International Environmental Modelling and Software Society (iEMSs)
7th Intl. Congress on Env. Modelling and Software, San Diego, CA, USA,
Daniel P. Ames, Nigel W.T. Quinn and Andrea E. Rizzoli (Eds.)
http://www.iemss.org/society/index.php/iemss-2014-proceedings

Automating Data Management and Sharing within a Large-Scale, Heterogeneous Sensor Network

Jeffery S. Horsburgh[a], Amber Spackman Jones[b], Stephanie Reeder[c]

[a] Utah Water Research Laboratory, Utah State University, jeff.horsburgh@usu.edu; [b] Utah Water Research Laboratory, Utah State University, amber.jones@.usu.edu; [c] Utah Water Research Laboratory, Utah State University, stephanie.reeder@usu.edu

Abstract: Hydrology researchers are collecting data using *in situ* sensors at high frequencies, for extended durations, and with spatial distributions that require infrastructure for data storage, management, and sharing. Managing streaming sensor data is challenging, especially in large networks with large numbers of sites and sensors. The availability and utility of these data in addressing scientific questions related to water availability, water quality, and natural disasters relies on effective cyberinfrastructure that facilitates transformation of raw sensor data into usable data products. It also depends on the ability of researchers to share and access the data in useable formats. In this paper we describe tools that have been developed for research groups and sites conducting long term monitoring using *in situ* sensors. Functionality includes the ability to track equipment, deployments, calibrations, and other events related to monitoring site maintenance and to link this information to the observational data that they are collecting, which is imperative in ensuring the quality of sensor-based data products. We present these tools in the context of a data management and publication workflow case study for the iUTAH (innovative Urban Transitions and Aridregion Hydrosustainability) network of aquatic and terrestrial sensors. The iUTAH monitoring network includes sensors at aquatic and terrestrial sites for real-time monitoring of common meteorological variables, snow accumulation and melt, soil moisture, surface water flow, and surface water quality. We present the overall workflow we have developed and new software tools that we have deployed for both managing the sensor infrastructure and for storing, managing, and sharing the sensor data.

Keywords: Sensor network; Data management; Data sharing; Hydrologic information system.

1. INTRODUCTION

Hydrologic monitoring with *in situ* environmental sensors presents many challenges for data management, particularly for large-scale networks consisting of multiple sites, sensors, and personnel. The high frequency, extended duration, and spatial distribution of data collection efforts require cyberinfrastructure to support and facilitate research. Researchers and practitioners need tools for data import and storage as well as data access and management. In addition to addressing the challenges presented by the sheer quantity of data, monitoring networks need practices to ensure high data quality, including procedures and tools for post processing. Data quality is further enhanced if networks are able to track physical infrastructure such as equipment, deployments, calibrations, and other events related to site maintenance and associate these details with observational data. In this paper we present a case study of a workflow for streaming sensor data for the iUTAH (innovative Urban Transitions and Aridregion Hydrosustainability) ecohydrologic observatory. The iUTAH monitoring network consists of aquatic and climate sensors deployed in three Utah watersheds to monitor Gradients Along Mountain to Urban Transitions (GAMUT). The variety of environmental sensors and the multi-watershed, multi-institutional nature of the network necessitate a well-planned and efficient workflow for acquiring, managing, and sharing sensor data. We present the overall workflow that we have developed for GAMUT data management, the software tools that we have developed and deployed, and aspects of the data quality assurance and quality control plan that have been implemented. Features of the workflow include a data model and web interface for managing

sensor infrastructure, Python-based tools for performing quality control post-processing, and web-based applications providing access and visualization of the data. The tools presented will be useful for similar large-scale and long term monitoring networks.

2. THE iUTAH GAMUT NETWORK

iUTAH researchers have developed and deployed an ecohydrologic observatory to monitor Gradients Along Mountain to Urban Transitions (GAMUT). The GAMUT Network measures aspects of water inputs, outputs, and quality along a mountain-to-urban gradient in three watersheds that share common water sources (winter-derived precipitation) but differ in the human and biophysical nature of land-use transitions. GAMUT includes sensors at aquatic and terrestrial sites for continuous, real-time monitoring of common meteorological variables, snow accumulation and melt, soil moisture, surface water flow, and surface water quality.

