Best practices for sensor networks and sensor data management

Citation

ESIP EnviroSensing Cluster (2014). Community Wiki Document, “Best practices for sensor networks and sensor data management”, Federation of Earth Science Information Partners. http://wiki.esipfed.org/index.php/EnviroSensing_Cluster. (wiki document accessed 12-1-2014).

Contributors (as of December 2014)

Each chapter has a lead editor who is responsible for periodically compiling comments and contributions into stable versions of this document which will be archived as PDF versions and can be found here. If you contribute to this document by editing or adding text, images or comments you agree to the use of that material in the regularly published PDF versions of this document. Please add your name to the list of contributors if you feel you made a significant contribution.

- Corinna Gries, University of Wisconsin-Madison, North Temperate Lakes LTER
- Don Henshaw, USFS Pacific Northwest Research Station, Andrews Forest LTER
- Scotty Strachan, University of Nevada, Reno
- Renee F. Brown, University of New Mexico, Sevilleta LTER & UNM Sevilleta Field Station
- Christopher Jones, National Center for Ecological Analysis and Synthesis
- Christine Laney, University of Texas at El Paso, Jornada Basin LTER
- Branko Zdravkovic, University of Saskatchewan
- Richard Cary, University of Georgia, Coweeta LTER
- Jason Downing, University of Alaska Fairbanks, Bonanza Creek LTER
- Adam Kennedy, Oregon State University, Andrews Forest LTER
- Mary Martin, University of New Hampshire, Hubbard Brook LTER
- Jennifer Morse, University of Colorado Boulder, Niwot Ridge LTER
- Fox Peterson, Oregon State University, Andrews Forest LTER
- John Porter, University of Virginia, Virginia Coast Reserve LTER
- Jordan Read, US Geological Survey
- Andrew Rettig, University of Cincinnati
- Wade Sheldon, University of Georgia,, Georgia Coastal Ecosystems LTER

Scope

This document on best practices for sensor networks and sensor data management provides information for establishing and managing a fixed environmental sensor network for on or near surface point measurements with the purpose of long-term or “permanent” environmental data acquisition. It does not cover remotely sensed data (satellite imagery, aerial photography, etc.), although a few marginal cases where this distinction is not entirely clear are discussed, e.g., phenology and animal behavior webcams. The best practices covered in this document may not all apply to temporary or transitory sensing efforts such as distributed “citizen science” initiatives, which do not focus on building infrastructure. Furthermore, it is assumed that the scientific goals for establishing a sensor network are thought out and discussed with all members of the team responsible for establishing and maintaining the sensor network. i.e., appropriateness of certain sensors or installations to answer specific questions is not discussed. Information is provided here for various stages of establishing and maintaining an environmental sensor network: planning a completely new system, upgrading an existing system, improving streaming data management, and archiving data.

Below are chapters of a living document to which contributions can be made by anybody interested in this subject. Please post questions, answers, experiences with particular software/hardware/setup, comments, additions, edits, resources, and publications. Please use common online etiquette. If conflicting views arise they should be discussed in the EnviroSensing e-mail list.

back to EnviroSensing Cluster main page

Contents

Planning Process

Throughout this document it is emphasized that the initial planning is extremely important, as well as the inclusion of expertise in many different areas, scientific and technical, in the early discussion and planning phase before a proposal is written. If all fields of expertise are not consulted/incorporated prior to making location, budget, deployment, and timeline decisions, critical interdependencies are likely to be overlooked (e.g. power requirements, topographic constraints, construction tools required, etc.).

Although, the discussion here is geared toward maintaining sensor networks over an extended period of time, planning is equally important for short term installations. Experience has shown that many short term installations have become long term even if that was not intended initially and many small installations have been expanded to cover more area or measure more parameters.

Clearly, sensor network deployments are driven by ambitious science questions. However, good planning can help anticipate limitations and prevent time issues from becoming the driving force. Focusing on the overarching imperatives of good design, proper placement, organized data flow, and a well trained and motivated team, will result in successful implementation and continued maintenance. Compromised installations diminish the impacts of the original study, can drain operating budgets unnecessarily, and inhibit leveraging of the science for future work and funding.

Implementation Feasibility

During the experiment or project design phase, defining the primary measurement objective is the first step to planning an observation site and platform. Answering these general questions is helpful before addressing specific technical issues:

- Where is the geographic area of interest?
- What are the measurements of interest?
- What is the desired accuracy and frequency of measurement?
- How critical are the sensor measurements? Can data gaps be tolerated? Is sensor redundancy necessary?
- What type of experimental manipulation is desired (if any)?
- What types of localized topography are likely to yield “representative” measurements at the time frequency of interest?
- What is the total funding amount for personnel, travel, tools/equipment, fees, and science instruments?
- What is the expected scope/lifetime of the deployment? Will it be expanded in the future? Consider scaling possibility (more sites, more sensors) even if it is not the immediate goal.
- Evaluate commercial turnkey installations vs. systems developed from commercial or open source components. Considerations: cost, skills, maintenance, longevity of the company providing the whole system or each component, functionality, interoperability, access to continued support.

Assembling the Team

Several very different areas of expertise are required to successfully plan, install, and maintain sensing systems. Some of these roles/skill sets can certainly be provided by a single individual or individuals.

Roles within a team establishing and maintaining an environmental sensor network:

- **Scientist** - determines the type of data and sampling frequency needed to answer the scientific questions within budget limitations.
- **Sensor system expert** - knows the types of sensors and platforms, their installation and programming needed to answer scientific questions. Is familiar with specific climate and terrain issues and QA/QC approaches in the field.
- **Field logistics expert** (for major site construction) - familiarity with transport, construction, weather, tools, and supplies for construction
- **Field construction and fabrication expert** - understands concrete, metal structure, tower design, fencing, underwater anchors, floating devices, load estimates
- **Field workers/assistants** - many people are needed for remote construction tasks, sensor wiring, initial site setup, cable management
- **Field technician** - familiar with maintenance tasks including minor repairs, maintaining a calibration schedule, other regular sensor maintenance tasks. Field technicians need to have a good understanding of the science application and the end user, they need to be comfortable with technology, and applying knowledge from one area to another, have creative problem solving and critical thinking skills and pay attention to detail. They should have basic electrical and mechanical knowledge (e.g., multimeter use, basic equipment installation, repair and programming). Depending on site conditions they also need to be certified in tower/rock/tree climbing, boat handling, SCUBA diving, respective safety training, and enjoy skiing, hiking, off-road driving etc. plus need to be skilled in GPS orienteering, navigation, and basic map making.
- **Communications/data transport expert / Licensed Commercial radio operator (ideal, but not required)** - needs to be familiar with moving digital data over wired or wireless networks from remote points to project servers and should have basic knowledge of radio communication (e.g., technician-level amateur radio license, basic antenna theory, IP networking)
- **Network administrator / System administrator** - is responsible for network architecture, redundancy of systems from data center to field sites, backup, data security
- **Software developer** - skills in preferred programming language
- **Data manager** - needs to be familiar with means of documenting procedures for maintaining communication between all roles involved, specifically, means for documenting field events and their ramification for the data quality. Needs to know approaches/software for managing high frequency streaming data, standard QA/QC routines for such data, approaches to documenting data provenance and data archiving (space requirements, backup, storage of different Q/C levels) and have database/software package programming/configuration expertise
- **Data technician** - needs to be thorough and reliable during tasks like ‘eye on’ quality control, manual data entry etc.

Overview of Chapters

The following chapters contained in this guide are structured to provide a general overview of the specific subject, an introduction to methods used, and a list of best practice recommendations based on the previous discussions. Case studies provide specific examples of implementations at certain sites.

- **Sensor Site and Platform Selection** considers environmental issues, site accessibility, system specifications, site layout, and common points of failure.
- **Data Acquisition and Transmission** is concerned with the acquisition of sensor data from the field, while ensuring the integrity of those data. Also, remote control of the system.
- **Sensor Management Tracking and Documentation** outlines the importance of communication between field and data management personnel as field events may alter the data streams and need to be documented.
- **Streaming Data Management Middleware** discusses software features for managing streaming sensor data.
- **Sensor Data Quality** discusses different ways sensor data may be compromised, how to automatically control for it in the data stream..
- **Sensor Data Archiving** introduces different approaches and repositories for archiving and publishing data sets of sensor data.

Sensor Site and Platform Selection

Considers environmental issues, site accessibility, system specifications, site layout, and common points of failure.

back to EnviroSensing Cluster main page

Fig. 1 Typical environmental sensor deployment with science, support, and communication systems. Photo 2013 Scotty Strachan, NevCAN Sheep Range Blackbrush station

Contents

Contacts

The lead editors for this page may be contacted for questions, comments, or help with content additions.

Scotty Strachan - Department of Geography, University of Nevada, Reno - scotty at dayhike.net
Adam Kennedy - Andrews Forest LTER - adam.kennedy at oregonstate.edu

Reviewers

This page was reviewed by:

Jason Downing, Bonanza Creek LTER Information Manager, on 5/2/2014
Richard Jasoni, Associate Research Ecologist, Desert Research Institute, on 6/30/2014

Overview

Selection of exactly where and how to acquire data via in-situ sensing efforts is a crucial point in the science process where environmental research is concerned. Decisions made when choosing sites, sensor packages, and support infrastructure in turn place boundaries on what the final science deliverables can be. Data types, quantity, and quality are more or less set in stone during this process. Initial costs, timeframes, and sustainability are also determined by these choices. Selections need to be made based on the desired science products, but also in consideration of a wide array of variables including land ownership, access, equipment budget, long-term maintenance capability, previous research, and construction/deconstruction logistics.

Setting up terrestrial sensing systems is a major infrastructure/personnel commitment with budgetary and environmental concerns, and every effort towards maintaining a robust, low-impact, and long-term data stream should be made. Because each region possesses unique geography, there is no “one size fits all” solution. Instead, a series of decisions needs to be made, with the goals and capabilities of the research team defined in the context of clearly-articulated science questions and objectives.

Introduction

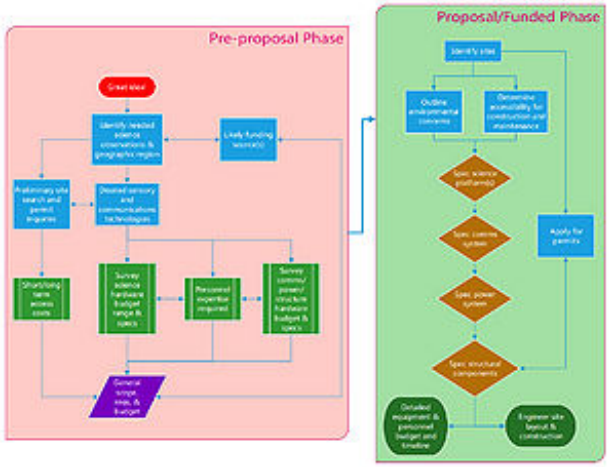

Fig. 2 Progression of work in selecting a site and designing a science deployment.

Identifying both the deployment strategy (site, process) as well as the physical hardware (sensor platforms and support infrastructure) for environmental sensing is usually a daunting task. A key objective of the research team should be to keep the science context in view during this process, as logistic realities will often clash with "ideal" scientific conditions. Very often the decision tree for choosing exact locations and deployment schemes is dependent on interacting factors (such as permitting/geography/access; Fig. 2). There is also a vast array of possible sensor/hardware packages available for a multitude of science applications.

It is critical that Principal Investigators (PI’s), logistical techs, and sensor specialists work together to develop specific deployment plans and alternatives, ideally in the pre-proposal stage. Planning topics must include science objectives, operating budgets, proposed locations, seasonal weather patterns, power sources, communications options, land ownership, distance from managing institutions, available personnel/expertise, and potential expansion/future-proofing. All of these categories are equally critical for discussion as proposed instrumentation projects move towards implementation.

Methods

Site visits, permit/agreement negotiations, equipment specifications, and deployment timelines need to be

initiated concurrently because all phases of deployment are interdependent (Fig. 2). The P.I., together with the technical personnel, should identify sites for sensor and equipment deployment based on science needs, local topography, permit/agreement availability, logistical access, and availability of services (such as power and communications). Portions of the plan (such as some purchasing decisions) should remain flexible until the precise sites, permits/agreements, and data flow plan have been positively determined.

Environmental concerns

Environmental conditions have considerable bearing on science application, platform design, construction logistics, access restrictions, equipment reliability, and maintenance cost/longevity. Conditions for in situ sensing can vary tremendously from region to region; therefore, site and equipment selection must be concidered on a case-by-case basis.

- Local topographic variables include: northern versus southern exposure, which can affect hours of direct sunlight and snow persistence; and valley/sink versus ridgeline settings, which can affect daily temperature cycle and wind characteristics. The differences in airflow, wind exposure, cold sinks, snow drifts, sky exposure for solar panels, and possible radio/communications pathways are all important variables when selecting a site and what type of equipment will be deployed.

- Dominant vegetation conditions and potential long-term growth can alter sensor readings via shading effects, affecting temperature, radiation, and snow-related measurements. Radio communications are also affected by vegetation, with most microwave frequencies used by high-speed data radios being strongly attenuated by trees and brush. Vegetation can also be a long-term hazard in the forms of fire fuels and deadfalls.

- Visibility and the visual impact of deployments should be considered for both security and aesthetic considerations. Sometimes reduction of visual impact is required by landowners, but in general it is simply good practice. Metal structures can be camouflaged with paint to reduce visibility, structure heights may be reduced to blend with vegetation, and ground disturbance can be kept to a minimum to avoid biasing certain types of measurements and erosion.

- Dominant weather conditions determine what levels of seasonal access are available, what structural designs should be used, and what sort of equipment should be purchased. Extreme temperatures, tropical storms, lightning, snow depth, riming/ice, UV exposure, high humidity, wind speeds, salt water exposure, flooding, and stream depth variation are all examples of conditions which will influence design and deployment plans.

- Wildlife can provide hazard considerations or be affected by proposed deployments. Bird perching and flight paths, cattle, soil invertebrates, rodents, and large mammals can all disturb or be affected by sensors and equipment installed in the field. Landowners will have regulations or preferences concerning these factors, and proactive steps are necessary on the part of the science team to minimize these hazards.

- Sensitivity to local political and social issues need to be considered, as objective science data should constructively serve the local populations as well as the scientists and funding agencies.

- Site security is a primary concern when planning to deploy sensors and equipment into the field. Human theft/vandalism is a potential cause of sensor disturbance or failure. While remote deployments are nearly impossible to secure physically, measures such as camouflaging, informative signs, fencing, and lockboxes may be employed to mitigate hostile or irresponsible passers-by.

- Hazards to sensors include natural disturbance/disasters such as wildfire, flooding, extreme winds, and mass wasting. Planners should be aware of all these possibilities and at least examine the likelihoods of these event at sites which have been evaluated from the scientific point of view.

Site accessibility

Fig. 3 Seasonal access may vary highly depending on location, limiting the types of maintenance possible at any given time.

Locations for in-situ sensing must be accessed for data collection, survey, construction, and maintenance over the life of the project. Seasonal conditions, roads, and topography determine what types of access may be used during different times of the year. Categorical considerations include:

Vehicular access. Commercial vehicle/equipment, 2WD auto, 4x4 truck, ATV, snow machine, boat, helicopter.
Non-motorized access. Hiking, skiing, pack animals, snowshoeing.
Access improvements. Road building, trail building, trail demarcation, safety rails, harness anchor points.
Seasonal access. Define access by spring/summer/fall/winter seasons. This is directly related to local weather/topographical conditions.
Construction access. Heavy equipment, special equipment, heavy loads, and heavy foot traffic are all likely possibilities depending on monitoring design.
Minimal impact considerations. Can traffic/access be directed in a way to minimize environmental impact (e.g. erosion, vegetation)? Solutions include boardwalks, bridges, raised steps, delineated pathways.

