CONVERTING DATA TO INFORMATION: COUPLING LAB-LEVEL DATABASE FUNCTIONALITY WITH PRIMARY LTER DATA ARCHIVING SYSTEMS

Adam M. Kennedy[1], Suzanne M. Remillard[1], Donald L. Henshaw[2], Lawrence A. Duncan[3], Barbara J. Bond[1]

[1] Department of Forest Science, Oregon State University, Corvallis, OR 97331, USA.
[2] U.S. Forest Service, Pacific Northwest Research Station, Corvallis, OR 97331, USA
[3] Orion Imaging, http://orionimaging.net, Portland, OR, 97202, USA

Abstract

Developing and operating a data management program to support dynamic terrestrial and aquatic sensor networks is challenging. The database architecture needs to be robust and extensible, and must maintain flexibility in response to frequent changes in sensor array configurations in the field. The objective of this paper is to describe a database application developed for the Forest Ecophysiology and Ecohydrology Laboratory (FEEL) research program at the Andrews Experimental Forest in Oregon, USA. We discuss the fundamentals of a lab-level, web-based, and open source database application, and summarize the database architecture, methods of user-entered metadata, generation and storage of data mappings that provide the flexibility to handle changes in the incoming raw data streams, and methods to couple the lab-level database tables to the archival-level tables for seamless data flow and scheduled updating. This web-based database application enables small labs to handle large and streaming sensor arrays locally. The architecture is flexible and can adjust on-the-fly to changes in data file and field configurations. We detail a robust, user-friendly, and open source database environment that permits metadata generation and handling, low-level sensor tracking, dynamic data streams, general data processing, basic visualization, user-defined queries, and data routing to the primary long-term data repository.

Keywords: climate, sensor array, Andrews Experimental Forest, LTER, Forest Ecophysiology and Ecohydrology Lab, open architecture, environmental data

1. Introduction

Developing and operating near real-time terrestrial and aquatic sensor network data streams presents special challenges. The database architecture needs to be robust and extensible, and must maintain flexibility in response to frequent changes in sensor array configurations in the field. In competitively funded projects that are part of a larger umbrella network, such as the Long-Term Ecological Research (LTER) Network, the increased data stream poses significant challenges in that they are beyond the scope and intensity of what an information manager at an LTER site can accommodate. Most research labs lack the infrastructure and personnel to develop and or maintain a robust data management system. However, advances in data collection technology (Selker et al. 2006) and the associated increases in the volume of raw data streaming from the field, require new tools to handle these new data streams.

One solution is designing robust "lab-level" databases that are extensions of the primary Network database infrastructure. This type of hierarchical database schema provides users at the lab-level the flexibility to change sensor suites, add new sensors, and track more specific information relating to a sensor array apart from the primary database system, but still enables

easier subsequent integration to the Network-wide databases than if there were no prior coordination of schemas. Tools for these robust systems could include a secure multi-level and hierarchical user login system, functionality for user-entered metadata prior to sensor deployment, near real-time quality control of the data, basic visualization, and the ability for users to query the database for specific information without the need for expensive and memory intensive software installed on local or field computers. Such a lab-level system becomes critical to research success and longevity, and the ability to convert these raw data into useable information. Additionally, these systems can be designed to be tightly coupled with primary, long-term archiving systems, such as those represented within the distributed LTER sites nationally.

This paper focuses on the design and implementation of a lab-level system, which we defined here as the local lab domain operated at the lab-level and designed to have direct communication with the LTER site database domain. While it may generally operate at a small spatial scale, its contribution to the larger database domain remains integral to the success of the entire LTER site. It is at this level that the research programs funded by external grants operate and should maintain tight data relationships with the LTER site data managers.

