

Architecting Climate Change Data Infrastructure for Nevada

Michael J. McMahon Jr.¹, Sergiu M. Dascalu¹, Frederick C. Harris Jr.¹, Scotty Strachan², Franco Biondi²

¹ University of Nevada, Reno, Department of Computer Science & Engineering,
1664 N. Virginia St., Reno, NV, USA 89557
{mcmahon, dascalus, fredh}@cse.unr.edu

² University of Nevada, Reno, Department of Geography, DendroLab, MS 154,
1664 N. Virginia St., Reno, NV, USA 89557
scotty@dayhike.net, franco.biondi@gmail.com

Abstract. The NSF EPSCoR-funded Nevada Climate Change Project seeks to create a central, reusable, extensible infrastructure that can be used to collect geospatial climate data and information. Housing climate data for Nevada and its surrounding regions during the initial construction phases, the newly created system (with its central component: the Nevada Climate Change Portal) will ultimately be capable of storing any kind of geospatial data for multiple types of research, education and outreach activities. In order to meet the varied needs of the climate researchers, educators, students, and policy makers involved in the project, it was necessary to research and implement a new system architecture. The novelty of this architecture is that it addresses, in an extensible and robust manner, the end-to-end needs of all project stakeholders, implementing multiple sub-levels of architectural design that incorporate data acquisition from sensor networks, data storage using high-performance geospatially-enabled databases, asset tracking and management to improve data validation and verification, metadata collection and management, data curation, and advanced web-based data management and retrieval. The paper describes the proposed system architecture, discusses the major design challenges encountered, addresses some implementation points, and highlights the capabilities of the Nevada Climate Change Portal.

Keywords: Climate change, infrastructure, system architecture, web services, sensor network, climate research, education, public policy.

1 Introduction

Funded in 2008 by an NSF EPSCoR grant, the Nevada Climate Change Project (NCCP) seeks to address the topic of climate change with relation to Nevada while simultaneously building a reusable infrastructure for such research. Specifically, members of the project (organized into six components) interact to create a reusable, extensible infrastructure to gather geospatial climate data that is made available to researchers and interested entities. Simultaneously, other members will use that data and information to enable educational outreach, inform public policymakers, and fulfill the

raw data requirements of climate modelers and scientists. In essence, the project involves the creation of a data infrastructure while answering key scientific questions.

The responsibility of meeting the computing infrastructure and data needs of other project components falls largely to the Cyberinfrastructure component. This group of individuals – in conjunction with key members of other components – has collaborated to research and architect an end-to-end system that connects climate measurements and information from field equipment and domain experts to interested members of the public and scientific community. Amongst other aspects that are unique to this architecture are: the ability to collect potentially high-frequency climate measurements over a high-speed communication network, the central maintenance and storage of measurements in a geospatial database, the selective retrieval and representation (i.e. format) of data by/for consumers (i.e. researchers, modelers, external evaluators, etc.), and the incorporation of asset tracking and metadata to support long-term data management and curation. This architecture provides a high level of reliability, performance, and data validity and verification when compared to other climate-related data architectures and systems, such as those in Section 2. In lieu of a large number of implementation details, the current iteration of the system based upon this architecture is available at <http://sensor.nevada.edu>, though it may be password-protected until publicly released in 2011.

The remainder of this paper is dedicated to elaborating the architecture established for this project on multiple levels: overall, data collection, data processing and storage, and data availability. Section 2 provides background on the topic of climate research, the challenges of such activities, and similar attempts to create suitable solutions. In Section 3, the architectures (at various levels [1]) are described, with particular emphasis placed on the novelty and research involved in creating each optimized sub-architecture and its ultimate implementation. In Section 4, a discussion of challenges related to implementing each architecture is presented. Section 5 concludes with an outline of the future developments and enhancements that will be made to both the architectures and implementations of the system.

2 Background

Climate research is a broad and multi-faceted topic that frequently requires the talents of many individuals to produce both raw and interpreted data in a multitude of formats. In a typical scenario, field technicians install equipment – often consisting of power systems (e.g. solar panels), monitoring sensors (e.g. wind, radiation, rainfall, etc.), sensor processing equipment (e.g. data loggers), data storage (e.g. local and flash memory), and communication infrastructure (e.g. network protocols and interfaces, radios and cables, Internet connectivity and other telemetry services, etc.) – that climate scientists determine is necessary to answer specific research questions or monitor particular physical systems. The data collected by the sensor networks is then acquired and provided to the research scientists for analysis and evaluation.