Science platform selection

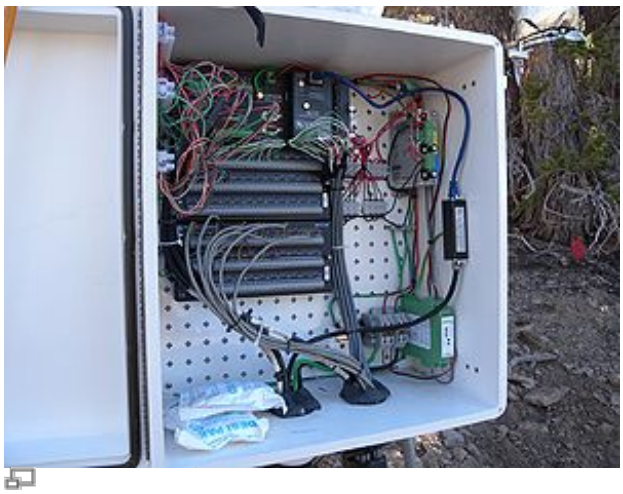

Fig. 4 Science instrumentation specification must be driven by science questions and environmental/logistical constraints.

Once the science questions have been established and site conditions are known, an itemized list of sensor and support system platforms/hardware may be assembled that best fits the application and budget. Primary considerations include reliability, comparability with other similar field systems, technological

(e.g. programming) requirements, budget, and system flexibility (e.g. upgrades, expansion, telemetry options). Accuracy, precision, and expected period of use prior to calibration or replacement may also be a consideration. In some fields of study, there are only one or two alternatives to choose from in terms of scientific instrumentation, whereas in others there can be many choices. Options can be narrowed by researching what equipment/standards are used by existing installations to which comparability is desired. Once a data acquisition platform and sensor array is chosen, remaining support systems are then designed around this core equipment.

Support system specification

The subsystems of infrastructure, electrical power supply, and data communications should all be designed to best support the science platforms in all seasons over the long term. While some vendors offer "all-in-one" packages supplied with standard instrumentation, it is best for the research team to assess whether these solutions are adequate for their chosen site and objective. Quite often several science questions are being addressed in larger deployments, and multiple hardware solutions from several vendors must be combined into one deployment. The support systems should be specified and scaled appropriately.

- Physical infrastructures – these are the building-blocks of any remote data acquisition site, including tripods, towers, poles, buoys, solar panel racks, storage boxes, fencing, concrete pads, and the like. Quite often a single tripod or tower does not have adequate space or structural integrity to support all of the sensors, antennas, solar panels, batteries, and other items, so a typical site design incorporates multiple structural components.

- Power generation and storage – for sustaining long-term reliable data streams, power independence is critical. Stations should be capable of generating and storing their own power locally, as well as taking advantage of any grid or other available power that is within budget and design criteria. Because the majority of related electronics are ultimately powered by DC voltage, having a power generation system and DC battery bank for every site (and sometimes discrete subsystems) is recommended to minimize the loss of power and the resulting data gaps. Independent generation sources are most commonly PV arrays (solar), wind, or water turbines. For reasons of cost, reliability, and maintenance issues, PV (solar) is recommended as the primary on-site generation source if environmental conditions allow. Incorporating simplicity, redundancy, and excess capacity is important for long-term reliability.

- Data communications – Use of real-time communication (in addition to local storage capacity) is desirable in order to transmit data, monitor system health performance, troubleshoot problems, and minimize data gaps. This is usually performed using radio communications (whether vendor-specific or building a general-purpose field IP network). Communications systems need to be robust, secure, and should have low power requirements (refer to "Data Acquisition and Transmission" Best Practices for further detail).

- Construction details – When selecting and designing the sensor and support systems, many details need to be considered when generating specifications and purchasing hardware. Wires should be protected in conduit and storage enclosures to avoid exposure to damage and seasonal degradation. Wire lengths, enclosure sizes, and mounting locations should be planned for accordingly. Anchoring for support structure should be designed to withstand worst-case weather/environmental conditions. Use of corrosion-resistant metals for structure and hardware such as galvanized steel and aluminum will greatly reduce failure or ongoing maintenance problems.

Site layout

Fig. 5 Carefully planning a site layout in advance can prevent surprises and setbacks during installation.

Site layout at first might seem trivial, but is very important when considering interactions of the various subsystems that can influence sensor/equipment reliability and data quality. Science questions/objectives should drive the placement/separation of sensors to optimize measurement quality, followed by placement of support systems and additional structure. Solar arrays need to be angled for sun exposure, minimal shading, and snow shedding. The impacts of site structure on measurements such as wind eddies, incoming/outgoing radiation, camera viewsheds, or precipitation catch zones need to be considered as well as aesthetic impacts if located in a region that is frequently visited by the public. Power and data cable runs should be protected and kept as short as possible; voltage drop over long runs can be a consideration in layout and design. Stipulations in site permits may be drivers of site layout and construction. Once the site layout is designed and mapped, specification of construction materials, sensor cables, and other supplies may be optimized.

Best Practices

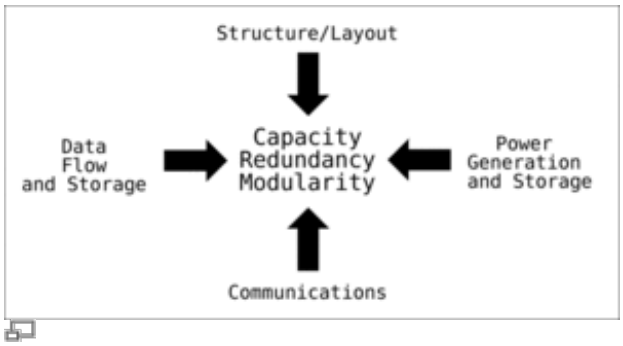

Fig. 6 The approach to deployment should be as durable, reliable, and flexible as possible to accommodate unforeseen conditions and changing science questions or technology improvements over the long term.

Selection of deployment sites, sensor packages, and support systems are interacting processes which can require some iteration before arriving at the final determinations. Unless the science questions are extremely narrow or exceptional in nature, it is unlikely that any one of these decisions can be made in a vacuum without considering the others. With this in mind, the following overarching recommendations should be emphasized:

- P.I. consultation with system/hardware/construction specialists while in the proposal phase will minimize budget surprises or platform compromise later in the process.

- Data quality and longevity should be the ultimate goals when designing the deployment. Making choices for more robust and widely-used core systems and sensors will ensure that data comparability is maximized and hardware problems corrupting data or creating gaps are minimized. Purchase of reliable and known equipment is not as expensive as repairing/replacing equipment halfway through the study or losing valuable data.
- When data quality and continuity is paramount, use of replicate sensors or stations may be required.
- Planning for real-time connectivity is crucial for reducing field maintenance time and data gaps.
- Optimal site selection to answer science questions can often be impeded by permit requirements and landowner preferences. Starting the conversation with landowners early in the process may improve the chance of getting the locations/deployment types that are desired.
- Standardizing sensor and support hardware, software/programming, and structural designs across multiple sites minimizes maintenance issues as well as construction costs and design time.
- Assessing access capabilities to the sites will allow for planning of emergency maintenance access, procedures, and costs.
- Overbuilding structure, power capacity, and site infrastructure (e.g. cabling, networking) will prevent problems in the case of unforeseen events or site expansion.

Common Points of Failure

- Power problems are one of the most frequent causes of total system failure. Battery fatigue, loose connections, and electrical shorts need to be anticipated and prevented where possible. Power systems need to be protected, over-engineered, and replicated wherever possible.
- Temperature extremes of heat or cold can cause electronic or mechanical failure of individual sensors and systems. Insulating enclosures, ventilating enclosures (active or passive), and placement of equipment in sheltered zones can help alleviate these problems.
- Humidity and condensation can be a serious issue for electronics longevity and circuit performance (including accuracy). In zones of high average humidity, sealing enclosures and providing some means of reducing humidity (e.g. desiccant packets) is desirable.
- Sensors can be disrupted by wildlife. Hardening of sensor systems (e.g., armoring cables, fences) can help with some problems. Near-real time data feeds allow rapid detection of problems that will occur.
- Lightning strikes or near-misses are a common problem at exposed or mountainous sites. Extensive grounding (e.g. exposed copper wire network) and use of surge protection throughout the power system and at ends of long power and data cable runs will compartmentalize the site electrically and protect as many components as possible.
- Lack of data storage replication can cause loss of data. Incorporating high capacity storage on-site (datalogger) as well as off-site (database), this problem can be mitigated.
- Personnel turnover coupled with lack of process and hardware documentation can lead to data discontinuity or equipment failure (see Sensor Management and Tracking for additional details).

Case Studies

- NevCAN Transects or Walker Basin (Scotty) --- To be completed, will include a station design and systems, maintenance/access plan, data flow, and some photos/diagrams.
- Andrews Research Sites (Adam) --- To be completed
- Sevilleta - Renee to complete with multiple case study examples

Data Acquisition and Transmission

Concerns the acquisition of sensor data from the field and remote control of the system, while ensuring the integrity of those data.

back to EnviroSensing Cluster main page

Contents

Overview

Traditionally, environmental sensor data from remote field sites were manually retrieved during infrequent site visits. However, with today's technology, these data can now be acquired in real-time. Indeed, there are several methods of automating data acquisition from remote sites, but there is insufficient knowledge among the environmental sensor community about their availability and functionality. Moreover, there are several factors that should be taken into consideration when choosing a remote data acquisition method, including desired data collection frequency, bandwidth requirements, hardware and network protocols, line-of-sight, power consumption, security, reliability and redundancy, expertise, and budget. Here, we provide an overview of these methods and recommend best practices for their implementation.

Introduction

The classic method of acquiring environmental sensor data from remote field sites involves routine technician site visits, in which s/he connects a laptop to a datalogger, an electronic device that records sensor data over time, and manually downloads data recorded since the last site visit. Once the technician returns to the lab, s/he is then responsible for manually uploading these data to a server for later processing and archival.

While manual acquisition methods are generally effective, there are many reasons to automate environmental sensor data acquisition. For instance, if the site is not visited frequently enough, the datalogger memory can become full and depending on how the datalogger is programmed, sensor data will either overwrite itself or stop recording entirely. This situation often occurs at remote sites that become periodically inaccessible due to environmental conditions, such as heavy winter snow pack. Second, the burden of responsibility for not only the successful retrieval of the sensor data, but also the subsequent upload to a server for safekeeping, lies solely on the technician. Moreover, with any instrumented site, there is the inherent potential for sensor or power failure. Automated data acquisition systems allow technicians to learn of such issues prior to visiting the field site, reducing the potential for data loss. Finally, automated data acquisition methods save hundreds of person hours and vehicle miles that would have otherwise been spent manually acquiring data or troubleshooting unanticipated problems, thus improving the overall quality of the data.

Bidirectional communication methods have the additional advantages of allowing technicians to remotely change system settings, test configurations, and troubleshoot problems. These methods also open the field to a wide variety of devices that may be deployed at a remote field site, such as controllable cameras, on-site wireless hotspots, and IP-enabled control or automation equipment.

Considerations

The decision of which sensor data acquisition method to use at a given site requires the careful consideration of many factors, for which we provide an overview here.

Collection Frequency

What is the desired collection frequency? How important is real-time accessibility? For instance, the data could be retrieved in near real-time (every few minutes to every few hours) or just once or twice per day. High frequency datasets or images should be collected more frequently.

Bandwidth

Bandwidth can be an important consideration, particularly when high frequency data are being collected. Will cameras be utilized at the site? Where is broadband point of presence (POP) located? Does equipment work with required bandwidth? More frequent collection intervals require less bandwidth per transmission are are recommended for high frequency datasets or for images.

Protocols

Hardware

Many dataloggers only have serial (RS232) ports, therefore requiring a serial-to-ethernet converter to interface with automated acquisition instrumentation. USB.

Network

Public IP networks are advantageous over private IP networks in many cases because they can be managed from anywhere there is a connection to the Internet. Remote access to private IP networks requires advanced network expertise to provision port forwarding in firewalls and/or VPN.

Line-of-sight

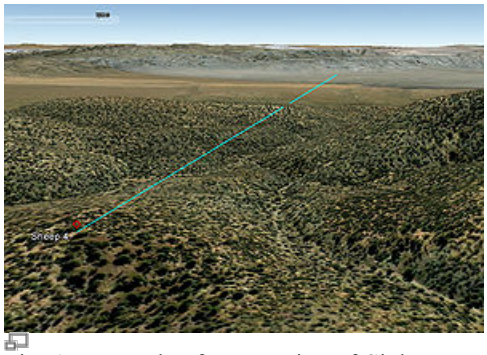

Fig. An example of a near-Line-of-Sight (nLoS) condition, where intervening terrain and/or vegetation can interfere with the radio signal. In this case, the antenna heights at both ends are actually at the 8 m level, mitigating the effect somewhat. The link is operational, albeit with a reduced Received Signal Strength Indication (RSSI) due to the presence of an obstruction in the links Fresnel zones.

Evaluation of environment, topography, and vegetation. Can be initially determined using LOS calculators, which use DEM models, but must be ground truthed. Often requires a repeater infrastructure. Choosing repeater locations involves many of the same considerations for choosing site selection. Distance to repeater is a factor. Automated sensor data acquisition methods require many of the same site selection considerations discussed in Sensor Site and Platform Selection, particularly when selecting repeater sites.

Power

How important is real-time accessibility? (e.g., what is desired collection frequency?). What are the transmission type power requirements, onsite buffer size. Redundancy is preferred, especially in very remote sites. If power is disrupted, will system resume operations?

Security

Physical Security

For physical security considerations, refer to Sensor Site and Platform Selection.

Network Security

It is recommended that encryption keys, such as WPA2 encryption, be configured to prevent unauthorized access of data acquisition equipment or sensor data. A private IP network can further help to prevent unwanted access, but also prevents easy remote management by network administrators unless a VPN is installed.

Reliability and Redundancy

of transmission mode and of equipment. Also, network infrastructure.

Expertise

Some acquisition methods are plug-n-play with substantial vendor and/or community support, while others require a fair amount of hardware and network expertise to configure and maintain. All acquisition methods require fundamental knowledge of IP networking along with basic electronics, radio, and antenna theory.

Budget

Costs of implementing a data acquisition and transmission method depend on existing infrastructure, initial setup costs including personnel, personnel costs, specifically technician maintenance, and recurring costs, such as monthly recurring costs with cellular transmission.

Methods

There are three general categories of remote sensor data acquisition methods: manual, unidirectional telemetry, and bidirectional telemetry. Each has advantages and disadvantages in terms of infrastructure, cost, reliability, required expertise, and power consumption.

Manual

This method involves scheduled visits to the site by a field technician, who uses a serial-to-computer connection and/or flash memory transfer of environmental sensor data to their laptop or similar device. Upon returning from the field, the technician is responsible for manually uploading these data to a server. This acquisition method is simple and may be the only option when site instrumentation generates large data files. However, this method provides no real-time data access and therefore, no knowledge of instrumentation failures. Moreover, the reliability of this method is completely dependent on the technician.

Unidirectional

Unidirectional sensor data acquisition methods involve regularly scheduled wireless data transmission from a remote site to a server, with no offsite ability to control or change sensor settings. These include...

Geostationary Operational Environmental Satellite (GOES)

Fig. A typical circular-polarized GOES antenna for one-way burst transmission of limited data

This method is preferred in very remote and potentially rugged areas where other automated transmission methods would not work. While it does not require line-of-site to a repeater like most other transmission methods, it does require a view to the southern sky. Additionally, the GOES method has a low power requirement. However, GOES has several disadvantages, including a high initial investment (<$5K) and requires training and licensing. Moreover, less than 100 values can be transferred per hour, making it disadvantageous for sites that sample at high frequencies.