This lab-level system consists of an online terrestrial database application that is currently deployed by the Forest Ecophysiology and Ecohydrology Lab (FEEL) at Oregon State University (http://oregonstate.edu/feel) and is being tested to handle near real-time data streams emerging from a small, steep-walled, forested watershed within the HJ Andrews Experimental Forest. The objectives of this paper are to summarize some of the techniques employed by this lab-level database ideology, and to serve as a building block for similar lab-level systems, as future sensor networks come online and create the need for solutions to growing data collections such as data storage, processing, and retrieval tools.

2. Methods

The climate and carbon research program within the FEEL increased in both data quantity and data type diversity during its four years of funding with the number of total continuous records approaching 50 million. This research program has nine plots, measures a full suite of environmental variables, and collects samples at intervals ranging from 1-minute to hourly. This example presents a clear need for robust applications to convert raw data points into useful information. The relational database selected for the FEEL application was MySQL (http://mysql.com), which provides a free, robust, relatively easy to use, and open source computing architecture. The platform permits advanced relational database functionality and can be coupled to other commercial database infrastructures using readily available MySQL ODBC drivers. The metadata entry component of this package was built using the PHP programming language with user authentication routed through online hypertext transfer protocol over a secure socket layer connection (HTTPS), which is used to indicate a secure HTTP connection. The PHP layer allows a user to avoid interacting with the database directly, and permits metadata entry and data processing to occur over a HTTPS connection through dynamic drop down lists and text boxes that insert the required data into the database in the proper format. Each piece of this application was developed to perform a specific task, but integrating the pieces has resulted in a robust application capable of managing a sensor network. While the database schema is not included in this paper, a summary of the FEEL application online metadata and data import procedures is shown in Figure 1.