The GAMUT network consists of sites within three watersheds in northern Utah, USA: the Logan River, Red Butte Creek, and the Provo River. The monitoring infrastructure within each watershed was built by separate universities (i.e., Utah State University, University of Utah, and Brigham Young University), and each university employs a full time watershed technician to manage the infrastructure. The multi-watershed, multi-institution nature of the GAMUT network presents challenges related to tracking the physical monitoring infrastructure (e.g., sensors, dataloggers, etc.), managing the data in a consistent way across the three watersheds (including quality assurance and quality control), and combining the equipment management with the data management to ensure that essential information about physical infrastructure is linked to the data streams from each monitoring site. For example, it is important to record the dates on which sensors were deployed, calibrated, serviced, replaced, etc. as these activities may affect the data recorded by the sensors.

3. THE GAMUT DATA WORKFLOW

Management of the GAMUT data is centralized at Utah State University, where the data from all of the remote sites are aggregated, loaded into operational databases, and shared on the Internet via a suite of web applications. Figure 1 shows the overall data management workflow for GAMUT. Each of the components is described in more detail in the following sections.

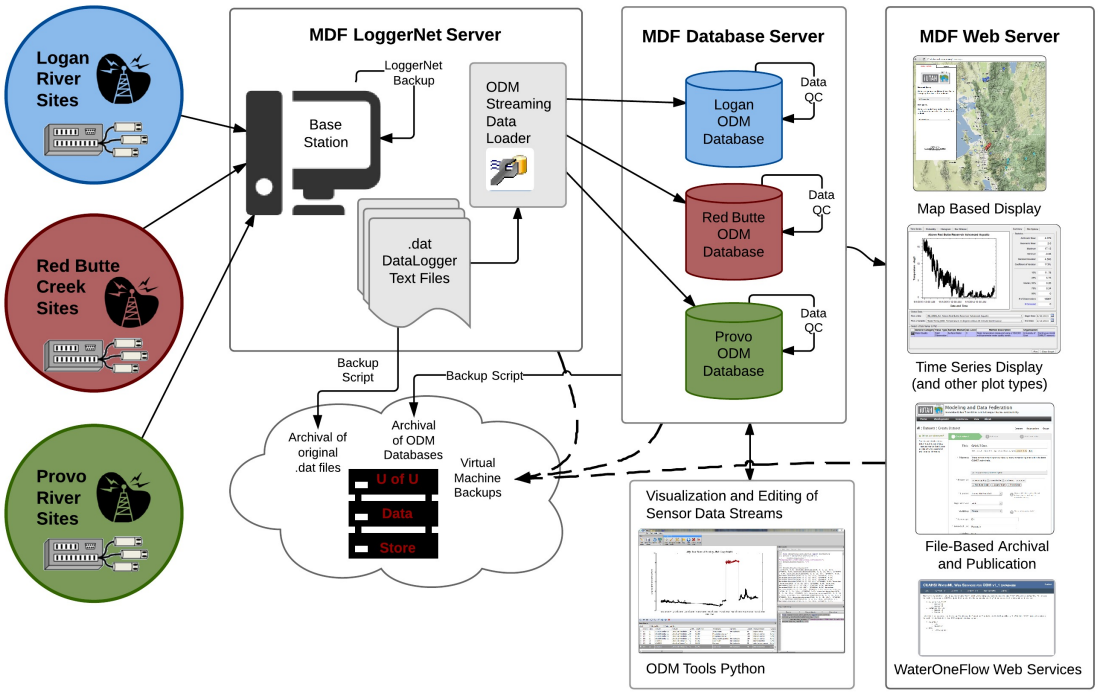

Figure 1. The GAMUT data management workflow.

3.1 Monitoring Site Design and Communications

Each of the sites in the GAMUT network includes a suite of sensors that are connected to a Campbell Scientific, Inc. datalogger. The datalogger provides programming logic to operate the site as well as onsite data storage. GAMUT uses a variety of telemetry connections, including radio frequency, cellular, and TCP/IP communications. A single instance of Campbell Scientific's Loggernet software was deployed to enable automated communication with each site, scheduled download of data, functionality to send new programs and instructions to the site, and a variety of communication and data collection diagnostic utilities. Using Loggernet, data are downloaded from each site on a regular schedule and are stored in comma-separated values (.dat) files on the Loggernet server (labelled "Base Station" in Figure 1.