Data transfer speed for GOES systems is typically limited to 1200 bits per second with 10 second transfer assignments occurring once every hour. During each 10 second period, one can transfer up to 1500 bytes of data (12,000 bits / 8) including the 53 byte GOES header string. In other words, maximum 1447 bytes with time stamps and measured values can be transferred to the satellite during one transmission interval. Most often, GOES messages are organized in a time ordered format similar to the following example:

```
0105E59013190131824G30+1NN196WXW00517
0 13:00:00 23.7,43,5,245,-55.1,5,245,23.7,23.7,12.8
1 12:30:00 23.7,43,-55.1,204,1011.09,0.000,0.0,24.7,0.270,-0.456,-0.997,-0.416,-2.687,23.5,0.00,214.81,0.00,5,245
1 12:45:00 23.7,43,-55.1,204,1011.11,0.000,0.0,24.7,0.249,-0.468,-0.994,-0.436,-2.650,23.5,0.00,214.82,0.00,5,245
```

Here, first line represents the GOES header string that includes the address, date and UTC time of the transfer (13:18:24), signal information, satellite information, message length and some other characters. In the example above, the lines that follow carry the time stamp and value information from the sensor sets 0 and 1. As the length of each character in the sensor set string is 1 byte, we can see that our GOES message has approximately 280 bytes used from 1447 bytes that are a theoretical maximum for the transfer. However, in order to accommodate the possible differences between the station sending time, decoders, and scheduled reception time, we never want to reach this value.

Prospective users of the GOES system must fill out the System Use Agreement (SUA) form and, upon approval, receive and sign the Memorandum of Agreement (MOA) from the NOAA's Satellite and Information Service (NESDIS). After the MOA is approved, NESDIS will issue a channel assignment and an ID address code to the applying organization. Non-U.S. government and research organizations must be sponsored by a U.S. government agency in order to apply for this permission. Upon approval, all users must purchase equipment that has been certified to be compatible with the GOES Data Collection System. As of May 2013, GOES transmitters must conform to the certification standards version 2 (also known as CS2). This change was implemented to double the number of GOES channels on the same bandwidth. As a result, old GOES transmitters that are only compatible with the CS1 standard cannot be used for new NESDIS assignments. For assignments obtained prior to May 2012, CS1 transmitters will be supported until 2023. If you consider buying the used equipment for GEOS transmission, make sure the transmitters are compliant with the CS2 standard.

Meteor Burst Radio

Like GOES, this method does not require line-of-sight and has a low power requirement. However, it requires a large antenna, arrangement of service, and has a very slow transmission rate. It works by reflecting VHF radio signals at a steep angle off the band of ionized meteorites that exist 50 to 75 miles above the Earth. See SNOTEL and ITU Case Studies for more information.

Iridium Satellite service

Iridium provides the only complete global satellite coverage. The new Iridium Pilot is available until 2016. The next generation of Iridium is expected to be implemented around that time frame. The Pilot is very easy to install and maintain with a waterproof body and USB interface. With this simple interface a laptop can be connected and surfing the web within minutes. Recently, the cost has become more affordable with a per data usage cost structure. Since Iridium operates in the L band it is nearly impervious to weather. Iridium is used primarily for marine communication.

Bidirectional

This method involves bidirectional (and typically wireless) transmission of data from a remote site to a server, with the ability to modify datalogger programs and/or sensor settings remotely. These methods generally require line-of-sight and security considerations (both network and physical). Sometimes, can be purchased from an Internet Service Provider (ISP) if there is commercial coverage in the area, or can be manually installed in remote areas. Often, connectivity can be extended to computers onsite. Combination of several methods may be required in certain situations.

ISM band radio network

Fig. Three different antenna types used for bi-directional microwave band communication: a) 5.x GHz 24" dual-polarity dish, 30dBi gain; b) 2.4 GHz single-polarity grid, 24 dBi gain; and c) 5.x dual-polarity panel, 23 dBi gain. The higher the gain value, the more narrow the antenna directivity, increasing signal strength in the desired direction and rejecting adjacent

interference. Each of these designs has pros
and cons, depending on the application.

(unlicensed, 900 MHz, 2.4 GHz, 5.x GHz): The ISM band radios are commonly referred to as "WiFi" radios (even though these are generally used as backhauls and not wide-area access points) and come in a variety of frequencies. This method has many advantages in that it is nonproprietary, has no recurring costs, uses public radio frequencies, allows transmission of large datasets, utilizes inexpensive hardware, is not restricted to a single vendor or device type, and has increasing compatibility with many devices. However, it requires line-of-site (LoS) or near-line-of-sight (nLoS), a network interface on loggers and devices, and basic to advanced network administration skills. These radios are also subject to interference, particularly in more populated areas, and can have higher power requirements than other transmission methods.

Cellular

This method has prolific coverage and minimal ongoing maintenance. However, it requires a reliable cellular network be present and comes with monthly recurring costs. Occasionally a contract may be required unless can be negotiated through university or organization.

Vendor-specific radio network

Vendor specific radio networks use proprietary protocols and are typically more expensive than some other acquisition methods, but have the advantage of being relatively easy to set up and maintain. For example, Freewave

Satellite internet

This method can get limited 2-way connectivity into a remote site, albeit at high monetary costs and significant power consumption. It has slow uplink speeds, high latency, requires a subscription, and on-site vendor setup is required.

Licensed radio

This method is expensive and requires a purchase of a licensed frequency.

Mesh Networks

A mesh network is a network topology in which each node relays data throughout the network. A mesh network whose nodes are all connected to each other is a fully connected network. Due to the inherent redundancy in mesh network design, mesh networks are typically quite reliable, as there is often more than one path between a source and a destination in the network. Mesh networks are typically wireless, but can be wired. Mesh networks are not very common, especially at large spatial scales, since every device must be connected to every other device. The initial investment to build such a network is considerably higher than other acquisition methods. Mesh networks, either partially or fully connected, are most commonly used in distributed sensor networks.

Wired

While all methods discussed utilize wireless transmission protocols, wired bidirectional transmission may be possible via in-ground or aerial copper or fiber optics.

Best Practices

- Think about data acquisition as part of site design. It is more expensive to add telemetry to a preexisting site than to integrate with initial site construction. Make sure to include acquisition method power consumption in the site power budget, or a separate power system will be required.
- Use software tools with radio or a handheld spectrum analyzer to survey RF conditions on-site. For instance, urban areas are typically noisier with respect to RF interference, and for Wi-Fi transmission methods, 5 Ghz frequencies are preferred.
- Use a bidirectional transmission method to provide more control and flexibility.
- Over-engineer power system, especially when powering repeaters and other sites in hard to reach areas.
- Use equipment that can conserve power (sleep mode)
- Provide adequate local storage for disrupted transmissions. Adequate “off logger” local storage is recommended to avoid losing data when/if logger is reset.
- Provide redundancy, such that when one link goes down, the site is still remotely accessible. This is related to network architecture planning - multiple geographic/hardware paths along backhaul routes to field hubs is highly desirable. Examples include: parallel backhauls, multiple internet points of access, "failover" paths. Having a "back door" into the network, even over reduced speed links, can allow a tech to remotely troubleshoot problems on the main links.
- Standardize transmission protocol across all sites to provide easier network management.
- Match radio band, power, antenna, and bandwidth to application. For instance, when a site generates high frequency sensor data, high bandwidth and high data collection frequency are recommended.
- Use a narrow bandwidth for your RF devices/coordinate frequencies between radio systems
- Thoroughly document all site coordinates, IP addresses, maps, radio azimuth, zenith.
- When using an IP based acquisition method, use public IP addresses for easier remote management of devices.

Case Studies

- NevCAN: Nevada Climate-ecohydrological Assessment Network - University of Nevada, Reno (UNR); Desert Research Institute (DRI), University of Nevada, Las Vegas (UNLV)
- Sevilleta Wireless Network - Sevilleta Long Term Ecological Research (LTER) Program and Sevilleta Field Station; Department of Biology; University of New Mexico (UNM), Albuquerque, New Mexico, USA
- Virginia Coast Reserve LTER Wireless Network - Virginia Coast Reserve Long Term Ecological Research (LTER) Program
- SNOTEL
- ITU

Resources

GOES

- New NESDIS Assignments
- CS2 Standard Compliance

References

Sensor Management Tracking and Documentation

Outlines the importance of communication between field and data management personnel as field events may alter the data streams and need to be documented.

back to EnviroSensing Cluster main page

Fig. Documentation of sensor installation, maintenance, and related systems is critical to long-term data usability.

Contents

Overview

Automated observation systems need to be managed for optimal performance. Maintenance of the overall sensor system include anything from repairs, replacements, changes to the general infrastructure, to deployment and operation of individual sensors, and seasonal or event driven site clean up activities. Any of these activities in the field may affect the data being collected. Therefore, consistent and uniform records of maintenance, service, and changes to field instrumentation and supporting infrastructure serve as metadata for long term quality control and evaluation of the sensor data.

In this chapter, we describe the types of management records that should be kept and the various methods for collecting, maintaining, communicating, and connecting this information to the data. It is important to create tracking and documentation protocols early on because these protocols will support and guide communications and work between field and data management personnel.

Real time monitoring of system health and alerting systems are discussed in the middleware, quality control, and transmission sections of this document. Although some of these parameters do not affect the actual data quality, tracking of these system performance diagnostic data may be helpful to detect patterns

and prevent future data loss, intervene remotely, and schedule site visits more effectively. Calibration procedures and schedules, maintenance activities, and replacement schedules are hardware specific and will not be covered here in detail.

Introduction

Data are collected to detect changes in the environment, effects of treatments, disturbances etc., and in all data collection great care is taken to not mask the signature of events of interest with impacts from unavoidable, sampling related disturbances. Field notes are usually associated with the raw data to be able to discern a natural event of interest from a management event. Data collection approaches using automated sensing networks are becoming more complex with many people involved in the data gathering, management, and interpretation activities, and communication among all involved parties is becoming more important and more challenging. Field notes can be a useful vehicle for this communication. Everyone using older long-term data knows the value of field note books to help understand and interpret a dataset. Field notes are equally valuable to future users for a sensor data stream, particularly if the notes are interpreted such that information is integrated with the data via data qualifying flags and method description codes.

Currently there are no standards for flag code sets or for defining which events should be flagged and how to efficiently communicate with data users. Here we attempt to present a list of events that are useful to track and that have been helpful in the past to guide data users in the interpretation and evaluation of the data. To manage this information the concepts of a '*logical sensor*', a '*physical sensor*', a '*method*' and '*event codes*' have proven useful.

A '*logical sensor*' or a sensor data stream can be defined by a location, height/depth, and measurement parameter, regardless of what exact physical sensor or hardware is used to log measurements. An example would be 'air temperature at 3 m above the ground at site A'. However, over time the '*physical sensor*' will have to be calibrated, eventually replaced, and a new type of sensor may be chosen to provide more accurate measurements. If hardware is swapped out for technical reasons, the data stream still represents the site location for that measurement, and the notion of a '*logical sensor*' allows identification of a consistent data stream over time.

Changes in the type of sensor or '*method*' might be tracked with a method code associated with the logical sensor. Of course should a replacement sensor be significantly different such that the past and new data stream are not comparable, then a new logical sensor stream should be initiated. Events such as routine calibration might be flagged with an '*event code*' rather than a change in '*method*', even if this event has lasting effects on the data, i.e., more accurate data. An event code may serve as a means to link to individual field notes for the event. '*Physical sensors*' should also be individually identifiable by location and tracked through a calibration or replacement schedule.

Methods

What should be tracked

Basic information on the site and hardware configuration need to be recorded at installation time. During normal operations event tracking can be done at several levels of granularity with respect to a research program. For example, it may be done at the level of the entire infrastructure, at a site, or at a sub-component of a site. The information about each event needs to be propagated or connected to all relevant data streams. Following are examples of what should be tracked at each of the above levels, in terms of impact on the recorded data:

Documentation at setup time

- Location lat, long, elevation (and/or depth), direction (e.g. camera facing north), Location from a certain reference point (e.g. tower base)
- Site description
- Site photos with metadata, photos of procedures (how do you change ...), photo of sensor (so others can easily recognize)
- Manufacturers specs and ID of instruments (make, model, serial number, range, precision, detection limit, calibration coefficient)
- Instrumentation (e.g. datalogger, multiplexer, sensor) wiring diagrams (this should be part of the logger program comments, a header section with the wiring description channel by channel)
- Power wiring diagrams (e.g. how many solar panels, are they hooked up in series or parallel, etc.)
- Network topology and IP addresses
- Software used for calculating measurements (other than datalogger)
- Instrumentation deployment date (the "go live" date)

Infrastructure events to track

- Changes to dataloggers, multiplexers, or datalogger programs (datalogger programs may be archived)
- Power problems, including battery voltage
- Enclosure temperature and humidity
- Platform maintenance (e.g., tower inspection, tramline leveling, etc.)
- Sampling protocol changes (e.g., timing, routine changing or upgrading of sensor parts, instrument change or replacement)
- RF/network performance degradation (prevents some/all data from being transmitted; track health/status of IP network devices using SNMP streams to Nagios, etc.)

Site level events to track

- Site disturbance (e.g., animal, human, weather caused)
- Site visits (presence of people may change measurements)
- Site maintenance (e.g., cutting brush, cutting trees, etc.)
- Changes to sensor network design, including additions or deletions of sensors

Sub-component events to track

Here, we include components like individual telemetry, power systems, instruments, sensor components, etc. While each component doesn't affect the whole system, they still may influence the interpretation of the measurements. To track individual components a system of IDs may be developed for all components and supported by Barcodes, Geo-Location Tags and Microchip Encoded Sensors.

- Sensor failures
- Sensor calibrations
- Sensor removal
- Sub-sensor addition, removal, or change (pluggable sub-sensor positions within the main sensor need to be noted and kept consistent)
- Sensor installation (replacement)
- Sensor maintenance (cleaning, change of parts)
- Sensor firmware upgrades
- Enclosure temperature and humidity
- Repositioning of sensor (e.g., move up during winter to be above snowline
- Normal (non extreme) disturbances as they are noted and removed (e.g., sticks in weirs)
- Methodology changes (e.g., temperature radiation shield change)

How to track the information

Minimally documenting or logging site events or problems might be in a table structure such as:

SiteID	DataloggerID	SensorID	date time begin	date time end	category	notes	person
					controlled vocabulary		

However, usually a lot more is recorded at each site visit - see use cases. A controlled vocabulary is very important to categorise the event for later interpretation and flagging in the data set and should be established as early as possible with project specific terms. Several database structures to maintain this information and connect to the actual data are currently being proposed and discussed below in use cases.

Best Practices

Establish and document procedures and protocols for site visits, installation of new sensors, maintenance activities, calibrations, communication between field and data personnel. Such protocols may include pre-designed field sheets or software applications on field data entry devices, both of which should be synchronized with a central storage system to which all parties have access. Observations in the field may also be made and recorded by researchers and field personnel not directly involved in the sensor system maintenance, and provisions should be made to capture that information and communicate it to responsible staff members.

In addition to capturing the field events mentioned above it is good practice for the data management staff to regularly monitor the data and confer with the field crew when anomalies are noticed. This frequently will bring up additional information that needs to be recorded in the field. It is also good practice to have the data management staff visit the site, periodically assist with field maintenance activities to better understand and interpret field notes and generally interact with the field staff.

All physical sensors should be uniquely identifiable. This may be achieved by recording a serial number, attaching a barcode, using intelligent sensors which are capable of storing their own metadata and which can be accessed upon connection. This is particularly important for sensors that are moved around or are pulled for mass calibration and redeployed. Sensor location and calibration schedules should be tracked by each sensor with ID.