This deceptively short description implies that climate research is a straight-forward, simple process of data collection. However, upon closer inspection (or during implementation), numerous subtle and important requirements (domain, functional, and non-functional) and questions emerge. For example:

- What are the sources of data for the system (e.g. sensor networks, data repositories, files, etc.)? Further, how is that data acquired or accessed, and is that access mechanism reliable, fast, intermittent, and/or slow?
- How and where are data stored and organized? Storage and retrieval decisions affect efficiency, effectiveness, safety/security, data redundancy (i.e. backups), and long-term extensibility. Climate measurements may cover a variety of spatial and temporal ranges, influencing the storage mechanism.
- How can the system effectively address the dynamic long-term needs of data validation, curation, metadata management, many of which are “fast-moving targets?”
- How are data made available to consumers (i.e. researchers)? Security, reliability, verifiability, and performance are all influenced by the access mechanisms utilized by consumers. Data formats and access mechanisms (e.g. web services, FTP access, etc.) must meet the interoperability needs of data consumers to ensure quality results and maximize utility. Further, different research areas may not require the retrieval of all data, or may require the use of aggregates of data over specific intervals.

Clearly, the broad nature of climate research requires that the architecture of any system designed to support its efforts address these issues, as does this. Failing to take these and other issues into consideration introduces the potential to invalidate or nullify the value of the research being performed, or create a system that satisfies immediate needs but that is unable to adapt to long-term changes. While it is often sufficient to assume that researchers would not falsify data or that the original data is accurate, when dealing with more controversial issues such as anthropogenically-modified climate change, data must be clearly accurate and verifiable, and the process by which it was gathered transparent and reliable.

The remaining areas in this section summarize existing attempts to resolve these and other issues related to climate research, as well as the challenges encountered deriving suitable architectures (and their sub-architectures [2], as appropriate).

2.1 Related Work

Many different agencies across the United States and around the world have funded climate research projects in efforts to better understand the physical world. The approaches of these agencies differ significantly, often emphasizing the optimization of one particular feature while sacrificing another (e.g. multiple climate data sources being aggregated into one location at the expense of uniformity and documentation).

Table 1 provides a summary of features that some of these systems provide as a result of their architectural decisions. These metrics indicate the primary type of data each agency manages, whether they provide or manage metadata, whether they enforce data uniformity, and the level of data search functionality they have made available. Each of the example entities in this section has made advances in the field of climate research, addressing one or more of the common problems in the field.

Table 1. A summary of features from various example climate data systems and projects. Here, basic search features are simple date and site search options; advanced include at least date, time, location, and individual parameter selection.

	Type of Data	Metadata	Uniform Data	Search Features
WRCC [3]	Atmospheric	Partial	Yes	Basic
NOAA [4]	Various	No	No	Basic
NCDC [5]	Various	Partial	No	Basic
CUAHSI [6]	Hydrological	Yes	Yes	Advanced
RGIS [7]	Various	Yes	No	Advanced

2.2 Challenges

The architecture of the NCCP is unique: it addresses the end-to-end needs of researchers, educators, policy makers, and climate data consumers. Unlike other architectures that are focused on a narrow set of data, users, or climate research, this architecture addresses the entire climate research process from data to public information. This broad focus introduced a plethora of challenges and issues that needed to be resolved, many of them with the architecture itself.

A chief concern faced was the creation of an extensible, flexible, long-term infrastructure that could be used by the NCCP and future research projects. This required the architecture to specify enough structure and interaction to leverage emerging technologies (e.g. NEON and Internet2) to address current project needs, while remaining flexible enough to adapt to future (25+ year) innovations. A great deal of interviews with key personnel helped to determine the non-functional requirements, such as connectivity and reliability, maintenance, verifiability, resilience, data capacity, interoperability with other systems, and data integration that needed to be fulfilled by the architecture and resultant systems.

Climate measurements themselves present a host of complex issues that need to be resolved, ranging from representation to metadata management. Not only is it common for measurements to utilize different units (i.e. metric or Imperial) and scales, but many data sources store string representations of values (e.g. longitude, latitude, etc.) that can vary significantly. Future climate measurements will be highly-precise, both spatially and temporally, as technological advances are made. Metadata values (e.g. sensor depth below the soil surface) must be selected for storage and later use in data curation activities. Most importantly, the measurements themselves must be handled and stored such that, at any given time, their authenticity and accuracy can be evaluated.