3.2 Server Infrastructure

The GAMUT data management workflow is spread across three virtual servers, each of which is running the Microsoft Windows Server 2008 R2 operating system. The first runs Campbell Scientific's Loggernet software and manages communication with and download of data from each monitoring site. The second server runs Microsoft SQL Server and hosts the operational databases into which the streaming sensor data are loaded upon download. The third is a web server on which several web applications are hosted for sharing the iUTAH data on the Internet. Although all of these software programs could be run on a single server, we have chosen to separate them into three separate virtual machines for security purposes. The Loggernet and database servers can be protected behind firewalls with very little external exposure to the Internet, whereas the web server provides unrestricted access to the data via web application interfaces (described below).

3.3 Streaming Data Loading

The data within the Loggernet text files (.dat files) downloaded from each of the monitoring sites are loaded into operational databases using the Streaming Data Loader installed on the Loggernet server. The Streaming Data Loader is a component of the HydroServer software stack (Horsburgh et al., 2010) developed as part of the Consortium of Universities for the Advancement of Hydrologic Science, Inc. (CUAHSI) Hydrologic Information System (HIS) (Horsburgh et al., 2009; Tarboton et al., 2009). The Streaming Data Loader can be configured to load any number of table-based datalogger files into a relational database that implements the CUAHSI HIS Observations Data Model (ODM) (Horsburgh et al., 2008). The Streaming Data Loader opens each datalogger text file, checks the latest date for which data were collected, compares the latest data in the file with the latest date in the ODM database and then loads any new data from the file into the database. The Streaming Data Loader can be configured to run with any frequency as a Windows Task using the Windows Task Scheduler.

3.4 Operational Databases and Data Quality Assurance

As depicted in Figure 1, three ODM instances were created to store the streaming sensor data, one for each watershed. The choice to create three databases instead of one larger database was somewhat arbitrary; however, it does have the benefit of providing an additional level of granularity on which to assign access control. For example, we could choose to give each watershed technician access only to the ODM database that stores the data for the watershed that they manage.

The databases are implemented on a virtual machine running the Microsoft SQL Server 2012 relational database management system (RDBMS). The ODM databases provide transactional access to the data using Structured Query Language (SQL), which facilitates both data quality assurance and quality control and publication of the data on the Internet (detailed below).

In addition to the field protocols specified in the GAMUT Quality Assurance/Quality Control Plan (e.g., site visits, calibrations, maintenance, etc.), we have developed automated data checks as stored procedures within the ODM databases that regularly scan the incoming data for anomalies or

problems. For example, new data are screened for battery voltages below acceptable thresholds, values that are outside acceptable ranges for each variable, values that are persistent, values that undergo unrealistic changes over small periods of time (e.g., where the difference from one value to the next is greater than a threshold set for a particular variable), etc. When problematic conditions are detected, the stored procedures send email alerts to the appropriate watershed technicians, notifying them of conditions that need to be investigated.

3.5 Data Quality Control

We have developed a new, Python-based software program to enable the iUTAH watershed technicians and others to visualize and perform quality control editing of the streaming sensor data. ODM Tools Python is an open source software application that allows ODM users to query and export, visualize, and edit data stored in an ODM database. It connects directly to an ODM database within a RDBMS. ODM Tools Python was built using wxPython for user interface design and implementation (http://www.wxpython.org/) and matplotlib for graphics (http://matplotlib.org/). It uses a SQLAlchemy-based database abstraction layer (http://www.sqlalchemy.org/) and additional Python modules as detailed on the ODM Tools Python GitHUB page (https://github.com/UCHIC/ODMToolsPython). Figure 2 shows the architecture for the ODM Tools Python software application.

