 2.0 of the paper summarizes the four stages of the LEWAS infrastructure. Section 3.0 describes the prior system and the challenges and limitations associated with it. Section 4.0 elaborates on the technologies used for the new system and their advantages over the earlier system. Then an overview of the new Raspberry Pi based system is given in section 5.0. The implementation of the redesigned data flow system is now in progress and is described briefly in section 6.0. Finally, section 7.0 summarizes the courses at various institutions that have utilized the LEWAS and its data for educational purposes.

2.0 The four stages of LEWAS

At the completion of the REU project in summer 2014, the LEWAS contained the following four stages (Figure 2): 1) data inputs which consist of environmental instruments at the field site including an acoustic Doppler current profiler[17] a water quality Sonde[18] and a weather transmitter[19], 2) data processing occurring locally on a Raspberry Pi for analysis and storage, 3) data storage on a remote server and 4) data visualization through the Online Watershed Learning System (OWLS) (www.lewas.centers.vt.edu/dataviewer/), through which end users can access the real-time high frequency LEWAS data for research and education.

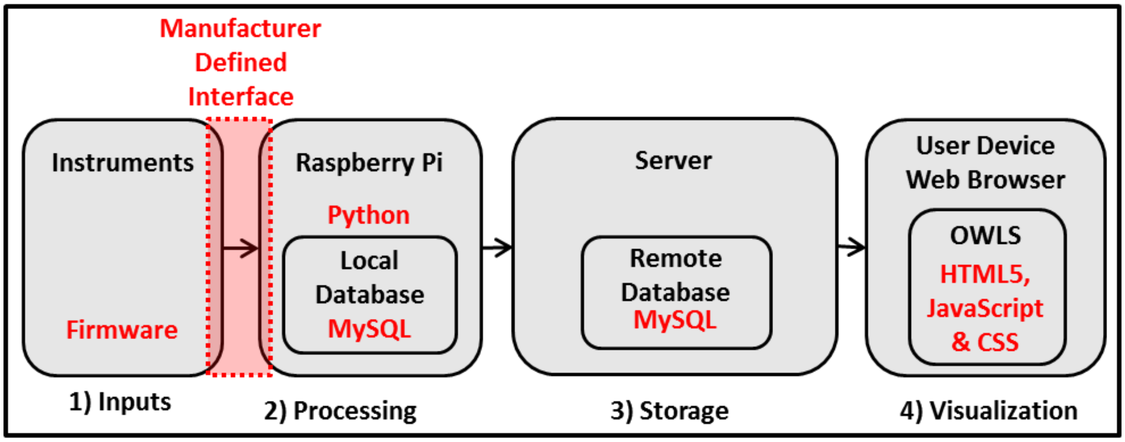

Figure 2. The four stages of the LEWAS and its data flow.

2.1 Stage 1: Inputs

This field site contains various instruments for collecting environmental data that take measurements every 1-3 min. The first is a SonTek Argonaut-SW Acoustic Doppler Current Profiler (ADCP), which measures stage and index velocity in a cross section of the stream. The second is a Hach Hydrolab MS-5 water quality Sonde which measures parameters such as pH, temperature, specific conductivity, dissolved oxygen (DO), oxidation reduction potential (ORP) and turbidity. The third is a Vaisala WXT520 weather transmitter mounted on a light pole near the field site. The weather transmitter measures air temperature, barometric pressure, relative humidity, precipitation and wind speed. There is also a Weathertronics Tipping Bucket rain gage which collects precipitation data and a Global Water WL705-02 ultrasonic level transducer that collects stage measurements in a weir just upstream of the site and provides additional flow measurements to increase data quality and accuracy. Only the three primary instruments, i.e. the ADCP, the Sonde and the weather transmitter, are currently connected to the processing unit (stage 2) for data collection and storage and work to connect the others is in progress. The site also has a camera to capture photos (http://www.lewas.centers.vt.edu/dataviewer/camera.html), which are used to enable remote users to visually assess the conditions at the field site during various events. Stages 1 is described in more detail in prior papers and a PhD dissertation[1,2,3,8].