Measurement collection from sensor networks (Figure 1) posed many unique challenges, largely requiring flexibility in the system architecture. Responsible for transforming sensor electrical information into usable numerical values and recording it, remote monitoring systems (often called “data loggers” or “loggers”) are resource constrained, having limited power and (often intermittent) connectivity. Retrieving data from these loggers frequently requires the use of hardware-specific software, whose output varies with each manufacturer. The long-term storage and utility of any original files retrieved from these networks also vary, affecting data verifiability.

Even the needs of various end-users present an array of difficulties that affect the architecture. For instance, different scientific users need particular subsets of the

collected data, not entire files. Further, the available search and selection functionality affects the data storage and retrieval mechanism. Flexible search and selection tools allow for innovative analysis methods to quickly explore large data collections. Per NCCP requirements, data and information collected by the various components was targeted at a public – not completely scientific – audience, making interface selection and design more difficult. The ability to support various output formats and units of measurement for end-users (especially scientists) is required to maximize the utility of the system, as is support of standards such as those of the OGC [8, 9, 10].

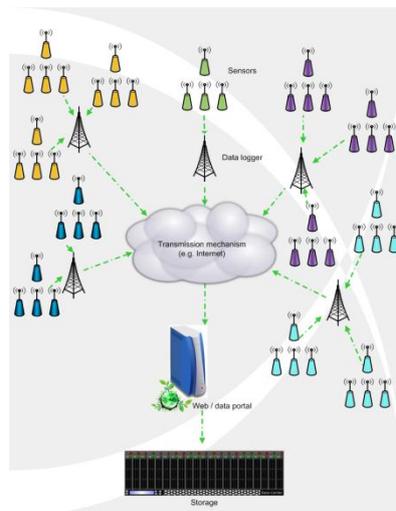


Figure 1. A general representation of the data collection process involving sensor networks.

Despite these challenges, the resultant architecture – as described in the next section – overcomes these hurdles, embodying a new, robust, extensible resource for climate research and, in effect, all fields that require reliable, verifiable geospatial data.

3 Architecture

After a great deal of research [1, 2, 11, 12, 13, 14, 15], the NCCP has been architected with a single, overarching architecture intended to facilitate the design and implementation of a robust, flexible, extensible infrastructure for climate research. In fact, this architecture is more broadly applicable to any kind of geospatial data system, not just climate research.

Similar to a tiered architecture or the TCP/IP model, the overarching architecture (Figure 2) addresses all the non-functional and domain requirements of climate research, creating a set of levels that communicate only with neighboring levels. Fulfillment of many of functional requirements is delegated to an optimized sub-architecture, while the interaction between the levels is dictated by the larger architecture. The sub-architectures then perform their role in the systems, communicating with necessary external resources and other sub-architectures in structured, predictable ways that isolate changes and ensure future extensibility.

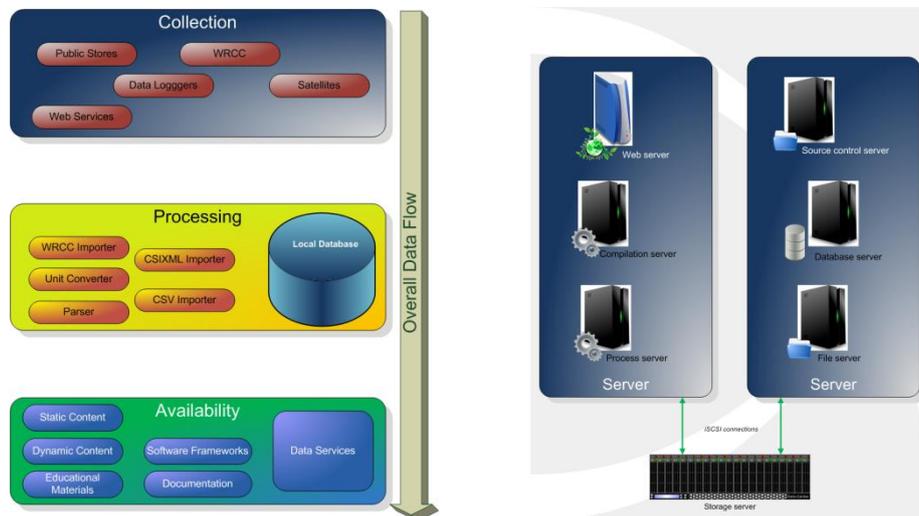


Figure 2. Overall architecture and data flow (left) and server architecture (right).