Previous versions of ODM Tools were developed in Microsoft Visual Studio .NET and included functionality to export data series and associated metadata, plot and summarize single data series, generate derivative data series, and edit data series using a set of simple tools. The new, Python-based version of ODM Tools adds a modernized graphical user interface (GUI), multiple platform support (Windows, Linux, and Mac), multiple RDBMS support (Microsoft SQL Server and MySQL), enhanced plotting and visualization, and automated scripting of quality control edits performed on data series through an integrated Python script editor and console. Integration of the Python scripting tools, multiple platform support, and multiple RDBMS support, which would have been very difficult to accomplish developing in Microsoft Visual Studio .Net, were all drivers for the decision to develop a new version of ODM Tools in Python. Additional improvements include customizable queries for data selection and export, the ability to plot multiple data series simultaneously with various plot types, and user-defined functions for data series editing and derivation.

A major goal in the development of ODM Tools Python was to add the capability to track all changes made to a time series during the quality control process. The new ODM Tools scripting interface records the user's actions in the GUI as they perform corrections and adjustments on data series in the quality control process. Each button click on the GUI executes a data editing function and fires a line of code to the Python script editor (Figure 3). This records the sequence of edits in a Python script that can be saved as a file, ensuring that the editing steps are traceable and reproducible. All edits are made on copies of the raw data in order to preserve the original data.

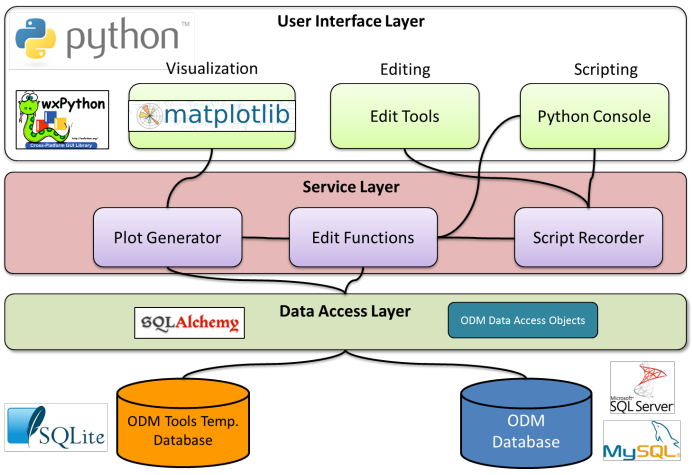

Figure 2. Architecture of the ODM Tools Python software.

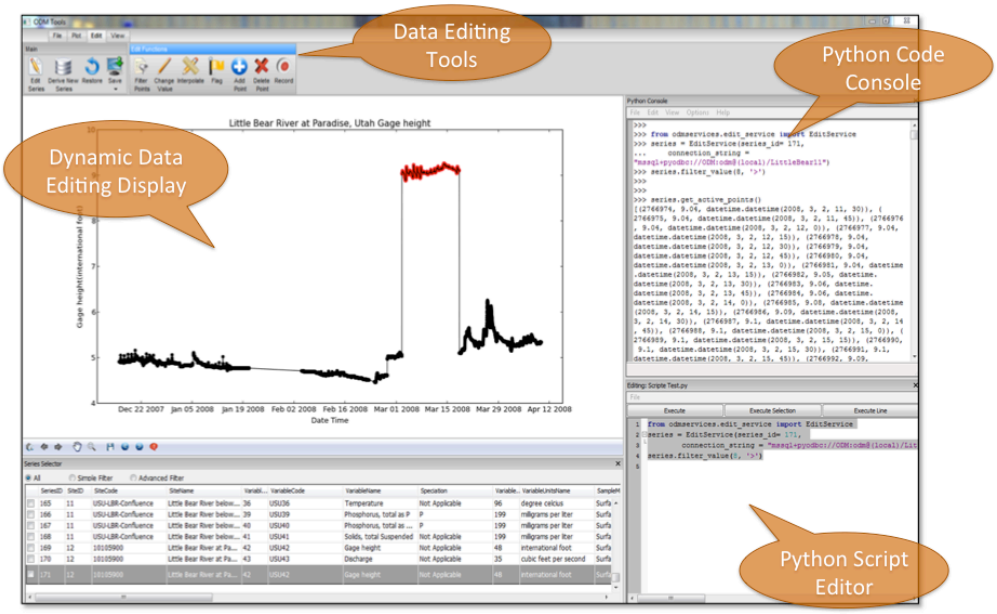

Figure 3. Screen shot of the ODM Tools Python graphical user interface showing the Python script editor and Python console.

