

CRWR Online Report 05-8

A Geotemporal Framework for Hydrologic Analysis

by

Jonathan L. Goodall, B.S.; M.S.

Graduate Research Assistant

and

David R. Maidment, Ph.D.

Principal Investigator

August 2005

CENTER FOR RESEARCH IN WATER RESOURCES

Bureau of Engineering Research • The University of Texas at Austin
J.J. Pickle Research Campus • Austin, TX 78712-4497

This document is available online via World Wide Web at
<http://www.ce.utexas.edu/centers/crwr/reports/online.html>

Copyright
by
Jonathan Lee Goodall
2005

**The Dissertation Committee for Jonathan Lee Goodall Certifies that this is the
approved version of the following dissertation:**

A Geotemporal Framework for Hydrologic Analysis

Committee:

David R. Maidment, Supervisor

Robert B. Gilbert

Daene C. McKinney

Bridget R. Scanlon

Zong-Liang Yang

A Geotemporal Framework for Hydrologic Analysis

by

Jonathan Lee Goodall, B.S.; M.S.

Dissertation

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Doctor of Philosophy

The University of Texas at Austin

August 2005

Dedication

To my wife for her love and support

Acknowledgements

I would like to thank my advisor, Dr. David Maidment, for guiding my graduate studies and for providing the vision for this research.

A Geotemporal Framework for Hydrologic Analysis

Publication No. _____

Jonathan Lee Goodall, Ph.D.

The University of Texas at Austin, 2005

Supervisor: David R. Maidment

The wealth of publicly available hydrologic data from observation networks, satellite-based sensors, global-scale modeling efforts, and digitized paper maps provide an excellent base-line of information for understanding hydrologic systems. It is difficult, however, to fuse this information into a holistic picture because there are many different formats and data models for storing, analyzing, and sharing such information. To address this problem, this research presents a prototype system for assembling local and remote data sources of various formats into a common geospatial-temporal (or simply geotemporal) framework for hydrologic analysis.

Starting from the basic concept of geographic information science that space consists of entities and fields, new concepts are derived for representing hydrologic space: geospatial time series and hydrologic flux coupler. A class library named HydroObjects is created that uses these new concepts to extend GIS

software for geotemporal visualization and processing of hydrologic data. The HydroObjects library provides interoperability between hydrology data because the attributes of an object can be populated from a variety of sources, formats, and data models.

As a case study, geotemporal visualization and processing extensions developed through this research are used to perform and visualize a water budget for subwatersheds of the Neuse River Basin, North Carolina. The water budget calculations are performed with data directly ingested from remote hydrologic data bases through the internet and then processed locally to align in space and time. The case study is an example of how hydrologists can gain access to remote information, fuse these data sources into a single picture of the geotemporal watershed environment, and use the information to formulate and test hypotheses regarding the hydrologic cycle.

Table of Contents

Chapter 1: Introduction	1
1.1 Background	1
1.2 Statement of Problem	4
1.3 Objective	8
1.4 Contribution to Science.....	9
1.5 Dissertation Outline.....	15
Chapter 2: Literature Review	17
2.1 Introduction to CUAHSI.....	17
2.2 Informatics in the Earth Sciences.....	24
2.3 Informatics in the Hydrologic Sciences	34
2.4 Representing Time in GIS.....	44
2.5 Summary	48
Chapter 3: Methodology.....	50
3.1 Hydrologic System Analysis.....	51
3.2 Representations of Space And Time	66
3.3 Designing a Geotemporal Framework	74
3.4 Hydrologic Object Classes (HydoObjects)	78
3.5 Summary	82
Chapter 4: Procedure of Application.....	83
4.1 Implementing HydroObjects	84
4.2 Example of Using the HydroObjects	107
4.3 Software Development Using HydroObjects.....	110
4.4 Accessing Remote Data	117
4.5 Summary	124

Chapter 5: Application	125
5.1 Defining the Hydrovolumes	126
5.2 Data preparation	131
5.3 Performing a Water Balance	138
5.4 Automating the Process Using ArcGIS Model Builder	142
5.5 Viewing Hydrologic Data in Space and Time	144
5.6 Summary	154
Chapter 6: Conclusions	155
6.1 Summary	155
6.2 Conclusions	157
6.3 Recommendations	160
Appendix A	164
Appendix B	168
Bibliography	188
Vita	192

List of Tables

Table 3.1: Hydrologic time series properties as defined by Maidment (2002)....	57
Table 4.1: The properties of a geospatial time series.....	87
Table 4.2 The enumerations within the geospatial time series object.....	87
Table 4.3: The methods for a geospatial time series.....	94
Table 4.4: The properties of a flux coupler class	105
Table 4.5: The methods of a flux coupler class.....	106

List of Figures

Figure 1.1: Twenty-four research groups from across the country have proposed watersheds to be funded as a CUAHSI Hydrologic Observatory (Image provided by CUAHSI).	3
Figure 1.2: The Great Salt Lake proposed Hydrologic Observatory website presents users two views of the basin: geospatial and temporal. One purpose of this research is to integrate these two separate views into a single geotemporal view of the watershed system.	8
Figure 1.3: Hydrologic values represented as geometries in space-time.....	10
Figure 1.4: The HydroObjects library facilitates the process of building software for visualizing and performing analysis with disparate data formats and sources.....	12
Figure 1.5: Geospatial Time Series are the combination of a georeferenced shape, an array of values and dates, and a collection of time series metadata properties.....	14
Figure 1.6: The Flux Coupler provides the ability to represent the exchange between discrete entities in space through their common interface.....	15
Figure 2.1: CUAHSI HydroView (Hooper et al. 2004)	19

Figure 2.2: Unidata’s Internet Data Distribution (IDD) network of servers (source: http://my.unidata.ucar.edu/content/software/idd/rtstats/index.html)	27
Figure 2.3: Map of the twenty-four LTER sites (source: http://ternet.edu/sites). 29	
Figure 2.4: The Long Term Ecological Research Network Information System (Baker et al. 2000).....	31
Figure 2.5: The GEON collaborators (Source: http://www.geongrid.org/)	32
Figure 2.6: UML for Arc Hydro Framework (Maidment 2002)	37
Figure 2.7: Arc Hydro Time Series component (Maidment 2002)	38
Figure 2.8: Arc Hydro facilitates integration of models through a common representation of the hydrologic system (Maidment 2004).	39
Figure 2.9: UML representation of a netCDF file (Nativi et al. 2004)	40
Figure 2.10: NetCDF files store multi-dimension variable fields such as surface evaporation (Source: North American Regional Reanalysis of climate visualized with Unidata’s Integrated Data Viewer).	42
Figure 2.11: Extending the Arc Hydro Time Series Component for additional spatial-temporal data types (Arctur and Zeiler 2004).	45

Figure 2.12: Minus 28°C Isosurface on December 11, 2000. The Integrated Data Viewer is an example of a spatiotemporal information system capable of visualizing 4-D fields. (Source: <http://www.unidata.ucar.edu/content/software/IDV/gallery/index.html>) 48

Figure 3.1: A conceptual picture of a control volume..... 52

Figure 3.2: Each feature within the hydrologic landscape can be considered a hydrologic system. 53

Figure 3.3: Stream discharge time series obtained from the USGS National Water Information System. 56

Figure 3.4: Different time series data types as defined by Arc Hydro (Maidment, 2002)..... 59

Figure 3.5: Common unit dimensions used in hydrology and the integrals used to convert between them..... 66

Figure 3.6: Viewing space as entities. Each catchment, river reach, and water body within the Neuse River Basin is a unique entity with properties such as a georeferenced shape or set of spatial coordinates. 68

Figure 3.7: Viewing space as fields. Space is a grid and variables are associated to specific locations within the grid. A field can represent either scalar or vectors. 69

Figure 3.8: A rain drop moving through the surface water system is an example of a dynamic entity in the Lagrangian viewpoint.....	73
Figure 3.9: Examples of spatiotemporal geometries	75
Figure 3.10: Coupling a geospatial time series and a hydrovolume occurs through the geospatial intersection of the two objects.....	81
Figure 4.1: A Unified Modeling Diagram (UML) of the HydroObjects class library.	84
Figure 4.2 This XML file is used to set the TSUnitType property	90
Figure 4.3: Algorithm for adding a geospatial time series object to a chart	96
Figure 4.4: Algorithm for adding a geospatial time series object to a chart space	97
Figure 4.5: Algorithm for changing the units of a geospatial time series	99
Figure 4.6: Algorithm for temporally rescaling a geospatial time series	101
Figure 4.7: Algorithm for writing a geospatial time series to an Arc Hydro geodatabase	102
Figure 4.8: Algorithm for generating the change in storage geospatial time series for a flux coupler object	107

Figure 4.9: TSPlotter: an extension to Arc Map for plotting local and remote time series associated to geographic features.....	112
Figure 4.10: The Space-Time Toolbox is an ArcGIS Geoprocessing Toolbox which can be used as tools in Model Builder (shown on left)	114
Figure 4.11: Modis data and NWIS Streamflow stations served by the USGS EROS data center as geospatial web services and viewed with ESRI's ArcMap. Neuse subwatersheds overlaying the remote data are stored locally in a geodatabase.....	119
Figure 4.12: Duke University's East Campus, which is in the Neuse River Basin, viewed using the TerraServer. The underlying relational database stores image data for the entire United States and allows client applications to access the images through web services.	122
Figure 5.1: The Neuse River Basin and a river reach within the Neuse as three dimensional hydrovolumes	127
Figure 5.2: The Neuse basin divided into watersheds.....	128
Figure 5.3: Two sub-watersheds of the Neuse River Basin.	129
Figure 5.4: Conceptualization of the water cycle (Source: http://www.usgcrp.gov/usgcrp/images/ocp2003/ocpfy2003-fig5-1.htm)	131
Figure 5.5: Interface for NWIS reader tool	134

Figure 5.6: Interface for NARR reader tool	134
Figure 5.7: Interface for Up-scale Attribute Series Tool	135
Figure 5.8: Interface for Batch Interpolation Tool.....	137
Figure 5.9: Interface for Batch Zonal Statistics Tool.....	137
Figure 5.10: The hydrologic flux coupling table and the TSType table for two watersheds within the Neuse Rive Basin	140
Figure 5.11: The Coupling table describes the relationship between a hydrovolume feature and its coupled exchanges of material represent by a geospatial time series vector.	141
Figure 5.12: Interface for flux coupler tool.....	142
Figure 5.13: An ArcGIS Model Builder model for performing a water budget analysis on the Neuse River Basin including automatic ingestion of web data, spatial and temporal rescaling, and coupling exchanges to watershed features for estimation of water storage through time.....	143
Figure 5.14: Visualizing monthly evaporation time series related to watershed features using the TSPlotter within ArcGIS.....	145
Figure 5.15: Explanation of the TSPlotter toolbar	146

Figure 5.16: The available water quality samples for USGS station 02091000, Nahunta Swamp near Shine, NC. For water quality data, an intermediate set is required because the list of available variables varies widely between sites.....	147
Figure 5.17: Changing the data source from a local Arc Hydro database to a web server.....	148
Figure 5.18: Cumulating streamflow to visualize the volume of water passing through the station over time.....	150
Figure 5.19: Plotting data from multiple sources, both local and remote, within the same chart space.....	151
Figure 5.20: Water balance for three subwatersheds within the Neuse River Basin calculated from precipitation and evaporation from the North American Regional Reanalysis (NARR) program and streamflow from USGS Gages.....	153
Figure 6.1: Hierarchy of hydrology software libraries or APIs (Application Programming Interface). Base level includes core data access and hydrologic processing routines. Application level incorporates base libraries and third party software libraries to customize commercial software for hydrologic analysis.	162

Chapter 1: Introduction

1.1 BACKGROUND

Current information and technology are insufficient to predict the effects of environmental change on hydrologic systems. Environmental changes are almost always multidisciplinary in nature, requiring information from many isolated databases to model the complete earth system and to adequately link causes with effects. In addition, the spatial and temporal variability of water plays a critical role in all the earth sciences and makes it difficult to adequately predict changes in the hydrologic and ecologic systems. Remediating these problems will require a coordinated effort from the hydrologic community in partnership with computer and information scientists to create the information infrastructure that will integrate hydrologic data and models for cross-scale, multidisciplinary hypothesis testing.

The level of effort and funding required to build the informatics infrastructure for hydrologic modeling at a river basin scale is beyond the scope of individual hydrologists. Therefore, the academic hydrologic community has recently united to create the Consortium of Universities for the Advancement of Hydrologic Science, Inc. (CUAHSI). CUAHSI was formed in 2001 as a non-profit organization representing nearly one-hundred universities with hydrologic interests (www.cuahsi.org). Its purpose is to address the present short-comings in hydrologic science by linking individual investigators in a coordinated effort to leverage financial support for large, interdisciplinary studies. Current plans are to

build a federated network of approximately twenty large-scale ($10^4 - 10^5$ km²) watersheds, called hydrologic observatories, located throughout the United States where retrospective data will be assembled and new data added to create “digital watersheds”. These digital watersheds will allow scientists to test hydrologic hypotheses on a scale and breadth rarely attempted (Band et al. 2003).

The creation of this network of hydrologic observatories is already underway; twenty-four prospective hydrologic observatories (Figure 1.1) were included in the first national workshop held in Logan, Utah during August 24 and 25, 2004. Of these twenty-four prospective hydrologic observatories, CUAHSI anticipates that the National Science Foundation (NSF) will begin funding two observatories with an initial budget on the order of five million dollars per year for five years (Hooper et al. 2004). Over the next twenty years, CUAHSI hopes to build the network of observatories to complement and coexist with already established watershed networks in the ecological (e.g. LTER) and agricultural (e.g. ARS) communities.

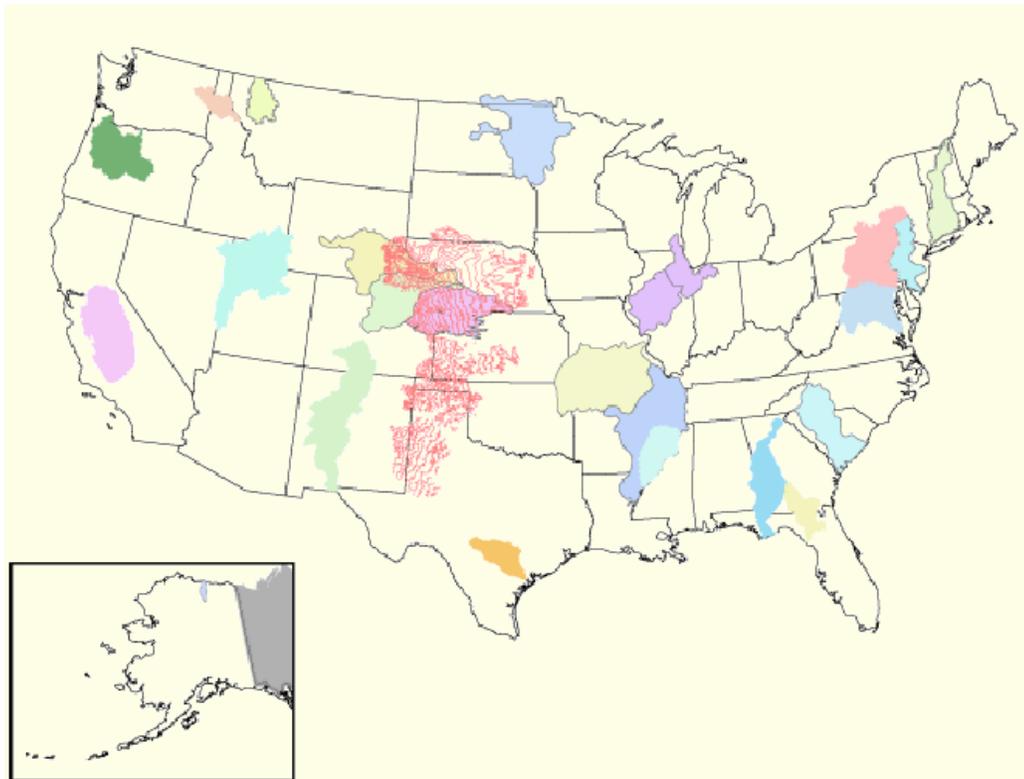


Figure 1.1: Twenty-four research groups from across the country have proposed watersheds to be funded as a CUAHSI Hydrologic Observatory (Image provided by CUAHSI).

To define the hydrologic observatory design and implementation process, CUAHSI scientists conducted a “paper prototype” study of the Neuse River Basin in North Carolina (Reckhow et al. 2004). Twelve scientists were involved in the project which resulted in an 84-page report outlining a vision for the network of hydrologic observatories. In this report, the scientists addressed such issues as the overall purpose of each observatory, what scientific questions the observatories may help to answer, what data and information will be collected by the observatories, and the role modeling will play in the hydrologic observatory.

In reflecting upon their efforts, the authors of the report saw four properties of water emerge as fundamental for understanding hydrologic processes: fluxes, flow paths, residence times, and mass balances. A flux (the amount of mass or volume that passes through an area during a period of time) describes water flow or pollutant transport rates; flow paths trace water through the atmospheric, surface, and subsurface environments to understand the fate and transport of contaminants; residence times give a measure of how long water remains in different parts of the hydrologic cycle; finally, mass balances allow hydrologists to estimate storage of mass within regions of space through time. These four properties – fluxes, flow paths, residence times, and mass balances – have emerged as key concepts for the science mission of CUAHSI.

1.2 STATEMENT OF PROBLEM

Quantifying the fluxes, flow paths, residence times, and mass balances of water in the natural environment requires the integration of vast amounts of data to drive, calibrate, and validate models at high spatial and temporal resolutions. Therefore, a key component in answering the hydrologic community's fluxes, flow paths, and residence time questions is the development of an information infrastructure for collecting, classifying, storing, and distributing both observed and modeled data. A well thought out integration of informatics and hydrology will ensure scientists are able to use the best information available in their studies, maximizing the benefit from expensive data collection and modeling efforts.

To address these issues, CUAHSI has promoted the development of Hydrologic Information Systems as one of the fundamental components of their

efforts (Hooper et al. 2004). In March of 2004, the National Science Foundation funded a two year CUAHSI Hydrologic Information System (HIS) prototyping project which is currently underway. The participants of the CUAHSI HIS project include researchers at the University of Texas at Austin, the University of Illinois at Urbana-Champaign, Drexel University, and the San Diego Supercomputing Center.

One of the first undertakings of the CUAHSI HIS was to survey hydrology researchers to better understand their need for information services (<http://www.iuhr.uiowa.edu/~cuahsi/his/>). Three themes that emerged from this survey and which provide the motivation for this research are:

1. better access to a large volume and variety of high quality hydrologic data through the internet;
2. access to visualization tools and data analysis software to inspect and assess data;
3. standardized datasets for hydrologic fluxes and state variables across the United States that can be used as benchmark datasets for both individual and community model development (Graham et al. 2002).

The unique aspect of a geoscience information system, as opposed to information systems developed for business management or library sciences, is the expressed need to organize data in both geographic space and time. Where a sample was measured and when it was collected are fundamentally important to

understanding the recorded value. Reckhow et al. (2004) saw this as a matter of great importance in the hydrologic observatory design process, claiming that integrating measurements in space and time “will provide a fundamental advance in data accessibility by data analysts, physical modelers, as well as policy and management investigators”. This statement identifies what is seemingly a very basic problem: inventorying observations across federal, state, and local agencies. However, there are a number of difficulties that must be addressed before being able to fuse hydrologic observations and modeling results into a common geospatial and temporal (or simply geotemporal) framework.

A fundamental challenge is to define the conceptual model for a geotemporal referencing system. To georeference data is commonly understood as giving a dataset “real-world” spatial coordinates. Geographic information systems (GIS) are widely used tools in hydrology and water resources engineering because of their ability to spatially integrate datasets into a common geospatial framework. Analogous to georeferencing, one can also temporally reference datasets along a timeline. The focus of this research, however, is the combination of these two previously mentioned ideas: the construction of a geotemporal framework for referencing hydrologic data to support analysis and modeling.

Although geographic and temporal information systems exist separately, there are few examples of information systems that provide an integrated space-time referencing system. Take for example the proposed Great Salt Lake Hydrologic Observatory (<http://greatsaltlake.utah.edu/>) where users are presented with two options for viewing hydrologic data: a map viewer and a time series

viewer (Figure 1.2). Both the map viewer and the time series viewer represent the state-of-the-art in terms of hydrologic data management and visualization, particularly over the web. At the same time, they also show the disconnection that currently exists between hydrologic information systems: users must choose either to view data in space or through time, but not both. However, the formulation of a geotemporal system provides a powerful tool for hydrologic and earth system science characterization. With such a system, hydrologists can isolate samples based on geotemporal queries and perform geotemporal processing to integrate, interpolate, or scale data over space and time.

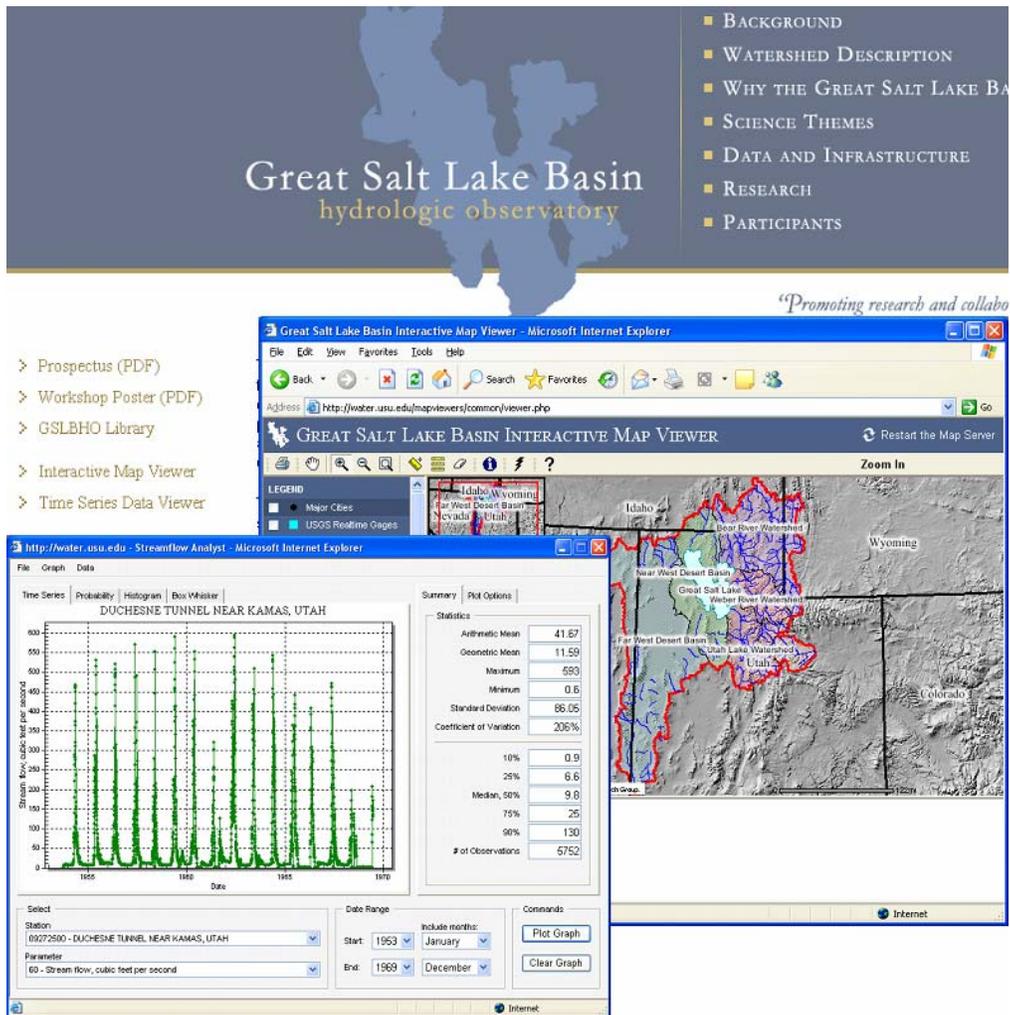


Figure 1.2: The Great Salt Lake proposed Hydrologic Observatory website presents users two views of the basin: geospatial and temporal. One purpose of this research is to integrate these two separate views into a single geotemporal view of the watershed system.