2.2 Stage 2: Processing

At the time of the REU project completion, the second stage of the LEWAS, the processing unit, is involved in reading the real-time data collected by the instruments. It communicated with the three primary instruments, read the data from these instruments, processed these data and stored them in a local database on the Raspberry Pi. This processing unit is a critical stage of the full system as all data collected from various instruments pass through it. Therefore, the design of this processing unit needs to be both efficient and stable so that it can collect accurate data uninterruptedly for extended periods of time with minimal maintenance.

2.3 Stage 3: Storage

The third stage of the LEWAS is the server with the remote database, which was not implemented during the 10 week summer 2014 REU project period. The combination of a background setting in the local database and an internet connection to the remote server would have allowed the processing stage to automatically update the recently added local data to be replicated in a remote database. The remote database was intended to store the data and provide access to both near real-time data and archived data that could be utilized by students and other end users via applications like the OWLS. The connection between the processing stage and the remote database is being implemented as part of the redesign.

2.4 Stage 4: Visualization

The first three stages of the LEWAS have been designed such that a wide variety of end-user applications can utilize the LEWAS data. The OWLS is one example of such applications. Basically, the OWLS is an interactive cyberlearning system that delivers integrated live and/or historical environmental data (atmospheric, hydrologic, geographical, visual, etc.) from the LEWAS to end users regardless of the hardware and software platforms used (Figure 3). The development of the OWLS, which was previously called the Platform-Independent Remote Monitoring System (PIRMS), is described in more detail in Brogan et al.[20]

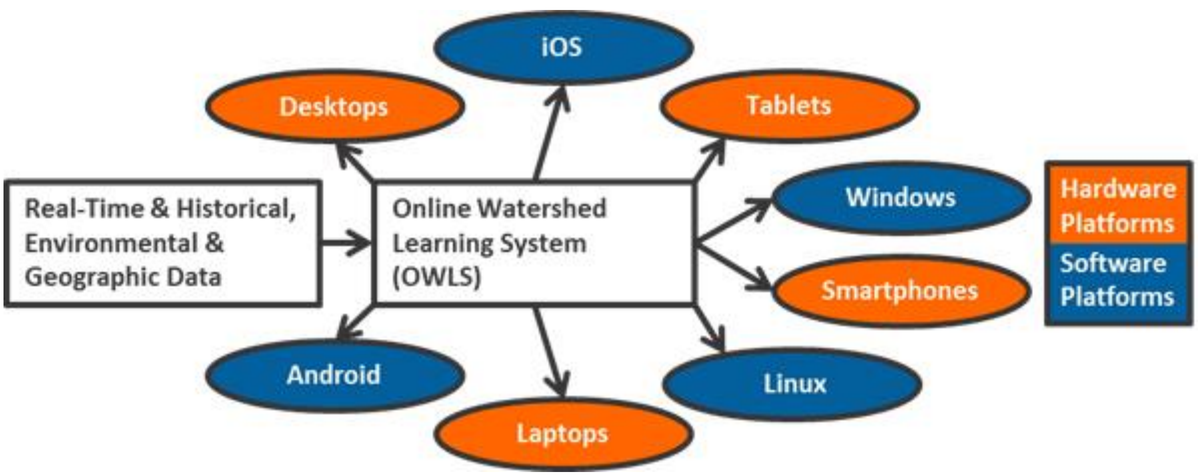

Figure 3. The OWLS reaches users across many platforms.

3.0 Prior work and shortcomings

Prior to the current Raspberry Pi work, the LEWAS used a cRIO (an embedded hardware system) and LabVIEW (a graphical programming language) to create the environmental monitoring system. The cRIO was set up in the field site to collect data from the three primary instruments in stage 1. The cRIO made these data available to end users in real-time via a LabVIEW-driven web interface. This version of the LEWAS had no data storage capability. Additionally, the combination of the cRIO model used and LabVIEW version 8.6 that was running on that cRIO prior to May 2013, did not allow more than one of the instruments to be connected at any given time. To address these shortcomings, an NSF/REU fellow worked on updating the system during the summer of 2013. The primary goal of this REU project was to implement a database management system and develop a user-friendly web-interface to the existing weather monitoring system with the cRIO using historical and real-time data[14]. It may, however, be noted that the summer 2013 REU project is not the focus of this paper.