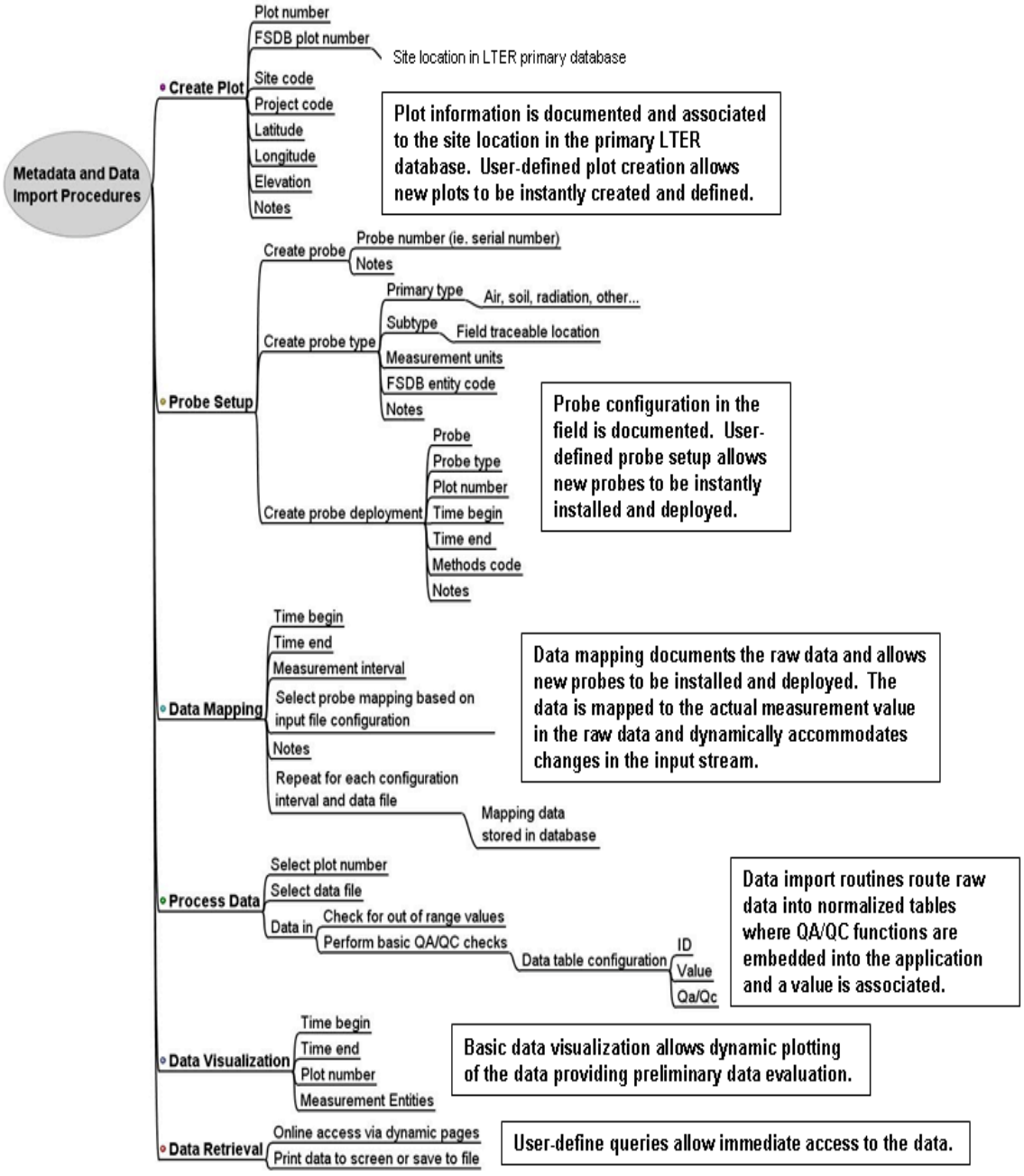

Figure 1. The FEEL database application was built to help manage detailed probe movement and calibration. The metadata and data import procedures and functionality include a secure user login system, user-entered metadata capabilities, near real-time quality control of the data, basic visualization, and the ability for users to query the database for specific information. The metadata entry component of this package allows the user to create plots, describe probe and probe deployment setup, and to handle dynamic field sensor configurations with on-the-fly data mapping setup tools. The metadata and data tables are linked such that the dynamic import of data streams accommodates changes in the input stream. Additionally, data visualization and retrieval functions allow immediate access to the data enabling preliminary data evaluation.

Defining valid and correct data mappings is an essential step of data processing, with the quality of the data mappings being critical to the quality of the extracted data. Our development

team has engineered a command line tool, *df_info* (a mnemonic for "data file information"), to assist in the initial assessment and analyses of raw data files, in preparation for defining data mappings. Df_info utilizes the PHP command line interface (CLI) (http://php-cli.com) framework to allow data files to be rapidly parsed, extracting all observation intervals that are present, while demarcating the line numbers on which changes in instrumentation occur. Once data mapping occurs, the data file is then ready to be processed with the FEEL application. Data mappings need only to be revised after a known configuration change has occurred.

The novel idea of data mapping allows for dynamic import of data streams into the application even when changes in the input stream exist mid-file. We tested the accuracy of this system by computing the fraction of import success over import failure. Import success rates are logged and available for retrospective analysis. On-the-fly plotting functionality is in place for preliminary data evaluation. The plotting module employs an off-the-shelf plotting application (http://jpgraph.com) coupled to the FEEL application and, as of this publication, is used only for preliminary evaluation or broad data quality checks. To connect the FEEL lab-level database to the primary LTER site SQL Server database, we employed Microsoft's SQL Server Integration Services (SSIS), which is embedded in SQL Server 2005 Management Suite (Figure 2). This tight coupling to the LTER site database at the lab-level promotes semi-automatic data warehousing and reduces the time required to continually update continuous sensor array data from the field.