1.3 OBJECTIVE

The primary question guiding this research is *how can hydrologists integrate observed and modeled data from various sources into a single description of the environment?* Thus, the objective of the research is to develop

a conceptual design for integrating hydrologic data and information into a common geotemporal framework. This will require three tasks to be completed: (1) identify the key hydrologic concepts for building spatiotemporal objects, (2) create a geotemporal referencing system for representing the physical environment, and (3) develop software classes based on the hydrologic concepts to populate the geotemporal referencing system.

1.4 CONTRIBUTION TO SCIENCE

The most significant contribution to science of this research is the development of a geotemporal framework for integrating geoscience data. This framework is based on geographic information science concepts, but is designed specifically for hydrology data types. Conceptually, all hydrologic observations and model output values are representative of some discrete or continuous geometry in space and time (Figure 1.3). The instantaneous temperature forecasted by a model is a point in space-time, the yearly-averaged streamflow at gaging station is a line in space-time, and the maximum precipitation over a watershed for a year is an area in space-time.

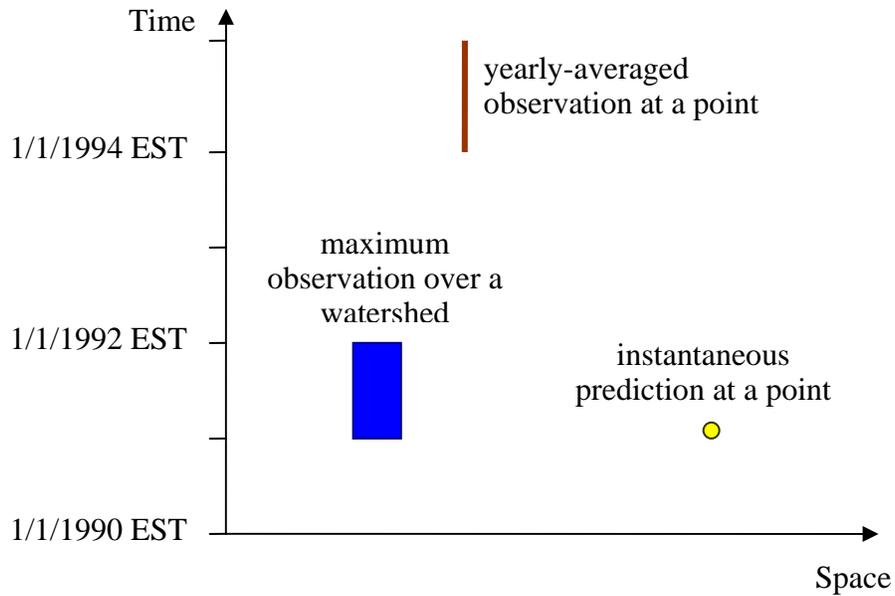


Figure 1.3: Hydrologic values represented as geometries in space-time.

By describing observed and modeled values in space-time, it is possible to integrate geoscience information into a common environment for visualization and processing. The values displayed in this geotemporal framework may be stored in a variety of formats (e.g. relational database, ASCII text file, binary flat file, etc.) and may come from both local and remote sources. These complications, however, are hidden from the hydrologist allowing him or her to concentrate on understanding the relationships between hydrologic flux and state variables in space and time.

To implement this vision requires the development of software classes to represent hydrologic concepts within the computer code. These classes, which are collectively named HydroObjects, provide the ability to place state and flux

variables within the geotemporal framework. They also allow basic hydrologic processing such as the transfer of fluxes between spatial objects through their geospatial intersection. Finally, the classes simplify the process of fusing data across formats and sources, as well as the process of transforming hydrologic variables in space and time.

ESRI ArcGIS applications are extended as an example of using the HydroObject classes to accommodate geotemporal visualization and processing of hydrologic data (Figure 1.4). The first extension, TSPlotter, adds a chart window to ArcMap for viewing time series related to geospatial features. By simply clicking on a feature in the map, a time series can be plotted within the charting window. The time series data can come from a local database formatted according to the Arc Hydro data model (Maidment 2002) or from remote data sources such as the National Water Information System (NWIS) and Ameriflux.

The second extension, Space-Time tools, is a toolbox of generic geotemporal processing tools implemented within the ESRI geoprocessing environment. With this toolbox, hydrologists can download data directly from NWIS or from the North American Regional Reanalysis (NARR) program for a particular geographic region and time interval. Hydrologists can also perform spatial interpolations and spatial averaging over time. The purpose of the toolbox is to ease the process of accessing and preprocessing basic hydrologic flux and state variables in order to parameterize hydrologic simulation models.

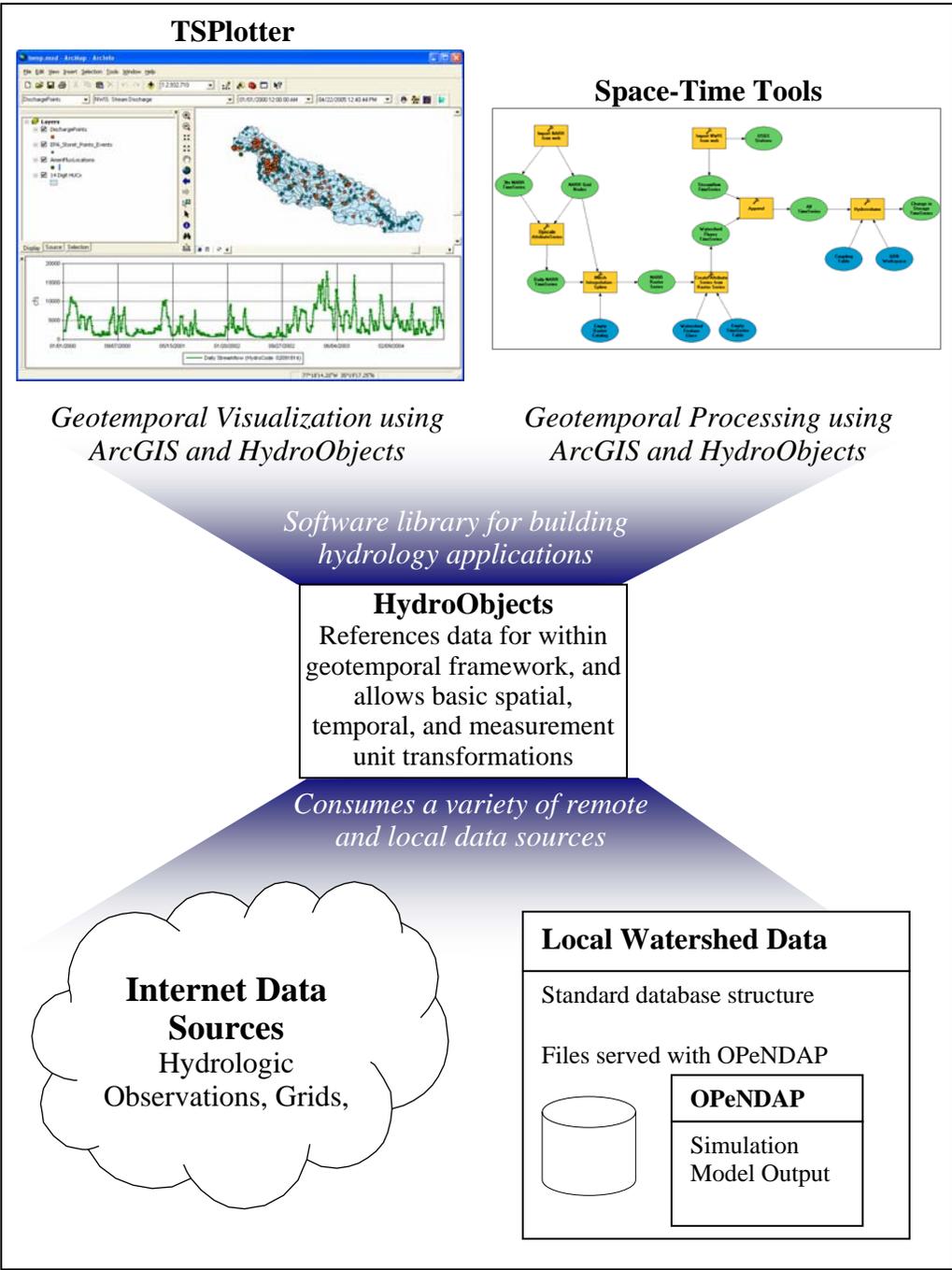


Figure 1.4: The HydroObjects library facilitates the process of building software for visualizing and performing analysis with disparate data formats and sources.

Both TSPlotter and Space-Time tools gain much of their functionality from the HydroObjects library. There are two classes within HydroObjects: geospatial time series and hydrologic flux coupler. A geospatial time series object has properties which describe the geospatial, temporal, and measurement dimensions of a flux or state variable (Figure 1.5). The object can be displayed on a chart as a time series or on a map as a feature. Hydrologists can easily convert the units and the measurement dimensions of a geospatial time series object. For example, if a geospatial time series object has $\text{kg/m}^2/\text{s}$ as its Units property and a polygon as its Shape property, then it is possible to automatically change the object's measurement units to mass flux dimensional units (e.g. kg/s) by using the polygon's area as the conversion factor. Finally, a geospatial time series object can be created using a variety of data formats and sources allowing one to visualize time series in a similar manner, despite their file storage format.

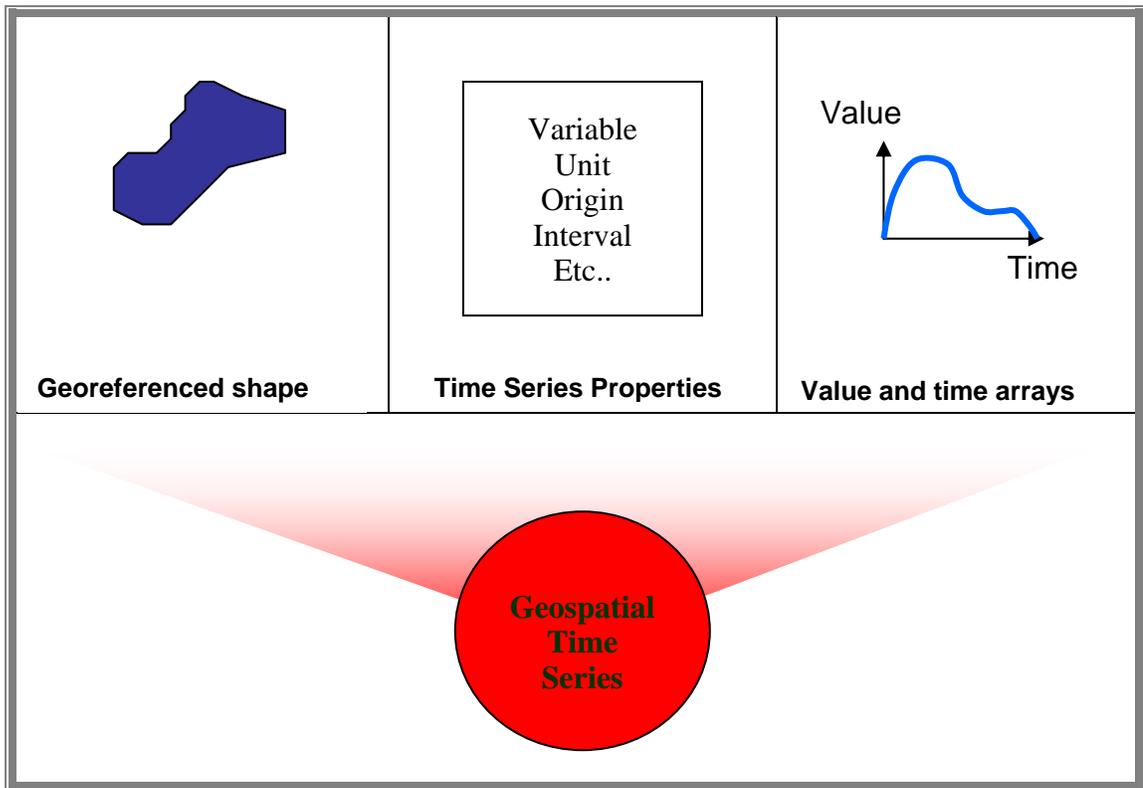


Figure 1.5: Geospatial Time Series are the combination of a georeferenced shape, an array of values and dates, and a collection of time series metadata properties.

The second class within HydroObjects, Hydrologic Flux Coupler, is a generalized structure for representing the hydrologic fluxes of water volume, mass, and energy passed between geospatially-discrete entities through time. The interaction of discrete entities will occur over the geospatial interface shared by the two geometries (Figure 1.6). For example, if a watershed is coupled to a river reach segment within its boundaries, then mass or energy is passed from the watershed to the river reach over the intersection between the two shapes. Likewise, two polygon entities, one representing a NEXRAD cell and the other a

watershed, can be coupled so that the NEXRAD cell passes a mass of water to the watershed by taking into account the intersection area between the two polygons. The flux coupler makes use of the geospatial connectivity between features for understanding the movement of mass and energy through the landscape.

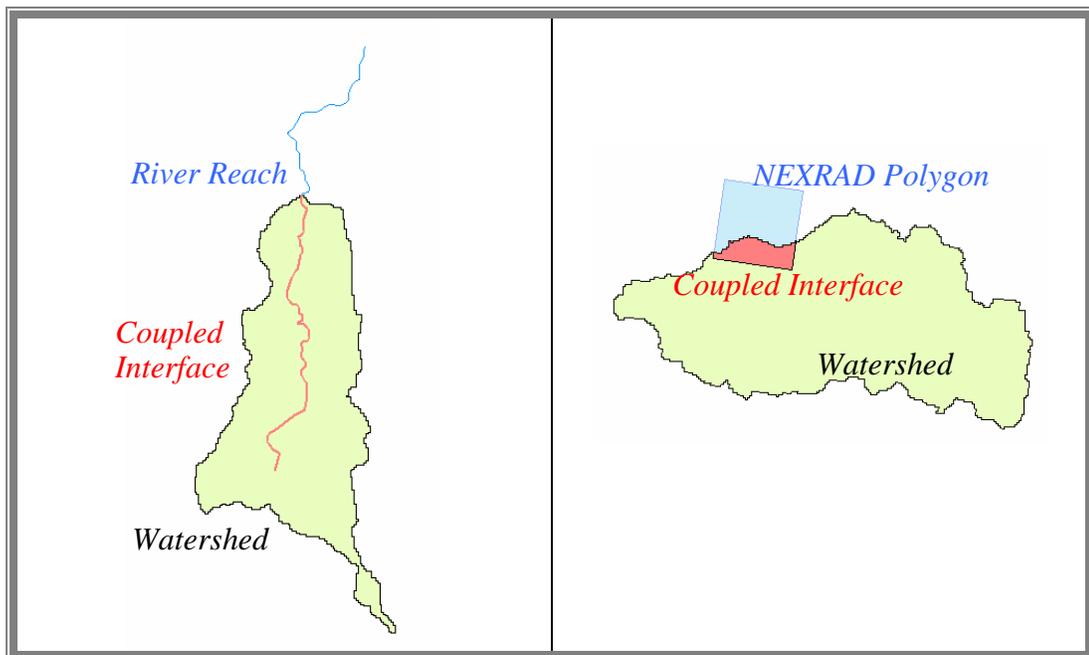


Figure 1.6: The Flux Coupler provides the ability to represent the exchange between discrete entities in space through their common interface.

1.5 DISSERTATION OUTLINE

The concepts introduced in this first chapter are further developed in the next five chapters of this dissertation. The second chapter reviews other relevant geoinformatics efforts that set the stage for creating a hydrologic information system. The third chapter provides the methodology for creating a geotemporal framework for referencing hydrologic data and describes the spatiotemporal,

hydrologic classes developed to populate this digital environment. The fourth chapter presents the procedure for application, the actual design of the HydroObjects in object-oriented programming terms. The fifth chapter is an application of the HydroObjects to the Neuse River Basin in North Carolina. The Space-Time Toolbox, which was created using the HydroObjects library to extend the ArcGIS geoprocessing environment, is used to (1) harvest data from the National Water Information System and the North American Regional Reanalysis program dynamically over the web, (2) transform these datasets to common geospatial and temporal dimensions, units, and scales, (3), and finally perform a water balance for a selected set of watersheds within the Neuse River Basin. Then TSPlotter, another custom software application built within ArcGIS using HydroObjects, will be used to visualize the time series data within a mapping environment. Finally, conclusions and recommendations are provided in the sixth chapter.

Chapter 2: Literature Review

This chapter reviews previous events and research in the fields of hydrology and geoinformatics beginning with a brief history of CUAHSI and its hydrologic information system component. Next, other earth science disciplines with current information technology oriented projects underway are reviewed to provide insight into the CUAHSI Hydrologic Information System (HIS) effort. Lastly, there is a brief review of the current trends in geospatial and temporal information technology within hydrology and water resources engineering.

2.1 INTRODUCTION TO CUAHSI

The Consortium of Universities for the Advancement of Hydrologic Sciences, Inc. was established in 2001 to provide support to hydrologic science research and education and now has a membership of nearly one hundred universities. Many hydrologists have concluded that the advancement of hydrologic sciences will require the community to adopt a more holistic view of the water environment (Council 2001, Hornberger et al. 2001, Reckhow et al. 2004). This holistic view must include interactions between the water cycle, the biogeochemical cycles, and social sciences. Without such a comprehensive view, it is impossible to adequately understand the relationship between hydrology and the environment, and, therefore, impossible to provide answers to the future of water resources.

Given this motivation, CUAHSI was incorporated in the District of Columbia on June 25, 2001 (<http://www.cuahsi.org/about/history.html>) with the expressed mission to:

foster advancements in the hydrologic sciences, in the broadest sense of that term, by: developing, prioritizing and disseminating a broad-based research and education agenda for the hydrologic sciences derived from a continuous process that engages both research and applications professionals; identifying the resources needed to advance this agenda and facilitating the acquisition of these resources for use by the hydrologic sciences community; and enhancing the visibility, appreciation, understanding, and utility of hydrologic science through programs of education, outreach, and technology transfer (<http://www.cuahsi.org/about/mission.html>).

2.1.1 HydroView: The CUAHSI Organization Structure

CUAHSI is implementing this mission by building a collection of elements referred to as HydroView (Figure 2.1) (Hooper et al. 2004). The central element of CUAHSI, as depicted by the HydroView, is the Hydrologic Observatory. A Hydrologic Observatory is an experimental watershed approximately 10,000 km² in size, or about the size of Maryland. This size of experimental watersheds was chosen to make use of satellite-based remote sensing data sources as well as regional and global climate models and is a

distinguishing feature between CUAHSI and other watershed-oriented research programs in the ecological and agricultural sciences.

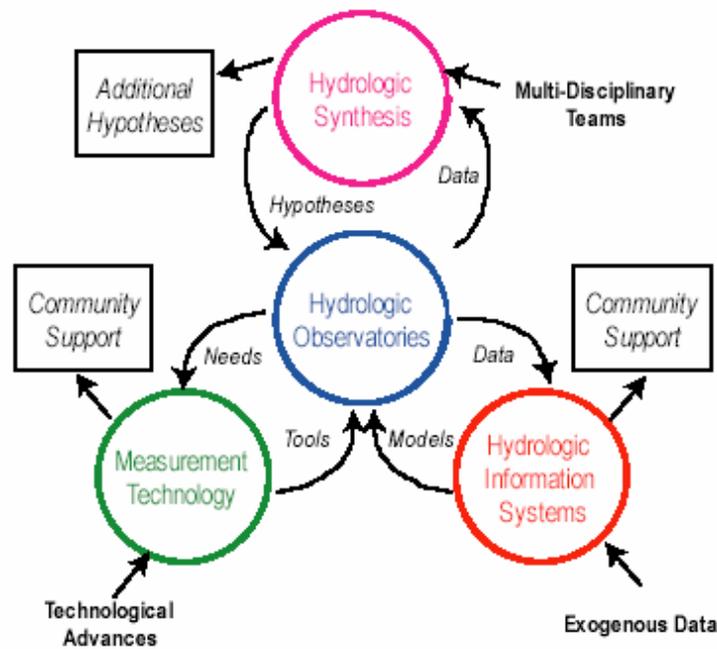


Figure 2.1: CUAHSI HydroView (Hooper et al. 2004)

The three other elements of HydroView are meant to support scientific studies within the Hydrologic Observatory. The Measurement Technology component is charged with collecting data through current and innovative technologies. The Hydrologic Information component is necessary to manage both data and models needed to adequately assess hydrologic processes over such a large spatial region. Finally, the Hydrologic Synthesis component is concerned with hypothesis testing and scientific analysis in order to better understand the

hydrologic systems. Together, these four elements comprise an unprecedented structure for hydrology research and education within the United States.

CUAHSI envisions a national network of these hydrologic observatories with each observatory in a unique hydrologic and ecological environment. The goal is to have a common infrastructure for data management and transmission, which will allow cross-Hydrologic Observatory studies. Providing interoperability between hydrologic observatories will be a challenge because it requires a balance between local and federal data management protocols (Graham et al. 2002). Local hydrologic observatory teams require the freedom to collect and store data as they see fit, however there must be an infrastructure in place to incorporate local data into a federated data system.

2.1.2 Hydrologic Observatory Prototype for the Neuse Watershed

To begin the formalization of the hydrologic observatory design, CUAHSI commissioned a paper prototype study of a hydrologic observatory for the Neuse River Basin in North Carolina which was completed in December of 2004 (Reckhow et al. 2004). This prototype hydrologic observatory is the template for designing a digital watershed capable of depicting a holistic view (atmosphere, soil, surface water, and groundwater phases) of the hydrologic cycle across traditional disciplinary boundaries and inclusive of multiple scales. The primary goal of the Neuse paper prototype was to answer the question: “for a watershed of this scale (10,000 km²), is it possible to carry out an observing strategy that would provide the basis for addressing critical sciences questions identified by the hydrologic science community, at a reasonable cost” (Reckhow et al. 2004).

The authors attempted to answer this question by first identifying the five themes of hydrologic science and then, from these themes, developing a vision for implementing a network of hydrologic observatories for the advancement of hydrologic science research and education. The five research themes identified by Reckhow et al. (2004) are:

1. linking hydrologic and biogeochemical cycles,
2. sustainability of water resources,
3. hydrologic and ecosystem resources,
4. hydrologic extremes, and
5. fate and transport of chemical and biological contaminants.

The hydrologic observatory is designed to be a natural laboratory where scientists can formulate and test hypotheses regarding these five science questions.

Taking a more fundamental view of hydrology, Reckhow et al. (2004) abstracted these five scientific questions into a vision for implementing a hydrologic observatory to answer how water and its constituents move between volume units in the surface, subsurface, and atmospheric environments. The authors referred to these units as stores. If one considers the environment as a series of these stores, then the goal of hydrology is to quantify the

1. mass in each store,
2. residence time within stores,
3. fluxes between stores, and
4. flow paths among stores (Reckhow et al. 2004).