3.6 Backups

As shown in Figure 1, there are multiple levels of backups within the GAMUT data management workflow. First, there is local data storage on the datalogger at each monitoring site. This ensures that if our centralized data management system goes offline, data collection will continue uninterrupted at each site. Next, Loggernet has a backup utility that is used to backup the configuration of the Loggernet network on a weekly basis. This ensures that, in the event of a failure of the Loggernet Server, we could start a new Loggernet Server at any time, import the network backup, and have the system running within minutes.

We regularly backup and archive the individual text files that are downloaded from each of the remote monitoring sites. This occurs daily for the active data files from each site. Additionally, any time a datalogger program is changed at a site, a new data file is initiated and the old data file is copied to an archive folder that is also backed up. This ensures that the original data files will always be available exactly as they were downloaded from the sites, providing an archival record of the data as downloaded. These files could be used at any time to reconstruct the full data record at each site.

The operational SQL Server databases are scheduled for weekly, file-based backups. This ensures that we could easily recover any of the operational databases from the file-based backup. Any data gap that may occur due to a database failure could be remedied without data loss by reloading any data collected after the last database backup from the original data files. Finally, each of the three virtual servers on which the system runs are scheduled for daily incremental backups and weekly full backups using the backup capabilities of the virtualization software. This ensures that if our physical hardware failed, we could move the virtual machine backups to a different physical host with little gap in services and no data loss. All of the backups described above are copied to an offsite storage resource at the University of Utah, insuring against a catastrophic event at the Utah State University Enterprise Data Center.

3.7 Data Sharing on the Internet

The data stored in the operational ODM databases are shared on the Internet using multiple mechanisms. First, the data are published using the CUAHSI HIS WaterOneFlow web services. WaterOneFlow web services connect directly to an ODM database and deliver data in WaterML

format in response to web service requests. The WaterOneFlow web services make the data accessible using the CUAHSI HIS HydroDesktop client application (Ames et al., 2012) – i.e., users can download the data directly into HydroDesktop via the WaterOneFlow web service. Additionally, we developed a Google Map based web application that shows the locations of the monitoring sites and links to a time series visualization tool that provides simplified access to the data through a web browser (Figure 4).

3.8 Equipment Management

A step that is often overlooked in the design of large-scale sensor networks is management of the inventory of data collection equipment (e.g., sensors, dataloggers, solar panels, etc.). This includes information about which equipment has been deployed at each monitoring site, but also includes a record of field activities and service events, including sensor calibrations, as well as routine factory service for sensors and dataloggers. This information is typically recorded in field notes or files, but is rarely linked directly to the data being collected. Yet, there are many scenarios where performing quality control of the data or even eventual interpretation of the data in analyses require consulting the record of field activities.

We developed a database for storing this type of information and a web application interface for the database that enables the iUTAH watershed technicians to use a web browser to enter information about equipment, where it is deployed, and field activities that have been performed. This database serves as a digital record of the deployment and maintenance activities that have occurred. The equipment management database was designed such that it extends the ODM database in which the observational data are stored. In this way, the equipment management information can be directly linked to the observational data. While the equipment management information is not currently viewable in the ODM Tools Python software, the equipment management web application enables the watershed technicians to query and view the equipment management information simultaneously while performing data quality control using ODM Tools Python. The schema for the current version of the equipment management database is shown in Figure 5, and Figure 6 shows screen shots of the equipment management web application, which manages the information stored in the database.