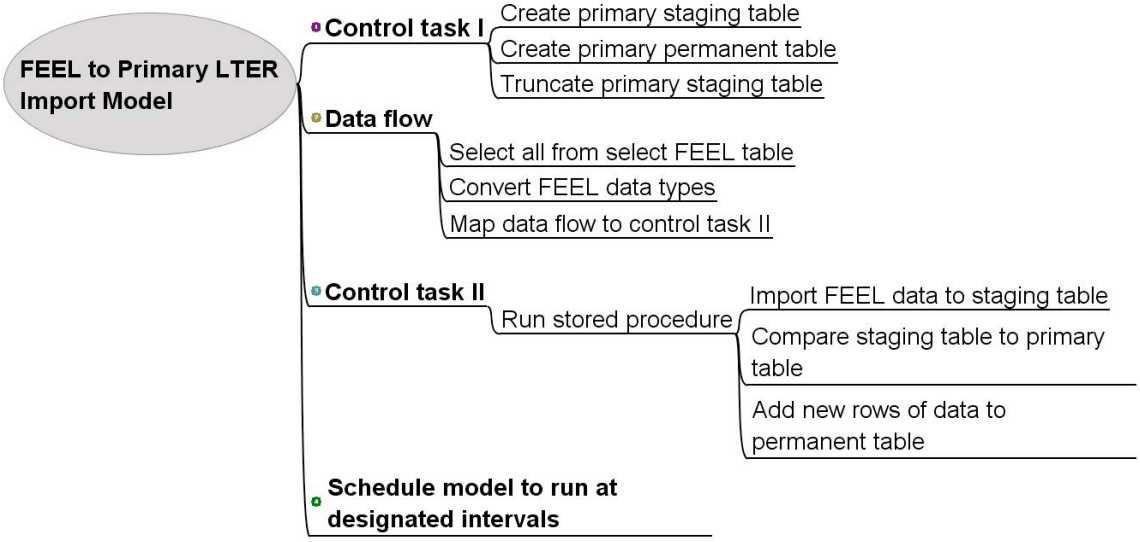

Figure 2. The FEEL database uses an SQL Server Integration Services algorithm to execute communication to the primary LTER database for long-term data archiving.

3. Results

Implementing the success of this system required several months of intensive but intermittent focus. The resulting database application achieves our objectives by handling the FEEL data needs; metadata (including sensor calibration notes, replacement, type, serial number, units, dates deployed, etc.) are accessible, individual sensors may be tracked in the field, input

file configurations can be appropriately mapped so that the database import features will know how to handle each record, and basic QA/QC functions are embedded into the application.

Attention was given to processing speed, long-term storage, writing optimized queries to insert and route data to their appropriate tables, and to return data to users via online data access portals. The user-entered metadata are stored in tables within the FEEL database and metadata are available through user-defined queries via a web portal. Data import functions route the raw data into normalized tables and a QA/QC field is associated with each record.

A general QA/QC model was developed for timestamp and min/max range checking as a function of month within the database environment that leaves data as is, but flags each questionable or missing value with an assigned value. Evaluating the data as a function of time and value instead of an absolute value tightens the band into which good values must exist during the QA/QC procedure, raw data are left joined to a "calendar" table so that all timestamps are included and missing raw records are inserted as null values.

Data mappings assigned to each raw input file result in an import success rate approaching 99.9%. When import failures occur, they are generally due to a temporal change in sensor configuration, or a duplicate timestamp due to data logger failures. While these count as "import failures", the former is isolated to the record where the configuration change occurred, and the latter is considered advantageous, in that our automated import routine will not import duplicate timestamps, thus preserving the integrity of our database. Files with no changes in sensor configurations typically have a 100% import success rate. More detailed error analysis will be performed in later releases of the FEEL database.

The FEEL application is flexible in that little additional programming is required to add new database fields as metadata needs evolve and portable in that it will run on a "localhost" or remote server with proper configuration. For our localhost environment, we currently employ XAMPP, which is an easy to install Apache distribution containing MySQL, PHP, and Perl (http://www.apachefriends.org). A complete installation using documentation averages less than one day. If the application were to be used by another lab, the main requirement is that the input files need to be comma delimited, and the first five fields must include (in this order) – array number (which describes the sampling interval), site id, year, day of year, hour/minute – with the remaining fields being the observed variables.