This vision of the hydrology environment as a set of stores is appealing because of its simplicity; it ignores distinctions of surface, subsurface, and atmospheric environments (traditional sub boundaries within hydrology) and presents a holistic view of the hydrologic cycle. It is also scale independent because stores can be defined at various scales and each store can be divided into sub-stores.

Hydrologic observations are a strong component of the CUAHSI scientific vision and are critical to developing and testing hypotheses regarding the five scientific questions posed by the Neuse Report. Hooper et al. (2004) stated that observational emphasis of CUAHSI arose because there are many untested conceptual hydrologic models. Focusing on creating an integrated, multi-scale representation of hydrologic observations will provide the most powerful means for testing these conceptual hydrologic models and advancing the science (Hooper et al. 2004). Models, of course, also play a critical role in the hydrologic observatory, in particular because it is often impossible or impractical to observe certain hydrologic variables. In such cases, hydrologists must turn to models to estimate a variable in space and time.

Reckhow et al. (2004) describe three requirements of hydrologic measurement approaches:

1. quantitative assessment of the fluxes and stores of water, sediment, and nutrients,
2. temporally and spatially integrated measurements of these fluxes and stores, and

3. acquisition of measurements in spatially stratified manner that allows for predictive understanding at the river-basin scale.

These three measurement approaches are a major theme of the Neuse Report and the authors spend a substantial portion of the report describing each approach.

(Reckhow et al. 2004) acknowledged a “lack of spatially and temporally integrated, comprehensive hydrologic observations” as a major impediment to the hydrologic observatory effort. It is not that current information technology limits the ability to integrated data into a common geospatial and temporal framework; it is that this ability currently does not exist within the hydrologic sciences. The main reason for this inability is the lack of a standard format for storing hydrologic observational data. Maidment (2005) provides a review of data formats between the U.S. Geological Survey (USGS), the U.S. Environmental Protection Agency (EPA), and the National Climate Data Center (NCDC) and presents a strong case for the need for a standard hydrologic time series data model to facilitate integrating these federal data sources into a complete picture of the watershed environment.

Without recent advances in information technology, the scale of the hydrologic observatory effort might not be possible (Reckhow et al. 2004). The hydrologic sciences are not alone in their increased dependence on information technology to advance scientific understanding. This is evident by the numerous informatics projects currently being funded by the National Science Foundation (GEON, CLEANER, NEON, etc.). The next section takes four of the earth sciences and explores their use of informatics.

2.2 INFORMATICS IN THE EARTH SCIENCES

One of the more recent endeavors by the National Science Foundation (NSF) to foster integration between informatics and earth sciences has been the Cyberinfrastructure program. Cyberinfrastructure refers to the technologic infrastructure required to

1. manage, preserve, and efficiently access data,
2. foster integrated, multidisciplinary scientific studies,
3. accelerate the pace of scientific discovery,
4. enable custom software to be easily written, shared, and preserved, and
5. provide access to high-end computational power and visualization tools to individual researchers (Keller 2003).

NSF is encouraging the development of cyberinfrastructure for the sciences as a means to link people, data, and tools across disciplinary and institutional barriers (NSF Advisory Panel on Cyberinfrastructure 2003).

The interest in building Cyberinfrastructure among the earth sciences will undoubtedly show great disparity in how each discipline uses information technology. There are many reasons for these differences. First, atmospheric and ocean sciences have community models that apply numerical modeling techniques at a global scale. Such models, because of their size, must be highly optimized in terms of data storage and input/output. Thus, these communities commonly sacrifice ease of use for computational speed. Hydrologists, ecologists, and geologists, for the most part, have not taken the approach of building community models for large-scale prediction through numerical schemes.

Instead, most work is observation driven and utilizes custom modeling and analysis techniques. Within these communities, where most time is spent on collecting and inventorying observation datasets, it has proven worthwhile to sacrifice computational efficiency for ease-of-use.

These differences suggest that the task of building a Cyberinfrastructure between the earth sciences will face many barriers when trying to build links between scientific communities. These barriers run deeper than one might anticipate and have permeated through the tools, computer languages, and even computer operating systems most common in different disciplines. Because these differences in informatics between earth science communities have evolved for practical purposes, the solution cannot be to force all hydrologists and ecologists into the atmospheric and ocean science informatics paradigm simply because these disciplines are more technologically advanced. The solution must instead merge the informatics employed by all earth sciences into an interoperable system allowing both ease of use for field scientists and computational efficiency for computer-oriented scientists.

In this section of the literature review, the focus will be on how informatics is being used within various earth science disciplines. Then in the following section, the focus will turn to hydrology and how a hydrologic information system fits with the atmospheric information infrastructure from a geospatial and temporal data management perspective.

2.2.1 Informatics in the Atmospheric Sciences

The University Corporation for Atmospheric Research (UCAR) was formed in 1960 to enhance computing and observational capabilities at U.S. Universities with doctoral programs in the atmospheric sciences (<http://www.ucar.edu>). There are three components to the UCAR overall mission: to support the university community, to improve the understanding of the atmospheric system and to foster knowledge and technology transfer. These three components are implemented through two organizations. The National Center for Atmospheric Research (NCAR) is charged with developing and maintaining state-of-the-art computer models, and the UCAR Office of Programs (UOP) is responsible for supplying real-time weather data to universities through Unidata, among other services.

The Unidata community includes over 150 Universities interconnected by an information infrastructure called the Internet Data Distribution (IDD) (<http://my.unidata.ucar.edu/content/software/idd>). The Unidata IDD delivers requested datasets to universities as they become available in near real-time (Figure 2.2). This is a unique configuration because typical data distribution systems require users to visit a centralized information system to obtain data through queries. The Unidata approach is referred to as a “push” technology, whereas access through an ftp site is a “pull” technology.

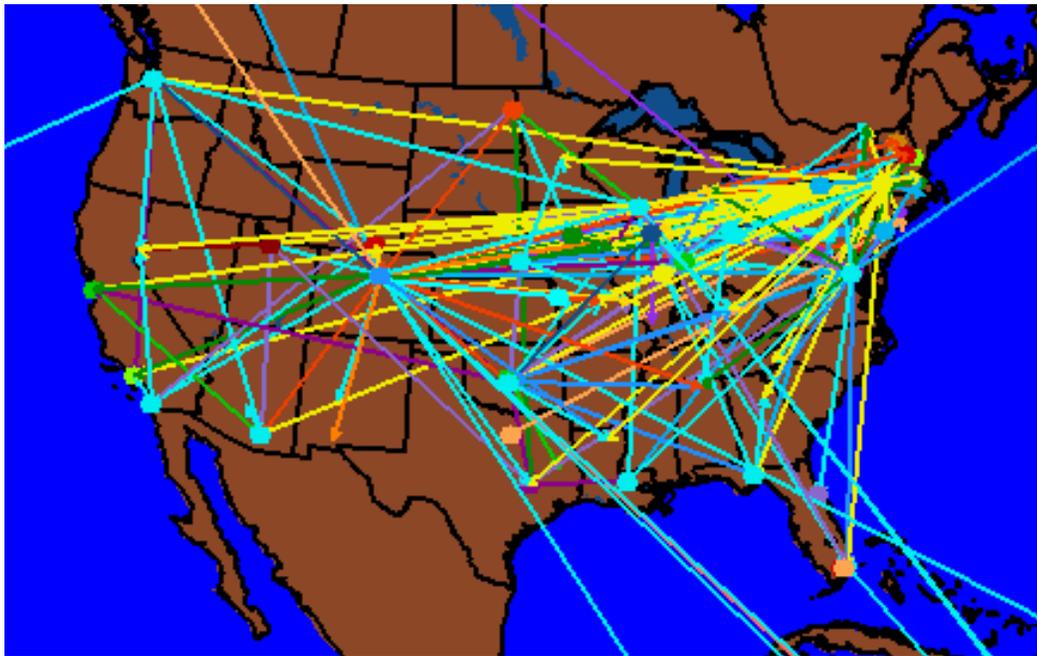


Figure 2.2: Unidata's Internet Data Distribution (IDD) network of servers
(source: <http://my.unidata.ucar.edu/content/software/idd/rtstats/index.html>)

One disadvantage of this system is that if a university wishes to capture Unidata information, they must first install Unidata Local Data Manager (LDM) software on a UNIX machine. Then, based on specifications established in the LDM software, near real-time information is sent to the university via the Internet Data Distribution (IDD). Thus, there is an overhead associated with obtaining atmospheric data through Unidata because of their chosen data distribution system as opposed the alternative “pull” technology where users query a database for particular information of interest. This overhead likely keeps potential data users from gaining access to real-time weather data for classroom and research purposes. At the same time, the volume of data generated through atmospheric modeling and observation campaigns requires such a data distribution system.

The data distributed by Unidata is stored in various formats, but one of the more popular is NetCDF (Network Common Data Form) (www.unidata.ucar.edu/packages/netcdf/). NetCDF is based on an early file format designed by NASA in 1985 named Common Data Format (CDF). It improved on the CDF format by providing a machine-independent structure that allows one to transport data across a number of computer platforms (<http://nssdc.gsfc.nasa.gov/cdf>). Its primary advantage as a file format for scientific data is that it allows direct access to large datasets stored as multi-dimensional arrays. Each array is described by a set of attributes. Typically, attributes include information such as the variable name and variable units. Each array also has one or more dimensions. For example, a variable within a netCDF could have the dimensions latitude, longitude, and time. In geographic information science terms, this would be representative of a 2D-space-time continuous field.

2.2.2 Informatics in the Ecological Sciences

One of the largest undertakings in the Ecological Sciences has been the United States Long Term Ecological Research (LTER) program. The LTER program began in the late 1970s to study long-term ecosystem dynamics through the establishment of local experimental sites throughout the United States. The number of research sites has grown from six in 1980 to 24 (Figure 2.3). Over 1,100 scientists and 700 students from 140 institutions contribute to the LTER program (Baker et al. 2000). The similarity between the LTER and CUAHSI suggest that understanding the strengths and weaknesses of the LTER program is important for the formulation and design of the hydrologic information system.

Despite beginning before the recent interest in data management for earth sciences, data management was a central issue from the beginning of the LTER project; the first workshop was held in 1981 (<http://intranet.lternet.edu/archives/documents/foundations/history.html>). Early data management issues, however, were concerned with individual sites and not communication between sites.

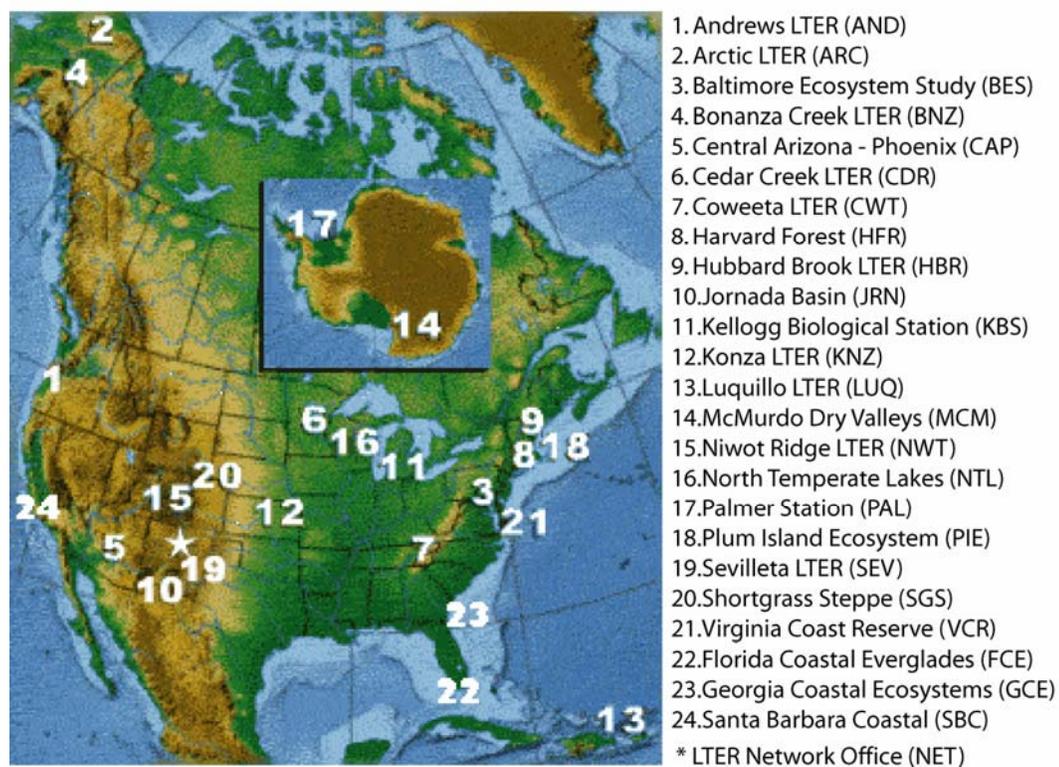


Figure 2.3: Map of the twenty-four LTER sites (source: <http://lternet.edu/sites>)

In 1996, the LTER program began to consider data exchange and synthesis between individual sites by establishing the LTER Network Information System (NIS). The LTER NIS was created as a “cooperative, federated database

system supporting multi-investigator, multi-site ecosystem studies created in collaboration with location information management personnel” (Baker et al. 2000). It was also designed to permit local site independence by providing a modular, flexible design that does not force sites to adopt a rigid information infrastructure. This was seen as a critical component because different sites have different objectives and, therefore different data requirements. The disadvantage of this plan was, of course, increasing the difficulty associated with sharing information between sites because of no universally accepted data model.

The conceptual model of the LTER Network Information System includes four components: individual sites, central sites, scientific communities, and the broader community. Raw data is collected by each site and each site has a data manager responsible for the facilitating, translating, and converting the incoming raw data to standards established by the community (Figure 2.4). The site data are then collected into a central, relational database. From this central database, the scientific community can draw from the raw data to produce publications, reports, or other required information. The website for the central database is <http://www.lternet.edu/DTOC/>. Meteorological data is stored in separated database and server through a separated website: <http://www.fsl.orst.edu/climdb/index.htm>. Both websites allow one to query for data by site name, by dataset name, and by time range.

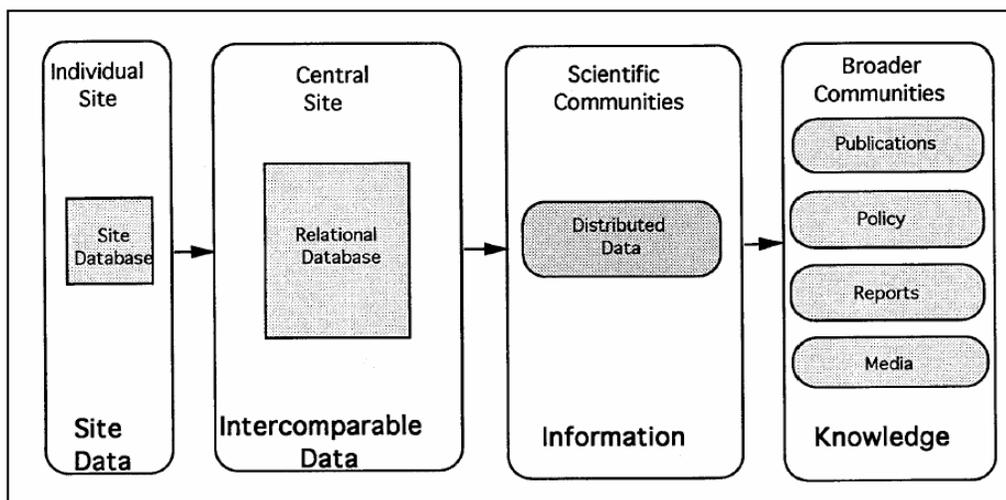


Figure 2.4: The Long Term Ecological Research Network Information System (Baker et al. 2000)

2.2.3 Informatics for the Geological Sciences

Broader in scope than CUASHI and more recent than both the UCAR and LTER programs, GEON (GEOscience Network) is concerned with developing the cyberinfrastructure for all of the geosciences. This is a new initiative and thus only the theoretical outline for the venture has been formulated. Its ultimate goal is to create a culture where data are shared, archived, and rapidly disseminated across disciplines to enable both large, multiscale research projects and smaller-scale research by individuals or small groups (Keller 2003).

GEON is a collaborative effort between earth scientists across the country (Figure 2.5) and computer scientists at the San Diego Supercomputer Center, also the technical partner for CUAHSI Hydrologic Information System, with the aim of better utilizing advances in information technology to serve the earth science communities. Such advances include the improved ability to manage and share

retrospective data and information collected by individual investigators and large government data collection ventures. Geoscientists, like hydrologists, have realized that current systems for data management and sharing are inadequate in efforts to advance scientific discovery. Traditional barriers to data sharing, such as format differences, lack of uniformity in terminology conventions, and the inability of individual researchers to store or share large data sets will be addressed through the GEON project (Tooby 2003).

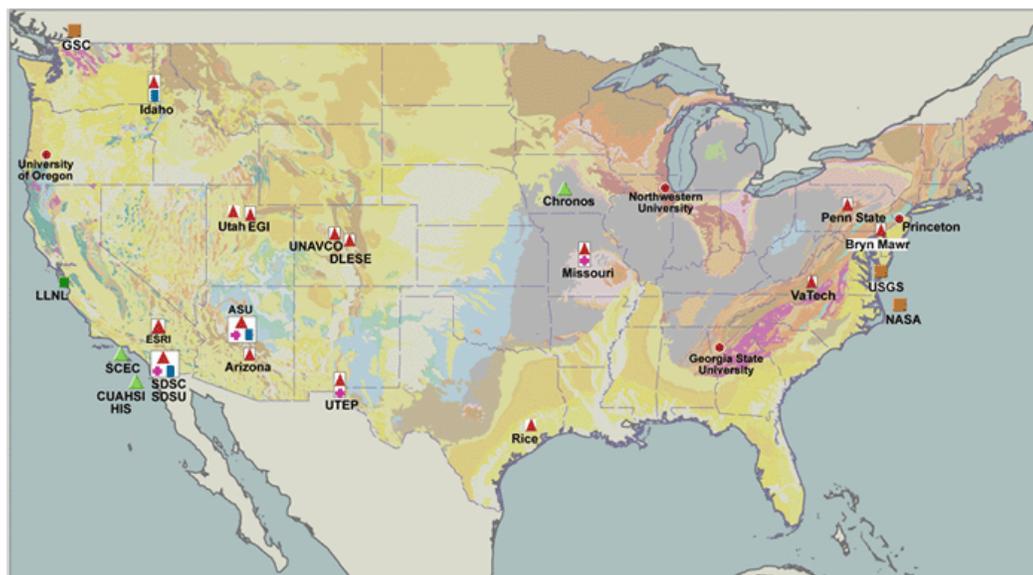


Figure 2.5: The GEON collaborators (Source: <http://www.geongrid.org/>)

The GEON project also stresses the need to explicitly store when and where data are collected (Tooby 2003). Often, data are not geospatially and temporally referenced to situate each measurement in 4-D space. This goes beyond simply including x,y,z and t columns in the data base for each record. In some cases, it is necessary to describe the temporal variation of discrete entities

with irregularly shaped geometries. Not only could the non-spatial properties of this discrete entity vary through time, but its location and shape could also be time-dependent. Moreover, the basic framework for describing geologic time can be very different that required to describe hydrologic time, because geologic time scales of most interest can be orders of magnitude larger than typical hydrologic time scales. Therefore, the issue of providing a spatial and temporal representation of geoscience data is nontrivial (Langran 1992, Peuquet 2001, Peuquet 2002) and is a major theme of this research.

Metadata also plays a central role in GEON to clearly and completely describe what information is contained in a dataset, how it was obtained, and the uncertainty associated with the information (Keller 2003). Processed information, such as the results from a model simulation, has uncertainties that not always clearly stated. Furthermore, two research groups may use different models to predict the same property. If a third research group requires the information produced by these two researchers, they must decide which dataset is best for their purposes. Such a task is impossible without adequate metadata.

Creating a culture where data are shared, archived, and rapidly disseminated across disciplines requires a framework for integrating the multitude of existing data sources including traditional databases, web information systems, digital libraries, or domain applications working on flat-files (Tooby 2003). Through the integration, one can query multiple data sources from a single interface, saving valuable time and effort. The integration of the data sets also

points to redundancy and gaps in data sources, providing guidance for future efforts to improve data collection efficiency.

2.3 INFORMATICS IN THE HYDROLOGIC SCIENCES

There are a number of proprietary and nonproprietary simulation models and data storage systems commonly used in hydrology. In general, these models and information systems have developed independently, resulting in little interoperability between them. Thus, the task of parameterizing a hydrologic or hydraulic simulation model with data from various sources can be a time consuming process. A challenge facing the CUAHSI Hydrologic Information System prototyping team is to, among other things, ease the process of parameterizing a hydrologic simulation model using data from national and local information systems. To do this, the prototype team must develop a generic hydrologic information system as a means for integrating data across the heterogeneous sources and formats common to hydrology.

An early step in building an information system is designing a data model (Hoffer and McFadden 2002). A data model is a template for storing data and should not be confused with a simulation model. Data models are not mathematical, but instead a framework for describing a subject and storing data about it (Maidment 2002). Organizations spend a significant amount of effort on modeling data because it insures logical data access for ingestion into analysis or mathematical modeling programs and, ultimately, more informed decision making. As was shown in the preceding sections, earth science sub-disciplines make use of different data models, and this leads to a lack of interoperability

between the earth sciences. To highlight these differences, the data models most common to the atmospheric and hydrologic sciences are emphasized here.

In the hydrologic sciences, geographic information systems are a critical tool for data management, parameter estimation, and visualization (Singh and Woolhiser 2002) . Geographic information systems, however, have been less widely used in atmospheric science (although there has been a substantial recent effort to change this (Betancourt and Wilhelmi 2003, Wilhelmi and Brunskill 2003, Habermann et al. 2004, Murray et al. 2004)). Atmospheric sciences commonly employ array-based data models more suitable for numerical modeling (Nativi et al. 2004, Ho et al. 2005). Thus, comparing the hydrologic and atmospheric science data models is, in many ways, a comparison of data models common to geographic information systems and data models common to numerical modeling.

2.3.1 The Arc Hydro Data Model

Perhaps the best example of a geographic data model designed specifically for the water community is the Arc Hydro Data Model developed by the GIS in Water Resources Consortium (Maidment 2002). Arc Hydro is a template for constructing a digital watershed; it is an object-oriented database design with over twenty spatial and non-spatial object classes ranging from watersheds to monitoring points to time series records (Figures 2.6 and 2.7). The design integrates the spatial and temporal properties of hydrographic and hydrologic data by georeferencing points, lines, and polygons and allowing time series to be related to monitoring stations or other hydrologic features.

The feature classes in the Arc Hydro framework schema are MonitoringPoint, Waterbody, Watershed, HydroEdge, and HydroJunction (Figure 2.6). MonitoringPoint features represent streamflow or water quality gages, Waterbody features represent large lakes or reservoirs, and Watershed features represent the drainage areas. The river network is described by the HydroEdge features, which represent river reach center lines, and HydroJunction features, which mark significant locations along the river network: for example a reservoir outlet, a diversion point, or the location of a streamflow gage.

TimeSeries information is stored in two tables: the TimeSeries table contains the actual values and the TSType table provides the metadata for each type of time series stored in the database. Because the time series typing of Arc Hydro is such an important component of building interoperability between hydrologic observations collected by different agencies, the details of the Arc Hydro time series component are provided in the methodology section of this paper.