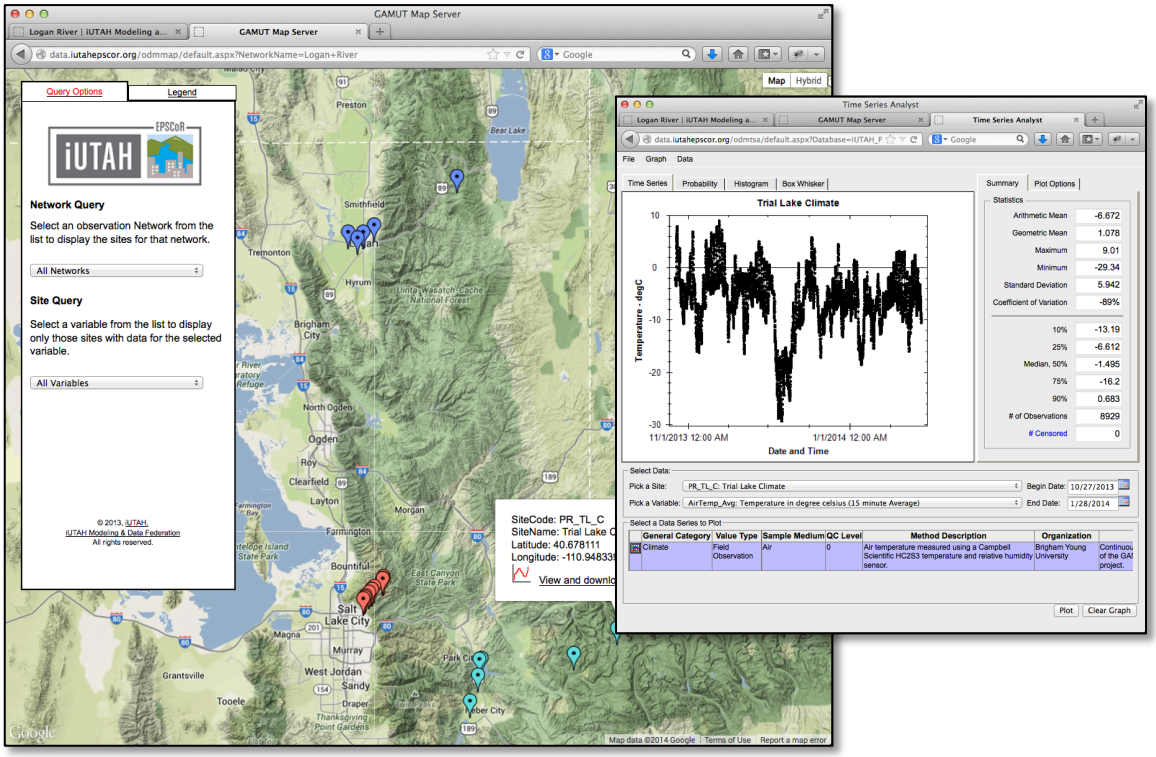

Figure 4. Google Map application and Time Series Analyst data visualization tool for sharing GAMUT sensor data.