Document specific information during normal operations

- Either a pre-designed field sheet or a data entry app on a field device (tablet, laptop, etc.) helps remember every detail to record. It is also helpful to define a list of terms to describe the most common problems in a consistent way for later analysis.
- Document site ID, date, time, person(s), site conditions, tasks performed every time a site is visited.
- When updating datalogger programs, use a new program name for every change. It is advisable to save old datalogger programs.
- Use a changelog section in a datalogger program comment header to note date, author, and description of differences from last datalogger program. i.e. versioning/revision control
- For sensor specific events note the sensor ID (Bar Codes, Geo-Location Tags, Microchip Encoded Sensors (NEON 'Grape'), or intelligent sensors that store and provide their own metadata upon connection).

Maintaining the records and linking to affected datastreams

As mentioned earlier, this record keeping is an effort in communication between field and data personnel as well as communicating events to future data users. Hence a good practice is to permanently link this information to the dataset. This may be achieved on different levels - a description in a metadata document, an indicator of a method for a data series or each data value, a flag indicating a one time event

at a certain data value. As a minimum affected data should be flagged in a different column within the data table.

Following the concept of a logical sensor, certain events should trigger the start of a new 'method' description when the data stream is affected more than regular corrections can accommodate (e.g., new sensor using a different methods of measurement). In this case it is good practice to run the old and the new sensor side by side for a while to compare. No hard and fast guidelines are available for deciding when a method change occurs and when a whole new logical sensor stream (i.e., different data set or data table) should be started. These concepts are well implemented in the CUAHSI ODM, please see those documents for further discussion.

Most events, however, can be handled by well documented flags (sensor calibration, site maintenance activities, disturbances, etc.). For documentation, flags in the data file should link to a database with more extensive explanations of the events.

Managing sensor configurations

A number of sensors provide core measurements, but will also provide the ability to expand the sensor via one or more pluggable ports. When a sub-sensor is connected, the data from the sub-sensor are usually added to the main datastream as a voltage measurement that gets converted to the measurement parameter units post-transmission. Track both the number of sub-sensors and their port positions, since a change to either may cause problems in processing the data stream in middleware applications. For instance, a water sampler like a CTD may provide ports to connect sub sensors for dissolved oxygen or turbidity measurements. Note that the DO sub-sensor should always be connected to, say, voltage port 1, and the turbidity sensor is always connected to voltage port 2, and voltage port 3 is empty.

See also middleware capabilities and QA/QC procedure documentation in those respective sections.

Case Studies

Case study: Data model for tracking sensors and sensor maintenance at the Utah Water Research Laboratory (J. Horsburgh, September 2013)

The database design diagram depicts the data model as it is used at the Utah Water Research Laboratory, Utah State University. It was developed by J. Horsburgh and his research team. Currently efforts are underway to extend the CUAHSI ODM to store this kind of metadata based on the experience with this data model.

Case study: Two example field sheets from the HJ Andrews Experimental Forest in Oregon.

- HJ Andrews stream gage check sheet
- HJ Andrews watershed check sheet

Streaming Data Management Middleware

Discusses software features for managing streaming sensor data.

back to EnviroSensing Cluster main page

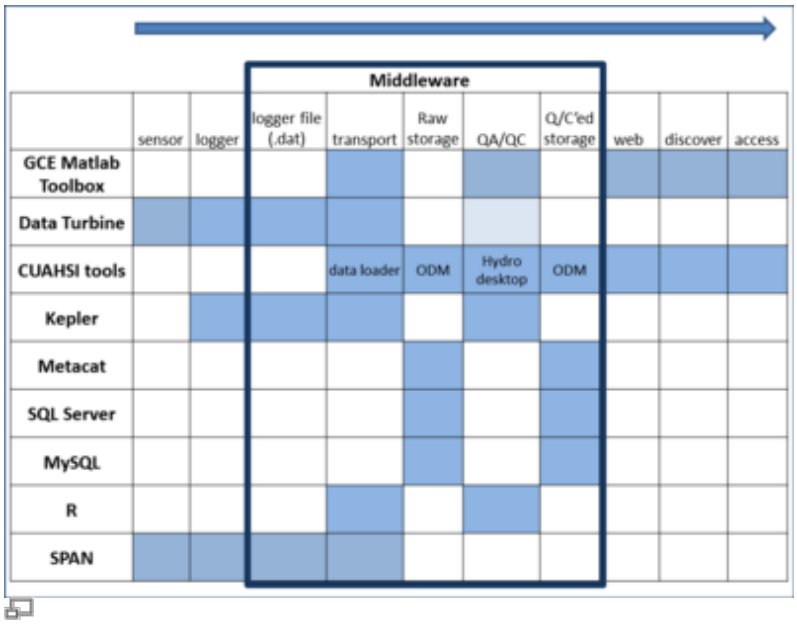

Fig. 1 The position of middleware in a generic sensor data management system.

Contents

Overview

Middleware are software packages and procedures that reside virtually between data collectors, such as automated sensors, and data 'consumers', such as data repositories, websites, or other software applications. Middleware can be used to perform tasks such as streaming data from data loggers to servers, archiving data, analyzing data, or generating visualizations.

Many middleware packages are available for developing a comprehensive, reliable, and cost-effective environmental information management system. Each middleware option can have a unique set of requirements or capabilities, and costs can vary widely. A single middleware package may be used if it includes all of the user requirements, or multiple middleware may be bundled into a data management system if they are compatible or interoperable with each other and the rest of the data collection and management system.

This section describes multiple middleware packages that are currently available, and provides examples of how different software and procedures are being used to collect, analyze, visualize, and disseminate sensor-supplied environmental data.

Introduction

There are multiple factors that may affect the choice, use, and performance of middleware. These factors may be classified according to a group's research agenda, technological requirements, and personnel skill sets.

Research Agenda

The research agenda of a group is a major determinant of the type of middleware system needed. A group focused on only one or a few narrowly focused research questions may need fewer types of sensors and consequently, fewer software modules may be adequate to streamline data processing from collection to the end goal. A team that investigates multiple questions spanning multiple research domains is likely to use more diverse and/or larger sets of sensors. There may not be a single middleware package that can meet all of the needs of a research group. In this case, multiple packages will need to be linked into a workflow.

Technological Requirements

The technological requirements of a research program may vary from simple to complex. If the research can be done with sensors from a single, well-managed company, the proprietary software packaged with the purchased sensor network may be adequate for at least a major portion of the information management system. For example, for Campbell Scientific dataloggers, their "LoggerNet" software integrates communication, data download, display and graphics functions. However, some dataloggers and sensors (particularly innovative ones, custom-built), may need custom-written software. It is important to plan time and budgets for required software upgrades, licensing, additional packages, support, and maintenance. Systems that cost less in the outset may not always be cheaper over the long run. It is also important to consider how to best meet infrastructure and bandwidth requirements, while deploying middleware on a variety of servers or laptop computers in the field or lab setting. Depending on the data and hardware infrastructure characteristics, each middleware option can introduce benefits or drawbacks to the overall system functionality.

Personnel Skills

Another key factor to consider is the skill set of the personnel. A complex data management system may require multiple people, each with a unique skill set such as database design, system architecture, web programming, etc. It is important to correctly identify each person's skill set and role in data management tasks. It may also be necessary to plan for additional hires or job-training to addresses various scenarios and solutions, to identify appropriate salaries, and to budget enough time for software development and system administration. More details about the personnel roles and skills can be found in the "Roles and required skill sets" section.

Middleware Classifications

Classification by functionality

Middleware can be classified with respect to the *functionality* they provide, such as:

> **Controlling instrumentation and data collection:** Modules may be used to control sampling intervals, manage the event-triggered (burst) or continuous sampling regimes, communicate and transfer data between the instrumentation and other system components.
>
> **Data monitoring, processing, and analysis:** Modules may provide alarm management, perform automated QA/QC on data streams, or run derivative calculations including averages, aggregation and accumulation, data shifting and transformation, filtering of time series records with respect to the dates, value range, location, station/variable type, or other criteria.
>
> **Export and publishing of data:** Modules may provide functionality to export sensor data to different

formats (e.g., ASCII, binary, or xml), different archives, make data discoverable through geospatial catalogues, or publish the data through web services.

Data visualization: Modules may provide visualization (e.g., tables, graphs, sonograms) of geospatial and/or time series data from sensor arrays or workflow structures.

Documentation: Modules may be used to document field events through paperless collection of field data, integrate sensor data and documentation (see sensor tracking & documentation section), or handle sensor calibration records.

Other supported functionality: Modules may be used to provide access to external data (e.g., ODBC, JDBC, OLE DB), to connect or chain other middleware components, or to implement mobile applications.

Classification by propriety and type

Middleware can also be classified by *software proprietary rights* and *whether they are considered applications or platforms.* Accordingly, we can identify different groups of middleware:

- Proprietary data management applications and platforms
- Proprietary research applications
- Limited open source applications (free packages that can be used with proprietary solutions)
- Open source data management applications and platforms
- Open source research applications and programming languages

Some of the applications and platforms listed above are often identified as a software of choice for many different organizations. More details about each of these components are provided in the next section of this document.

Best Practices

Choosing the middleware components that will best fit the tasks and work environment can be challenging. In addition to the personnel roles and skills, budget, and infrastructure considerations already discussed in the Introduction and other chapters of this best practices guide, it is important to be aware of the whole sensor management process in order to identify the suitable middleware components. In some cases, a proprietary middleware software will required as part of the information management system if the instrumentation only outputs data in a proprietary format. In other cases, multiple open source software packages may be suitable for chaining into a comprehensive system that manages data from collection to final archiving and sharing.

Some steps in selecting middleware are:

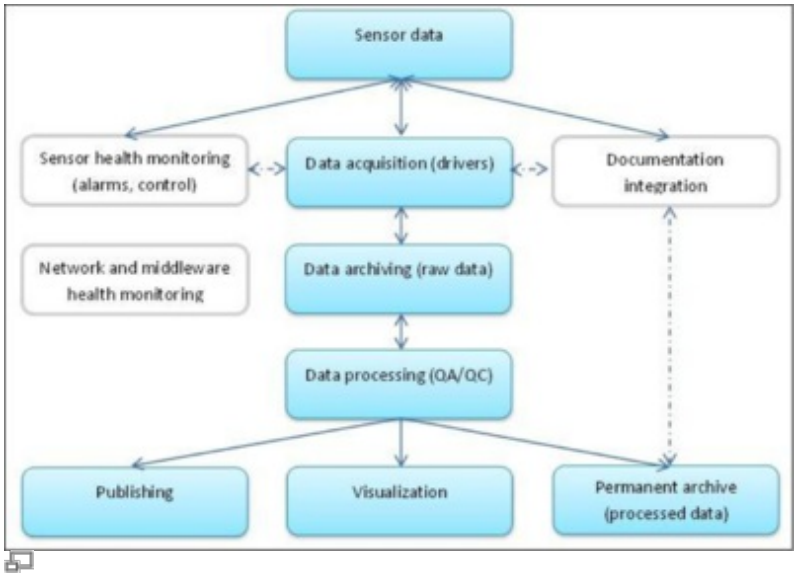

Fig 2. Sensor management workflow. Simple sensor management configuration is presented in blue; optional system components are shown in grey.

1. Identify your objectives. What do you want the middleware to do?
2. Assemble a list of candidate software.
3. Rate the candidates based on capabilities, cost (keeping in mind that a simple-to-use but expensive package may cut costs in the long-term), stability, and ease of use with respect to the personnel skills available on your team.
4. If no single software product can meet all the objectives, test to see how well different candidate software integrate with one another to perform the needed functions.

During this planning stage, consider the following recommendations:

- Identify workflow components and describe their functional requirements from the instrumentation to the archive level of organization (see Figure 2). Some components can be optional or part of the more complex solutions.
- Plan for robust execution and choose software and hardware components that can handle the loss of connectivity, power, or other failures related to harsh environmental or operational conditions.
- Choose reusable/sharable components.
- Keep field deployment of middleware as simple as possible (keep out of field if possible).
- Use as few middleware components as possible based on research group requirements.
- Document and diagram the entire workflow and update as needed.

Case Studies

We present several real world case studies in that vary widely in the types of ecosystems that sensors are deployed in and in complexity of the information management system. Some case studies include proprietary software only, some include free or open-source software, and some include both.

Marmot Creek Research Site, Rocky Mountains, Canada

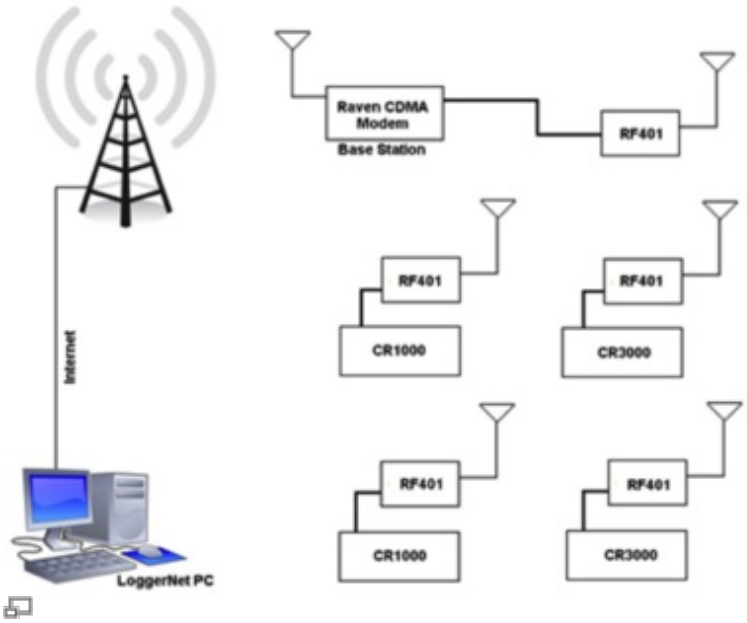

Fig. 3. Marmot Creek research site's PacBus Network with mixed data loggers and Raven to RF401 Base

Introduction Marmot Creek research site is located on the eastern slopes of Rocky Mountains in Alberta, Canada. The site is dominated by the needle leaf vegetation and poorly developed mountain soils. Precipitation, snow depth, soil moisture, soil temperature, short and longwave radiation, air temperature, humidity, wind speed, and turbulent fluxes of heat and water vapour data sets are collected and used for the hydrological modelling of the Marmot Creek Basin. Time series records are obtained at Hay Meadow, Upper Clearing , Vista View, Fisera Ridge, and Centennial Ridge hydro-meteorological stations equipped with different sensor configurations and Campbell Scientific data loggers.

Communication equipment and methods The telemetry network consists of one Raven CDMA cellular modem and RF401 spread spectrum radio modem located at the Upper Clearing base station, four additional RF401 modems located at each of the Meteorological stations serviced by telemetry, and the desktop computer located at the University of Saskatchewan. The radios connected to the data loggers at each of the meteorological stations talk to the base station on an ongoing basis. All of the data loggers and RF401 radios have PacBus addresses and they operate as PacBus Nodes. Also, data loggers are set to operate as routers enabling routing inside this network through the various paths. The telemetry network configuration is presented in Figure 3.

Data collection and processing At the intervals prescribed within the LoggerNet application running on a desktop computer, data is collected from the meteorological stations. The Raven CDMA transfers data utilizing a dynamic IP address and its static alias associated through the Airlink IPmanager software. The unique PakBus address is assigned to each of the dataloggers in this telemetry network. In most cases, logger data files at the off-site location will be appended on a daily, four-hourly and hourly basis. In addition to the scheduled intervals, field data can be downloaded on demand through the LoggerNet application.

LoggerNet “Task Master” utility is used to execute custom programs after each successful collection of the field data. Also, the utility can be used to start scheduled executions of different programs and operations. For Marmot Creek records, Task Master is used to rename the collected data logger files and upload them to the FTP server.

Data publishing Field data downloaded to the off-site computer are accessed by the RTMCPRO LoggerNet utility. Last measured values are mapped to the specified locations on a web page. The web server hosts different RTMC files for daily summary information, station data tables, alarms, and other records. The main interface contains individual windows for the main screen web page as well as the screens for individual stations, weekly data graphs, and site information. RTMC files interface with the web page via the RTMC Web Server desktop utility.