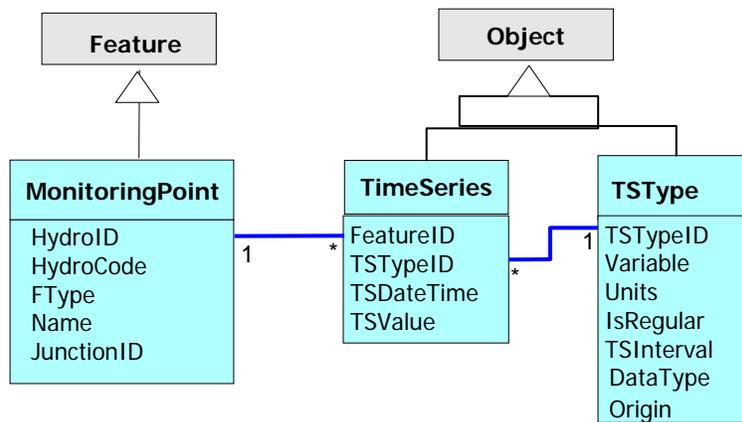


Figure 2.7: Arc Hydro Time Series component (Maidment 2002)

The advantage of Arc Hydro is that it provides the foundation for representing and connecting geospatial and temporal hydrologic information. Of course, different hydrologic simulation models need different descriptions of reality, but at the same time there will be a significant amount of repetition between these models. When there is repetition, there is an opportunity to reduce data storage by sharing a common data model. Furthermore, having a single data model for the representation of the hydrologic environment facilitates hydrologic simulation model integration through interchange points, where output from one

simulation model serves as input into a second simulation model (Figure 2.8) (Whiteaker et al. 2005). Arc Hydro seeks to insure compatibility between hydrologic simulation models by providing the standard for hydrologic information used in a variety of sources.

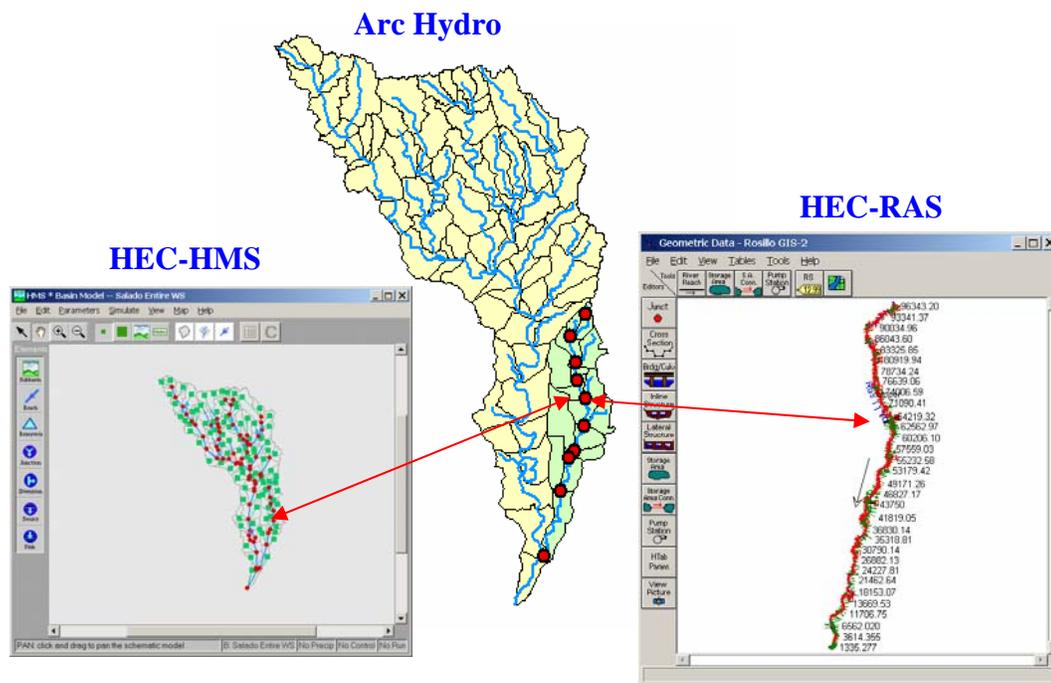


Figure 2.8: Arc Hydro facilitates integration of models through a common representation of the hydrologic system (Maidment 2004).

A data model for the hydrologic sciences may be different from a data model for water resources engineering. Scientists and engineers ask different questions and, therefore, require different information systems to provide answers to their questions. Engineers are driven by solving current problems facing society (water supply, water pollution, flooding), while scientists want to understand why these events happen. But at a fundamental level, both scientists

and engineers are attempting to digitally represent the same hydrologic system, so the differences in the questions the two groups ask may be less significant than the character of the hydrologic system they both work with.

2.3.2 The NetCDF Data Model

As mentioned in the review of atmospheric science informatics, netCDF is a format widely used in the atmospheric sciences. The data model behind a netCDF file consists of three components: variables, dimensions, and attributes (Figure 2.9) (Rew et al. 1997). Variables are the main component of the netCDF file because they contain the multidimensional array. Variables have a name, a shape, a type, and values.

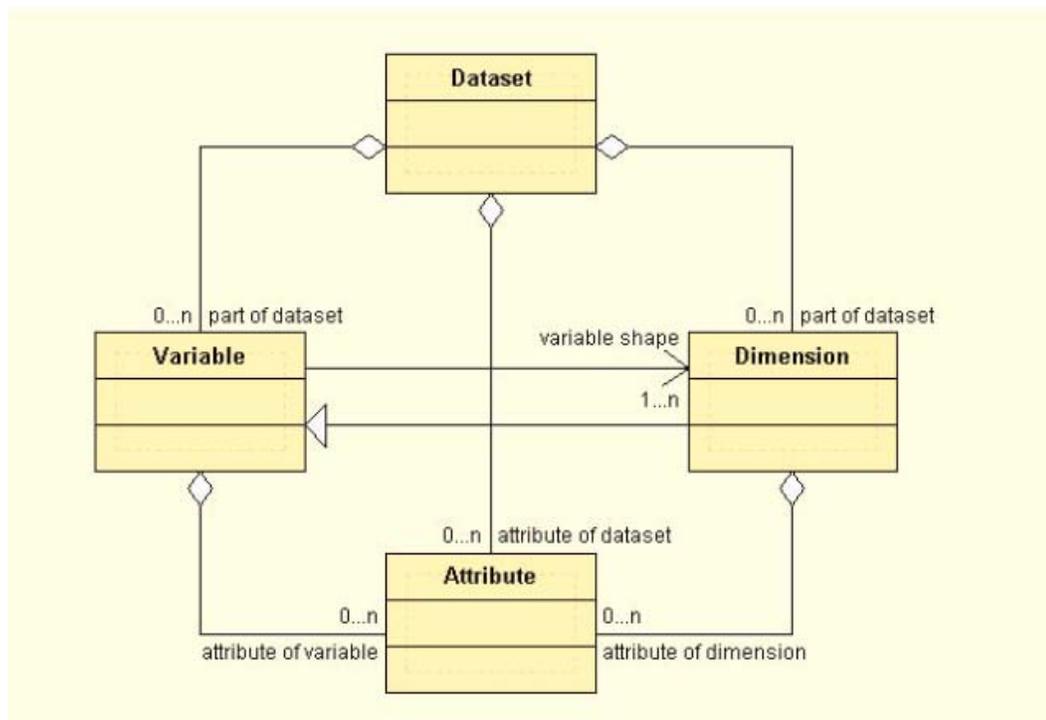


Figure 2.9: UML representation of a netCDF file (Nativi et al. 2004)

Each variable can have one or more dimensions. A dimension can be physically-based, such as latitude, longitude, height, or time, or it can represent an index, for example to a model-run. The total number of entries for a variable is equal to the product of its dimensions. For example, if a variable has two dimensions, latitude and longitude, and latitude equals 4 and longitude equals 5, then it represents a 2D matrix with 20 values. The actual latitude and longitude values can be stored as variables themselves, or, in some cases, the geospatial coordinates are stored as three variables within the netCDF file: (1) a start value, (2) a step size, and (3) a count. This second option is popular because it requires less memory for storing the geotemporal coordinate domain. Figure 2.10 shows an example of a variable stored within a netCDF file – three hour accumulated surface evaporation according to the North American Regional Reanalysis program.

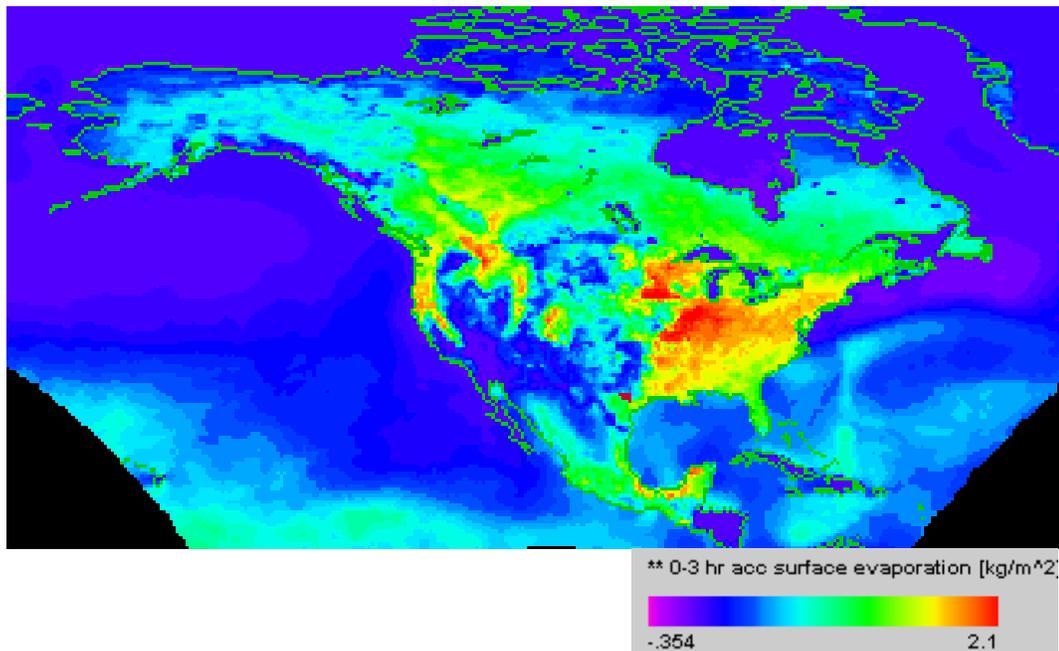


Figure 2.10: NetCDF files store multi-dimension variable fields such as surface evaporation (Source: North American Regional Reanalysis of climate visualized with Unidata’s Integrated Data Viewer).

The attributes component contains the metadata for the variable. Each variable has a unique set of attributes. There are no required standards for what metadata must be included with a certain variable type, but there is a list of standard conventions that are strongly recommended (<http://my.unidata.ucar.edu/content/software/netcdf/conventions.html>). The attributes for a variable type can be extended at a later point in time, so there is no need to fully anticipate all required metadata when variables are first created.

2.3.3 Differences between Arc Hydro and NetCDF Data Models

One significant difference between the Arc Hydro data model and the netCDF data model is the precise geo-referencing of the former and the emphasis

on multidimensionality of the latter (Nativi et al. 2004). This distinction has evolved within each field for practical reasons. Space in atmospheric science models is kept relatively simple so as to not overcomplicate the numerical model. On the other hand, geographic information systems are often applied in hydrology at a spatial scale which requires explicitly defined georeferencing coordinate systems, datums, and tools for reprojecting datasets to different geographic coordinate systems. Geographic information systems, however, contain no native means for handling temporal data

A deeper look at the data models common to atmospheric and hydrologic science shows that they have completely different concepts for organizing data. Atmospheric science data models, like netCDF, are array-based data structures whereas geographic information system data models, like the geodatabase, are table-oriented data structures. Array-based files are designed to provide optimal input/output of arrays of information or single values of an array at specified index (Rew et al. 1997). Table-oriented data structures, when part of a relational database management system (RDMS), are optimized to provide quick reorganization of information and to provide subsets of information given a list of qualifications (Hoffer and McFadden 2002).

Thus, when developing a digital watershed, the question becomes how to provide integration between the data models common to the atmospheric and hydrologic communities. Some researchers have suggested means for providing this interoperability by means of the raster data model common to geographic information systems (Nativi et al. 2004, Ho et al. 2005). The raster data model,

however, is limited to representing two spatial dimensions and has no intrinsic means for representing time or the z-dimension. Therefore, while conversion between netCDF and rasters may offer a quick means of interoperability, it is not a long-term solution. Instead, what is ultimately needed is an extension of the data models used in the atmospheric and hydrologic sciences to accommodate shortcomings in their representations of discrete and continuous in 3-D space-time. That is what is attempted in this research

2.4 REPRESENTING TIME IN GIS

Because this research has a hydrologic science emphasis, the goal is to extend geographic information systems to include temporal data. Arc Hydro provides a means for relating space and time through the relationship between a MonitoringPoint feature and its associated time series records. It does not, however, address dynamic attributes of river reaches or watersheds, nor mention features with dynamic shapes or time-indexed rasters.

Recent work at the Center for Research in Water Resources (CRWR) at the University of Texas at Austin has extended the integration between geospatial and temporal data by presenting three basic structures for geotemporal data within GIS: Attribute Series, Feature Series, and Raster Series (Figure 2.11) (Goodall et al. 2004). These three basic geotemporal structures, along with a generic time series structure, provide a data model for storing dynamic data within a GIS.

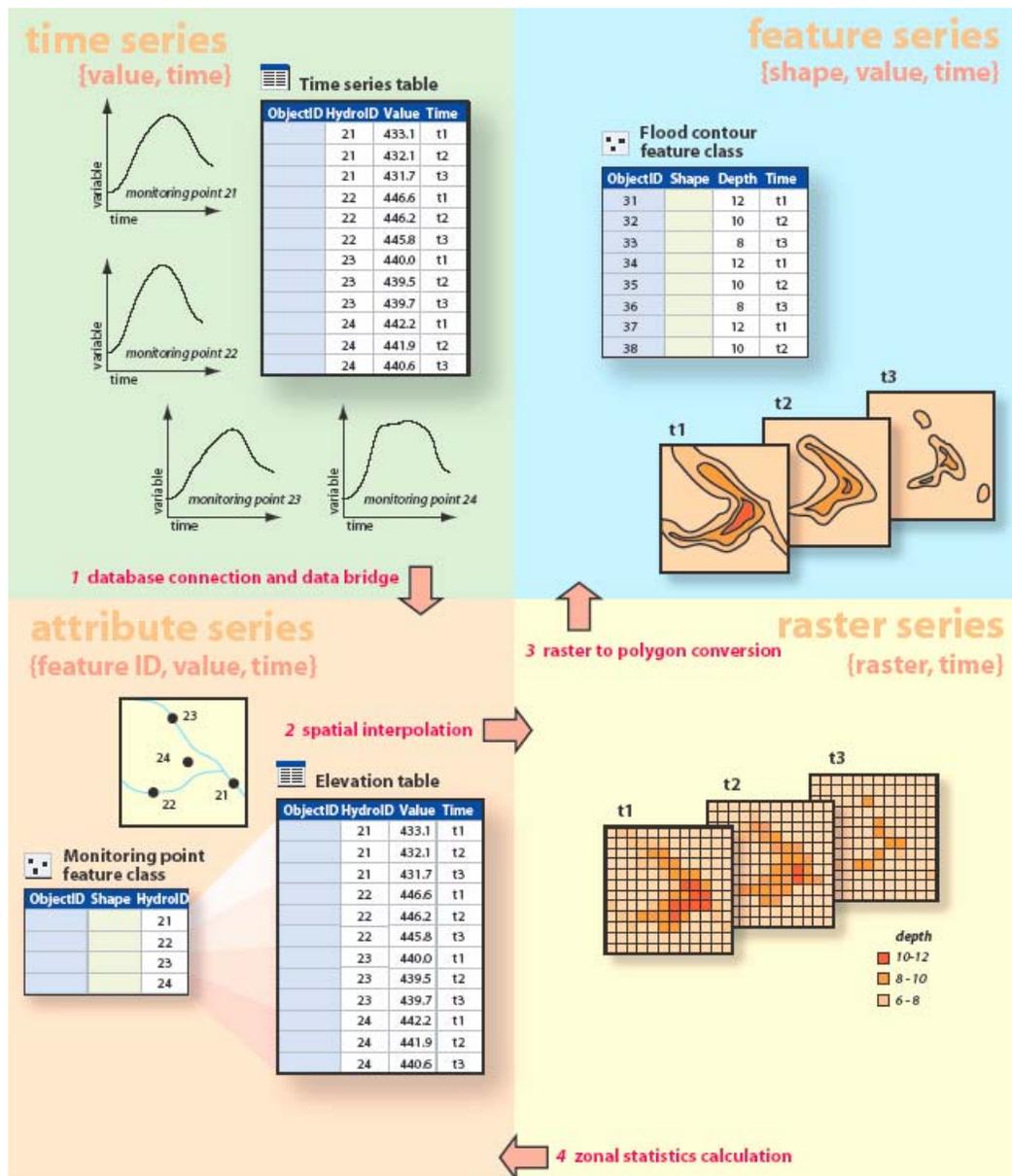


Figure 2.11: Extending the Arc Hydro Time Series Component for additional spatial-temporal data types (Arctur and Zeiler 2004).

An AttributeSeries is a feature class with a time-varying attribute. The MonitoringPointHasTimeSeries relationship within Arc Hydro is an example of

an Attribute Series. The recorded time series at the monitoring point is a dynamic attribute of that MonitoringPoint feature. A given feature can have one or more dynamic attributes. For time series, these are organized according to TSTypeIDs in the TSType table. An Attribute Series is not limited to point features; it may also represent dynamic attributes of polygon or line features as well. For hydrologic modeling, it is often useful to store the mass or energy fluxes occurring over catchment features as Attribute Series.

A Feature Series is a feature with a dynamic shape attribute. Each location and geometry of the feature is indexed by a particular time. This allows one to represent the dynamics of a feature moving through time, changing shape through time, or appearing or disappearing through time. Hydrologic modeling of flow control structures, flood plain inundation, and particle tracking all require the description of features with dynamic shapes. Combining the Attribute and Feature Series descriptions provides one with the flexibility to represent particles traveling through the landscape whose location and properties (chemical concentrations, for example) vary through time.

A Raster Series is a set of rasters indexed by time. At any given time, one of the rasters represents the current state of the system. The next raster will then update the system at the time at which it is indexed. This solution allows one to represent two-dimensional continuous space dynamics important to hydrology including fluxes like rainfall and evaporation, latent and sensible heat, and chemical deposition. One could also represent dynamic properties of the landscape including land use/land cover change. Having this information

organized with the database allows one to query for the present state of the system for a given time.

These data structures are the start to creating a spatiotemporal information system. There still needs to be a great deal of software development to implement the data structures and make them useful for modeling purposes. However, the potential impact of extending geographic information systems to include temporal dynamics is significant if the time dimension is handled as a core dimension and not simply an add-on viewing device. It may take some time before such a system is in commercial production, but ESRI is making the first steps with the forthcoming release of ArcGIS 9.2, which includes support for netCDF as a native file format.

As an example of what is possible from a spatiotemporal information system, consider the Integrated Data Viewer (IDV) developed and maintained by the Unidata. With the IDV, one can view 4-D data in a variety of ways including slices along planes or lines, time series at points, or isosurfaces (Figure 2.12). Because the IDV is a scientific research tool, it does not have the ease of use or the cartography options compared with ESRI software. Nonetheless, its underlying software structure as a 4-D environment is more geared to the needs of the earth science community than is current commercially available GIS software.

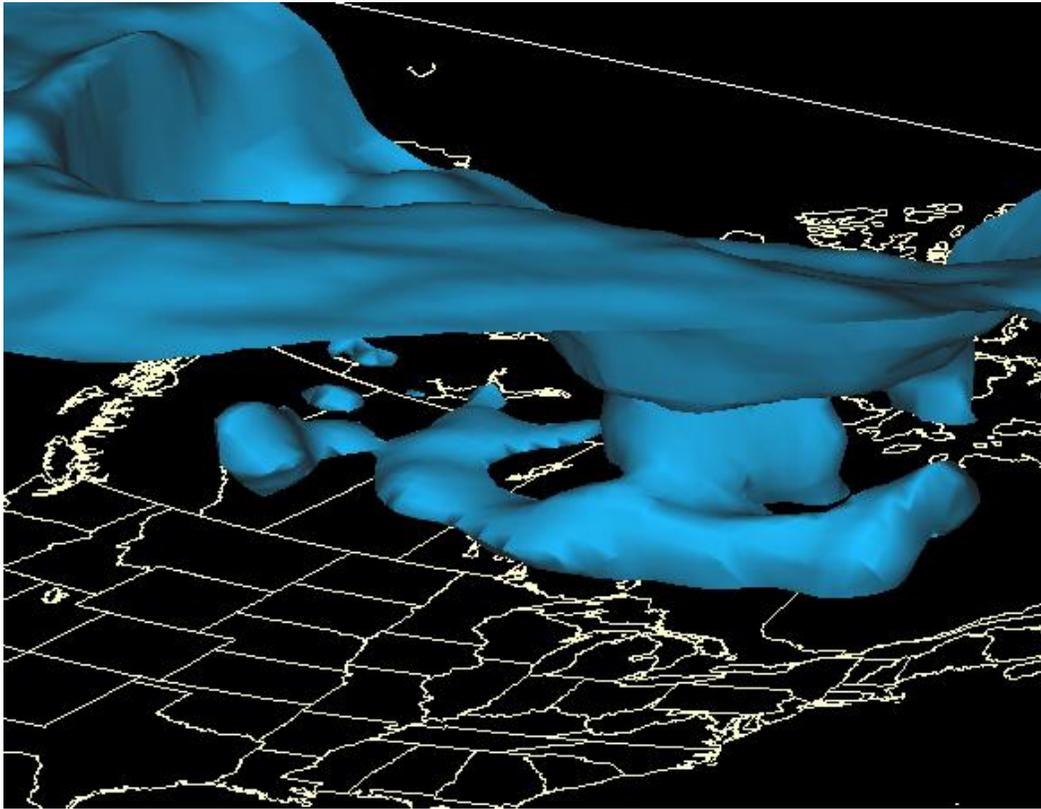


Figure 2.12: Minus 28°C Isosurface on December 11, 2000. The Integrated Data Viewer is an example of a spatiotemporal information system capable of visualizing 4-D fields. (Source: <http://www.unidata.ucar.edu/content/software/IDV/gallery/index.html>)

2.5 SUMMARY

CUAHSI, the Consortium of Universities for the Advancement of Hydrologic Science, Inc., seeks to unite hydrologic science research at U.S. universities through the establishment of a network of hydrologic observatories. These hydrologic observatories are experimental watersheds with unprecedented observing and modeling systems where scientists from around the world can gather to understand the water cycle at the basin scale. This is a major effort in

the hydrologic sciences represented by the nearly 100 university members whom have already joined CUAHSI in just four years.

A central component of the CUAHSI mission is an informatics infrastructure which allows the (1) management of locally collected datasets, (2) integration of federal hydrologic datasets, and (3) sharing of data and models among community members. Other similar efforts in the earth sciences provide guidance for the CUAHSI Hydrologic Information System effort and CUAHSI should seek to work closely with these efforts to ensure interoperability between data, models, and science. In particular, the Geological Network (GEON) is an important partner with CUAHSI because both efforts are utilizing the San Diego Supercomputing Center as a technology partner.

The focus of this research is to present a conceptual model for integrating data into a geotemporal framework referred to as a digital watershed. This digital watershed will allow one to reference observations and model results within the same space-time domain. The challenge in achieving this vision of a digital watershed is in extending traditional geographic information systems to include a temporal dimension. Goodall et al. (2004) provides a conceptual framework for representing geotemporal data. This research builds on this foundation and presents software tools which extend ArcGIS, a widely used commercial GIS produced by ESRI, to facilitate spatiotemporal hydrologic modeling.