This architecture – based upon the climate research process itself – organizes functionality into three layers, which are discussed next: Collection (Section 3.1), Processing (Section 3.2), and Availability (Section 3.3). Specifically, the typical climate data acquisition sequence follows the order: 1) collect data from sensors using sensor monitors, 2) transfer and integrate data from data repositories (i.e. sensor monitors) to central servers, and 3) distribute collected data to consumers. This mirrors the real-world process of collecting data, storing and processing it, and making it available for use.

3.1 Collection

This level encompasses all data sources – whether national repositories, sensor networks, data loggers, or files – that the system collects for centralized storage in the Processing level. The interaction between the Collection and Processing levels is not specific to any manufacturer or hardware, allowing any device or entity to provide data to the system, optimizing its own internal operation. Additionally, this allows the use of transparent network technologies, intermediate data services, and other similar features without explicit internal knowledge being passed to other levels. The only requirement is that data be *pulled* from that source by the Processing level.

The specification of a pull-interaction between levels addresses availability constraints common in sensor network deployments. Specifically, the fact that sensor networks may suffer from limited bandwidth and/or intermittent one- or two-way communication that result from environmental conditions or power constraints. This allows the Processing level to manage the retrieval process, handling disconnections or disruptions on its side. Although one-way communication (i.e. GOES satellite) is supported *within* this architecture, it must have an intermediate data service or layer that supports the pull (and, thus, two-way) interaction when providing data to the Processing

level. Any implementation must ensure reliable transmission / transfer of data internally and to the Processing layer, eliminating transmission-induced data aberrations.

The compartmentalization of data loggers and general data sources into this layer has several advantages. Firstly, it allows the data source to optimize its internal operations accordingly. In the case of data loggers and sensor networks, this isolates the transparent details of network optimization and most aspects of logger programming to the remote installation, exposing only communication concerns with the Processing level. Secondly, this isolation ensures that the measurements and data are tamper-proof, insofar as they are not exposed to the public or researchers before storage. By removing the end-use (i.e. researcher) from the collection and processing interactions, the opportunity for data alteration (or accusations thereof) is eliminated, allowing later verification of any derivative results. Third, this allows multiple connection mechanisms between the Collection and Processing levels, depending on the specific availability constraints of the implemented hardware. Specifically, this allows manufacturer-specific or optimized software residing in the Processing level to pull data from sources in this level.

Also incorporated into this layer are user-provided data. This includes manual observations, metadata, asset management information, maintenance logs, and administrative logging information. While these are responsible for a comparatively small amount of information that is provided via the interfaces of the Availability level, they still represent an abstract data source whose data is collected by the Processing level. As such, they are a part of this level/sub-architecture.

3.2 Processing

The Processing level (or sub-architecture) is responsible for retrieving data from the Collection level and incorporating it into its data systems. After this data is incorporated, it is made accessible by the Availability level via query requests. To support the advanced, high-performance data retrieval and search needs of the Availability level, this layer specifies the use of a geospatially-capable database or other efficient mechanism.

This level involves the use of manufacturer-provided or customized data retrieval protocols, routines, and software to connect to data sources in the Collection level. Via whatever synchronization or heuristic mechanism they wish, the software obtains the original measurements and stores them in a non-modifiable location – these non-modifiable files provide a mechanism for detecting any potential data alteration during import into the database, as well as providing the ability to perform long-term data validation through comparison.

Once located in read-only storage, the retrieved data is imported into the database using whatever mechanisms the database, storage engine, or system supports. As this is a part of the internal architecture of this level, it ultimately has no bearing on any interaction with the Availability level. The import process requires the use of software or routines to take the read-only data and decompose it into an efficient, format-neutral structure used by the database. Because each data source may provide different data formats or representations, the import routines must losslessly transform the original data into this format, applying any unit conversion or format parsing necessary.