This previous system update was not fully implemented for several reasons[14]. Firstly, difficulty was encountered in connecting to the cRIO with the desktop computer in the lab. This occurred, even though the cRIO was reformatted and the latest versions of various modules were installed on the cRIO (LabVIEW version 2012 Service Pack 1). Secondly, compiling the virtual instruments (VI's) failed several times using the "compile cloud" method before the method was changed to local compilation. Finally, once the connection was established, the operating system files for the 2012 version of LabVIEW were found to be too large for the cRIO model used. To address all these issues, stages 2 and 3 of the LEWAS are being redeveloped and implemented based on a Raspberry Pi computer. The next section describes the different technologies used for the various components of the revised system which started being implemented during the summer 2014 REU project.

4.0 Technologies used in implementing the new system

As part of the processing unit redevelopment, the cRIO was replaced with a Raspberry Pi, which can provide a large array of tools for collecting and storing the data. A Raspberry Pi is a system on chip computer that is slightly larger than a credit card and has all the main characteristics of a full desktop computer[15]. It can easily be placed at the LEWAS field site in the control box shown in Figure 1 and can be accessed remotely from the LEWAS' indoor lab space. It has several advantages over the cRIO. Firstly, it addresses the primary issue of power consumption by the earlier system by reducing the power consumption by fourteen watts compared to the cRIO (from 20 W down to 6 W). Secondly, the cRIO had the limitation of being able to connect to only one instrument at a time during real time data collection[20]. The Raspberry Pi instead can be connected to several instruments with the help of a simple USB hub, thus making the system more expandable. Thirdly, with the cRIO, the LEWAS was constrained to use LabVIEW as the only supported primary programming language, but the Raspberry Pi can use any programming language compatible with GNU/Linux. Fourthly, the cRIO model used has an internal storage capacity of 128MB whereas the Raspberry Pi's, which is used for the processing unit, can access up to 32Gb of storage on an SD card. Moreover, the connection and compilation challenges faced by working with the cRIO, were not encountered while working with Raspberry Pi. Thus the Raspberry Pi adds significant flexibility to the system and was, therefore, chosen as the best fit for data collection in the LEWAS. In this implementation, the Raspberry Pi is running a derivative of the open source Debian Linux Distribution optimized to run on the Raspberry Pi called Raspbian[21].

In this system the instruments are connected to the Raspberry Pi through RS-232 serial connections. During the summer 2014 REU program, individual Python[16] programs were developed for each instrument, connected to the instruments and tested for data acquisition over the serial connections. Python was utilized in this implementation because of its powerful abstractions and its ease of implementation. Python has a comprehensive standard library in addition to a very large selection of mature third party libraries. One of these called PySerial[22] was used to communicate over the serial connection with the instruments. PySerial retains the abstraction that is inherent to the language while still allowing full use of the serial port and related settings.

After the data had been read by each python program from each instrument, python functions parsed the data and inserted them into a MySQL database running locally on the Raspberry Pi. The MySQL[23] database system is a popular open source solution which has high scalability, performance and flexibility. It is the commonly used standard due to its high performance query engine and very fast and reliable insertion capability[24].

5.0 New system overview

For the summer 2014 REU project, the Raspberry Pi was placed in the control box of the field site with no peripherals (e.g., a keyboard, a mouse or a monitor) connected to it. The Raspberry Pi had a static IP address assigned to it. When controlling the device in the field, a Linux laptop was directly connected to the computer via an Ethernet cable. Linux laptop was also given a static IP address and then the user of the computer laptop could SSH (Secure Shell, a UNIX based command interface and protocol for securely getting access to remote computer) into Raspberry Pi. When it is connected to the campus wireless network the Raspberry Pi can be accessed remotely from the lab office or anywhere else. Remote terminal access to the Raspberry Pi allows for the remote start and stop of data collection and direct communication with the sensors via serial terminal software such as picocom[25].

The schematic in Figure 4 shows the intended data flow for stages 2 and 3. Much of it was implemented during the summer 2014 REU project including python programs to interface to each of the three instruments connected to the system. A few parts were not completed due to the limited time span. These included a Linux shell script that was planned for running all the python programs for each instrument for a user specified period of time. The script was needed to initialize the sensors and start data acquisition (shown in blue in Figure 4). In addition the mirroring of the local database to the remote was not completed. This had been planned to protect against data loss, an issue that must be addressed in the future.