Chapter 3: Methodology

The objective of this research is to design a geotemporal framework for hydrologic analysis. This framework allows one to establish the base-line of known (i.e. observed) and estimated (i.e. modeled) hydrologic state and flux variables within a common spatiotemporal coordinate system. The variety of data available through government sponsored observation networks, satellite-based remote sensors, weather and climate models, and the digitizing of national maps provide an excellent description of the natural environment. However, because these data are produced by different governmental agencies for use by different scientific communities, it is difficult to integrate the information into a common framework.

A geotemporal framework allows one to fuse data from heterogeneous formats and distributed sources into a common environment. Then, because these data are integrated into a common framework, it is possible to develop hydrologic analysis routines that operate on data based on its location in space and time, and not its original format, data model, or data source. The routines could search for referenced variables based on geospatial and temporal queries, or trace the connectivity of the hydrologic landscape from hill slopes, to river channels, to estuaries and oceans.

This methodology identifies issues in developing automated hydrologic analysis routines within a spatiotemporal environment. The basic principles of hydrologic analysis – conservation of mass, energy, and momentum – can be

coded into routines that operate on spatial hydrovolumes at any scale. These hydrovolumes are aware of their environment and can accumulate all exchanges of mass or energy across their boundaries through time.

3.1 HYDROLOGIC SYSTEM ANALYSIS

Hydrology has its roots in fluid dynamics: the study of the physical processes governing fluid motion. No matter if water is in the atmosphere, surface, or subsurface, its distribution and movement in space and time are subject to the conservation of mass and energy. These conservation laws can be applied to a volume of space, commonly called a control volume, to predict the transfer of water and its constituents through the natural environment.

3.1.1 The Systems Concept

Chow et al. (1989) show that the connection between hydrologic analysis and fluid mechanics is the concept of a control volume as a system. Using such an approach, one is able to simplify the complexity of the hydrologic cycle into a discrete volume of space with associated inflow, outflows, and internal changes (Figure 3.1). A catchment can be viewed as a system where the components of the system are the overlying atmospheric volume (i.e., the airshed), the surface and river channel volumes, and the subsurface volume underlying the catchment. Chow et al. (1989) provide a formal definition for a hydrologic system stating that it is “a structure or volume in space, surrounded by a boundary, that accepts water and other inputs, operates on them internally, and produces them as outputs.”

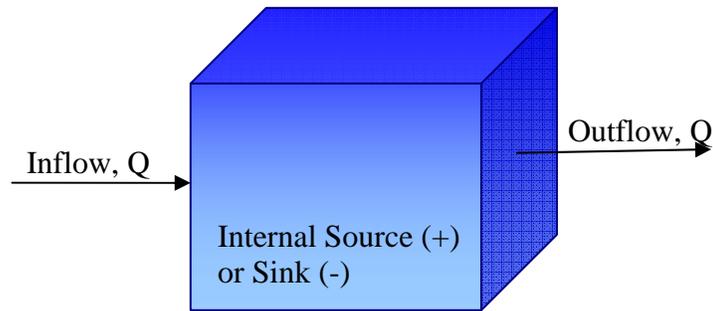


Figure 3.1: A conceptual picture of a control volume

Figure 3.2 shows a land surface system comprised of catchments, river reaches, and water bodies. Each of these features could be considered its own hydrologic system by defining the volume in space which the feature occupies and the inputs, outputs, and internal changes to the volume. One could perform a water balance on a river channel, waterbody, or catchment. Because the same flux or flow time series can represent output for one system and the input for another, time series provide a link between subsystems, uniting the landscape and the hydrologic cycle.

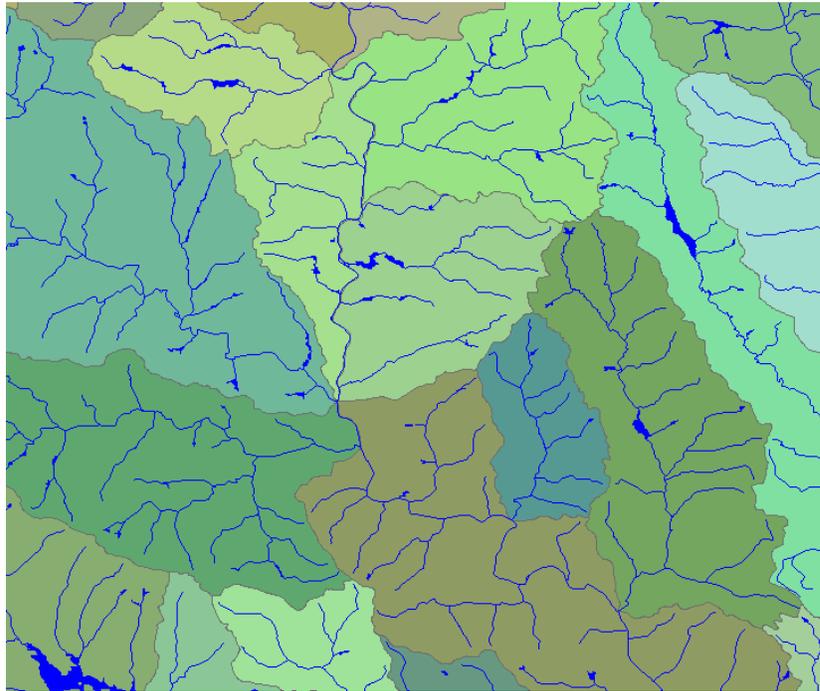


Figure 3.2: Each feature within the hydrologic landscape can be considered a hydrologic system.

One hydrologic system can be the conglomeration of smaller systems. Within a catchment system, one could define a set of embedded hydrologic systems including waterbodies, stream segments, or subcatchments. The benefit of this approach is that one can defined a hydrologic system at the scale appropriate for the level of detail required for a desired application. Many hydrologic models use the idea of hydrologic system analysis to estimate the movement of water and its constituents through the hydrologic cycle.

When performing hydrologic analysis on a system, the first step is to determine the geospatial boundary of the system. This boundary is the control surface of the system through which mass, energy, and momentum are exchanged.

The control surface can have zero or more associated inputs and outputs. In addition to these exchanges through the control surface, the hydrologic system can also have zero or more internal gains or losses of mass and energy. The exchanges of mass and energy through the system's control surface and the internal changes of mass and energy will equate to change in storage of the property within the system.

Once the gains and losses have been defined for a system, the next step is to sum these exchanges to estimate the net change in storage of mass or energy over time. If different data sources are used to perform this accounting, complications could arise due to input data being available for different temporal resolutions or for different temporal periods. The inflows, outflows, and internal changes could also be measured with different dimensions (flows, fluxes, and volumes) or they could have different units (cfs, m^3/s , etc.). Thus, to define the system is the first step, but to automate mass balancing using heterogeneous data sources, further steps are required to align the data in space, time, and measurement unit.

3.1.2 Hydrologic Time Series

The inflows, outflows, and internal changes to a hydrologic system are all considered hydrologic time series. A time series is a collection of value, time pairs with specific properties. Salas (1993) defines a hydrologic time series as a time series of a single hydrologic variable at a given site. A hydrologic time series has specific properties that identify the name of variable that was measured, how it was measured, and where it was measured.

For example, Figure 3.3 shows the metadata associated with a USGS stream discharge station. The metadata regarding the time series includes the collection agency (USGS), the unique identification for the particular gage (USGS station number), the units of measurement, the station name, and a description of what the time series represents (daily mean streamflow). If a time series is properly annotated with these properties as metadata, it is possible to isolate relevant time series from a large database for a particular study, and to develop a set of generic analysis routines for transforming and converting time series.

```

#
# U.S. Geological Survey
# National Water Information System
# Retrieved: 2005-05-03 10:14:44 EDT
#
# This file contains published daily mean streamflow data.
#
# Further Descriptions of the dv_cd column can be found at:
# http://waterdata.usgs.gov/nwis/help?codes_help#dv_cd
#
#
# This information includes the following fields:
#
# agency_cd      Agency Code
# site_no       USGS station number
# dv_dt         date of daily mean streamflow
# dv_va         daily mean streamflow value, in cubic-feet per-
second
# dv_cd         daily mean streamflow value qualification code
#
# Sites in this file include:
# USGS 08164300 Navidad Rv nr Hallettsville, TX
#
#
agency_cd      site_no dv_dt    dv_va    dv_cd
5s           15s      10d      12n      3s
USGS        08164300      1961-10-01      60
...

```

Figure 3.3: Stream discharge time series obtained from the USGS National Water Information System.

It is difficult, however, to define a generic list of attributes adequate for describing hydrologic time series. The United States Geologic Survey and the Environmental Protection Agency have different schemes for describing a hydrologic time series which, ultimately, makes time series from these two federal agencies incompatible (Maidment 2005). Despite these differences, it is possible to define a set of descriptors that are required for the majority of hydrologic time series and for the majority of hydrologic analysis.

Maidment (2002) isolated a set of universal hydrologic time series properties required to describe time series for hydrologic analysis and modeling (Table 3.1). Although particular applications may require additional properties, such as details about the measurement technique used to obtain the time series observations, this list provides a starting point for developing application specific time series properties.

Variable	The name of the hydrologic variable, e.g. streamflow
Units	Units of measurement
IsRegular	Whether data are regularly or irregularly measured in time
TSInterval	Time interval represented by each measurement
DataType	Type of time series data, e.g. instantaneous or cumulative
Origin	Origin of time series data

Table 3.1: Hydrologic time series properties as defined by Maidment (2002)

Without doubt, the variable name and measurement units are important properties of a time series. The variable name is of primarily use to end users for understanding what has been measured. Because so many variables could be measured, this property must remain unrestricted to allow full freedom in describing the time series. The Units property, on the other hand, must be restricted to a controlled vocabulary, i.e. a list of acceptable terms, to ensure software tools can understand and transform the units of a time series. However, these restrictions must be flexible and easily extended for specialized studies.

The IsRegular property is a Boolean variable, meaning it is either true or false. Many analysis programs and techniques require the time series to be recorded on regular intervals of time. Time series of water quality parameters, however, are often collected sporadically in time. The samples are collected either in response to a potentially harmful event or when it is convenient to travel to the sampling site. Having the IsRegular property allows one to quickly isolate regular time series from irregular time series prior to ingestion in analysis or modeling routines.

The TSInterval property distinguishes instantaneous time series from interval time series. Some variables can only be measured over intervals of time (the depth of rainfall) and some can only be measured at instants of time (temperature). However, it is common in hydrology to talk of statistics of instantaneous time series over time intervals (e.g. minimum daily temperature or daily averaged streamflow). When one is presenting a statistic of a continuous variable over a time interval, it is important to remember that this introduces uncertainty that should not be neglected in modeling or analysis of the information.

As stated in the last paragraph, time series can represent either instantaneous or interval time series values. An instantaneous time series can be either the variable itself or an accumulation of the variable over time. An interval time series can be an incremental value, precipitation was the example of an interval time series given in the previous paragraph, or a statistic of a continuous variable over a time interval. Maidment (2002) has summarized the various

classes of time series into six types: instantaneous, cumulative, incremental, average, maximum, and minimum (Figure 3.4). The DataType property can be any one of these six types.

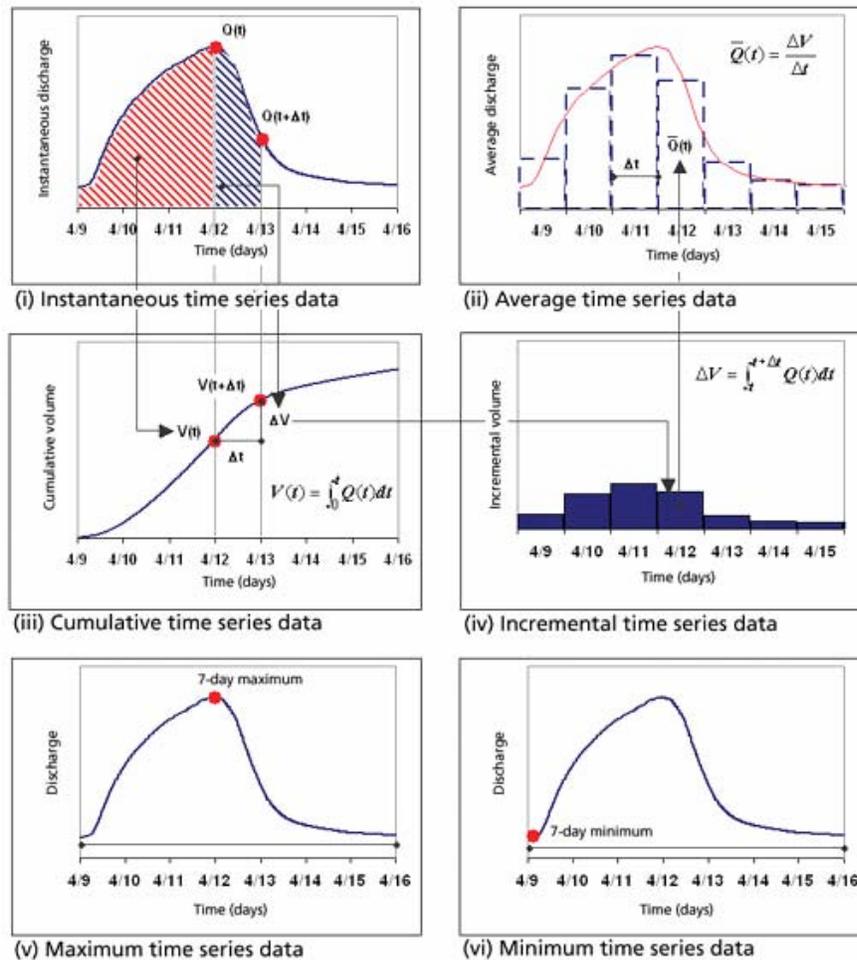


Figure 3.4: Different time series data types as defined by Arc Hydro (Maidment, 2002)

The DataType property is critical to hydrologic analysis and routines for transforming time series. One simple example of this is the development of a

generic accumulation routine. If the `DataType` property of the time series is instantaneous, the accumulation will operate differently than if the property is averaged (Figure 3.4). Furthermore, if the `DataType` property is cumulative, the routine should exit the accumulating routine without updating the time series values.

The `DataType` property has some means for describing statistics over an interval of time, but does not fully address the need to represent uncertainty associated with aggregating time series measurements. The average, minimum, maximum, and incremental data types all represent statistics of a time series over a particular interval of time. It seems logical to extend this list to include standard deviation, Skewness, and Kurtosis, as the moments of a probability distribution function, and other statistics useful to hydrologists (e.g. median, mode, range, coefficient of variation, etc).

Finally, the `Origin` property indicates whether the time series was observed or generated. In broader terms, the origin of a time series should provide information about the original source of the data, not simply whether the data was recorded or measured. As an example, suppose that the same variable is observed by two different agencies. Under the present system, it is possible that the time series type will be exactly the same for both of these variables, and the distinction of the two collection agencies would be lost. However, if the `Origin` field is used to store the name of the collection agency, it is possible to perform a query to extract the time series for different collection agencies or all time series independent of collection agencies.

Maidment (2002) has proposed a time series typing system capable of uniquely classifying heterogeneous time series. There are some potential shortcomings of this time series typing system – namely, the TSInterval and Origin properties are too restrictive and the DataType only include four statistical types – yet it is nonetheless a helpful tool in attempting to inventory the multitude of hydrologic time series and in achieving interoperability between time series collected by different federal agencies and those generated by hydrology and hydraulic models.

Once the hydrologic time series have been properly categorized, the next step for automating a mass or energy balance using heterogeneous data is to develop a library of generic time series processing routines for manipulating measurement dimensions or rescaling a time series in space or time. This library of routines allows one to more easily prepare time series for hydrologic system analysis or to manipulate the dimensions of time series to a consistent set of units for exploratory data analysis. Unit and dimension conversions are a simple, yet error prone processes, and automation of the process would help save time and avoid mistakes.

3.1.3 Spatial, Temporal, and Measurement Dimensions

Dimensions of space, time, and measurement units of a time series are critically important to understanding its representation of either a state and movement of mass and energy. A time series has a spatial dimension which represents the point, line, area, or volume to which the measurement is applied. Likewise, its temporal dimension represents the instant or interval of time for

which the measurement is valid. Finally, the measurement dimensions of a time series are an indication of the quantity being observed (mass, energy, etc.).

One of the difficulties in achieving interoperability between hydrologic time series is that details of the time series' spatial, temporal, and measurement dimensions are not always explicitly recorded. This leads to misconceptions about what the time series values truly represent. In addition, there are rules for acceptable combinations of these three dimensions (e.g. a volume cannot be defined for a point in space). Converting time series properties should operate and update all three of these dimensions. For example, converting a flux to a flow changes the spatial dimension of the time series from an area to a point in space.

Creating a generic routine capable of converting the dimensions of a time series must begin with a routine for converting between units of the same dimension. The most obvious algorithm is to expressly provide the conversion factor between every combination of units. The advantage of this approach is that it allows one to use exact conversions when available and it is an intuitive approach. The disadvantage is its inefficiency: for n units, it will require 2^n conversion factors to express the conversion factors between all possible combinations of units.

A second approach is to provide conversion factors from every variable to a standard unit for each dimension. This approach is more efficient because it requires only n factors to convert between n units. The limitation of this approach is that it will not always include exact conversion factors. Taking length as an example, if meter is the standard unit and one wishes to convert from inches to

feet, the first approach would provide 12 in per foot as the conversion factor, while the second would provide some approximation of 12 by going through meters (the standard length unit) to derive the conversion factor. If a sufficient number of significant figures are carried through the conversion process, the inexactness introduced by the second method could be minimized to an acceptable level.

With the ability to convert between units of a dimension, the next step is to provide the ability to convert the spatial, temporal, and measurement dimensions of a time series. While all mass and energy either enters or exits a control volume through an area, these exchanges are often expressed in dimensions which imply an input through a point (flow), a line (line flux), or an area (areal flux). Conversion between fluxes and flows requires the spatial integration and differentiation of the time series values. Likewise, to convert fluxes and flows to volumes of water requires the ability to temporally integrate a time series. Thus, converting the dimensions of a time series is the process of performing calculus on the time series.

Consider the following example of the conversion of an areal flux to a flow. If the flux is a function of the x and y dimensions, $f(x, y)$, then its integral over the x and y dimensions is equal to a flow (Equation 3.1).

$$Q = \iint_{y,x} f(x, y) dx dy \quad 3.1$$

If the flux is constant over the area of integration, Equation 3.1 can be simplified to Equation 3.2

$$Q = fA \quad 3.2$$

where A is the area of integration. Thus, the flow is simply the product of the constant flux and the area of integration.

Likewise, a line flux is the amount of mass, energy, or momentum entering a control volume per unit length, per unit time. One can calculate the flow represented by a line flux by integrating the flux over its length (Equation 3.3).

$$Q = \int_L f(L)dL \quad 3.3$$

The geometry of the line can be in one, two, or three dimensions and the line flux can vary as a function of the length of the line. Again, if the areal flux is constant over the line, then Equation 3.3 reduces to Equation 3.5.

$$Q = f_L L \quad 3.5$$

Thus, the conversion factor between a constant line flux and an area flux is the length of integration.

In some cases, the amount of mass, energy, or momentum entering a control volume has been integrated over a time interval as well as over a spatial domain. In this case, the time series would not be expressed in “per time” dimensions, but as simply a mass, mass per unit length, or mass per unit area. Following the same argument above, it can be shown that the conversions between these dimensions and the flow, line flux, and area flux dimensions

requires the time interval of integration as the conversion factor, assuming the flow, line flux, or area flux is constant over the time interval.

Figure 3.5 summarizes the conversion between dimensions. If the flows and fluxes are constant over space and time, the conversion between dimensions is simplified from calculus to arithmetic. One way of representing spatially variable fluxes is as a uniform grid of constant fluxes. Then the conversion between flux and flow dimensions is simply the product of the flux and the grid cell size. However, with geographic information systems, it is possible to define spatially discrete geometries with constant fluxes. This removes the restriction of uniform grids and allows greater flexibility in geographically representing fluxes in space, an idea that will be further explored in the following section.

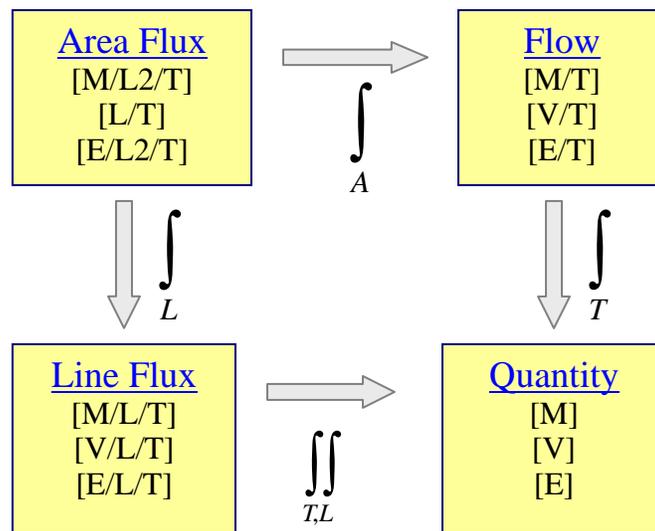


Figure 3.5: Common unit dimensions used in hydrology and the integrals used to convert between them

3.2 REPRESENTATIONS OF SPACE AND TIME

A geotemporal information system should build off the foundation laid by geospatial information systems because the referencing of data in geographic space is far more complicated than the referencing of data in time. This is primarily true because space is three dimensional while time is one dimensional. Moreover, georeferencing data requires one to construct a mathematical model to represent the Earth as a geoid, and, for many types of analysis, to project the Earth from a latitude/longitude coordinate system to a projected x/y coordinate system. Thus, the next sections provide a conceptual design for adding a temporal dimension to geographic information systems.

3.2.1 Entities and Fields

The distinction that GIS software makes between vector and raster data is an important concept in Geographic Information Science. In a broader sense, this distinction in data types says that there are two views of space: entities and fields. To view space as entities is to see discrete objects with properties and behaviors, while to view space as fields is to see a defined coordinate domain, often, but not limited to, Cartesian coordinates. Many geographic information systems support both views of space, and a hydrologic information system must also support both. Atmospheric data and terrain data are often represented as fields, while the watersheds and stream networks are most often represented as entities. Thus, the ability to work with both views of space is critical to modeling the movement of water and its constituents through the atmosphere and land surface environments.

Entities are aggregations of space meaning that they have an associated geospatial shape (Figure 3.6). They can be composed of multiple connected points used to make a line, multiple lines used to make a polygon, or multiple polygons used to make a volume. In water resources geospatial databases, volumes are used to represent river channels or water bodies, polygons to represent watersheds, and lines to represent rivers. In this sense, entities are geospatial objects, each with a unique set of properties.