In this architecture (shown in Figure 3, using a database) a layered database schema has been developed to ensure long-term performance, maintainability, and extensibility

[16, 17, 18]. This schema consists of a core set of tables that store scalar geospatial measurements. All import routines decompose vector or other composite measurements into scalar values, storing them in these core tables. The structure of these tables does not change, allowing the efficient, selective retrieval of individual measurements, categories, or other aggregations of data. Rather, as the system requirements or needs evolve, additional supplemental or support tables are added. For example, import routines may utilize information stored in some of these tables; when new import functionality is required, additional tables may be added without affecting the query functionality required by the Availability level. In this way, the core tables and related performance requirements are shielded from breaking changes while the ability to extend functionality to encompass any data source is retained.

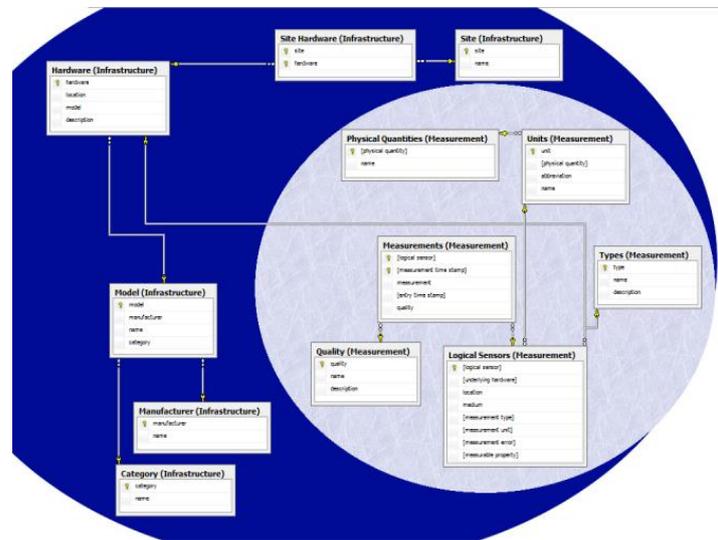


Figure 3. Core database tables (gray) and extensibility-supporting buffer tables (blue).

This database schema organization, while independently developed, is similar in some ways to that of the CUAHSI system [19, 20, 21]. Although designed to handle hydrological data, some developments in the CUAHSI system were later integrated into the database schema – particularly, the use of “controlled vocabularies” to categorize and represent measurements.

Isolation of these operations to this architectural level has several advantages. First, direct access to the data sources is not allowed from the Availability level, securing the original data from tampering. Second, quality assurance and quality control operations are centralized within the data store, eliminating the need for complex interactions with other components. Third, the storage engine is free to be optimally configured (i.e. performance) without affecting the operations of any other level. Fourth, data management activities such as metadata management and data curation can be carried out within this level, isolated from any other operations. Fifth, asset tracking and management information stored in this level can be correlated to sensors and measurements, adding verifiability to the measurements. Finally, the mechanism by which data is retrieved from this level by the Availability level is left for

implementation, allowing optimal communication mechanisms to be used. All that is required is for this level to provide the data retrieval and management functionality necessary for the Availability level.

3.3 Availability

The availability sub-architecture is responsible for retrieving data from the processing sub-architecture to satisfy the needs of data consumers (i.e. researchers, scientists, modelers, etc.). The communication mechanism between the processing and availability sub-architectures is implementation-specific, allowing database-specific features to be utilized optimally.

In large part, this layer exists to shield the processing layer from direct public access and to fulfill formatting and selection requests from end-users. Data selections are supported based upon the type of scalar value (i.e. solar radiation, soil moisture, etc.), the timeframe, geospatial region, physical property measured, originating project, monitoring station, and other features that implementers wish to support. This level also performs formatting (e.g. HTML, CSV, Excel, and XML) and conversion (i.e. units) operations when fulfilling data requests.

Data accessibility is also implemented in this level. Specifically, the transport and request protocols and communication mechanisms that support data selection and formatting requests located in this level. Any type of required industry-standard data access protocols/formats/methodologies are implemented as a part of this sub-architecture. This may include future and current access protocols such as REST or SOAP, or standards such as FGDC or OGC.

As with the other levels, organizing user activities in this level has many advantages. First, security and data access efforts are centralized, preventing direct access to the data store of the Processing level. Second, any data access technology can be implemented in this layer without affecting the Processing level – any format or protocol requirement can be fulfilled by using structured queries into the Processing level and transforming the results. This allows future data access standards to be implemented without requiring extensive system refactoring activities. Third, the isolation of user interfaces to this level allows the use of any technologies to meet the needs of users. For example, this level would house a web portal for public information, web services for climate modelers, FTP services for some researchers, and advanced search interfaces for other researchers. This layer is the primary point for public extensibility and system expansion.