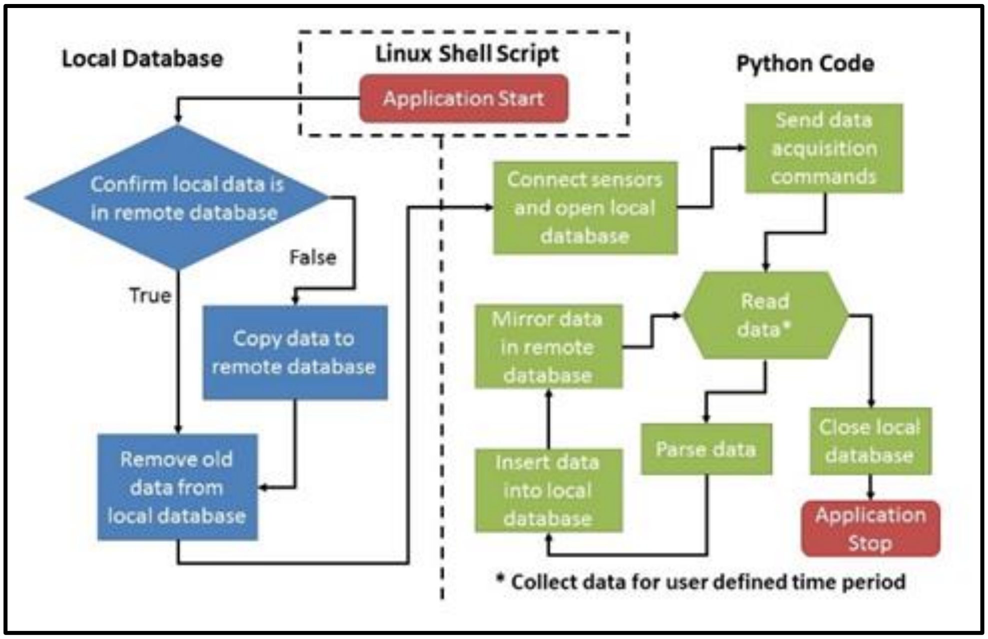

Figure 4. Data flow diagram of the processing unit

As shown in Figure 4, after the data preservation steps (shown in blue) were completed, blocks shown in green started data acquisition. Each of the python programs for the three instruments first established a connection to each instrument and opened a connection to the local database before sending instrument-specific data acquisition commands to each sensor to begin data collection. Each program read data from the instruments' output buffer when it became available. After a line of data had been acquired from an instrument, it was parsed into components that represented the individual data values. These parsed values were then inserted into the local database. As mentioned earlier, the solution for mirroring the data to the remote database remained incomplete. Thus the data had to be accessed directly from the local database at the completion of the REU project. The acquisition, parsing and insertion of data values continued for each Python program for a user defined period of time, the deployment period. These programs would later be set to run continuously. However, for testing purposes, to check for issues that may arise during the pilot phase of the system, the programs were run only for a specific period of time.

Connections to all the devices were made through TTY connections in the Linux kernel on the Raspberry Pi. Though all of the sensors communicated over RS-232, the Raspberry Pi did not have a native RS-232 serial connector, only USB. To resolve this, a USB to RS-232 converter was used to allow each sensor to connect to the Raspberry Pi via USB. The Linux kernel of the Raspberry Pi assigned input device identifiers based on the order in which instruments were detected rather than what the instrument was. To send and receive data from the correct instrument the python programs needed to know the device name associated with each instrument, which could change when the system was rebooted. To overcome this issue and provide consistent names for each device, each converter had a custom rule written that allowed the kernel to assign a name based on the particular device, regardless of the order in which they were detected.

The instruments' data collected by the python programs were stored in a local MySQL database on the Raspberry Pi. This local database on the Raspberry Pi had seven data tables; two for the acoustic Doppler current profiler, four for the weather transmitter and one for the water quality Sonde. Some instruments used multiple tables to store data in a way that reflected how data was retrieved from them. All of the tables in the local database had an index column that served as the primary key, a column with time stamps from the system time and columns of data values. The data values were stored as floating point numbers to preserve precision.