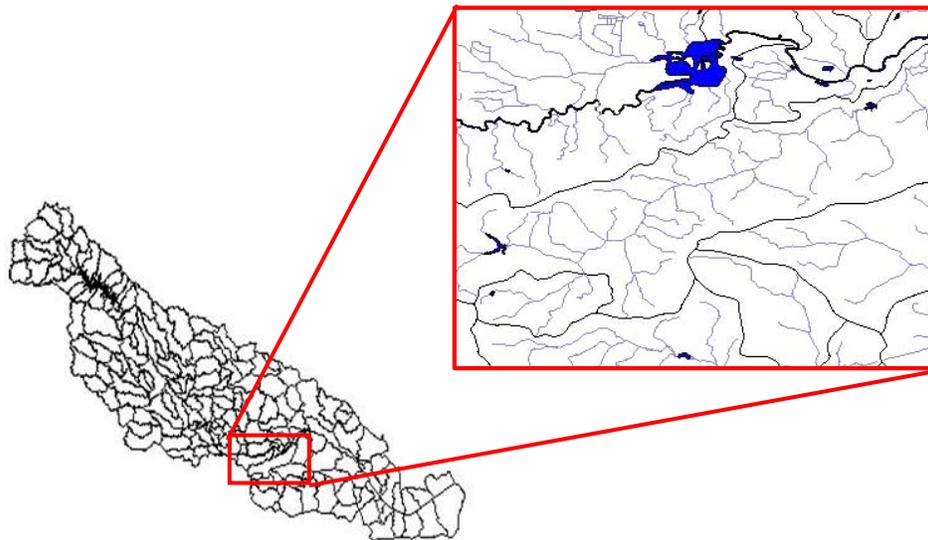


Figure 3.6: Viewing space as entities. Each catchment, river reach, and water body within the Neuse River Basin is a unique entity with properties such as a georeferenced shape or set of spatial coordinates.

Fields are spatially continuous domains where values are referenced to points in a coordinate domain (Figure 3.7). The field view is often used to represent physically continuous variables which are being digitally stored as discrete pixel or voxels. It is logical to store many environmental variables as fields because these variables are continuous in space. Rainfall, terrain, and atmospheric deposition are all examples of continuous environmental datasets best represented as fields. In addition to these scalar fields, it is also possible to have vector fields where each point in space has two values: a direction and magnitude.

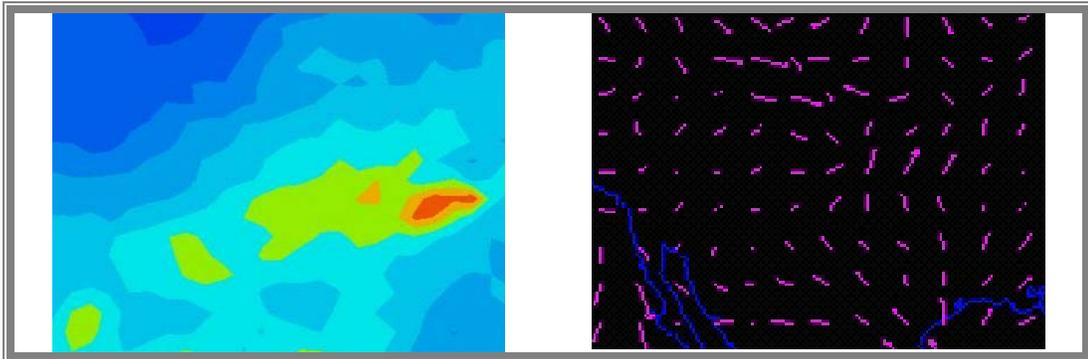


Figure 3.7: Viewing space as fields. Space is a grid and variables are associated to specific locations within the grid. A field can represent either scalar or vectors.

This is not to say entities could not be used to represent rainfall, terrain, or atmospheric deposition; there are examples of scenarios where a rain storm can be considered a discrete entity moving through space. A single variable does not always have to be represented using the same view of space. It is the intended application, and not the variable, which determines the appropriate view of space. For hydrology, it is often useful to have the ability to represent data as either entities or fields and to transform variables between the two views of space (e.g. transform precipitation fields to watershed rainfall entities.).

A hydrologic information system, therefore, ought to represent the two geographic space views of space in an integrated environment. Accommodating both views of space means having a geospatial framework where not only can weather precipitation model output and watershed polygons co-exist, but where the two are aware of each other. A volume entity can then summarize the

properties of a field within its perimeter, or a point entity could extract a time series from a 4-D field. Providing this functionality facilitates the parameterization of distributed scientific models and the aggregation of these models' output to real-world objects that determine policy.

3.2.2 Fields and Time

To supplement the field view of space with time is a natural extension of the field data model, making the extension relatively straight forward. The field data model, as discussed before, views space as a coordinate domain and values are associated to a particular point within the coordinate domain. The coordinate domain is, theoretically, not restricted to only two-dimensional space. Therefore, to expand to a three or four dimensional domain does not require a departure from the underlying data model, although it would require additional software to manage and visualize the temporal domain.

Within geographic information systems, the most used field based data format is a grid. Grids are essentially image files where each pixel of the image contains a single value. The simplest way to represent dynamic fields, therefore, is to store a series of grids, each with a time stamp indicating the instant or interval over which the grid is valid. The type of time series described by the grid can be indexed using the same time series typing system used by other hydrologic time series. The time series type defines the variable being measured, the units of measurement, and other time series properties outlined by Maidment (2004).

There are some low-level data management issues that come into play when constructing the field-time dataset. It requires a significant amount of data

to store the entire state of a region for each time step. One could optimize the data storage, particularly for relatively static variables, by storing one grid and providing updates to the grid at particular instants in time. Under this scheme, the grid cells would be valid continuously through time and at any point in time, one or more grid cell could be replaced by a new value (Langran 1992).

To move past a two dimensional view of space into three dimensional space requires the abandonment of the grid data format and the adoption of a new format, such as netCDF. Using the netCDF format, one could store a hydrologic variable as in four-dimensional space-time (x,y,z,t) as an array. Each entry in this array is a realization of a continuous field. Combining the netCDF variable with spatiotemporal interpolation routines completes the representation of a multidimensional continuous field by providing a unique value of the variable at any location within space-time.

NetCDF is the primary data format used by the Integrated Data Viewer, a software system developed by Unidata for the visualization of weather and climate data. The Integrated Data Viewer surpasses ArcGIS in its ability to visualize multidimensional data. This is in part because netCDF is more capable of representing multidimensional space-time variables. Thus, it is useful for ESRI, the developers of ArcGIS, to investigate incorporating netCDF as a data format and tools for visualizing the four-dimensionality of a netCDF file.

3.2.3 Entities and Time

Extending the entity space view to include time requires a new data model because of the complexities involved with representing entities that vary through

time. The data model for entities in geographic information systems views entities as static objects with static attributes. Currently available geographic information systems include no native support for allowing shapes or attributes, to depend on time. There is an extension available for ArcGIS called Tracking Analyst that allows one to view temporal data, but it does so by manipulation of native GIS formats and not through a redesign of the basic data models within the software. This leads to performance issues resulting from inefficient memory use that greatly limits to functionality of the Tracking Analyst for large datasets.

Conceptually, there are two possibilities for the temporal nature of entities: (1) the entity could have one or more dynamic attributes, (2) the entity could have a dynamic shape (where shape is a combination of geometry and location). Examples of hydrologic entities with dynamic attributes are observation stations where hydrologic time series are recorded. Examples of hydrologic entities which have dynamic shape are flood inundation polygons or a raindrop traveling within or between hydrologic systems (Figure 3.8). Dynamic entities, therefore, could be used to represent both the Eulerian and Lagrangian view points, i.e. entities could be fixed systems through which water travels, or as volumes of water which travel through the hydrologic system.

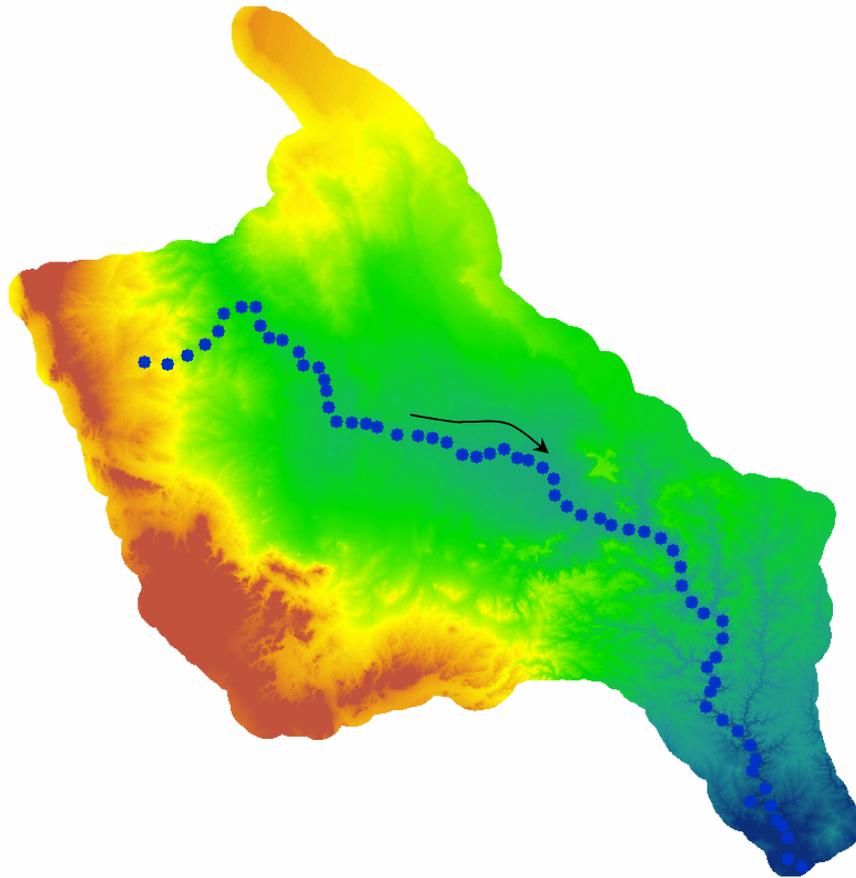


Figure 3.8: A rain drop moving through the surface water system is an example of a dynamic entity in the Lagrangian viewpoint

Although this is a very powerful idea which could offer great potential to geospatial environmental and hydrologic modeling, it is not the focus of this research. Instead, the focus of this research is primarily on representing the temporality of stationary features (fluxes on watershed features, hydrologic time series at gaging stations, etc.). Future research should extend this research to representing features which move and change shape through time.

3.3 DESIGNING A GEOTEMPORAL FRAMEWORK

The purpose of this section is step back from the practical extension of current GIS software to handle temporal data and to explore a truly spatiotemporal information system where time is a core dimension of the system. There are important similarities between space and time, and the analogy of time as a spatial dimension is useful to providing the conceptual design for a temporal geographic information system as the backbone for creating a digital watershed.

3.3.1 Geospatial and Temporal Geometries

Geometries can be defined for both geospatial and temporal dimensions (Sumrada 2003). Geographic space is three dimensional, thus there are four possibilities for geometries which can be built within geographic space: points, lines, areas, and volumes. Time is one dimensional, limiting the possible temporal geometries to points and lines. A temporal point is an instant in time and a temporal line is an interval of time. An interval of time is defined either by an instant and duration or as a beginning and ending instant of time.

In addition to simply geospatial and temporal geometries, it is also possible to construct spatiotemporal geometries. Consider the simple example of a set of point observations taken at different locations over time. In a geospatial domain, each observation is a point referenced by its latitude and longitude coordinates. In a temporal domain, each observation is also a point referenced by its date and time of observation. Taking the geospatial and temporal coordinates together, the observation is a three dimensional point in a latitude, longitude, time coordinate system.

Combined geotemporal geometries are not limited to points; lines, areas, and volumes also have significance. Mixing geospatial and temporal dimensions can present patterns in data not visible by simply viewing the data with maps and charts. One could construct a vertical line through the domain to represent a history at a single location (Figure 3.9a) or a horizontal line to represent changes through space at a single time. Likewise, a vertical plane through the domain would represent a history for a region of space, while a horizontal plane would represent the state at a single instant of time (Figure 3.9b). One could also construct a volume of the domain to represent a time history of an area of space.

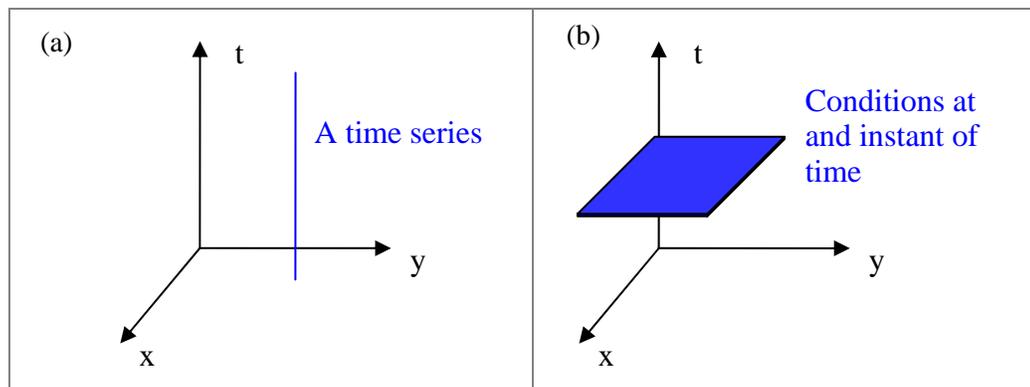


Figure 3.9: Examples of spatiotemporal geometries

If the point measurements are all recorded in a single geospatial coordinate system and the time values all from the same time zone, then there are visualization techniques possible with software like Matlab or Excel, which do not require the aid of a geographic information system. The problem with this approach arises when one wishes to create a map with data referenced according to a different coordinate domain. Because there are so many geospatial

coordinate systems, this is a very common occurrence when working with geographic datasets. In fact, for many applications, the primary benefit of a geographic information system is bringing data defined in different coordinate systems together into a common environment.

3.3.2 Geospatial and Temporal Coordinate Systems and Projects

The ability of a GIS to bring data together into a common environment is dependent on an underlying library of coordinate systems and project routines. Geospatial coordinate systems represent space either as a globe or as a plane. The global dimensions, such as latitude and longitude, are useful for finding locations in geographic space, whereas the projected coordinates are useful for deriving distances or directions in space. The process of creating a flat representation of the Earth, however, introduces unavoidable inaccuracies. This is why there are so many projections: many of the routines quickly lose accuracy as one departs from the project routine's reference lines. Thus, the multitude of available geographic coordinate systems ensure that one can maintain accuracies in spatial direction and distance over particular regions of the Earth.

Temporal coordinate systems are much simpler than geospatial coordinates because time is, for most practical purposes, linear. However, there are still some very important properties of time which must be considered. For example, hydrologic information systems must be able to handle data collected in multiple time zones. Likewise, some states in the U.S. do not observe day light savings time and some do, making it even more difficult to place samples from different sources along a common time line.

In the scientific community, the typical approaches for overcoming these difficulties are either to reference all data to a single temporal coordinate system (e.g. Universal Time Standard) or to use Julian days (the fraction of days that has passed since a stated date and time). While these are appropriate solutions for referencing world-wide datasets, it may cause confusion to reference regional or local data according to Universal Time or to Julian Days. One would have to keep a conversions factor in his or her head in order to properly interpret the information. Thus, a conversion from these temporal coordinate systems to the local time zone would be necessary.

The routines to convert between temporal coordinate systems are not complicated. Assuming adequate metadata defining the temporal coordinate system, it would be possible to devise a temporal referencing system capable of placing heterogeneous datasets along a common time line. Microsoft Windows has a time zone property which identifies the computer's time zone and, using this information, an application could automatically "re-project" all temporal data to the user's time zone. This avoids the necessity to convert all data to a single time zone prior to ingestion into the hydrologic information system.

The current version of ArcGIS is capable of doing geospatial coordinate system conversions "on-the-fly" like that being discussed for handling discrepancies in temporal coordinate systems. When one adds data to a map, if that data's coordinate system does not match the map's coordinate system, ArcGIS runs a conversion routine to remedy the coordinate system mismatch problem. This use of automatic conversions, although useful for quickly

displaying information, can lead to confusions if the conversion process is not adequately explained to the end user and documented in user guides.

3.4 HYDROLOGIC OBJECT CLASSES (HYDOOBJECTS)

A product of this research is a prototype design of two hydrologic classes built from the concepts of hydrologic system analysis and temporal GIS discussed in the previous sections: geospatial time series and hydrologic flux coupler. As a brief introduction to object-oriented programming, a software class is used as a template for creating objects within the program. Every object has an associated set of properties and methods which are defined by the class. A property is an attribute of the object (e.g. Units is a property of the geospatial time series class) and a method is an action that can be performed by an object (e.g. AddToChart is a method of a geospatial time series object).

3.4.1 Geospatial Time Series Class

A geospatial time series describes a hydrologic time series associated with a georeferenced geometry. It can be used to describe either a hydrologic flux or state variable, meaning it can store the properties of water within a region of space or the movement of water through a region of space. In the former, the geospatial time series presents the properties of a point or volume of space, such as the concentration of a contaminant within a lake or river reach. In the latter, the geospatial time series describes a flux of material either through an area or a line or a flow of material between volumes of space.

In relation to the spatiotemporal object classes presented in the literature review, a geospatial time series can be thought of as describing one feature within

an attribute series (Goodall et al. 2004). That one feature is related to a particular type of time series, as described by Arc Hydro's TSType table. If a geospatial time series object is constructed for each feature within a feature class and for a single type of time series, then it can be considered an attribute series.

A geospatial time series represents a hydrologic time series with both geospatial and temporal geometries. The geospatial geometry of the object can be displayed on a map, intersected with other geometries, or used to calculate the distance between time series. With the temporal geometries (each value has its own associated temporal geometry) one could add a geospatial time series to a chart, query for the values that fall within a specified date range, or calculate the interval of time between two values.

The geospatial time series class is a template for constructing a geospatial time series object. The object can be populated with attributes from an Arc Hydro geodatabase, a webpage returned from a National Water Information System (NWIS) web query, or, with some effort, any other hydrologic time series with the correct metadata description. The behavior of the object will be the same despite its data source. One can change the temporal scale of a time series (e.g. from daily averaged to monthly averaged) using the same routine, even if the object is populated directly from an NWIS webpage or from a Arc Hydro geodatabase stored locally. The geospatial time series class, therefore, provides interoperability between time series formats and allows one to create software capable of ingesting multiple sources of hydrologic time series information.

3.4.2 Hydrologic Flux Coupler Class

A hydrologic flux coupler class describes a hydrologic system as presented in the first section of the methodology chapter. It could represent a region of space in the subsurface, surface, or atmospheric environment and can be defined at any scale in space. The flux coupler object is not meant to derive the exchanges of material between itself and its surrounding environment; instead, it is designed to summarize observed and modeled exchanges from different sources in order to understand the movement and storage of material within the landscape.

To do this, a flux coupler object is coupled to a one or more geospatial time series where each geospatial time series describes an exchange between the flux coupler and its surrounding environment. The flux coupler has the ability to transform the dimensions of coupled geospatial time series into flow dimensions (using the intersection area or line between the geometric shape of the flux coupler and geospatial time series) and to, from these flows, calculate the rate of storage change through time.

The coupling between geospatial time series and a flux coupler occurs over the geospatial intersection of the two features. When an area geospatial time series with flux measurement units is coupled to a hydrovolume, the exchange occurs over the intersection area between the two shapes (Figure 3.10a). If a line geospatial time series is coupled to a hydrovolume, then the exchange of material occurs over the line common to both geometries (Figure 3.10b). Finally, if a point geospatial time series is coupled to the hydrovolume feature, the exchange

is simply ingested into the hydrovolume and the intersection shape is not computed.

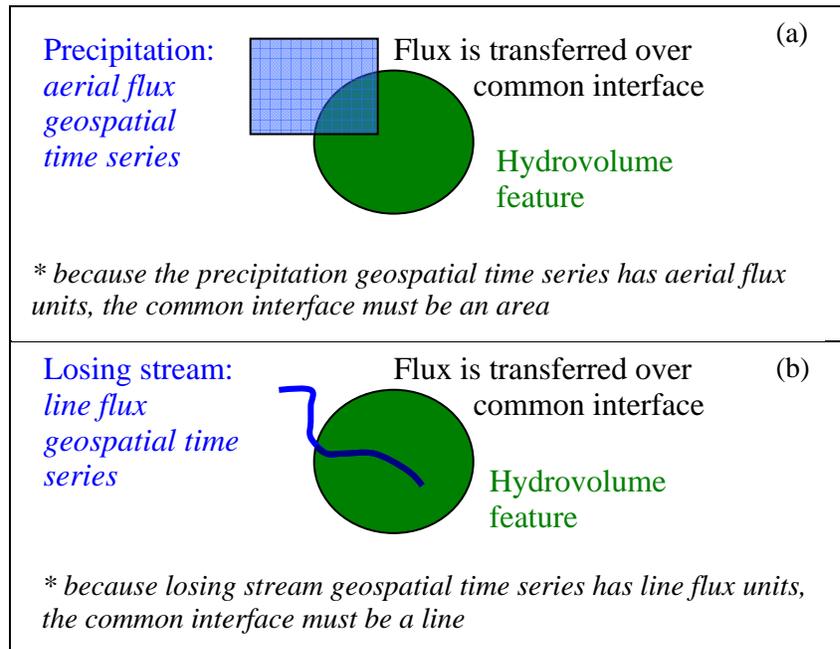


Figure 3.10: Coupling a geospatial time series and a hydrovolume occurs through the geospatial intersection of the two objects

The intersection occurs when the hydrovolume is constructed. The coupled geospatial time series is cloned, meaning a new object is creating in memory with precisely the same properties as its parent, and the shape property of the clone geospatial time series is updated. Thus, when the area or line flux is transferred to a flow dimensions within the mass or energy balancing procedures, the correct area or length is used for the conversion.

3.5 SUMMARY

Taking fundamental hydrologic concepts like time series, and system analysis, it is possible to customized basic geographic information science ideas and concepts to better accommodate the needs of the hydrologic community. These needs include the ability to pass fluxes between geometries in space, to visualize and process spatiotemporal data, and to work across data formats, data models, and data sources.

Given this vision, the task remaining is to implement the hydrologic concepts of a geospatial time series and a hydrologic flux coupler as software classes. These software classes can then be used to extend ESRI ArcGIS software for geotemporal visualization and analysis of hydrologic data. The next chapter, Chapter 4, presents the implementation of these classes in object-oriented programming terms and concepts. Chapter 5 then shows an application of the object classes for building hydrologic software tools with-in the ArcGIS software system.

Chapter 4: Procedure of Application

Conceptually, hydrologists view the water system as stores and fluxes. To implement this conceptual view of the hydrologic system, it is necessary to develop programming object classes which allow the manipulation of time series information and the estimation of water and energy balances. This research presents two object classes for such tasks: geospatial time series and hydrologic flux coupler. These classes are the beginning of a class library, name HydroObjects, which aid in the development of custom software tools for the hydrologic visualization and processing within a geotemporal framework (Figure 4.1).

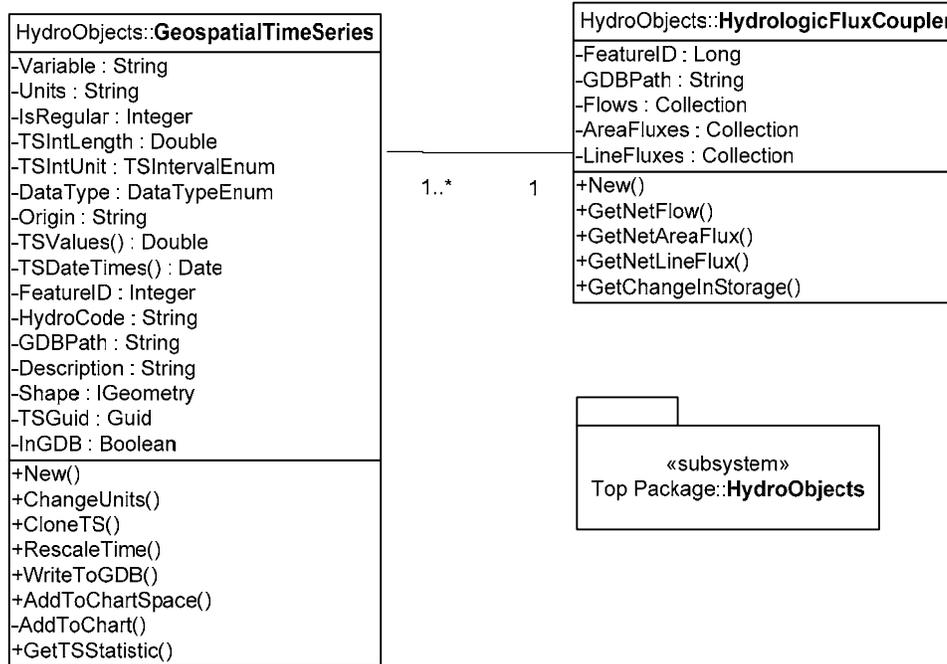


Figure 4.1: A Unified Modeling Diagram (UML) of the HydroObjects class library.