4 Discussion

The derivation of an appropriate architecture to support climate monitoring activities is not a simple matter. While the general domain requirements seem relatively simple, the analysis of hardware feature variability and the limitations of supporting software quickly complicate the schema. Incorporating the varied needs of end-users (e.g. verifiability, specific research areas, etc.) and collaborating entities (i.e. data synchronization/sharing) present additional difficulties that necessitate the careful creation of an architecture that meets all requirements, yet avoids any specific implementation-level restriction to retain the greatest level of flexibility.

The architecture we have composed and implemented for the NCCP addresses these complex and diverse needs in a flexible, extensible, consistent manner. As described, the overall architecture effectively addresses the non-functional and domain requirements of the systems required to collect and disseminate data from a disparate collection of climate data sources to researchers and other data consumers. The division of functionality in this architecture allows each individual sub-architecture to optimize and arbitrarily extend its performance and features, communicating with adjacent sub-architectures in structured ways.

In implementing this architecture for the NCCP, several important, yet optional, decisions were made that improved system performance and extensibility. These decisions and their impact are discussed in Sections 4.1, 4.2, and 4.3.

4.1 Standards

As described in Section 3, the architecture defines the general interaction and functionality between different levels. While there is no restriction or requirement to use a particular standard at any level, the implementation of policies and standards within the NCCP has simplified implementation efforts and improved the verifiability of collected data. The policies responsible for these improvements target the Collection and Processing sub-architectures.

Chief amongst these policies is the decision to make all data stored in the database of the Processing level immutable. That is, it can only be removed by a system administrator, never altered by anyone. In cases of manual measurements, erroneous measurements must be deleted and re-entered; all modification is disallowed. Any corrective action undertaken by a system administrator is logged within the system.

Within the Collection sub-architecture, programming policies have ensured that the data loggers record data values using consistent units (i.e. metric values), simplifying import efforts. This standardization allows the data loggers to perform a single conversion to the sensor voltage readings they measure, removing the possibility of compounded calculation rounding. In addition, this ensures that the original data values retrieved from the loggers match those stored in the database, simplifying data verification and validation activities.

Further, the clocks of all data loggers are required to support synchronization to a central server to the greatest degree possible – for two-way communication, clocks are not allowed to drift by more than a few hundred milliseconds at any given time (less for highly-precise measurements or experiments). This policy ensures that the collected values can be correlated, allowing for highly-accurate and inter-related data analysis by researchers.

4.2 Performance

Critical to the performance of the overall system is the performance of the Processing sub-architecture, which is responsible for the long-term storage and retrieval of data. While the mechanism by which data is stored and retrieved is not dictated by the architecture directly, the optimization of the sub-architecture merited the use of a geospatially-aware relational database.

The relational database at the heart of the Processing sub-architecture uses a core schema that does not change. Rather, other support tables are added to facilitate parsing and import routines for any required external source. This decision optimizes the performance of searches on the core tables (which are indexed appropriately), eliminating the gradual changes that would otherwise occur to these tables. Such changes are generally required as new data sources are added and additional, import-specific information is required to import values. Not only does this preserve data retrieval and search performance, but also ensures system extensibility by isolating additions from existing information.

4.3 Public Interaction

To meet the requirements of the NCCP, a web portal was implemented to provide information to project members, researchers, and the public (<http://sensor.nevada.edu>). By creating a central, visible repository for project information, researchers and component members are able to make their efforts visible to the public, encouraging interest and attracting future funding opportunities.

5 Future Enhancements

The NCCP seeks to integrate and transform data from many disparate sources and integrate future projects that wish to utilize this infrastructure. In essence, this means that there is always room for enhancement and improvement – the system is designed for it.

One set of enhancements involves the incorporation of additional data sources into the system, beyond the data loggers deployed as a part of the NCCP. Depending on the data source (e.g. NOAA and NCDC), either data must be incorporated into the internal database or data queries be forwarded to the remote source and processed by appropriate routines when retrieved. Data access routines for the Western Regional Climate Center have already been developed; import routine development is underway. After this source is incorporated, others will be added depending on the priority given them by researchers and other data consumers.

The implementation of data synchronization routines with the collaborating entities of New Mexico and Idaho will be completed to allow the replication of collected data between the partners. Although these are also general data sources, the collaborative development efforts of the three institutions will yield a more cohesive, integrated data access system than with other data sources that have completed their development cycles.