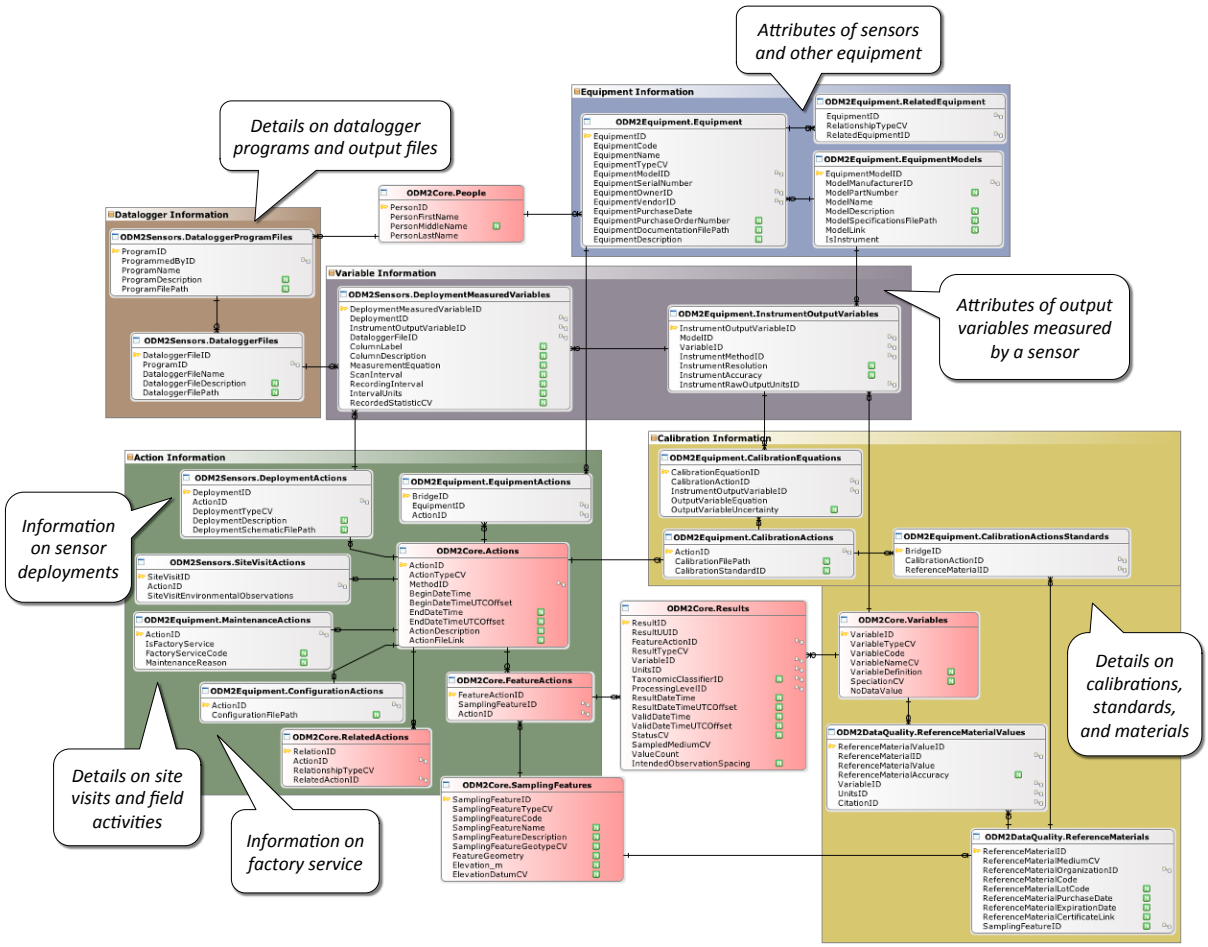

Figure 5. Relational schema for the equipment management database.