4.1 IMPLEMENTING HYDROOBJECTS

The object classes produced in this research are written in Visual Basic .Net and make use of the ESRI ArcObjects Library for geospatial processing and analysis. The concepts and overall design of the objects can be implemented in any object-oriented programming language. This chapter provides an explanation of the class library in terms of the reasoning behind certain classes, properties, and methods. If a more complete description is required, Appendix B contains documentation of the classes within HydroObjects.

4.1.1 Geospatial Time Series

A geospatial time series object derives many of its properties from the Arc Hydro TimeSeries and TSType tables. It also has a shape property derived from the feature associated to the time series. It is, therefore, the combination of the properties from three Arc Hydro object classes and provides a complete picture of a hydrologic time series in both time and space. In addition to the Arc Hydro properties, the geospatial time series class has additional properties added for managing multiple geospatial time series objects in software applications. Because the basic Arc Hydro time series properties were described in the Methodology chapter, the focus here is on the properties not included in the TimeSeries and TSType Arc Hydro Tables: GeneratedDescription, GeodatabasePath, InGDB, Shape, TSIntervalLength and TSIntervalUnit, TSUnitType and UID. The full list of geospatial time series class properties is given in Table 4.1 and the enumerations, which are essentially coded value domains for particular variables, are provided in Table 4.2.

DataType <i>DataTypeEnum</i> (see Table 4.2)	Type of time series data e.g. instantaneous, cumulative, averaged, etc.
FeatureID <i>Long Integer</i>	HydroID of the feature described by the time series (HydroID is a unique ID used in the Arc Hydro data model).
Description <i>String</i>	Description of how a time series was generated, the addition of two other geospatial time series objects, for example.
GDBPath <i>String</i>	Gives the geodatabase path, if the object was constructed from Arc Hydro
HydroCode <i>String</i>	The public identifier of the object, for example the USGS Station ID or the NDCD CoopID.
InGDB <i>Boolean</i>	Read only property that indicates if the object was constructed from an Arc Hydro geodatabase and has not since been modified.
IsRegular <i>Boolean</i>	Whether data are regularly or irregularly measured in time
Origin <i>String</i>	Description of the source for the time series
Shape <i>ESRI.ArcGIS. Geomtery.IGeometry</i>	The shape of the feature associated to the time series
TSDateTimes <i>DateTime()</i>	The date/times of measurement
TSLength <i>Double</i>	The length of time represented by each measurement. This is used with TSIntervalUnit to represent the TSInterval.
TSUnit <i>TSIntervalEnum</i> (see Table 4.2)	The unit of time (second, minute, hour, etc.) represented by each measurement. This is used with TSIntervalLength to represent the TSInterval.
TSTypeID <i>Long Integer</i>	Identifier for the type of time series
TSUnitType <i>UnitTypeEnum</i> (see Table 4.2)	Gives the dimensions of the measurement unit
TSValues <i>Double()</i>	The time series values
TSGuid	Read only GUID for the object.

<i>GUID</i>	
Units <i>String</i>	Units of measurement
Variable <i>String</i>	Description of the time series being measured, e.g. Daily Streamflow

Table 4.1: The properties of a geospatial time series

DataTypeEnum	Instantaneous – <i>A condition at a given instant of time</i> Cumulative – <i>The accumulated value since the beginning of time</i> Incremental – <i>The difference in cumulative values at the beginning and end of a time interval</i> Average – <i>The average rate over a time interval, calculated as the incremental value divided by the duration of the data interval</i> Maximum – <i>The maximum value of a variable in a time interval</i> Minimum – <i>The minimum value of a variable in a time interval</i> Definitions from (Maidment 2002)
TSIntervalEnum	Second, Minute, Hour, Day, Week, Month, Year
UnitTypeEnum	Length, Area, Volume, Time, Mass, Energy, Concentration, MassPerArea, MassPerLength, MassFlowrate, MassAreaFlux, MassLineFlux, VolumeFlowrate, VolumeAreaFlux, VolumeLineFlux, EnergyPerVolume, EnergyPerArea, EnergyPerLength, Power, EnergyAreaFlux, EnergyLineFlux
StatisticEnum	Count, Minimum, Maximum, Median

Table 4.2 The enumerations within the geospatial time series object

Two of the most important properties added to the geospatial time series are Shape and TSUnitType. The Shape property is the geometry of the feature

associated to the time series. Obviously, if the shape of time series is simply a point, there is no need to have a shape property – a simply latitude and longitude would do. However, if the time series is related to a line or polygon geometry, for example if it is a flux of material over a line or area of space, it is beneficial to have the shape property to perform intersections between geospatial time series objects when considering the amount of mass or energy transferred between the features.

The shape property is an ESRI class (IGeometry). Using an ESRI class as the shape property allows ArcGIS geospatial functions to operate on the geospatial time series. For example, the IntersectionOp class within ArcObjects operates on the shape property of a geospatial time series object, just as it operates on the shape property of a feature. This means it is possible to utilize the classes behind ArcGIS software to perform hydrologic analysis in custom and creative ways – ways which ESRI may never develop themselves. The disadvantage is that ArcGIS is not open source, so one can not manipulate the internal algorithm of the intersection routine and one must have an ArcGIS license to use the tools.

The TSUnitType property is a coded value domain, or an enumeration data type variable, with a set number of dimensions. An XML document is used to find the correct dimension for a particular unit string (Figure 4.2). Thus, this TSUnitType property is automatically set from the Units property of the geospatial time series object. For example, if a geospatial time series object is used to store the daily streamflow recorded at a USGS gage, then its Units property will be the string 'cfs'. When a Geospatial Time Series object is

initialized and the Unit property is set to 'cfs', the code will then set the TSUnitType property by opening the XML file, looking for the string 'cfs', and then finding the parent node which tells the unit type: volume flowrate. Knowing the dimensions of a geospatial time series object is important when performing mathematical routines on the object, such as adding or subtracting two time series, accumulating a time series, or integrating a time series over space.

```

<?xml version="1.0" ?>
- <Conversions>
+ <Length>
- <Area>
  - <Square_centimeter>
    <abbrev>cm2</abbrev>
    <factor>0.0001</factor>
  </Square_centimeter>
  - <Square_meter>
    <abbrev>m2</abbrev>
    <factor>1</factor>
  </Square_meter>
  - <Hectare>
    <abbrev>ha</abbrev>
    <factor>10000</factor>
  </Hectare>
  + <Square_kilometer>
  + <Square_foot>
  + <Acre>
  + <Square_mile>
</Area>
+ <Volume>
+ <Time>
+ <VolumeFlowrate>
+ <VolumeAreaFlux>
+ <MassAreaFlux>
+ <EnergyAreaFlux>
+ <Energy_per_time>
+ <Energy_per_area>
+ <Energy>
+ <Mass_per_area>
</Conversions>

```

Figure 4.2 This XML file is used to set the TSUnitType property

The benefit of exposing the unit dimensions in an XML document is that it allows one to add to the library of acceptable units and have the class be aware of the new unit definitions without recompiling the project. For example, if one wishes to use the length dimension ‘furlong’, which is currently not in the list of

acceptable units, then he or she can add furlong to the list of length units and provide the conversion between furlong and meters (the standard length unit). The program will automatically be able to convert a time series with furlong units to any other length unit, even if the application was open when updating the XML file.

The disadvantage of this system for describing units and the conversions between them is primarily in the way composite unit types, i.e. units that are some combination of length, time, mass or energy units, are described. In the present system, each composite unit is treated just as a base unit, where a base unit is simply length, time, mass, or energy. For example, cubic feet per second is given an abbreviation and a conversion factor to the standard unit for volume flowrates: cubic meters per second. A better way to handle composite unit types is to describe the base units that make up that composite unit, i.e. cubic feet per second is a composite unit made up of the base units ft, ft, ft, and second. If the units string is restricted to a particular structure that can be easily parsed with code, then the code can translate the composite unit to its base dimensions and then calculate the correct conversion factor.

To make this alternative system more flexible in terms of the acceptable unit strings, one could maintain a complementary XML document of composite unit types and various abbreviations. This library could, for example, translate “cfs” to “cubic feet per second” prior to using the conversion routines to convert cfs to a different flow unit. Thus, the overall system would be robust in how unit conversions are applied to composite unit types, and flexible in the way units are

abbreviated within the database. It is advised that future revisions to the unit conversion routines and XML document produced through this research follow this new path for creating a robust unit conversion library for hydrology.

The TSIntUnit and TSIntLen are two properties which replace the Arc Hydro TSInterval property. The Arc Hydro TSInterval property is limited to 20 interval types and if the time series did not have one of these intervals, for example if the interval was 25 minutes, then the Arc Hydro data model has to be reconfigured for that particular application. By replacing the TSInterval property by both a TSUnit and TSLength property, one can represent any time series interval. The Unit specifies a dimension of time (e.g. second, minute, hour, day, week, month, year) and the Length an amount of that unit (i.e., if the interval is three hours, the TSUnit would be hour and the TSLength would be three). If an Arc Hydro database does not have these new fields within its TSType table, backwards compatibility is achieved by populating the TSIntervalUnit and TSIntervalLength from the TSInterval field (although the WriteToGDB method will not always be able to convert the new TSIntervalUnit and TSIntervalLength fields to the old TSInterval field prior to writing the time series to an old version of the TSType table).

The properties GDBPath and InGDB are necessary for reading and writing from an Arc Hydro geodatabase. The Description property provides a place to record how a new time series was generated. Finally, the UID is a unique identifier for a Geospatial Time Series object. It is a GUID (Global Unique Identifier) object which is a 32 digit alpha numeric string which almost certainly

will never be repeated. These properties are useful when many geospatial time series objects are in memory, and in particular when new geospatial time series objects are created by mathematical manipulation of other geospatial time series objects.

There are eleven methods defined on the Geospatial Time Series class (Table 4.3). They perform such actions as adding a time series to a chartspace (a collection of charts), changing the units of a time series (and likewise its TSValues), and rescaling a time series (as of now, only up scaling is supported, but downscaling through interpolation could also be added in future versions). There is also the ability to write a geospatial time series from memory to an Arc Hydro geodatabase.

AddToChart	Adds the time series to a chart within the chartspace.
AddToChartSpace	Adds the time series to a Microsoft Office Web Components (OWC10) ChartSpace object.
ChangeUnits	Creates a new time series object with new units
Clone	Clones the object with the exception of the TSUID.
GetTSSStatistic	Returns a statistic of the time series
RescaleTime	Produces a new time series by upscaling the original time series. Supported statistics are defined by the StatisticEnum (Table 4.2)
WriteToGDB	Writes the TS object to an Arc Hydro geodatabase

Table 4.3: The methods for a geospatial time series.

The majority of code in the geospatial time series class is devoted to four of the methods: AddToChart, ChangeUnits, RescaleTime, and WriteToGDB, thus these four methods are discussed in greater detail than the others. The AddToChart method requires the installation of Microsoft Office Web components version 10 which is part of Microsoft Office XP. If the user does not have this version of Office, but does have Microsoft Office installed, it is possible to install just the Office Web Components version 10 from Microsoft's website for no charge.

The AddToChart method takes a time series and adds a new series on the chart for that time series (Figure 4.3). Its companion method, AddToChartSpace, decides whether to add the time series to an existing chart (if the value axis units

match the time series units property) or to create a new chart within the chartspace for the time series object (Figure 4.4). The AddToChart method plots the time series as either a point or interval (i.e. a start and end point connected by a line) based on the DataType property. If the DataType property is instantaneous or cumulative, the time series object is plotted as points. If the DataType property is incremental, average, maximum, or minimum the series is plotted as an interval. To generate the intervals, the AddToChart method calls a function which creates interval TSDatetime and TSValue arrays from the original arrays. Thus, the number of values in each array is doubled minus one.

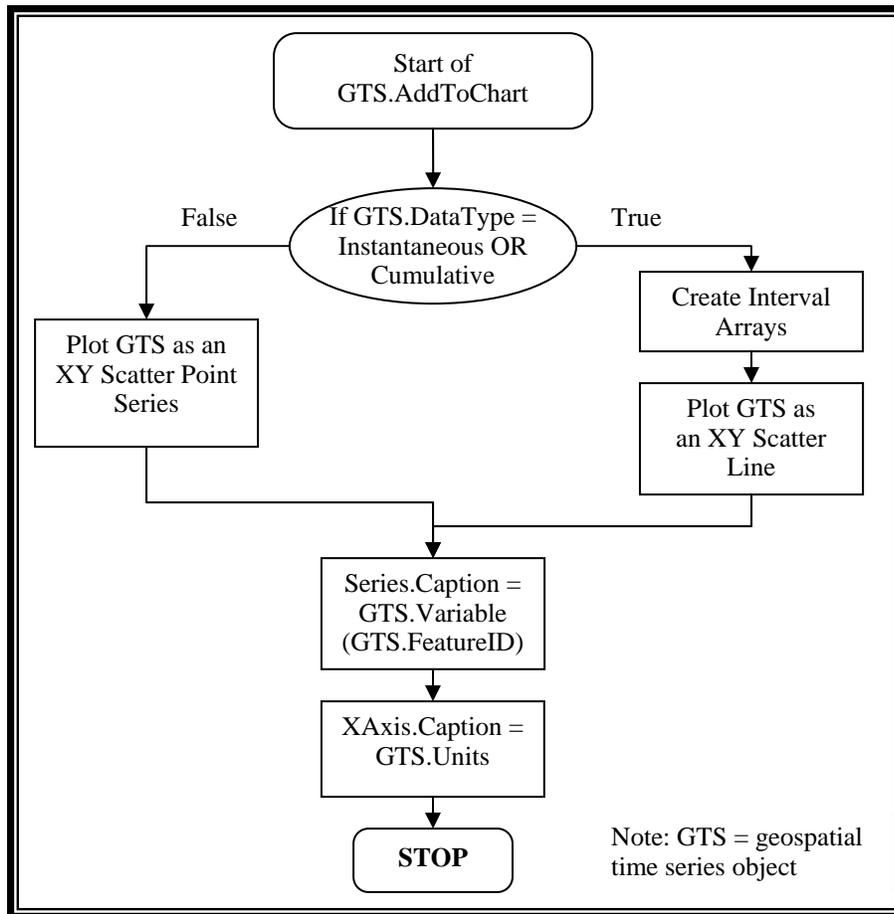


Figure 4.3: Algorithm for adding a geospatial time series object to a chart

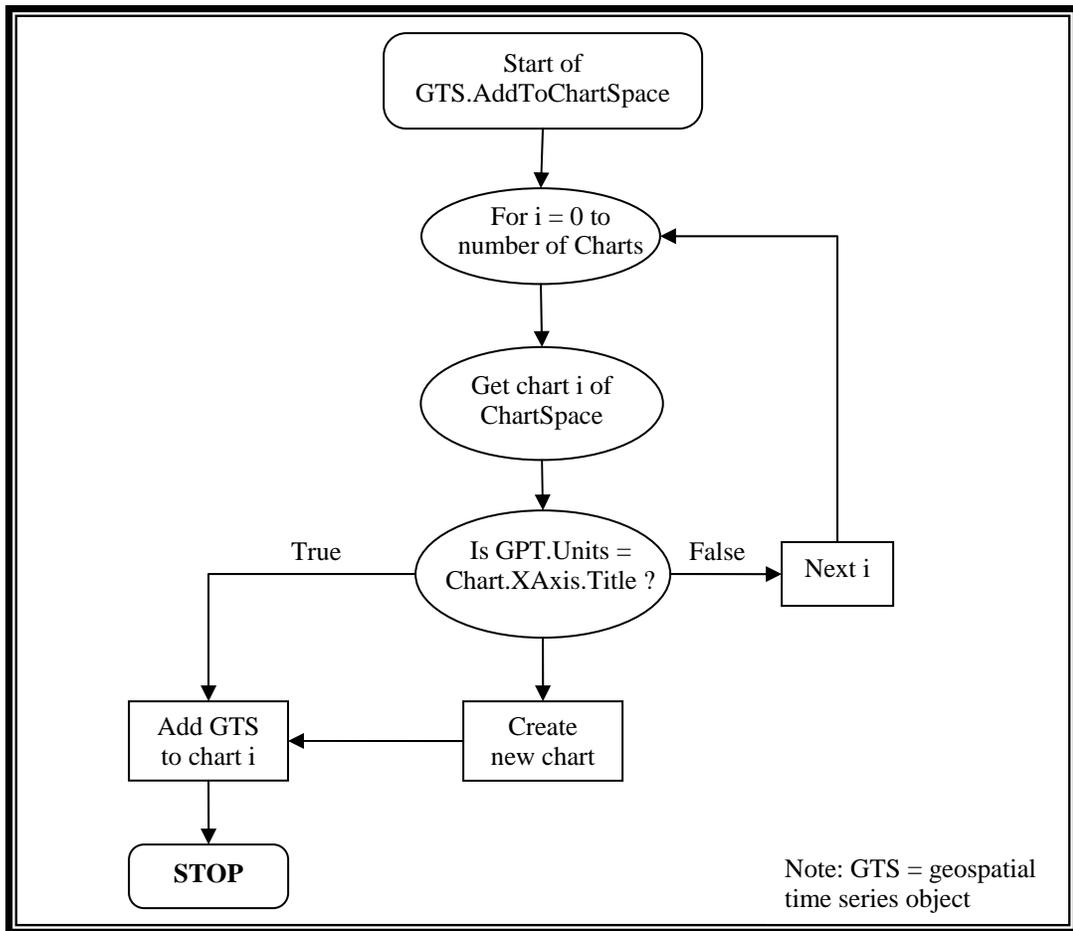


Figure 4.4: Algorithm for adding a geospatial time series object to a chart space

One challenge which arose was plotting a time series which represented intervals of time (for example, daily averaged streamflow). There is currently no chart type within the Microsoft Office charting component appropriate for plotting intervals given a start date and time period (as Arc Hydro does). Column charts provide one alternative, but columns are centered on a date/time and cannot be created by giving a start date/time and an interval. A second alternative, and the method used within the class, is to represent an interval as two points (start

and end date/time) connected by a straight line. This is not an ideal solution, and moving away from Microsoft Office Web Components to a charting control more designed for science applications is recommended for future versions of the class library.

The ChangeUnits method will create a new geospatial time series object with the updated time series values and the new units property (Figure 4.5). Changing units of a geospatial time series is not restricted to units of the same type – it is also possible to convert a geospatial time series between dimensions. For example, if a geospatial time series has a polygon shape and an areal flux unit type, then one can transform the units to a flow time series. The ChangeUnits method, in this case, will create a new geospatial time series which is a clone of the old TS, but with an updated TSValues property.

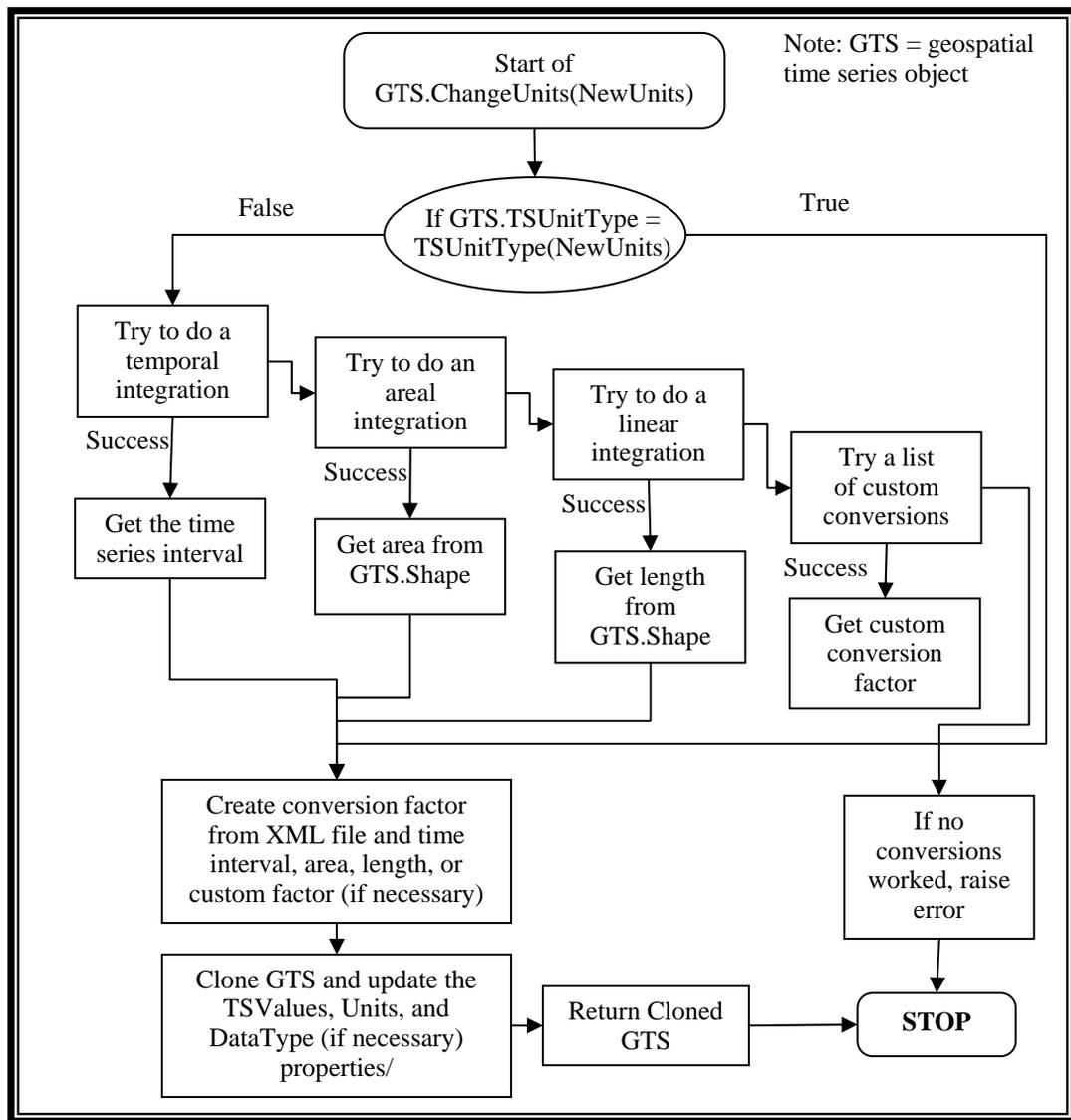


Figure 4.5: Algorithm for changing the units of a geospatial time series

It is unclear if a unit conversion should also update the shape property of a geospatial time series or if it should keep the shape as in the original geospatial time series object. To convert the shape provides consistency between geospatial and measurement dimensions (area fluxes are related to polygons, lines fluxes to

lines, and flows to points or volumes), and this simplifies mass and energy balancing in a geotemporal domain with heterogeneous measurement units. However, converting the shape introduces an additional level of complication and, perhaps, confusion. The ideal solution would be to allow future versions to provide the option of updating the shape property when calling the `ChangeUnits` method.