6 Acknowledgements

The material presented in this paper is based upon work supported by the National Science Foundation under Grant No. 0814372.

7 References

1. Fowler, M.: *Patterns of Enterprise Application Architecture*. Addison-Wesley Professional, New York (2002)
2. Microsoft Patterns & Practices Team: *Microsoft Application Architecture Guide*, 2nd Edition. Microsoft Press, Redmond (2010)
3. Western Regional Climate Center, <http://www.wrcc.dri.edu> (2011)
4. National Oceanic and Atmospheric Administration, <http://www.noaa.gov> (2011)
5. National Climatic Data Center, <http://www.ncdc.noaa.gov> (2011)
6. Consortium of Universities for the Advancement of Hydrologic Science, Inc., <http://www.cuahsi.org> (2011)
7. New Mexico Resource Geographic Information System Program, <http://rgis.unm.edu> (2011)
8. Open Geospatial Consortium: OGC Web Map Service Interface, Version 1.3.0. http://portal.opengeospatial.org/files/?artifact_id=4756 (2004)
9. Open Geospatial Consortium: Web Feature Service Implementation Specification, Version 1.1.0. http://portal.opengeospatial.org/files/?artifact_id=8339 (2005)
10. Open Geospatial Consortium: Web Coverage Service (WCS) Implementation Standard, Version 1.1.2. http://portal.opengeospatial.org/files/?artifact_id=27297 (2008)
11. Jansen, A., Bosch, J.: *Software Architecture as a Set of Architectural Design Decisions*. In: *Fifth Working IEEE/IFIP Conference on Software Architecture (WICSA'05)*, pp. 109--120. IEEE Press, New York (2005)
12. Schmidt, M.-T., Hutchison, B., Lambros, P., Phippen, R.: *The Enterprise Service Bus: Making service-oriented architecture real*. In: *IBM Systems Journal*, Volume 44, Issue 4, pp. 781--797. IBM Software Group, Hursley Park (2005)
13. Lu, X.: *An Investigation on Service-Oriented Architecture for Constructing Distributed Web GIS Application*. In: *2005 IEEE International Conference on Services Computing*, vol. 1, pp. 191--197. IEEE Press, New York (2005)
14. Tsonis, A.A., Roebber, P.J.: *The Architecture of the Climate Network*. In: *Physica A: Statistical and Theoretical Physics*, vol. 333, pp. 497--504. Elsevier, Maryland Heights (2004)
15. Rew, R., Davis, G.: *NetCDF: An Interface for Scientific Data Access*. In: *IEEE Computer Graphics and Applications*, vol. 10, issue 4, pp. 76--82. IEEE Press, New York (1990)
16. Delaney, K., Randal, P., Tripp, K., Cunningham, C., Machanic, A.: *Microsoft SQL Server 2008 Internals*. Microsoft Press, Redmond (2009)
17. Ben-Gan, I., Sarka, D., Wolter, R., Low, G., Katibah, E., Kunen, I.: *Inside Microsoft SQL Server 2008: T-SQL Programming*. Microsoft Press, Redmond (2009)
18. Aitchison, A.: *Beginning Spatial with SQL Server 2008*. Apress, New York (2009)
19. Piasecki, M., Ames, D., Goodall, J., Hooper, R., Horsburgh, J., Maidment, D., Tarboton, D., Zaslavsky, I.: *Development of an Information System for the Hydrologic Community*. 9th International Conference on Hydroinformatics (HIC 2010) (2010)
20. Ames, D. P., Horsburgh, J., Goodall, J., Whiteaker, T., Tarboton, D., Maidment, D.: *Introducing the Open Source CUAHSI Hydrologic Information System Desktop Application (HIS Desktop)*. In: Anderssen, R.S., Braddock, R.D., Newham, L. T. H. (eds.) *18th World IMACS Congress and MODSIM09 International Congress on Modelling and Simulation* Modelling and Simulation Society of Australia and New Zealand and International Association for Mathematics and Computers in Simulation, July 2009, pp. 4353--4359 (2009)
21. Tarboton, D. G., Horsburgh, J. S., Maidment, D. R.: *CUAHSI Community Observations Data Model (ODM), Version 1.1, Design Specifications*. <http://his.cuahsi.org/documents/ODM1.1DesignSpecifications.pdf> (2008)