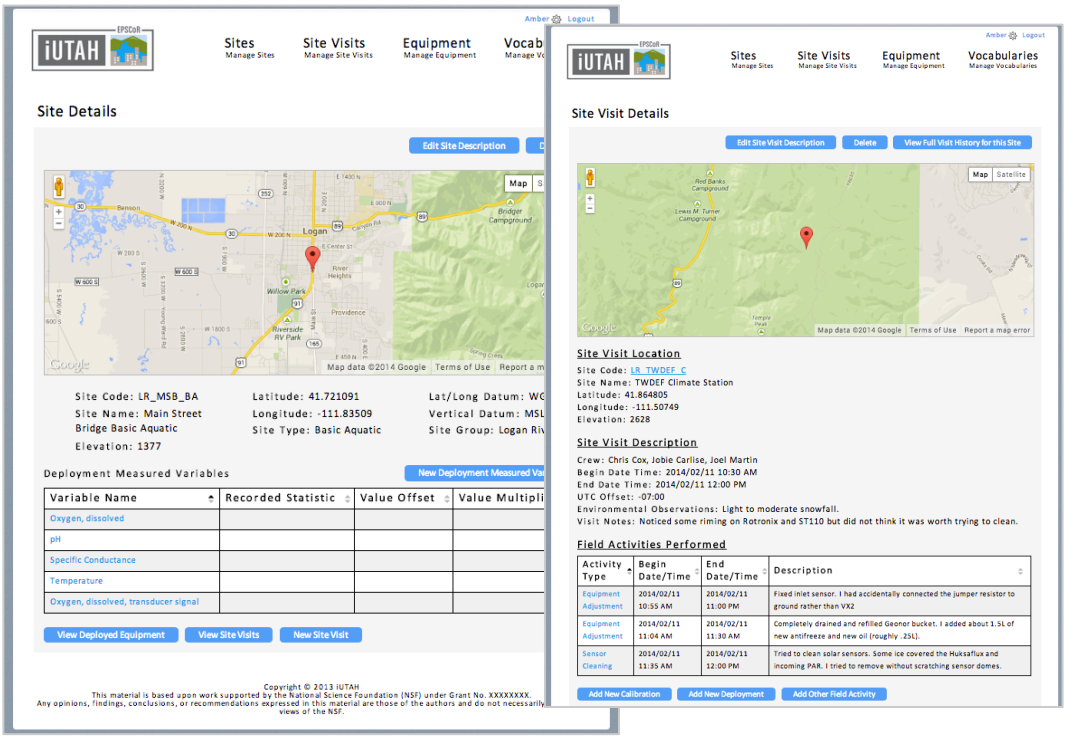

Figure 6. Screen shots of the GAMUT equipment management web application.

4. CONCLUSIONS AND RECOMMENDATIONS

It is becoming more common for researchers to collect data using *in situ* sensors at multiple sites, with high frequencies, and for extended durations. In this paper we have described a workflow and cyberinfrastructure that meet the data storage, management, and sharing needs of a large, multi-watershed, multi-institution sensor network. The workflow and cyberinfrastructure described here standardize data management across all three watersheds and institutions. They address the often-overlooked aspects of protection and backup of the operational sensor data and supporting systems, as well as formally storing and managing information related to monitoring equipment and its use in data collection activities. They also address many aspects of quality assurance and quality control of the data with specific tools for notifying data managers of potentially problematic conditions and software tools for performing data quality control in a repeatable way.

Finally, they integrate with the CUAHSI HIS and make the data accessible on the Internet using relatively simple data visualization and download tools. We anticipate that the workflow and tools we have developed will potentially be useful for other research groups developing new data collection, management, and sharing systems. Future work will be focused on adding additional data editing tools and algorithms to the ODM Tools Python software as well as new enhancements to the web-based data visualization tools.

5. SOFTWARE AVAILABILITY

Most of the software programs described in this paper are available in open source code repositories. The CUAHSI HIS HydroServer software stack, including ODM, the Streaming Data Loader, the WaterOneFlow web services, and the Time Series Analyst, are available via the HydroServer Codeplex website and code repository (http://hydroserver.codeplex.com). The ODM Tools Python application and is available on GitHUB (https://github.com/UCHIC/ODMToolsPython). For information on any other software or code described, contact the authors.

6. ACKNOWLEDGMENTS

This was funded by the National Science Foundation through grant EPS 1208732. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

7. REFERENCES

Ames, D.P., Horsburgh, J.S., Cao, Y., Kadlec, J., Whiteaker, T., Valentine, D., 2012. HydroDesktop: Web services-based software for hydrologic data discovery, download, visualization, and analysis, Environmental Modelling & Software, 37, 146-156, http://dx.doi.org/10.1016/j.envsoft.2012.03.013.

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, W05406, http://dx.doi.org/10.1029/2007WR006392.

Horsburgh, J.S., Tarboton, D.G., Piasecki, M., Maidment, D.R., Zaslavsky, I., Valentine, D., Whitenack, T., 2009. An integrated system for publishing environmental observations data, Environmental Modeling and Software, 24, 879-888, http://dx.doi.org/10.1016/j.envsoft.2009.01.002.

Horsburgh, J.S., Tarboton, D.G., Schreuders, K.A.T, Maidment, D.R., Zaslavsky, I., Valentine, D., 2010. HydroServer: A platform for publishing space-time hydrologic datasets. In: Proceedings of the AWRA Spring Specialty Conference on GIS and Water Resources, Orlando, FL, March 29 – 31.

Tarboton, D.G., Horsburgh, J.S., Maidment, D.R., Whiteaker, T., Zaslavsky, I., Piasecki, M., Goodall, J., Valentine, D., Whitenack, T., 2009. Development of a community Hydrologic Information System. In: Anderssen, R. S., R. D. Braddock, and L.T.H. Newham (eds.) 18th World IMACS Congress and MODSIM09 International Congress on Modelling and Simulation, Modelling and Simulation Society of Australia and New Zealand and International Association for Mathematics and Computers in Simulation, July 2009, pp. 988-994, ISBN: 978-0-9758400-7-8.