Reference Centre for Hydrology, University of Saskatchewan. University of Saskatchewan Hydrology Field Data Retrieval and Management Manual. 2009. PDF file.

University of Texas at El Paso's System Ecology Lab, Jornada research site, NM

Middleware used: Hobolink, MySQL, R, ArcGIS, HTML5/Javascript website

Introduction The Systems Ecology Lab (SEL) at the University of Texas, El Paso studies patterns and controls of land-atmospheric water, energy, and carbon fluxes in both arctic and desert biomes. At the USDA-ARS Jornada Experimental Range in southern New Mexico, SEL's research site collects data using >100 automated sensors (made by Campbell, Onset, Decagon, PPSystems, and others), and manual field observations. Sensors are mounted on an eddy covariance tower, eight connected mini-towers (which together form a wireless sensor network), and a cart mounted on a 110m long tramline. >4 GB of data is collected per week from micromet, hyperspectral, and gas flux sensors, as well as cameras (detect changes in phenology). This research site is also used to help develop and test new cyberinfrastructure and information management concepts and tools. For this case study, we focus solely on measurements made by the 8-node wireless sensor network.

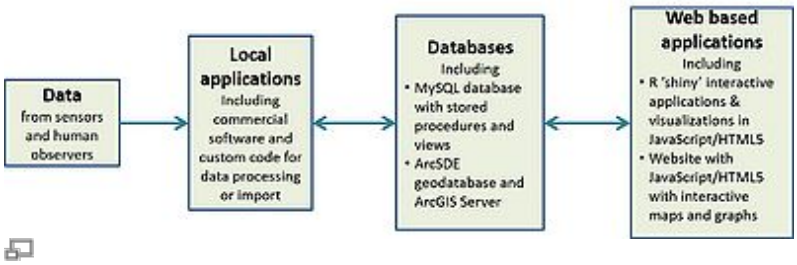

Fig. 4. SEL-Jornada research information system framework in terms of data flow (symbolized by arrows). Web services are used by web-based applications to query data from databases.

Data Collection and Processing The 8-node wireless sensor network is composed exclusively of Onset's Hobo data loggers (8) and sensors (62). Each data logger is powered by its own solar panel. Sensors measure precipitation, leaf wetness, PAR, solar radiation, and soil moisture. Data are relayed to the Jornada Headquarters and sent to Onset. The data are available for visualization and downloading via Hobolink (http://www.hobolink.com). The online service allows users to set up alerts for system malfunctions and automated reporting of data. Our team developed a database schema (using MySQL for implementation) that uses a core common concept among all datasets - a measurement on a focal entity by an observer at a specific location and time - to organize data. Around this core, there are other tables to store metadata such as project information, maintenance records, and related files. The wireless sensor network is imported into the database via custom SQL scripts. Within the database, scheduled queries can do basic data quality-checking and flagging, and generate tables of summarized data (e.g., daily means) that can be accessed shortly after the raw data were imported.

Data Publishing Our team wrote web services in Python to query the summarized data and deliver the results (in JSON format) to a JavaScript/HTML5 website in which AmCharts' JavaScript Charts library (http://www.amcharts.com/; free for non-commercial use) are used to plot the data and generate reports including the data and pertinent metadata. This website also renders a map of the site and sensors via an ESRI ArcSDE geodatabase and ArcGIS Server.

Middleware Package Descriptions

In this section, several middleware packages are described as a basic introduction to the packages. No specific endorsement or criticism is implied, and it is important to keep in mind that software packages are often revised.

> ***Aquatic Informatics Aquarius:*** AQUARIUS is a software for water time series data management that provides functionality to correct and quality control time series data, build rating curves, transform and visualize the hydrological data, as well as to publish the field data in real-time. To accomplish these tasks, AQUARIUS uses three main components. Data Acquisition System enables accessing the real time data from the field instrumentation either as the extension to the EnviroSCADA or through the hot folders. Aquarius Server is a web controlled data management platform that enables centralized access

to the database stored data sets. Also, the server is used to publish the data either through the public web portal or as a Representational State Transfer (REST) web service that supports the WaterML representation of the data. Finally, AQUARIUS Workstation provides a set of data processing tools to import and process the data, create rating curves, and apply QA/QC procedures to the time series records.

Aquarius system components
Aquarius modeling approaches

Campbell Scientific LoggerNet: LoggerNet is the main Campbell Scientific software application used to set up, operate, and manage a sensor network that uses Campbell Scientific equipment. LoggerNet uses serial ports, telephony drivers, and Ethernet hardware to communicate with data loggers via cellular and phone modems, RF devices, and other peripherals. LoggerNet also includes a suite of tools such as text editors for creating Campbell Scientific datalogger programs, and methods for real-time monitoring, automated data retrieval, data post-processing, data visualization and monitoring of retrieved information, and data publishing options. More advanced features, such as export to to MySQL or SQL Server databases, are also offered through additional LoggerNet applications not included in the standard version (LoggerNet Database, LoggerNet Admin, LoggerNet Remote etc.).

LoggerNet 4.1 Instruction Manual

CUAHSI HIS: The Consortium of Universities for the Advancement of Hydrologic Science, Inc.'s Hydrologic Information System (CUAHSI HIS) is an advanced web service based system created to share the hydrologic data. The system is comprised of hydrologic databases running the CUAHSI Observations Data Model (ODM) and servers hosted by different organizations that are connected through the web services. Centralized HIS modules are used for data publication, access, and discovery, while local (and central) modules provide tools for data analysis and visualization. Overall, CUAHSI HIS is used to store the observation data in a relational data model (ODM), access the data through internet-based Water Data Services that publish the observations and metadata using a consistent Water Markup Language (WaterML), index the data through a National Water Metadata Catalog, and provide a discovery of data through a map and keyword search system.

CUAHSI HIS components
CUAHSI HIS list of publications
Development of a Community Hydrologic Information System

Data Turbine Initiative: DataTurbine is a real time streaming data engine that acts as a black box to which data providers (sources) send data and consumers (sinks) receive data from. DataTurbine is implemented as a multi-tier java application with servers accepting and serving up the data, sources loading the data onto the servers, and sinks pulling the data for visualization and analysis purposes. Each of these components can be located on the same machine or different computers and can communicate with each other over the internet. Data is heterogeneous and the sinks could access any type of data seamlessly. While new data is loaded to the server(s), old data is being erased in order to free the receiving buffers.

DataTurbine – Sensor Networks Workshop
Understanding DataTurbine

GCE Data Toolbox: The Georgia Coastal Ecosystems (GCE) Data Toolbox is a software library for metadata-based processing, quality control, and analysis of environmental data. It is designed and maintained by Wade M. Sheldon, Jr. of the Georgia Coastal Ecosystems LTER and is available free of charge, but does require a MATLAB license. The Toolbox can be used for a wide variety of environmental data management tasks such as: importing raw data from environmental sensors for post-processing and analysis; performing quality control analysis using rule-based and interactive flagging tools; gap-filling and correcting data using gated interpolation, drift correction and custom algorithms/models; visualizing data using frequency histograms, line/scatter plots and map plots; summarizing and re-sampling data sets using aggregation, binning, and date/time scaling tools; synthesizing data by combining multiple data sets using join and merge tools; mining near-real-time or historic data from the USGS NWIS, NOAA NCDC, NOAA HADS or LTER ClimDB servers;

harvesting and integrating channel data from Data Turbine servers. This software is highly modular and can be used as a complete, lightweight solution for environmental data and metadata management, or in conjunction with other cyber infrastructure. For example, newly acquired data can be retrieved from a Data Turbine or Campbell LoggerNet Database servers for quality control and processing, then transformed to CUAHSI Observations Data Model format and uploaded to a HydroServer for distribution through the CUAHSI Hydrologic Information System.

GCE Toolbox overview (Georgia Coastal Ecosystems LTER)

Kisters WISKI: WISKI software package is a tool for hydrological data management. WISKI is a Windows based client/server system hosted through the MS SQL or Oracle databases. The software combines data management features with tools to collect, store, analyze, visualize, and publish the observation data. Typical data input sources are remote data collected from the field data loggers, data imported from third parties via input files in different formats, records obtained from digitization of graphical charts, or manual inputs. Main WISKI module incorporates the data management functionality as well as the discharge and rating curve tools that work closely with other Kisters software components including KiWQM (water quality), KiWIS (data publishing through web services), SODA (telemetry hardware module for remote data collection), KiDSM (task scheduler), Modeling apps (Link-and-Node and statistical forecast), ArcGIS extensions, Web Public and Web Pro (web server publishing applications).

WISKI system overview
WISKI modules

NexSens iChart: NexSens iChart is a Windows-based data acquisition package designed for environmental monitoring applications. iChart supports interfacing both locally (direct connect) and remotely (through telemetry) with many popular environmental products such as YSI, OTT, and ISCO sensors. Additionally it can interface with a NexSens iSIC and submersible data loggers. The software simplifies and automates many of the tasks associated with acquiring, processing, and publishing environmental data.

[http://nexsens.com/pdf/nexsens_wqdata_spec.pdf NexSens WQData and iChart software overview
iChart software product spotlight (Lake Scientist)
NexSens data website (Bucknell University)
iChart quick start guide

Onset Hobolink and Hoboware: Onset has two main software applications to support its Hobo data loggers and sensors.. Hobolink is an online services that provides 5-minute data from its data loggers, multiple graphs of data streams, customizeable interface, settings for automated alerts for sensor malfunction, and customizeable data reporting features. Hoboware is a downloadable package that provides more functionality, such as line charts for more than one data stream, charting types that are unavailable in Hobolink, etc.

HOBOware Pro vs. HOBOware Lite List of features
HOBOware® User's Guide (Data visualization and analysis)
HOBOlink® User's Guide (Data access and control of HOBO devices)

Vista Data Vision: VDV is a data management system with tools to store and organize data collected from a variety of data logger "dat" files. The software offers different visualization, alarming, reporting, and web publishing features. Data logger files are parsed, imported, and stored into the MySQL relational database from where the data can be custom queried and exported or published on a web server. Numerous access control options are available so VDV users can have customized access to specific station or sensor data.

Vista Data Vision brochures and manuals
Vista Data Vision version comparison
Vista Data Vision Review (LTER)

YSI EcoNet: EcoNet software works with YSI monitoring instrumentation. The software offers delivery of data from the field directly to the YSI web server. No desktop applications are used and all data are

stored on the remote YSI computer. System users can access visualization, reports, alarms, and email notification tools directly on the YSI server.

EcoNet system overview
Embedding EcoNet data

The following tables describe features of middleware packages known to the authors of the wiki. These tables do not imply endorsement or criticism of any given product, and may reflect older versions of products than currently exist.

- Basic: The software has built-in but basic features compared to the overall market.
- Standard: The software has built-in features that are standard with comparison to the overall market.
- Advanced: The software has built in advanced features compared to the overall market.
- Custom: The software doesn't have built-in features, but a programmer can develop them.
- None: The software doesn't have the feature, and it cannot be custom-developed.
- Has: The software has built in features, but the level compared with the overall market is unknown currently.
- Unknown: The capacity of the software is unknown.

Table 1. Middleware basic features: licensing, cost, input and export data formats, and required level of programming expertise.

Program	Licensing	Cost	Input data format	Export data format	Needed programming expertise
Antelope Orb	Proprietary	Pay	ASCII, Binary	ASCII, Binary	Advanced
Aquarius	Proprietary	Pay			Advanced
ArcGIS	Proprietary	Pay	ASCII, shapefiles	ASCII, shapefiles	Advanced
B3	Open source	Free	ASCII	ASCII	None to Basic
BigSense and LtSense	Open source	Free	Binary	CSV, JSON, TXT, XML	Advanced
Cosm					
CUAHSI HIS	Open source	Free	ASCII	XML, WaterML	Standard
DataTurbine	Open source	Free	ASCII, Binary	ASCII, Binary	Advanced
EddyPro	Proprietary	Pay	Binary	ASCII, Binary	Standard
GCE Toolbox	MATLAB is proprietary, Toolbox is open source	MATLAB is pay, Toolbox is free	ASCII, Binary, database	ASCII, Binary, .mat, database	Toolbox is Standard, MATLAB is Advanced
Hobolink (Onset)	Proprietary	Free	Proprietary	ASCII, Proprietary	None
Hoboware (Onset)	Proprietary	Pay	Proprietary	ASCII, Proprietary	None
Kepler	Open source	Free	ASCII, Binary	ASCII, Binary	Basic to Advanced
Lake Analyzer	Proprietary/Open source	Free	ASCII	ASCII	Basic
LoggerNet (Campbell)	Proprietary	Pay	Proprietary	ASCII, database	Standard

Nexsen's Technology	Proprietary	Pay	Unknown	Unknown	Unknown
Pandas	Python is Free, Pandas is Free and Open Source	Binary, encoded, np.array, database, markup	Binary, encoded, np.array, database , markup	Advanced and Custom	
Pegasus	Unknown	Unknown	Unknown	Unknown	Unknown
R	Open source	Free	ASCII, Binary, database	ASCII, Binary, database	Standard to Advanced
SAS	Proprietary	Pay	ASCII, Binary, database	ASCII, Binary, database	Standard to Advanced
Taverna	Open source	Free	Unknown	Unknown	Standard to Advanced
Vista Data Vision	Proprietary	Pay	ASCII	ASCII	Unknown
VizTrails	Open source	Free	ASCII	ASCII	Basic to Advanced
WaterML support	Unknown	Unknown	Unknown	Unknown	Unknown
WISKI	Proprietary	Pay	ASCII	ASCII	Advanced
YSI EcoNet	Proprietary	Pay	Unknown	Unknown	Unknown

Table 2. Middleware data handling features: hardware communication, ability to do quality assurance and control (QA/QC), ability to stream data to archives, data visualization, data transformation and analysis, and ability to generate custom SQL queries or other scripting.

Program	Hardware communication	QA/QC capacity	Capacity to stream to archive	Data transformation and analysis	Data visualization	Custom SQL queries/Scripting
Antelope Orb	Custom	Custom	Custom	Custom	Custom	Custom
Aquarius	Has	Advanced	Advanced	Advanced	Advanced	Advanced
ArcGIS	Unknown	Advanced	Unknown	Advanced	Advanced	Standard to Advanced
B3	None	Advanced	None	Has	Has	None
BigSense and LtSense	Custom	Custom	Has	Has	Unknown	Unknown
Cosm	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown
CUAHSI HIS	Custom	Advanced	Advanced (ODM)	Advanced (HydroDesktop, ODM Tools, TSA)	Advanced (HydroServer TSA, HydroDesktop, external programs)	Advanced (HydroDesktop, ODM Tools)
DataTurbine	Custom	Custom	Custom	Basic (NEES RDV)	Basic (NEES RDV)	Has
DataFrames.jl	Through C and Python libs	Has, stats.jl, numpy	Has through code.native	Has through Gadfly, Matplotlib, D3, or Winston	Has	Has
EddyPro	Unknown	Has	Unknown	Has	Has	Unknown
GCE Matlab	Custom	Advanced	Standard to Advanced	Advanced (with Matlab)	Advanced	Advanced

Hobolink (Onset)	Basic	None	Basic	Standard	None	None
Hoboware (Onset)	Advanced	Has	Unknown	Advanced	Standard	None
Kepler	Custom	Custom	Custom	Custom	Custom	Custom
Lake Analyzer	None	Basic	None	Has	Has	None
LoggerNet (Campbell)	Advanced	Basic	Basic	Basic	None	None
Nexsen's Technology	Has	None	Basic	Basic	None	None
Pegasus	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown
R	Custom with open-source tech	Custom	Custom	Advanced/Custom	Advanced/Custom	Custom
SAS	None	Custom	Custom	Advanced	Advanced	Custom
Taverna	Unknown	Custom	Custom	Custom	Custom	Custom
Vista Data Vision	None	Basic	Standard	Standard	Standard	Basic
VizTrails	Unknown	Custom	Unknown	Custom	Custom	Custom
WaterML support	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown
WISKI	Has	Advanced	Advanced	Advanced	Advanced	Advanced
YSI EcoNet	Has	None	Basic	Basic	None	None

Table 3. Middleware other features: Task automation, capacity for multi-tier architecture, website publishing, streaming through web services, support for modeling.