6.0 System modifications

To accommodate future growth and improve maintenance and expansion of the LEWAS, a redesign of the data flow system began in the fall of 2014. The redesign is divided into three components: refactoring the instrument parsing code, restructuring the database schema and implementing a REST API. Currently, the LEWAS still maintains the four stages but there are some modifications in the design as shown in Figure 5 and described in the sub-sections below.

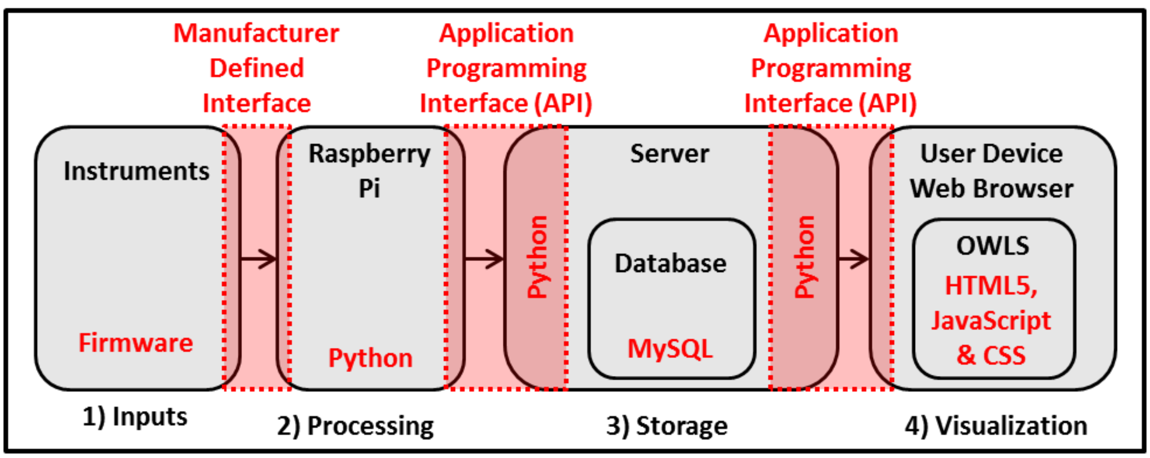

Figure 5. Data Flow Diagram of the Revised System.

6.1 Instrument data parsing

The software written during the summer 2014 REU project, which had to interface with the individual instruments, contained a mix of code and routines specific for each instrument along with more general code that was replicated for each instrument. The first step of the redesign process was to define the kinds of operations that all instruments would have in common, e.g. connect, read data, write data, parse data and refactor those components into a shared module. While each instrument has a parsing step that must accept textual data from the instrument and manipulate it into a form suitable for storage, the details of parsing depend on individual instrument data formats. To accommodate this, a parsing module is written to accept a regular expressions and other parameters that are specific to each instrument. These parameters are stored in a configuration file so that a common code base can be shared across multiple instruments.

6.2 Database schema

The original database schema from the summer 2014 REU project specified a table for each type of data packet an instrument would produce. For example, the weather transmitter has four types of data packets: pressure/temperature/humidity, precipitation, supervisor and wind speed and direction data packets and there were separate tables for each type. This design was not easily expandable: as instruments were added or removed from the system the database schema had to change to match, which in turn forced changes in system code. This tight coupling between components of the system reduced the ease of modifying any one component since a change to one component would require a change to all other components.

After conducting a preliminary survey of database schemas used for similar applications[26], a star shaped schema was chosen. In this design, commonly used in data warehouse applications, a *fact table* contains a list of values along with foreign key references to additional tables called *dimension tables* which describe the value. In the LEWAS the primary data are the values of the environmental (water and weather) parameters that are recorded from each instrument. In the new design the fact table contains observation values stored as unitless numeric values. Each row in the fact table contains foreign key references to dimension tables such as units, variable, instrument which describe the value stored and provide sufficient context to interpret row of the fact table. That is, the dimension tables store the metadata that describe the values stored in the fact table. Crucially, specific information about instruments is not encoded into the database schema but rather is contained as data in the tables themselves. The structure of the program that interfaces with the database is tied to the database schema, so a schema that does not need to change often makes code maintenance significantly easier. The current schema is structured such that a wide array of instruments could be used without altering the database schema. While this schema may seem more complex due to separating meta data (units, instrument used, etc.) from values, it is structured in a way that is easy for a program to understand. This is sufficient as there is generally no need for an end user to view the database directly.