4. Discussion

Lab-level tools need to maintain the strict metadata protocol built into the primary LTER database, but also embrace the abundance of raw data streaming from the field and have the ability to convert the raw data streams into a complete set of information used to answer questions from all levels of students, the research program personnel, and policy makers. While the LTER network and many LTER sites have developed integration tools and techniques for core datasets at the inter-site level, like ClimDB/HydroDB (http://www.fsl.orst.edu/climhy), and the site level (GCE (Sheldon 2003)), lab-level components, derived from the recent onset of data rich and externally funded research programs, are missing from the LTER tool suite.

New products, both commercial and open source, such as DataTurbine (http://dataturbine.org), Antelope (http://brtt.com), SensorBase (http://sensorbase.org), and GSN (http://gsn.sourceforge.net/) are being developed to provide means to handle large amounts of streaming data. Commercial products were not considered financially feasible for the FEEL lab and most of the open source applications were still in development when the lab-level application was being developed. The FEEL lab decided to develop a custom tool with the intention of creating a congruent system that could be easily adapted by other labs. An

immediate advantage of this application is its ability to integrate directly with the existing LTER site database. Improved interoperability between the major data management tools and existing data management programs could improve the effectiveness of these tools.

There are two fundamental ideas presented in this paper that make tools like the FEEL database necessary. First, management of sensor array data is critical for maintaining and assuring data quality in near real-time. Preserving low-level metadata regarding collected attributes, array configurations, probe placement and sensor calibration is critical to track field sensor array history in long-term studies. Raw data streams and low-level metadata must also be available in a useable format to researchers for regular quality checks and sensor calibration history. Secondly, lab-level databases at the LTER sites can accommodate intensive studies and manage associated sensor arrays that are outside the scope of the primary information management system, but still integrate final data into the system

As presented in this paper, it is necessary that the lab-level system be managed independently from the LTER primary database but must still function in conjunction with the long-term data archive to gain benefits from that system such as metadata-driven data validation, compliance with network-wide metadata standards, generation of Ecological Metadata Language, and participation in network-wide databases.

Acknowledgements

We would like to acknowledge the Orion Imaging programming staff for their consultation and interest in this project and the system administrators at Oregon State University Central Web Services and the OSU College of Forestry. Gabriel Shea of Idea Pivot provided the initial expertise needed to complete the FEEL to LTER database coupling. Finally, we would like to thank H. Barnard, N. Czarnomski, J. Gabrielli, R. Hopson, Z. Kayler, C. Phillips, T. Pypker, S. SanRamoni, B. Wilson, and E. Wyckoff who have helped test the FEEL application during the implementation of the FEEL system. Funding for this work was provided by the NSF "A Wireless Network of Battery-Free Sensors for Atmosphere-Biosphere Studies in Complex Environments" grant (DBI0529223) and the NSF Andrews LTER grant (DEB0218088).

References

Selker, J.S., L. Thévenaz, H. Huwald, A.Mallet, W. Luxemburg, N. van de Giesen, M. Stejskal, J. Zeman, M. Westhoff, 2006. Distributed fiber-optic temperature sensing for hydrologic systems. Water Resources Research 42: W12202 1 – 8. doi: 10.1029/2006WR005326.

Sheldon, W.M., 2003. Presentation: Software tools for automated metadata creation, metadata-mediated data processing and quality control analysis – real time processing solutions for real-time data. 2003 LTER All-Scientists Meeting. Sept. 18-21, Seattle, WA.

Proceedings of the Environmental Information Management Conference 2008 (EIM 2008)

September 10-11, 2008
Albuquerque, NM

Editors:
Corinna Gries
Central Arizona-Phoenix LTER
Arizona State University

Matthew B. Jones
National Center for Ecological Analysis and Synthesis (NCEAS)
UC Santa Barbara

Copyright