Program	Task automation	Multi-tier architecture	Website publishing	Streaming through web service	Support for modeling
Antelope Orb	Has	Standard	Custom	Custom	Unknown
Aquarius	None	Advanced	Advanced	Unknown	Unknown
ArcGIS	Advanced	Unknown	Advanced	Advanced	Advanced
B3	Unknown	None	None	None	Has
BigSense and LtSense	Has	Has	Has (via RESTful services)	Has	Unknown
Cosm	Unknown	Unknown	Unknown	Unknown	Unknown
CUAHSI HIS	Advanced (ODM SDL)	Advanced	Advanced (HydroServer, Website, HydroSeek)	Advanced	Has (through external programs)
DataTurbine	Has	Standard	Basic	Standard	None
EddyPro	Has	Unknown	Unknown	Unknown	Unknown
GCE Matlab	Advanced	None	Advanced	None	Advanced (with Matlab)
Hobolink (Onset)	Basic	None	Advanced	Has	None
Hoboware (Onset)	Basic	None	None	None	None
Kepler	Custom	Unknown	Custom	Custom	Advanced/Custom

Lake Analyzer	Custom (Matlab)	None	None	None	Basic (link to GLM model)
LoggerNet (Campbell)	Standard	Basic	Basic	None	None
Nexsen's Technology	Basic	Basic	Basic	None	None
Pegasus	Unknown	Unknown	Unknown	Unknown	Unknown
R	Custom	None	Custom (*shiny* package)	Custom	Advanced/Custom
SAS	Custom	None	Custom	Custom	Advanced/Custom
Taverna	Custom	None	Unknown	Custom	Advanced/Custom
Vista Data Vision	None	Standard	Advanced	None	None
VizTrails	Custom	None	Custom	Custom	Advanced/Custom
WaterML support	Unknown	Unknown	Unknown	Unknown	Unknown
WISKI	Advanced	Advanced	Advanced	Advanced	Has
YSI EcoNet	Basic	None	basic	None	None

References

"OGC WaterML Standard Recommended for Adoption as Joint WMO/ISO Standard." Open Geospatial Consortium, 10 Dec. 2012. Web. 14 May 2013.

Sensor Data Quality

Discusses different ways sensor data may be compromised and how to control for it in the data stream.

Return to EnviroSensing Cluster main page

Contents

Contacts

The primary editors for this page may be contacted for questions, comments, or help with content additions.

Don Henshaw – U.S. Forest Service Research, Pacific Northwest Research Station – don.henshaw at oregonstate.edu
Mary Martin – Hubbard Brook LTER, University of New Hampshire – mary.martin at unh.edu

Overview

A new generation of environmental sensors and recent major technological advancements in the acquisition and real-time transmission of continuously monitored environmental data provides a major challenge in providing quality assurance (QA) and quality control (QC) for high-throughput data streams. Deployments of sensor networks are becoming increasingly common at environmental research locations, and there is a growing need to access these large volumes of data in near real-time. However, the direct release of streaming sensor data raises the likelihood that incorrect or misleading data will be made available. Additionally, as research applications begin to rely on real-time data streams, the continual and consistent delivery of this information will be essential. This increasing access and use of environmental sensor data demands the development of strategies to assure data quality, the immediate application of quality control methods, and a description of any QA/QC procedures applied to the data.

Traditional QC systems tend to operate on file-based collections of environmental data from field sheets, field recorders or computers, or downloaded datalogger files. Manually applied tools and techniques such as graphical comparisons are used to provide data validation. Documentation is typically not well-organized and not directly associated with data values. The application of these systems must balance the need for release without months or years of delay versus the delivery of well-documented, high quality data. However, with increasing deployment of sensor networks, these older systems fail to scale or keep pace with user needs associated with high volumes of streaming data. Comprehensive and responsive QC systems are needed that are designed to reduce potential problems and can more quickly produce high quality data and metadata. Methods described here for building a QC system will include identification of:

- preventative measures to be taken in the field
- quality checks that can be performed in near real-time
- necessary data management practices

Introduction

A team approach is necessary to build a QC system and multiple skills and personnel are needed. The QC system will begin with system design and preventative measures taken in the field and continue through data quality checking and data publishing. A lead scientist will propose research questions and describe the types of data and necessary quality. Expertise in field logistics, sensor systems and wireless communications will play a role in site design and construction. A sensor system expert will provide knowledge of specific sensors and programming skills to establish quality control checking. Field technicians with strong knowledge of the overall scientific goals and communication skills can help to articulate issues and discover solutions. A data manager will be needed to guide delivery and archival of documented data products. Communication among all parties is necessary for the most timely delivery of well-documented and high quality data.

All team members will be needed to define a QC workflow that is useful in describing procedures and personnel responsibilities as the data flows from field sensors to published data streams. A QC system must allow for an iterative, quality management cycle to accommodate feedback to policies, procedures, and system design as data collections continue over time. A system will depend on communication among team members to assure that noted sensor data collection and transport issues and problems are addressed quickly and documented in the data stream. An active, well-documented QC system will help to establish user-confidence in data products.

Automated or semi-automated QC systems are needed that can adequately review and screen source data and still provide for its timely release. Automated quality control processes such as range checking can be performed in near real-time and a system can assign data qualifier codes, or flags, for any sensor value when problems or uncertainty occurs in the data stream. However, these processes can often only indicate potential problems in the data stream that still require manual review. A comprehensive QC system is only achievable as a hybrid system demanding both automated QC checks and manual intervention to assure highest data quality.

For this chapter we will define quality assurance (QA) as those preventative processes or steps taken to reduce problems and inaccuracies in the streaming data. These will include sensor network design, protocol development for routine maintenance and sensor calibration, and best practice procedures for field activities and data management. Quality control (QC) primarily refers to the tests provided to check data quality and the assignment of data flags and other notations to qualify issues and describe problems. QC system refers to this complete set of QA/QC preventative and product-oriented processes.

Methods

Sensor Quality Assurance (QA)

Quality assurance (QA) refers to preventative measures and activities used to minimize inaccuracies in the data. For example, scheduling regular site visits and maintenance procedures, or continuously monitoring and evaluating site sensor behavior can prevent sensor failures or lead to early detection of problems. Designing networks with redundant sensor measurements provides an additional means to quality check sensor data and assure continuity of measurement. Of course, the time and expense to conduct high-level maintenance procedures or implement efficient and redundant designs may be limited by project budgets, but may be warranted by the importance of the data. Here we describe QA measures categorized by design, maintenance, and practices:

1. Design
    a. Design for replicate sensors. Co-located sensors independent of the datalogger and included in the data flow can be useful checks. For example, check temperature measurements might be made alongside a Campbell thermistor with a HOBO pendant, SDI-12 temperature sensor, or analog thermocouple. Ideally, three replicate sensors are used so that sensor drift can be

detected (with two sensors it may not be obvious which sensor is drifting).

b. Assure an adequate power supply. Power considerations might include adding a low voltage cutoff (LVD) to prevent logger "brown-out", or adding power accessories with switched power supply (e.g. CSI logger, IP relay) to programmatically control optional devices (radios, power-cycle loggers).
c. Protect all instrumentation and wiring from UV light, animals, human disturbance, etc. such as with flex conduit or enclosures.
d. Implement an automated alert system to warn about potential sensor network issues or certain events, e.g., extreme storms. For example, automated alerts might signal low battery power, indicate sensor calibration is needed, or indicate high winds or precipitation.
e. Add on-site cameras or webcams. Webcams can be used to record weather or site conditions, animal disturbance or human access.

2. Maintenance
   a. Schedule routine sensor maintenance. Routine site visits following standard protocols can assure proper maintenance activities.
   b. Standardize field notebooks, check sheets or field computer applications to lead field technicians through a standard set of procedures and assure that all necessary tasks are conducted. These notebooks or applications can serve as an entry point for technical observations regarding potential problems or sensor failures.
   c. Schedule routine calibration of instruments and sensors based on manufacturer specifications. Maintaining additional calibrated sensors of the same make/model can allow immediate replacement of sensors removed for calibration to avoid data loss. Otherwise, sensor calibrations can be scheduled at non-critical times or staggered such that a nearby sensor can be used as a proxy to fill gaps.
   d. Anticipate common repairs and maintain inventory replacement parts. Sensors can be replaced before failure where sensor lifetimes are known or can be estimated.
   e. Assure proper installation of sensors (correct orientation, clean wiring, solid connections and mounting, etc.). Protocols for installing new sensors will also assure that key information is logged regarding a sensor's establishment (See Management section).
3. Practices
   a. Maintain an appropriate level of human inspection. Develop the capability to easily view real-time data and examine regularly (daily/weekly). Regular inspection can help identify sensor problems quickly and might allow for fewer site visitations. Certain problems such as visible extreme spikes, intermittent values, or repetitive values can be easily viewed in raw data plots.
   b. Spot check measurements with a reference sensor can be routinely used for some measurements, i.e. temperature, snow depth, etc. to verify the performance of in situ sensors.
   c. A portable instrument package that can be rotated among sensor sites can be useful in identifying problems. The portable package might run alongside installed sensors over a fixed period (daily or longer cycle) to inspect for drifting or failing sensors. This type of co-location might be done to audit sensor performance on an annual or periodic basis.
   d. Record the date and time of known events that may impact measurements (see Management section). Ideally, these notes can be entered or captured for automated access. For example, sensors are known to demonstrate alternative behavior during site visits or maintenance activities, and light or trip sensors might be used in recording sensor access.
   e. Routinely synchronize the time clock on dataloggers with the public Network Time Protocol (NTP) server (http://www.ntp.org/).
   f. Provide a reference time zone and avoid changing data logger timestamps for daylight savings time. Many would argue the best practice is to output data in Coordinated Universal Time (UTC), which is particularly useful when data spans multiple time zones. However, most local users of the data prefer seeing output in local standard time because it corresponds to local ecological conditions, i.e., ocean tides or solar noon, and may ease troubleshooting or field-based checking. Another strategy is to provide the local offset from UTC within the data stream to allow simple conversion to UTC, or allow users to query the data and choose whatever time zone they would like to receive the data in. ISO 8601 (http://www.iso.org/iso/home/standards/iso8601.htm) is an international standard covering the exchange of date and time-related data and provides timezone support. For example, 2013-09-

17T07:56:32-0500 provides the offset from an EST timezone, however, lack of support in many instruments and software packages is a drawback to its use. Recently, REST services are constructed to allow the return of datetime values with an implicit timezone offset enabling convenient sharing of data with timestamp flexibility.

g. Ensure that files stored on the logger are transmitted error-free to the data center for import (use error-corrected protocols like FTP, Ymodem and HTTP). Schedule manual file download and post-import checks if non-error-corrected protocols are used as an interim measure.

Quality Control (QC) on data streams

Quality Control of data streams involves automated or semi-automated processes whereby values and associated timestamps are cross-checked against predetermined standards and separate concurrently-collected data streams. QC takes place post-collection during the streaming process or after data is assimilated into a central database. Some processes can be performed in "near real-time", or at the time the data streams are brought into the database, and data can be released as "provisional" after this initial inspection to satisfy immediate user needs. Other processes may require some delay such as trend analysis for sensor drift detection. Results of these tests are typically accounted for in a data qualifier flag for each value. Manual inspection and resolution of suspect or problem data is also a necessary step before data is released with "provisional" tags removed. Revised or corrected data versions can be published at a later date, and it is important to provide documentation on the types of quality checks conducted with each release of these data.

Three categories of automated or semi-automated QC processes can be described:

1. independent evaluation, whereby a single data point is checked against predetermined standards (such as range checks)
2. point-to-point evaluation, whereby a single data point is compared to other concurrently-observed data points (such as replicate sensors)
3. many-point, or trend analysis, where some timeframe of observations are examined statistically or against other data trends. The first two are essentially near real-time checks, whereas the third can involve timeframes several orders of magnitude longer than the measurement interval.

Near real-time processing involves automated checking of each data point and its associated date and time. Data qualifier codes, or data flags, will be assigned based on these checks. These automated checks and flag assignments are essential in processing the mass volumes of data streaming from sensor networks, but are not sufficient. Human inspection of data is critical and particularly might focus on data points that are flagged by an automated system. The following terminology corresponds with quality control tests listed in Campbell et al. 2013.

The most common and simplest checks to implement

1. Timestamp integrity checks – ensures that each date-time pair is sequential. With fixed interval data it is possible to cross-check the recorded and expected timestamp.
2. Range checks - ensures that all values fall within established upper and lower bounds. Bounds can be established based on the specific sensor limitations, or can be based on historical seasonal or finer time-scale ranges determined for that location. Separate flags might be assigned to qualify impossible values (based on sensor characteristics) versus extreme values that are outside of the historic norms but within the sensor operating range.

Other checks can be employed for near real-time or in post-streaming QC

1. Persistence - checks for repeated, unchanging values in measures where constant change is expected.
2. Spike detection - checks for sharp increases or decreases from the expected value in a short time interval such as a spike or step function. These tests often employ statistical measures such as the standard deviation of the preceding values in detecting outliers or spikes that exceed 2-3 sigma (standard deviations) from what is expected. An alternative algorithm is to check to see that the

median value of points t, t+1 and t-1 is not more than a fixed magnitude from point t.

3. Internal consistency – plausibility checks for consistency between related measurements such as that the maximum value is greater than the minimum value, or that snow depth is greater than its snow water equivalence. These checks may also examine values that are not possible under known conditions such as incoming solar radiation recorded during nighttime.
4. Spatial consistency – checks for sensor drift or failure based on intersite comparisons of nearby identical sensors. The integration of several data streams may be possible in post-processing and drifting may be detected based on known correlations or prior conditioning with redundant or nearby sensors.

Data qualifiers (data flags)

The QC system must be able to assign one or more codes to each data point based on the result of QC tests or other available information. Data flags may be assigned during the initial QC tests that are intended to guide local review in identifying erroneous or problematic data (e.g., invalid values out of range or below detection level), or might be flags that indicate site-specific events (e.g., low battery voltage, an icing or other event or site condition, or notification of a due date for sensor calibration). These internal flags may use a richer vocabulary of fine-grained flags than what is necessary to share publicly. Reviewing internal flags is necessary to resolve issues that may be evident in the data before these data are made available in final published versions. Some systems might employ a "rejected" flag as a means of preserving an original value but allow capability to withhold that value from public use.

External flags provided in published data will likely be a more general, simpler suite of flags better suited for public consumption. Multiple internal flags would be mapped into this more general flag set. While many vocabularies are in use, an example suite of external flags follows:

- A: Accepted
- E: Estimated
- M: Missing
- Q: Questionable
- Specification of uncertainty

The "Accepted" flag should be assigned to values where no apparent problems are discovered, but the QC tests that were applied should be described. The "Accepted" flag is likely less commonly used than simply leaving the flag blank. If the blank flag is used it should be included in the list of flags and defined, e.g., "no QC tests were applied" or "no recognizable problems" or "provisional data". A blank flag can be included in an enumerated listing of valid flags but may not be the best practice within some metadata standards. A "Provisional" flag is not listed here but may be appropriate. Alternatively, "provisional" data might be indicated within a "quality level" attribute on the record level or file level rather than associated with an individual measurement (See Data Quality Level section below).