6.3 Introduction of a REST API

In the preliminary system design, the end-user interface, the OWLS, relied on querying the database directly. This coupled the design of the user-side code to the database schema. While this may not seem to be a major restriction in the development of the OWLS, decoupling the two components would provide a more robust system design. Furthermore The LEWAS Lab would like to provide an opportunity for end users to develop their own applications that make use of LEWAS data at the same time. A REST API is a common way to provide controlled access to data sets to third party developers[28]. By defining a public REST API for the LEWAS, an opportunity for additional educational experiences related to data-driven application development is created.

7.0 Educational application of the LEWAS lab

The LEWAS Lab and its data have the potential to educate both students and the general public about the need and implications of high frequency environmental monitoring. From 2009, this lab has been utilized in various courses at Virginia Tech and in VWCC as shown in Table 1. This table also shows the course number and the number of students who were introduced to the LEWAS and in several cases used LEWAS-based modules for learning about high frequency environmental monitoring[1,2,6-12]. Several of these students participated in various class activities such as hands on exercises, case study review, field trips, data analysis and report writing, etc. addressing various concepts of sustainability, environmental monitoring, various programming applications, etc. Various studies relating to these activities assessed students' actual learning, self-perceived learning and motivation. Results from a study of 150 participants in a fresman engineering course (EngE 1024) determined that providing real-time access to environmental parameters can increase student interest and their perception of the feasibility and relevance of environmental monitoring[1,9]. Successful implementation of LEWAS into EngE 1024 led to the utilization of LEWAS in many other courses including a senior level hydrology course at Virginia Tech in order to help students learn water quality and quantity concepts through "activity learning strategy" using LEWAS-based modules and field trips. Students were assessed and their feedback was taken to improve the modules and its implementation[6,7,9]. LEWAS based modules showing real world application of LabVIEW programming, data acquisition and data analysis were also introduced to the freshman courses in the VWCC. Students in these courses were also introduced to sustainability concepts, basics of water quality monitoring and Microsoft Excel software using LEWAS data. Results of pre- and post-tests on both these hydrology and freshman courses showed positive learning gains[6,7], and student blogs[29,30] show their active participation in the LEWAS-based environmental monitoring modules. Implementation of the LEWAS-based modules in the hydrology and community college courses is continuing in the spring 2015 semester[10.] In addition, a design project using the LEWAS data is being implemented in another freshman level engineering course (EngE 1216, Foundations of Engineering II) at Virginia Tech, where more than 1400 students learn about the environmental monitoring in a project-based environment.

Semester, Year	Course Number	Course Name	Institution	No of Students per semester
Fall 2009 - Spring 2014	EngE 1024	Engineering Exploration	Virginia Tech	~1700/year
Spring 2015	EngE 1216	Foundations of Engineering II	Virginia Tech	~1500
Fall 2012 Spring 2014 Spring 2015	CEE 4304	Hydrology	Virginia Tech	~ 30 to 35 (10% graduate student)
Spring 2013	CEE 3314	Water Resources Engineering	Virginia Tech	~ 20
Spring 2015	CEE 7534	Urban Hydrology	Virginia Tech	~ 14
Fall 2013	GEOS 4804	Groundwater Hydrology	Virginia Tech	~ 20
Fall 2013	GEOS 1124	Resources Geol. Lab	Virginia Tech	~ 20
Spring 2015	CSES 4314	Water Quality	Virginia Tech	~ 20
Fall 2013- Spring 2015	EGR 124	Introduction to Engineering	Virginia Western Community College	~ 40
Spring 2013	EGR 120	Introductory Engineering and Engineering Methods	Virginia Western Community College	~ 40

Table 1. LEWAS used in various courses[1,2,6-12].