Examples of Quality Flag Sets (listed codes may only represent a subset of each flag set)

Andrews LTER	WISKI (Univ. of Saskatchewan)	HFR LTER	VCR LTER	SeaDataNet
A - Accepted	10 - Rejected	M - Missing	Blank - OK	0 - no QC
E - Estimated	15 - Disregard	E - Estimated	Q - Questionable	1 - Good value
M- Missing	20 - Manually edited	Q - Questionable	M - Missing	2 - Probably good
Q - Questionable	25 - Simulated		R - Range Error	4 - Bad value
Measurement specific, e.g., B - Below detection	30 - Filled		S - Data Spike	6 - Below Detection

The evaluation of extreme values may benefit from "expert inspection" that can be built into the QC

system. Historical ranges can be developed for sites with long-term sensor measurements at annual, seasonal or finer time scales. For remote sites that are data sparse these ranges may be a primary tool for ascertaining data quality, and, for example, a QC system may flag values that fall outside of two standard deviations of long-term means. Where other nearby in situ measurements are available or where national surface station networks are available, quality checks may be improved through comparison of values. Access to multiple climate elements may provide the ability to create relationships among stations and allow specification of uncertainty for all values. Evaluation of a QC system's performance in determining uncertainty or in estimating values will be important in making system improvements and potentially allowing a retrospective re-application of quality control (Daly et al. 2005).

Where specifications of uncertainty cannot be determined, values may be deemed "Questionable" by an automated system. Ultimately, manual evaluation may be required and a decision made as to whether a data point can be released as "Accepted" versus removing from the data stream and listing as "Missing" versus leaving the value flagged as "Questionable". As Daly et al. 2005 points out, "in the end, the fundamental dilemma with nearly all quality control is a tension between the relative merits and costs of accidentally rejecting good data, or accidentally accepting bad data, and a tradeoff is usually involved".

Where data are missing, an option might be to fill gaps with "Estimated" data. From Campbell et al. 2013, "filling these gaps may enhance the data's fitness for use but can possibly lead to misinterpretation or inappropriate use, and can be a complex endeavor. The decision about whether to fill gaps and the selection of the method with which to do so are subjective and depend on factors such as the length of the gap, the level of confidence in the estimated value, and how the data are being used".

Data quality level

The level of QC testing applied to a set of data should be well-described and transparent to the data user. Publishing of data is independent of data quality, and users need to be able to quickly identify its quality level, for example, to discern whether the data is unchecked, raw data vs. thoroughly inspected and reviewed. Groups such as NEON and CUAHSI have assigned a quality level to data products including original raw data, initially inspected and flagged raw data, published raw data, and estimated, gap-filled or other synthetic products involving model-based or scientific interpretation (See references in data_quality_level.pdf). While these groups do not necessarily agree on the actual level assignment, there are some general concepts of quality level that can be agreed upon and are represented here:

Level 0 (raw) − Unfiltered, raw data, with no QC tests applied and no data qualifiers (flags) applied - Typically, these are original data streams that are not published but that should be preserved. Data quality flags are not assigned. Conversion of raw measurement values to more meaningful units may be acceptable, e.g., thermocouple table conversions of millivolts to degrees C.

Level 1 (provisional)− Provisional data released in near real-time with initial QC testing applied - Preliminary QC tests or data calibration are applied, potentially in near real-time through automated scripts. Data qualifiers are assigned and may be for internal use intended to guide further review of the data (See Data qualifiers subsection). All data qualifiers should be well-defined. Range and date-time checking are commonly applied to this provisional level. The QC tests applied should be well-described.

Level 1 (published) - Published data with a delayed release after automated and manual review - QC testing is complete and suspect data has been inspected and flagged appropriately. Each value is assigned a data qualifier and the set of flags may be a more simple set devised for public use of the data. Impossible or missing values would be assigned an appropriate missing value code and a data flag of "Missing". Data would no longer be considered provisional and would be unlikely to change.

Level 2 (gap-filled) - Gap-filled or estimated data involving interpretation - This is quality enhanced data where careful attention has been applied to estimate or fill gaps in data or to otherwise build derived data to accommodate data user needs, for example estimate gaps in a sensor stream using a nearby sensor. As gap-filling typically involves interpretation and may employ multiple models or algorithms, other versions of level 2 data may be used in practice. Methods employed in gap-filling or deriving data should be well-described.

Aggregating data from one time-step to another, e.g., creating daily summary data from 10 minute data, that does not involve any interpretation in that simple means, maximum, and minimums are determined would not necessarily alter the quality level. That is, mean daily temperature determined from level 1 (published) data would still retain a quality level 1. However, interpretation may be involved when determining an appropriate qualifier flag for the daily mean. For example, if some of the 10 minute observations are missing at what point does the daily mean also become missing (e.g., more than 20% are missing) or become questionable (e.g., more than 5% are missing). This type of processing may yield daily mean values that are best described as Level 2 as interpretation is involved.

Data collection interval

Data loggers offer the capability to easily output mean data values at multiple time steps, e.g., 10 minutes, hourly, daily. Saving values at multiple time steps may present an extra complication in the QC process as separate tables are usually stored for each timestep. When a single sensor measurement is reported at separate time steps, conflicting QC results may occur if both streams are QC'd independently. One strategy to simplify this problem is to output most or all data in the shortest common timestep and use post-processing to statistically aggregate the data at longer time steps. For example, a system might QC and output the 10 minute data and then aggregate hourly and daily values from this finer resolution 10 minute data stream. Dataloggers might typically calculate and output daily (24-hour) data streams, but accurate QC may be impossible as the exact values used in this aggregation are unknown, and the aggregation may be only representing a subset of values, e.g., if there was a power discontinuity to the logger. However, there may be cases where the output of daily values by the logger are important. For example, an instantaneous maximum or minimum value based on a single logger sample would not be captured through this aggregation, and a daily minimum or maximum based on a 10 minute or hourly mean output may differ significantly from the instantaneous value.

Data Management

Timing of QC system processes

Automated QC system procedures provide the most timely and efficient processing of streaming data. The use of system procedures provides consistent assignment of data flags and removes much of the subjectivity inherent in manual assignment. Ideally, the QC system will be employed every time data is acquired, e.g., every 10 minutes, and secondarily operate on hourly or daily time periods. More comprehensive visual or programmatic checks or the assignment of uncertainty using nearby or other related sites might occur at a later time. The frequency and timing of a manual or visual review processes will depend on the data flow at the site, software stack, and data processing capabilities. The necessary timeframe for data delivery of provisional versus fully processed data should be considered.

Documentation of the QC processes

The documentation of QC processes should identify the near real-time streaming QC methods including assumptions and thresholds, and additional algorithms or visual methods applied. If no QC is applied that should be made apparent. Descriptions of data processing and QC workflows are also useful in describing data provenance and all workflow versions should be retained (See example workflow). Data measurement attributes and qualifier flags should be defined.

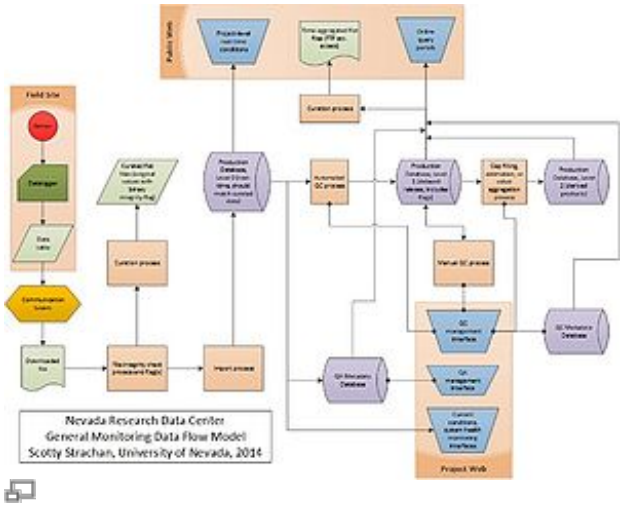

Fig. 1 Example of a general monitoring data flow model from Nevada Research Data Center, Scotty Strachan, University of Nevada, 2014

The application of the QC tests employed or any algorithms applied to aggregate, estimate or gap-fill data should be described for all data levels, and data levels can potentially be defined in conjunction with a data release policy. Ideally, data at each level should be locally archived. Level 0 raw data should be retained locally in its original, unmanipulated state. Level 1 (published) or level 2 data may be the best candidates for more formal archiving. Data sets should be transparently tagged with a data quality level as data are released.

Sensor data documentation

Develop and use a common vocabulary and syntax for sensor measurement attribute names and file naming conventions. Research organizations with multiple sensor sites measuring common sets of parameters can greatly improve efficiency and more easily employ automated methods when a common vocabulary is employed. These naming conventions should be planned from the outset into datalogger programs and other software employed within the data flow.

Data qualifier flags provide documentation for each measured value and should be placed alongside the value as data files are produced for archival storage. An additional attribute or method code may also be added to note shifts in method or instrumentation or other key changes in collection procedures. Inclusion of a method code directly within the data file places key documentation close to the data value and is more visible to the data user. In long-term data streams where the quality level may change over time, e.g., periods of time where gap-filling is employed, a data quality attribute might be used to assign data quality at the record or measurement level.

Best Practices

Reorganized from: Campbell et. al. 2013.

Sensor Quality Assurance (QA)

- Maintain an appropriate level of human inspection
- Replicate sensors, n=3 is optimal
- Schedule maintenance and repairs to minimize data loss
- Have ready access to replacement parts
- Record the date, time, and timezone of known events that may impact measurements
- Implement an automated alert system to warn about potential sensor network issues

Quality Control (QC) on data streams

- Ensure that data are collected sequentially
- Perform range checks on numerical data
- Perform domain checks on categorical data
- Perform slope and persistence checks on continuous data
- Compare data with data from related sensors
- Use flags to convey information about the data
- Estimate uncertainty in the value, if feasible
- Correct data or fill gaps if it is prudent

Data management

- Automate QA/QC procedures
- Retain the original unmanipulated data
- Indicate data quality level with each release of the data
- Provide complete metadata
- Document all QA/QC procedures that were applied and indicate data quality level
- Document all data processing (e.g., correction for sensor drift)
- Retain all versions of workflows and metadata (data provenance).

Case Studies

- *We are looking for case studies that will describe some complete QC systems, QC processing and general setup (e.g., number and type of sensors, dataloggers, telemetry, etc.)*
- *Examples using GCE Toolbox, Vista Data Vision, R, etc. would be useful*
- General workflow example from Nevada Research Data Center

References

Campbell, JL, Rustad, LE, Porter, JH, Taylor, JR, Dereszynski, EW, Shanley, JB, Gries, C, Henshaw, DL, Martin, ME, Sheldon, WM, Boose, ER. 2013. Quantity is nothing without quality: Automated QA/QC for streaming sensor networks. BioScience. 63(7): 574-585. http://www.treesearch.fs.fed.us/pubs/43678

Taylor, JR and Loescher, HL. 2013. Automated quality control methods for sensor data: a novel observatory approach, Biogeosciences, 10, 4957-4971 doi

Daly, C, Redmond, K, Gibson, W, Doggett, M, Smith, J, Taylor, G, Pasteris, P, Johnson, G. 15th AMS Conf. on Applied Climatology, American Meteorological Soc. Savannah, GA, June 20-23, 2005. pdf

Resources

QC Resources

- Campbell et al. 2013 Bioscience http://www.treesearch.fs.fed.us/pubs/43678
- Taylor and Loescher 2013. Biogeosciences http://www.biogeosciences.net/10/4957/2013/bg-10-4957-2013.pdf doi
- Daly et al. 2005. 15th AMS Conf. on Applied Climatology, Amer. Meteorological Soc. http://ams.confex.com/ams/pdfpapers/94199.pdf
- *CUAHSI* http://wdc.cuahsi.org/wdc/Docs/ODM1.1DesignSpecifications.pdf
- *NOAA Satellite and Information Service (National Climate Data Center)* http://www.ncdc.noaa.gov/oa/climate/ghcn-daily/
- *Carbon dioxide information analysis center* http://cdiac.ornl.gov/epubs/ndp/ushcn/daily_doc.html
- *SeaDataNet* http://www.seadatanet.org/Standards-Software/Data-Quality-Control
- *Data Quality Assessment: Statistical Methods for Practitioners* http://www.epa.gov/quality/qs-docs/g9s-final.pdf

Flag set examples

- NOAA National Climatic Data Center http://www.ncdc.noaa.gov/oa/hofn/coop/coop-flags.html
- Campbell et al. 2013 Bioscience (See p. 580) http://www.treesearch.fs.fed.us/pubs/43678

Data quality level

- NEON http://www.neoninc.org/documents/513
- CUAHSI http://his.cuahsi.org/documents/ODM1.1DesignSpecifications.pdf, pp. 19-20, 57-58
- Ameriflux http://public.ornl.gov/ameriflux/available.shtml
- Earth Science Reference Handbook http://eospso.gsfc.nasa.gov/sites/default/files/publications/2006ReferenceHandbook.pdf (p.31)
- ILRS Data products: (CODMAC - Committee on Data Management, Archiving and Computing) http://ilrs.gsfc.nasa.gov/about/reports/9809_attach7b.html

Sensor Data Archiving

Introduces different approaches and repositories for archiving and publishing data sets of sensor data.

Return to EnviroSensing Cluster main page

Contents

Overview

Archiving data snapshots and using appropriate metadata and packaging standards can increase the longevity and discovery of data immensely. However, these local curation techniques are still susceptible to threats to the projects or institutions that maintain the local archive. People in critical technology positions that maintain archives may change careers or retire, projects can lose funding, and institutions that seem solid can dissolve due to changes in political climate. For these reasons, partnering across institutions to provide archival services of data can greatly increase the probability that data will remain accessible for decades or into the next century.

In this chapter, we discuss techniques and issues involved with archiving data on a multi-decadal scale. For sensor data, we promote the use of periodic data snapshots, persistent identifiers, versioning of data and metadata, and data storage formats and strategies that can increase the likelihood that data will not only be accessible into the future, but will also be understandable to future researchers.

Introduction

A data archive is a location that has a reasonable assurance that data and the contextual information needed to interpret the data can be recovered and accessed after significant events, and ultimately after decades. Data archives should be maintained through backup strategies such as redundancy and offsite backup, in multiple locations and through institutional partnerships. Archiving activities should have institutional commitment, and ideally cross-institutional commitment. Archives may be locally maintained, may be part of a national or network-wide archive initiative, or both. For raw data, an archive can be a local or regional facility, whereas quality controlled, ‘published’ data should be archived in a community-supported network archive and available online.

Environmental research scientists are in need of accessing streaming data from sensor networks both provisionally in near real-time, after QA/QC processing, and in final form for long-term studies. Without appropriate archiving strategies, data are at great risk of total loss over time due to institutional memory loss, institutional funding loss, natural disasters, and other accidents. These typically include near-term accidents and long-term data entropy due to career and life changes for the original investigator(s) [Michener 1997]. Data, and the methods used to generate and process them, are often insufficiently documented, which may result in misinterpretation of the data or may render the data unusable in later research. Likewise, lack of version control or use of persistent identifiers for all files causes downstream

confusion, and hinders reproducible science.

Data managers are increasingly asked to both preserve raw data streams and to additionally provide automated, near real-time quality control and access to provisional data from these sensors. Typically, these provisional data streams undergo further visual and other quality checking and final data sets are published. Commonly, further interpretation occurs where some missing data are gap-filled through imputation procedures, or faulty data are removed. There is a strong need to archive these data streams and provide continued access, which ultimately safeguards the investment of both time and money dedicated to collect the data in the first place. There are a number of organizational, storage, formatting, and delivery issues to consider. However, four main archiving strategies should be used: creating well documented data snapshots, assigning unique, persistent identifiers, maintaining data and metadata versioning, and storing data in text-based formats. These practices, described below, will increase the longevity and interoperability of the data, and will promote their usefulness to current and future researchers.

Methods

Publishing of Snapshots

Generating periodic snapshots of near real-time sensor streams allows the data to be stored and described in a deterministic manner. The rate that snapshots are produced depends on the needs of the community using the data, but typically snapshot files are organized using hourly, daily, weekly, monthly, or annual datasets. It also depends on the sample rate and sample size. Producing thousands of tiny data files, or one file with gigabytes of data, would decrease the usefulness of the data from a transfer and handling perspective. Make it easy on the researchers using the data, and size the snapshots appropriately.

Without detailed documentation of the contextual information needed to interpret individual measurements, even well-archived data will be rendered unusable. Develop metadata files to accompany the data using a machine-readable metadata standard appropriate to the community using the data. Common standards include the ISO 19115 Geographic Information Metadata [ISO/TC 211, 2003], the Content Standard for Digital Geospatial Metadata (CSDGM) [FGDC, 1998], the Biological Profile of the CSDGM (FGDC, 1999], and the Ecological Metadata Language [Fegraus et al., 2001]. Also consider documenting sensor detailed deployment settings and processes with SensorML [OGC, 2000].

Likewise, snapshots of data that represent a time-series should be documented and packaged appropriately such that the relationships among files are clear. Many of the above metadata standards have their own means of linking data with metadata, however they are all implemented differently. Federated archiving efforts such as DataONE have adopted ‘resource maps’ [Lagoze, 2008] to describe relationships between metadata and data files in a language-agnostic manner. (See DataONE packaging in the Resources section, and the Open Archives Initiative ORE primer). Consider publishing resource maps of your data and metadata relationships to improve interoperability across archive repositories.

Once data collections are sufficiently described, delivery can also be a challenge. While providing resolvable links directly to the metadata and data files is a good practice, scientists often would like to be able to download full collections. Providing a service that packages files into a downloadable zip file is commonplace, but relationships between data and metadata can be lost. Consider using the BagIt specification (see BagIt in the Resources section) [Boyko, 2009], which provides simple additions to zip files such as a manifest file that maintains the machine-readable relationships between the items in the collection, while still allowing researchers to download data packages directly to their desktop.

Persistent Identifiers

The above snapshot archiving strategies hinge on the ability to uniquely identify each file or component of a package in an unambiguous manner. File names can often collide, particularly across unrelated projects. So, assigning unique, persistent identifiers to each file, and the originating sensor stream, is paramount to successful archiving. A persistent identifier is usually a text-based string that represents an unchanging set of bytes stored on a computer. Persistent identifiers should be assigned to science data objects, science

metadata objects, and other files that associate the data and metadata together, such as resource maps. Opaque identifiers tend to be best for persistence and uniqueness (like UUIDs), but can be less memorable. Systems such as the Digital Object Identifier service (DOI) and EZID can help in maintaining unique, resolvable identifiers (see UUIDs, DOIs, and EZID in the Resources Section). Each version of a file or products derived from files (see versioning below) should also have a persistent identifier. If snapshots of data are being extended with new data, a new version of the dataset needs to be published. Shorter identifiers are best, and avoid using spaces and other special characters in identifiers to increase compatibility in file systems and URLs. Ultimately, the use of persistent identifiers allows associated metadata to track the provenance of cleaned, quality assured data or other derived products, and promotes reproducible science and citable data.

Versioning

Data from sensor streams are usually considered 'provisional' until they have been processed for quality control, and multiple versions of the data may be generated. However, provisional data are often used in publications and are cited as such. That said, in order to support reproducible science using sensor data, each version should be maintained with it's own citable identifier. Overwriting files or database records with new values or with annotated flags will ultimately change the underlying bytes, and effectively break the 'persistence' of the identifier pointing to the data. This applies to metadata or packaging versions as well, and so care must be taken to plan in versioning within your storage system. Your versioning strategies of raw data will be dependent on your snapshot strategies (e.g. appending to hourly files, then snapshotting and updating metadata files, or alternatively, say, producing daily, weekly, monthly, or annual packages that include data files and metadata files for the time period of covered). However, by making citable versions, researchers will be able to access the exact bytes that were used in a journal publication, and peer review of studies involving sensor data streams will be more robust and deterministic.

Data Storage Formats

Sensor data may be stored in different structures, each with its own advantages and disadvantages. A suite of variables from one station and collected at the same temporal resolution may be stored within one wide table with a column for each variable, each time being one record of several variables. Alternatives might be a table for each variable or one table of the format of [time, location, variable, value]. This latter system may be value centric with metadata attached to each value or series centric with metadata attached to a certain time interval for one variable (e.g., a time series of air temperature between calibrations). No matter how you organize your data, long-term, archival storage file formats need to be considered. In the digital age, thousands of file formats exist that are readable by current software applications. However, some formats will be more readable into the future than others. As an example, Microsoft Excel 1.0 files (circa 1985), are not readable by Microsoft Excel 2012 since the binary format has changed over time in a backward-incompatible manner. Therefore, unless these files are continually updated year after year, they will be rendered unusable. The same is true for database system files (.dbf) that hold the relational table structures in commonly used databases such as Microsoft SQL Server, Oracle, PostgreSQL, and MySQL. Database files must be upgraded with every new database version so they do not become obsolete. A good rule of thumb is to archive data in formats that are ubiquitous, and are not tied to a given company's software. Archive data in ASCII (or UTF-8) text files preferably, since this format is universally readable across operating systems and software applications. If ASCII encoding would cause major increases in file sizes for massive data sets, consider using a binary format that is community-supported such as NetCDF. NetCDF is an open binary specification developed at the University Cooperative for Atmospheric Research (UCAR), and provides open programming interfaces in multiple languages (C, Java, Python, etc.) that are supported by many scientific analysis packages (Matlab, IDL, R, etc.). By choosing an archive storage format that isn't tied to a specific vendor, data files will be readable in decades to come even when institutional support for maintaining more complex database systems falls short.

Data Storage Strategies

For local archives, the most common storage strategy is to just directly store files in a hierarchical manner

on a filesystem. Many data managers use a mix of location, instrument, and time-based hierarchy to store files into folders (e.g. /Data/LakeOneida/CTD01/2013/06/22/file.txt). This is a very simple, reliable strategy, and may be employed by groups with little resource for installing and managing database software. However, filesystem-based archives may be difficult to manage when volumes are large, or the number of instruments or variables are growing and don't fit a straight hierarchical model.

Many groups use relational databases to manage sensor data as they stream into a site's acquisition system. Well known vendor-based solutions like Oracle Database and Microsoft SQL Server are often used, as well as open source solutions such MySQL and PostgreSQL. These systems provide a means of data organization that can promote good quality control and fast searching and subsetting based on many factors, beyond the typical location/instrument/date hierarchy described above. Databases also provide standardized programming interfaces in order to access the stored data in standard ways across multiple programming languages. From an archive perspective, the use of databases may help in managing data locally, but should be seen as one component of a workflow to get data into archival formats.

Local databases used for managing data should be backed-up regularly, ideally to an offsite location, and the underlying binary database file formats should regularly be upgraded to the newest, supported versions of the database software. Ultimately, data stored in databases should be periodically snapshotted and stored in archival file formats, described above, with complete metadata descriptions to enhance their longevity.

Although local filesystems and databases are the most practical means of managing data, they are often at risk of being destroyed or unmaintained over decadal scales. Natural disasters, computer failure, staff turnover, lack of continued program funding, and other risks should be addressed when deciding how to archive data for the long term. One of the strategies for best data protection is cross-institution collaborations that provide storage services for their participants. These sorts of arrangements can guard against institutional or program disollution, lack of funding, etc. Consider partnering with community supported archives such as the LTER NIS, or federated archive initiatives such as DataONE to archive snapshots of streaming sensor data (see both in the Resource Section).

Best Practices

The following list of best practices are taken from the above recommendations, as well as additional considerations when archiving sensor-derived data.

- Develop and maintain an archival data management plan such that personnel changes don't compromise access to or interpretation of data archives (potentially through University Library programs)
- Employ a sound data backup plan. Archived data should be backed up to at least two spatially different locations, far enough apart that they won't be affected by the same destructive events (natural disasters, power or infrastructure issues). Perform daily incremental backups and weekly complete backups that may be replaced periodically, and annual backups that won't change. (Crashplan, acronis)
- Generate periodic snapshots of near real-time sensor streams (acronis)
- Develop metadata files to accompany the data using a machine-readable metadata standard
- Assign persistent identifiers to science data objects, science metadata objects, and other files that associate the data and metadata together
- Maintain versioned files with their own citable identifier
- Preferably archive data in ASCII (or UTF-8) for text files, or community supported formats like NetCDF for binary format

- Archive all raw data, but all raw data do not necessarily need to be available online.However, assign a persistent identifier to each raw data file to be able to document provenance of the published, quality controlled data.
- Partner across institutions to provide archival services to mitigate programmatic losses
- Preferably make data publicly available that have appropriate QA/QC procedures applied.
- Assign a different persistent identifier for published datasets of different QC levels in an archive. In the methods metadata, document the provenance and quality control procedures applied.
- Document contextual information for each data point. i.e., in addition to assigning a quality flag, assign a methods flag which documents field events like calibrations, small changes, sensor maintenance, sensor changes etc. Include notes that handle unusual field events (e.g., animal disturbance etc.) Encode metadata for sensor-derived data using community and or nationally accepted standards.
- Ensure the timezone for all time stamps is captured. Datalogger are being manually set to a certain time. Consider daylight savings.
- Establish the meaningful naming conventions for your variables taking into account the type of observation that is archived, adjectives describing the location, instrument type, and other necessary variable determinants.
- Determine the precision for your observation values in advance
- Preferably follow a naming convention or controlled vocabulary for variables (See the Resources Section)
- Avoid using databases for archival storage, but use them for management and quality control. However, if databases are used for managing sensor data then periodic snapshots into ASCII or open binary data formats are recommended.
- Track changes to data files within metadata files to maintain an audit trail

Case Studies

1. LTER NIS

The NSF Long-Term Ecological Research Network Information System (LTER NIS) is the central data archive for all data generated by LTER research and related projects. All data including sensor data are submitted with metadata in the Ecological Metadata Language (EML). Data are publicly available through this portal and through DataONE, of which the LTER NIS is a member. Specific approaches to archive streaming sensor data are following the best practice recommendations given in this document: Datasets are submitted as snapshots in time and it is up to the site information managers to decide the length of time in each snapshot, i.e., how frequently a new dataset is submitted. Minimally quality controlled data sets are submitted, while the raw data are archived at each site. As of this writing no standards for quality control levels or data flagging have been adopted by the LTER community.

Features of the LTER NIS include:

- Public availability of data and metadata
- Congruence check - quality check of how well the metadata describe the structure of the data
- Use of persistent identifiers
- Strong versioning of metadata and data files in the system
- Member Node of DataONE
- Support of LTER and related projects data storage using access control rules

- Replication of data and metadata across geographically dispersed servers

2. KNB

The Knowledge Network for Biocomplexity (KNB) is an international network that facilitates ecological and environmental research on biocomplexity. For scientists, the KNB is an efficient way to discover, access, interpret, integrate and analyze complex ecological data from a highly-distributed set of field stations, laboratories, research sites, and individual researchers. The KNB repository has been storing and serving data for over a decade, and stores over 25,000 data sets.

Features of the KNB include:

- Public availability
- Metacat, an open source data management system
- Morpho, an open source, desktop metadata editor
- Support for any XML-based metadata language, but optimized for the Ecological Metadata Language
- Use of persistent identifiers
- Strong versioning of all files in the system
- Support for cross-metadata packaging using resource maps
- Cross-institutional partnering with the LTER and DataONE
- Support for both public and private data storage using access control rules
- Replication of data and metadata across geographically dispersed servers
- International participation, and support for multi-language metadata descriptions

Recent developments of the KNB include support for the DataONE programming interface (API) in both the Metacat and Morpho software products. This API promotes interoperability of archival repositories, and enables federated access to environmental data. Since the KNB products support this open API, anyone can create their own web or desktop applications that are optimized for their research community.

3. GIWS, University of Saskatchewan WISKI data archive

Global Institute for Water Security (GIWS) at the University of Saskatchewan is directly involved in the collection of the field data from different research areas including Rocky Mountains, Boreal Forest, Prairie, and others.

In addition to the “in-house” managed data, GIWS uses external data sets from organizations such as Environment Canada and Alberta Environment. Data management platform on which GIWS currently operates is the Water Information System Kisters (WISKI). This system is used together with Campbell Scientific LoggerNet software and custom .NET modules in automated tasks that handle data collection, centralized data processing, storing, and reporting. After processing, the environmental data sets are published and made available to specific groups of users through the Kisters WISKI Web Pro web interface and KiWIS web service. Both applications can query the centralized database and return data in the formats that are used for visualization or further processing and dissemination purposes.

Features of the GIWS system include public availability, use of persistent identifiers, support for cross-institutional partnering, data access control for different groups of users, support for OGC WaterML2 data format. [See GIWS in the Resources Section]

Resources

BagIt Zip file format: https://wiki.ucop.edu/display/Curation/BagIt

DataONE: http://www.dataone.org/what-dataone and http://www.dataone.org/participate

DataONE Packaging: http://mule1.dataone.org/ArchitectureDocs-current/design/DataPackage.html

Digital Object Identifier (DOI) System: http://doi.org

Ecological Metadata Language: http://knb.ecoinformatics.org/software/eml/

FGDC Metadata Standards: http://www.fgdc.gov/metadata/geospatial-metadata-standards

GIWS: http://giws.usask.ca/documentation/system/GIWS_WISKI.pdf

ISO 19115 metadata Standard - Geographic Information
http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=53798

EZID Identifier Service: http://n2t.net/ezid

LTER Network Information System: http://nis.lternet.edu

Open Archives Intiative Object Reuse and Exchange (Resource Maps) Primer:
http://www.openarchives.org/ore/1.0/primer.html

Universally Unique Identifiers (UUID): http://en.wikipedia.org/wiki/Universally_unique_identifier

CF Metadata http://cf-convention.github.io/

References

Biological Data Working Group, Federal Geographic Data Committee and USGS Biological Resources Division. 1999. CONTENT STANDARD FOR DIGITAL GEOSPATIAL METADATA, PART 1: BIOLOGICAL DATA PROFILE. FGDC-STD-001.1-1999
https://www.fgdc.gov/standards/projects/FGDC-standards-projects/metadata/biometadata/biodatap.pdf

Boyko, A., Kunze, J., Littman, J., Madden, L., Vargas, B. (2009). The BagIt File Packaging Format (V0.96). Retrieved May 8, 2013, from http://www.ietf.org/id/draft-kunze-bagit-09.txt

Fegraus, E. H., Andelman, S., Jones, M. B., & Schildhauer, M. (2005). Maximizing the value of ecological data with structured metadata: an introduction to Ecological Metadata Language (EML) and principles for metadata creation. Bulletin of the Ecological Society of America, 86(3), 158-168.
http://dx.doi.org/10.1890/0012-9623(2005)86%5B158:MTVOED%5D2.0.CO;2

Lagoze, Carl; Van de Sompel, Herbert; Nelson, Michael L.; Warner, Simeon; Sanderson, Robert; Johnston, Pete (2008-04-14), "Object Re-Use & Exchange: A Resource-Centric Approach", arXiv:0804.2273v1 [cs.DL], arXiv:0804.2273

Metadata Ad Hoc Working Group, Federal Geographic Data Committee. 1998. CONTENT STANDARD FOR DIGITAL GEOSPATIAL METADATA. FGDC-STD-001-1998.
http://www.fgdc.gov/standards/projects/FGDC-standards-projects/metadata/base-metadata/v2_0698.pdf

Michener, William K., James W. Brunt, John J. Helly, Thomas B. Kirchner, and Susan G. Stafford. 1997. NONGEOSPATIAL METADATA FOR THE ECOLOGICAL SCIENCES. Ecological Applications 7:330–342. http://dx.doi.org/10.1890/1051-0761(1997)007[0330:NMFTES]2.0.CO;2

Open Geospatial Consortium, Inc. (OGC). Url: http://www.opengeospatial.org (2010).