8.0 Summary and future work

This paper discusses the continuing development and application of the LEWAS on the campus of Virginia Tech. High frequency water (quality and quantity) and weather data from a site on an on-campus stream are sensed, stored, and used in various educational and research applications. The continuing development of the processing stage (stage 2) and the storage stage

(stage 3) were the defined goals of the REU project during of summer of 2014. Python programming with its PySerial library provides efficient communication with the instruments and controlled devices over RS-232 serial connections. A MySQL database enables storing and archiving LEWAS data for future use. During the REU project, python programs were developed to acquire the real time data from the instruments and store it in database tables locally on a Raspberry Pi. Because the remote database was not set up during the 10-week program, the automatic mirroring process remained untested. The shell script to run all the python programs also remained unimplemented during that period.

In fall of 2014, developmental work for the system focused around making the system more flexible, robust and maintainable in order for the LEWAS to become easily extendable and adaptable to different environments. Redesign of the partially completed project is incorporating the following three new components: 1) refactoring the instrument parsing code, 2) restructuring the database schema and 3) implementing a REST API. This system development will ultimately help to provide smooth operation of user applications like the OWLS. Future work in the lab includes fully integrating and testing of the Raspberry Pi-based components of the LEWAS with the remaining instruments and the camera and developing and making public a REST API to easily access data.

The educational applications of the LEWAS are expanding. In addition to the courses listed in this paper, the lab has multiple pending proposals at NSF which seek to use the LEWAS in other courses in biology, geosciences and engineering at Virginia Tech, and Virginia Western Community College. In addition, the LEWAS team is collaborating with K-12 teachers in the region to bring the LEWAS-based modules into their curricula. Through this expansion, the LEWAS has the potential to educate a wide range of students and increase their awareness of several interdisciplinary concepts such as rainfall-runoff process, environmental monitoring, sensor interfacing, data analysis, measurement errors, data management and visualization.

Beyond classroom applications, the LEWAS team is collaborating with their colleagues in India and Australia for extending and developing similar systems there, so that the students/faculty from these countries can learn about environmental monitoring issues at a different geographic region in a virtual environment. As the LEWAS team pursues its goal of providing remote access of the LEWAS data and the LEWAS field site to promote global education of environmental and other interdisciplinary concepts, it is essential that the LEWAS operates as an efficient and easily maintainable system. The development of the LEWAS into a robust and reliable system will help students/faculty not only from this university but also from various parts of the world to use the lab and its data for education and research. Lastly, this work demonstrates the interdisciplinary nature of engineering education research and hopefully sets a good example of how research in engineering education impacts the practice in engineering education.

9.0 Acknowledgement

The authors would like to thank various undergraduate and graduate students who assisted in the development and implementation of the LEWAS lab at Virginia Tech. We acknowledge the support of the National Science Foundation through NSF/REU Site Grant EEC-1359051 and NSF/TUES grant 1149467. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Bibliography

1. Delgoshaei, P., (2012). , PhD Dissertation submitted to Virginia Tech, Dec. 2012.
2. Delgoshaei, P., & Lohani, V. K. (2012). Implementation of a real-time water quality monitoring lab with applications in sustainability education. *Proc. 2012 Annual Conference of American Society for Engineering Design and Implementation of a Real-Time Environmental Monitoring Lab with Applications in Sustainability EducationEducation*, June 10 - 13, 2012, San Antonio, Texas.
3. McDonald, W. M., Dymond, R. L., Lohani, V. K., Brogan, D. S. and Basu, D., (2014a). Insights and Challenges in Developing a Remote Real-Time Watershed Monitoring Lab. *121st ASEE Annual Conference & Exposition*, Indianapolis, IN, USA, June 15-18, 2014.
4. Virginia Department of Environmental Quality, (2006). *Upper Stroubles Creek Watershed TMDL Implementation Plan Montgomery County, Virginia*, Blacksburg, VA, USA.
5. Clarke, H., Mcdonald, W., Raamanthan, H., Brogan, D., Lohani, V. K., and Dymond, R., (2013). Investigating the Response of a Small, Urban Watershed to Acute Toxicity Events via Real-Time Data Analysis. Proceedings of Research, NSF/REU Site on Interdisciplinary Water Sciences and Engineering, Virginia Tech.
6. Dymond, R., Lohani, V.K., Brogan, D., and Martinez, M., (2013). Integration of a Real-Time Water and Weather Monitoring System into a Hydrology Course. *Proc. 120th ASEE Annual Conference & Exposition*. Atlanta, GA.
7. McDonald, W., Brogan, D., Lohani, V.K., Dymond, R., Clark, R., (2014). Integrating a Real-Time Remote Watershed Monitoring Lab into Water Sustainability Education. *121st ASEE Annual Conference & Exposition*, Indianapolis, IN, USA, June 15-18, 2014.
8. Delgoshaei, P., & Lohani, V. (2014). Design and application of a real-time water quality monitoring lab in sustainability education.*International Journal of Engineering Education (IJEE)*, 30(2), 505-519.
9. McDonald, W. M., Brogan, D. S., Lohani, V. K., Dymond, R. L. and Clark, R. L., 2015. Integrating a Real-Time Environmental Monitoring Lab into University and Community College Courses. *International Journal of Engineering Education (IJEE)*. Accepted.
10. McDonald, W. M., Brogan, D. S., Lohani, V. K. and Dymond, R. L. (2015). Assessing Cognitive Development and Motivation with the Online Watershed Learning System (OWLS). *122nd ASEE Annual Conference & Exposition*, Seattle, WA, USA, June 14–17, 2015. Accepted.
11. Lohani, V.K., Delgohaei, P., Green, C., (2009). Integrating LabVIEW and Real-Time Monitoring into Engineering Instruction. *Proc. 2009 ASEE Annual Conference*. Austin, TX.
12. Delgoshaei, P., Lohani, V.K., and Green, C., (2010). Introducing Dataflow Programming in a Freshman Engineering Course with Applications in Sustainability Education. *Proc. 2010 ASEE Annual Conference*. Louisville, KY.
13. http://www.ni.com/compactrio/ accessed on March 27, 2015.
14. Rai, A., Brogan, D., and Lohani, V. K., 2013. A LabVIEW Driven Real-time Weather Monitoring System with an Interactive Database, Proceedings of Research, NSF/REU Site on Interdisciplinary Water Sciences and Engineering, Virginia Tech.
15. http://www.raspberrypi.org/ accessed on Feb1, 2015.
16. https://www.python.org/ accessed Feb1, 2015.
17. http://pubs.usgs.gov/of/2000/of00-458/pdf/ofr200-458toolboxmanualv4_508.pdf accessed on March 27, 2015.
18. http://s.campbellsci.com/documents/ca/manuals/series_5_man.pdf accessed on March 27, 2015.

19. http://www.vaisala.com/Vaisala%20Documents/User%20Guides%20and%20Quick%20Ref%20Guides/AWS330_Users_Guide_M211296EN-B.pdf accessed on March 27, 2015
20. Brogan, D. S., Lohani, V. K. and Dymond, R. L., (2014). Work-in-Progress: The Platform-Independent Remote Monitoring System (PIRMS) for Situating Users in the Field Virtually. *121st ASEE Annual Conference & Exposition*, Indianapolis, IN, USA, June 15-18, 2014.
21. http://www.raspbian.org/ accessed Feb1, 2015.
22. http://pyserial.sourceforge.net/ accessed Feb1, 2015.
23. http://dev.mysql.com/doc/ accessed on March 27, 2015
24. http://www.mysql.com/why-mysql/topreasons.html accessed on March 27, 2015
25. http://linux.die.net/man/8/picocom accessed on March 27, 2015
26. Horsburgh, J. S., Tarboton, D. G., Maidment, D. R., & Zaslavsky, I. (2008). A relational model for environmental and water resources data. *Water Resources Research*, 44(5), W05406. doi:10.1029/2007WR006392.
27. https://go.oracle.com/LP=6740?elqCampaignId=13283&src1=ad:pas:bi:dg:soa&src2=wwmk14054202mpp013&SC=sckw=WWMK14054202MPP013 accessed Feb 1, 2015.
28. http://www.publicapis.com/search?keyword=REST accessed April 3, 2015.
29. http://vwcclewas.blogspot.com/ accessed on Feb. 4, 2014.
30. https://blogs.lt.vt.edu/cee4304f2012 accessed on Feb. 4, 2014.