The `Rescale` method creates a new geospatial time series object with a different time interval than previously (Figure 4.6). As of now, only up-scaling time series is supported, but future develop of this class library will support downscaling time series by interpolation as well. This means that it is possible to take a daily time series and produce a monthly or yearly averaged time series by calling this `Rescale` method. Up-scaling of a time series into time intervals introduces uncertainty which should be stored in the geospatial time series. For example, the rescale method could produce some sort of probability density function for the interval of time and not a simply an averaged value.

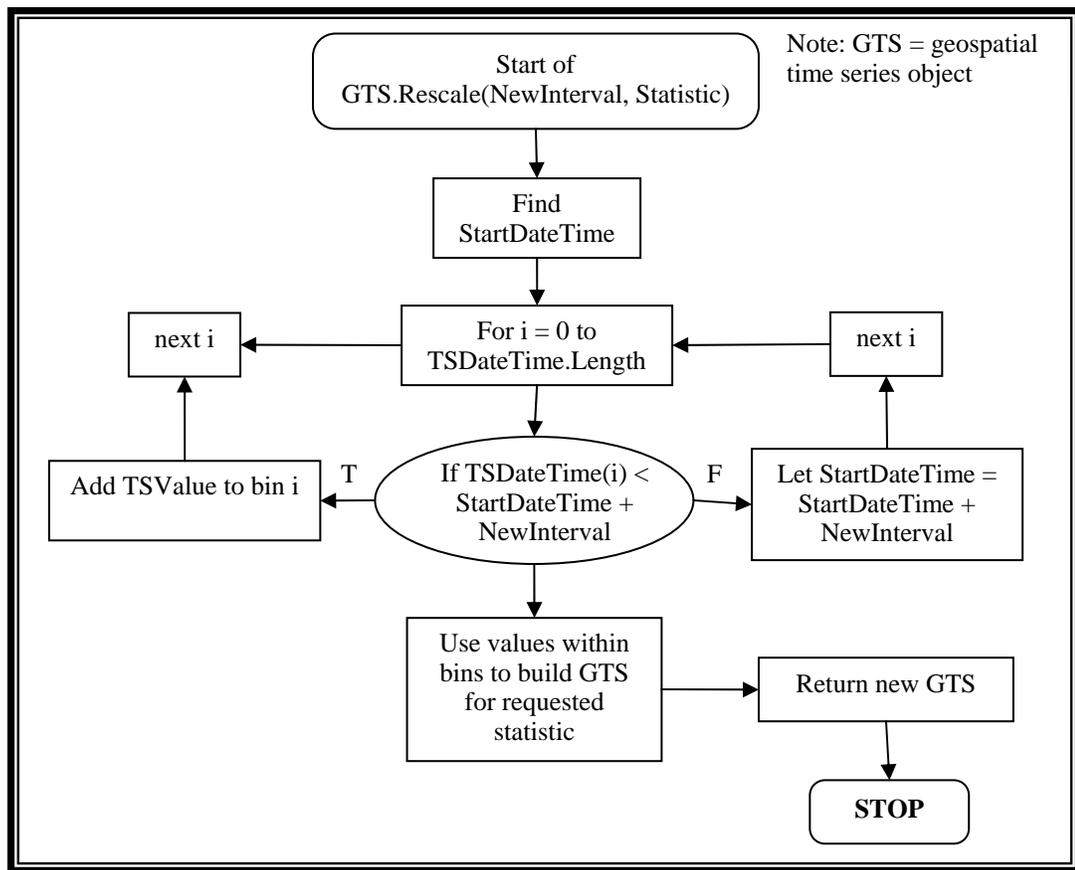


Figure 4.6: Algorithm for temporally rescaling a geospatial time series

Finally, the WriteToGDB method takes a geospatial time series object and writes its properties to an Arc Hydro geodatabase (Figure 4.7). The method checks the time series types in the TSType table to find the correct type that makes the geospatial time series properties. If one does not exist, the method creates a new TSType record from the properties of the geospatial time series object. The method also writes out the time series values and date-times to a TimeSeries table. Finally, if the user chooses to do so, the shape of the geospatial

time series can be used to create a new feature within the feature class related to the time series records.

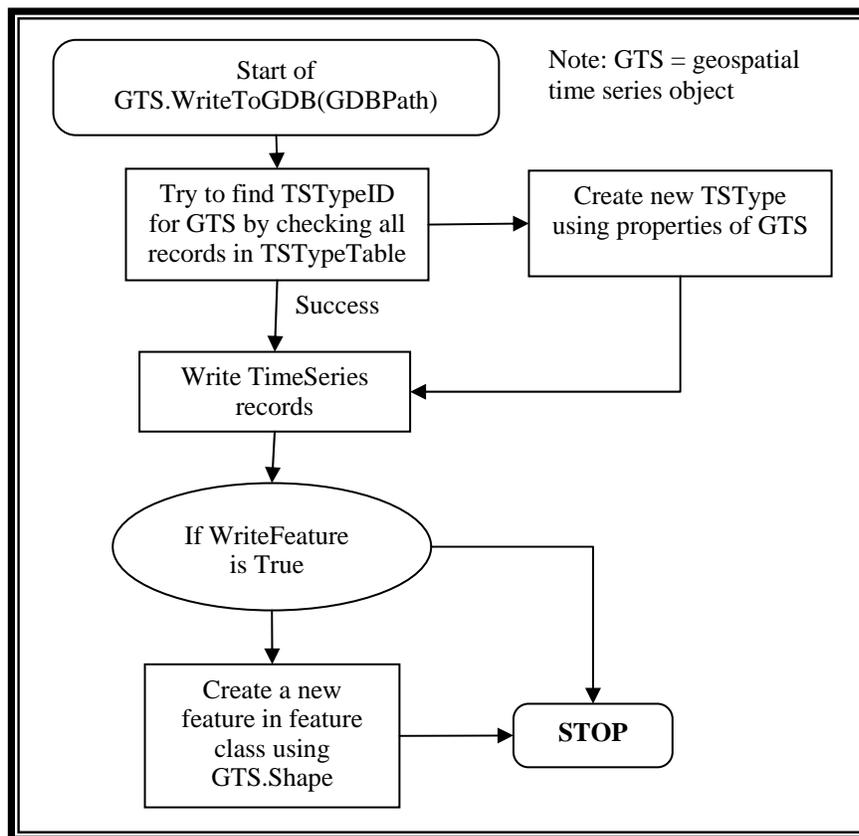


Figure 4.7: Algorithm for writing a geospatial time series to an Arc Hydro geodatabase

The geospatial time series, as of now, can be constructed from an Arc Hydro geodatabase or constructed as an empty object and each individual property assigned from a different data source. As an example of the second way of populating a geospatial time series object, the charting application originally designed to plot time series from a local Arc Hydro database was extended to

dynamically plot data from the National Water Information System (NWIS), Ameriflux, and EPA STORET databases by constructing a URL a particular station, type of time series, and date range, parsing the returned webpage, and populating the properties of an empty geospatial time series object from the parsed XML file. Once this object was populated, the AddtoChartSpace method was called to chart the time series. This example shows the benefit of object oriented programming in that, once the classes have been developed, it is much less time consuming to build and extend software for new applications.

4.1.2 Hydrologic Flux Coupler

The hydrologic flux coupler, or simply flux coupler, allows one to represent a hydrologic system as a hydrovolume of space with an associated collection of geospatial time series objects. Each geospatial time series has an associated direction which either makes the object a source or a sink of material to the hydrovolume system. Once a hydrovolume feature has been coupled to its associated geospatial time series exchanges, it is possible to derive new time series for the hydrovolume entity such as the rate of change in storage. Coupling of the geospatial time series to the hydrovolume entity means that means that the flux or flow of material occurs over the intersection between the geometry of each geospatial time series and the geometry of the hydrovolume feature. The Application Chapter provides an example of this concept.

The flux coupler is an analysis tool for the reorganization and summarization of observation and modeled data. It is designed to act as a generic geospatial object which can summarize data from disparate data sources. Thus, a

flux coupler object could be setup to calculate a mass balance where the exchanges of water come from a USGS stream gage, precipitation comes from NEXRAD, and evapotranspiration from a Flux Tower.

A flux coupler object is constructed from a coupling table. The coupling table links the hydrovolume feature with a collection of geospatial time series objects. All of the geospatial time series (meaning the features and their time series records) must be in the same geodatabase. The coupling table fields are FeatureID, SourceSinkID, TSTypeID, and Direction. FeatureID is the HydroID of the volume feature. SourceSinkID and TSTypeID together represent the geospatial time series object which is coupled to the hydrovolume feature. Direction points the geospatial time series either into or out of the hydrovolume feature. The Application Chapter provides an example of this concept.

The properties of a flux coupler object provides access to the geospatial time series coupled to that feature organized by their dimensions (Table 4.4). Geospatial time series with flow, area flux, or line flux measurement units can be coupled to a hydrovolume feature. If a coupled geospatial time series has area flux measurement units, then its series must have an area geometry. Likewise, if the coupled geospatial time series has line flux measurement units, then it also must have a line geometry. This allows the hydrovolume feature to convert all geospatial time series to flow units for the balancing algorithm.

FeatureID <i>Long</i>	Unique identifier of hydrovolume feature
GDBPath <i>String</i>	Gives the geodatabase path
AreaFluxes <i>Collection</i>	A collection of the related geospatial time series with area flux dimensions
Flows <i>Collection</i>	A collection of related geospatial time series with flow dimensions
LineFluxes <i>Collection</i>	A collection of the related geospatial time series with line flux dimensions

Table 4.4: The properties of a flux coupler class

The methods on a hydrovolume feature allow one to create new geospatial time series which describe net exchanges between the hydrovolume and its surrounding environment (Table 4.5). New geospatial time series can be the net of all flow, line flux, or area flux geospatial time series, or the change in storage for the hydrovolume feature. The calculation of net flow or fluxes is useful to understand the relative changes between vertical and horizontal exchanges of water, since vertical exchanges are often stored in area flux units and horizontal exchanges are stored with flow units. The change in storage method takes all coupled geospatial time series objects, converts them to flow units, and returns the sum of the converted geospatial time series.

GetNetFlow	Returns a geospatial time series object that is the summation of all coupled flow geospatial time series
GetNetAreaFlux	Returns a geospatial time series object that is the summation of all coupled area flux geospatial time series
GetNetLineFlux	Returns a geospatial time series object that is the summation of all coupled line flux geospatial time series
GetChangeInStorage	Returns a geospatial time series object that is the summation of all coupled geospatial time series

Table 4.5: The methods of a flux coupler class

The major routine within the flux coupler class deals with the summation of geospatial time series (Figure 4.8). The geospatial time series can be of many different types and this summation tool must be able to either convert the geospatial time series objects to a compatible type on-the-fly, or inform the user why it is impossible to perform the request action. Of course, the smarter the summation algorithm, that is the more conversions it is capable of performing to insure compatibility between geospatial time series, the more user-friendly the end application. However, one must exercise caution when providing the ability to automatically convert time series and insure that the internal conversion routines are documented and tool functionality can be controlled through the user interface to the application.

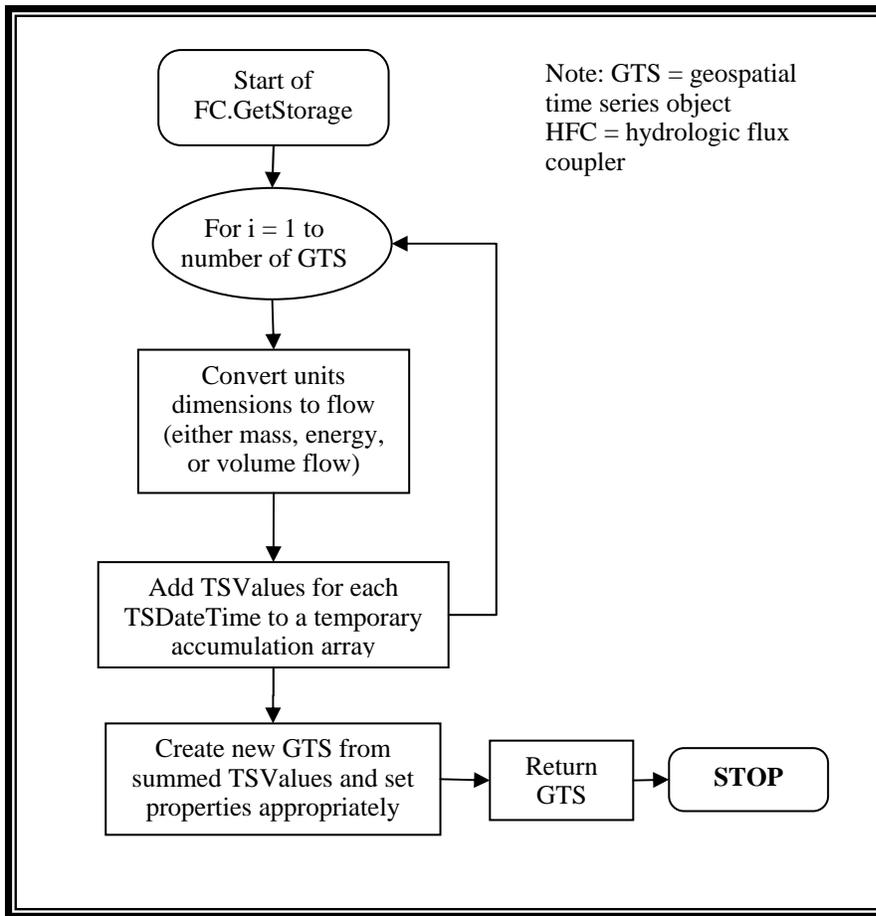


Figure 4.8: Algorithm for generating the change in storage geospatial time series for a flux coupler object

4.2 EXAMPLE OF USING THE HYDROOBJECTS

This section will provide a simple example of how one would use the hydrologic objects to program custom tools. The tools one may wish to create using the hydro objects can be anything from a generic hydrologic analysis program to a GIS-based data preprocessor for creating model input files. The hydrologic objects facilitate the process of writing such applications by providing

basic routines for reading, converting, and writing hydrologic data just as the ESRI object library facilitates the creation of custom geoprocessing tools by the manipulation of the Arc Objects library.

Suppose we have a hydrologic model and we need to have monthly streamflow to calibrate the model. The steps necessary to obtain this information are:

1. Query an Arc Hydro geodatabase for a stream discharge recorded at a given USGS station over a certain time period.
2. Upscale the data to monthly averaged values.
3. Write a monthly time series to format required by the hydrologic model.

The geospatial time series object can be used to quickly program an application which automates the process of gathering, converting, and outputting this data.

The first step is to construct a geospatial time series object from an Arc Hydro geodatabase. To do this, we must know the HydroID of the USGS station within the geodatabase. The HydroID is a unique identifier for the station feature within an Arc Hydro geodatabase. We must also know the TSTypeID for daily averaged stream discharge within the geodatabase. These two identifiers allow us to construct a geospatial time series for stream discharge at the given station with the following line of VB .Net code.

```
Dim GeoTS as New GeospatialTimeSeries(HydroID, TSTypeID, GeodatabasePath)
```

Next, we use the Rescale method of the geospatial time series class to transform the daily data into monthly data. This method creates an entirely new geospatial time series object with the same properties as the previous geospatial

time series, except for the `TSValue`, `TSDateTime`, `DataType` and `TSInterval` properties. These four properties will be updated by the routine from the previous time interval (e.g. one day) to the new interval (e.g. one month). The parameters of the `Rescale` method are the new time unit, new time length (in this case 1 month is the new temporal scale), and the request data type (e.g. average, minimum, maximum, or incremental sum).

Dim UpscaledGeoTS as GeospatialTimeSeries = GeoTS.Rescale(Month, 1, Average)

Now that we have a geospatial time series, the next step is to write the time series to an output file format for ingestion into a simulation model. The output format may be an ASCII file, a binary file, a database, and XML file, or any other file format. If one wishes, it is possible to extend this geospatial time series object by writing a method for outputting the time series to the specified file format. For major hydrologic models, like the HEC models, it would be worthwhile to write a method for writing the geospatial time series to a DSS file. If one does not wish to extend the geospatial time series object, another solution is to write a routine that creates the file and that accesses the properties of the geospatial time series object when necessary.

If one wishes to store this up-scaled time series to the geodatabase for future use, he or she can use the `WriteToGDB` method on the geospatial time series. As shown in the previous section, this routine will check with the `TSType` is already in the geodatabase by comparing the properties of the geospatial time series with the values in the `TSType` table. If the `TSType` is already in the geodatabase, it will take the `TSTypeID` of that `TSType`. Otherwise, the method

will create a new TSType within the geodatabase when outputting the time series. There is also the option for adding a feature to a feature class when output the geospatial time series to the geodatabase.

UpscaledGeoTs.WriteToGDB(GeoDBPath, TSTable, TSTypeTable, FeatureClass)

4.3 SOFTWARE DEVELOPMENT USING HYDROOBJECTS

This example provides a basic introduction to how one would create software that utilizes the hydro object classes. The objects facilitate software development because these basic hydrologic objects can be incorporated into any number of client applications. This ensures code reusability and enables one to build more complex hydrologic models because of the organization of the basic routines into logical classes.

The client applications which use HydroObjects are not limited to desktop applications; current research into web services further opens the door to using objects stored on a server in a server/client architecture. Four substantial benefits of web services compared to dynamic link libraries are that (1) rebuilds of the class library are automatically part of any client applications, (2) the server can make use of commercial software, like ArcGIS, to perform analysis, even if the user does not have a license of that software, (3) a web service and a client application do not have to be written in the same language because of the universal acceptance of web service protocol standards, and (4) the server can have more computational speed than a personal computer, thus allowing clients to conduct large-scale hydrologic processing utilizing the server's CPU. There are some technical complications which must be considered when implementing web

services, such as security, handling multiple users, and transferring data between clients and servers over the internet, but the potential warrants further investigation of this developing technology.

Provided in this dissertation are two examples of software which are built from the hydrologic objects: TSPlotter and Space-Time Toolbox. Both of these are extensions of the ArcGIS software system for handling dynamic hydrologic processing.

4.3.1 Time Series Plotter

The first software product developed through this research extends the ArcMap interface with the additional a toolbar and a dockable window (Figure 4.9). The toolbar provides the user the ability to query for time series related to features within the map. When the user sets these criteria and then clicks on a feature, the time series for that feature is plotted on the dockable window. The user has the option of selecting the underlying data source as a local Arc Hydro database, the National Water Information System (NWIS) database, or the Ameriflux database. NWIS is a U.S. Geological Survey maintained database with hydrologic time series like stream discharge, water quality, and groundwater level observations (<http://waterdata.usgs.gov/nwis>). Ameriflux is a network of micrometeorological towers that measure exchanges of CO₂, water, energy, and momentum between the land surface and atmosphere (<http://public.ornl.gov/ameriflux/>).

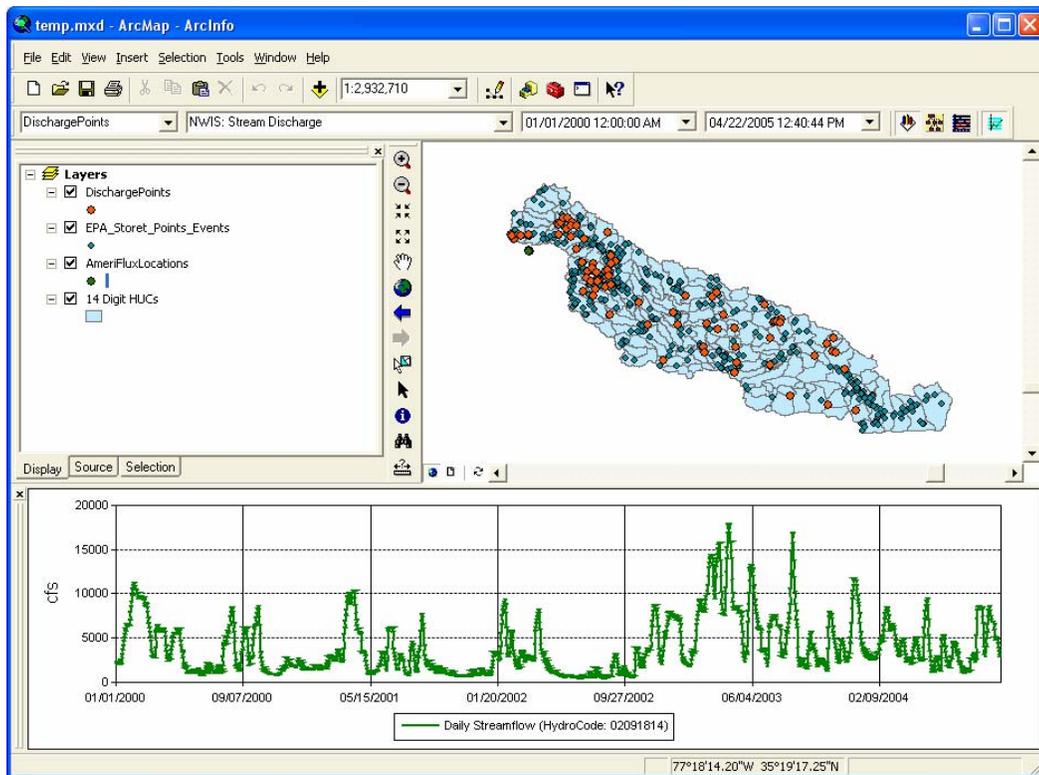


Figure 4.9: TSPlotter: an extension to Arc Map for plotting local and remote time series associated to geographic features.

When the data source is set to local Arc Hydro, the code builds a geospatial time series object for the selected feature and time series type from the underlying database. This populated geospatial time series object is then added to the chart space using the `AddToChartSpace` method. When the data source is set to a website, the code constructs a URL to query the remote database for the selected feature and time series type (in this case, the types are stream discharge, water quality, or groundwater levels). The web server returns a webpage with the requested data as defined by the URL. From this point, an empty geospatial time series object is created and the webpage is used to populate the relevant attributes

of the time series object. Once populated, the same AddToChartSpace method is performed to chart the time series.

Other capabilities provided with this plotting tool are the ability to manipulate the units of a time series, to integrate a time series over space or time, and the ability to export a chart from the chart space to EXCEL. The accumulation of time series over space or time uses the ChangeUnits method of the geospatial time series. Spatial integrations can be performed to convert a line or polygon geospatial time series object with measurement dimensions of “per unit area” to a point geospatial time series object with spatially integrated time series values.

The output to Excel routine works from the chart object itself and not a geospatial time series. With the exception of this routine, the majority of the work plotting the time series, querying the database for a time series, changing the units of a time series, etc is handled within the time series class code and not within the TSPlotter application. The TSPlotter application developer does not have to consider these basic tasks, just as when one works with the ArcObjects library, one does not code the procedure for how a feature layer is displayed on a map. The difference with hydrology classes, compared to the ArcObjects classes, is that the hydrology code should be open source so that one could manipulate the basic classes when building custom applications.

4.3.2 Space-Time Toolbox

The second example of an application that utilizes the hydrologic objects is a toolbox within ArcGIS for ingestion of data from national web servers,

temporal-geoprocessing, hydrologic analysis, and temporal scaling (Figure 4.10a). Because it is implemented within the geoprocessing framework (within ArcGIS 9.0), the tools can be accessed from the command line, the tool box, or linked with other ESRI tools in a model. This last option, which ESRI calls Model Builder, is particularly powerful because it allows one to create geospatial analysis models by combining individual tools in a workflow (Figure 4.10b). This toolbox is meant to extend the suite of already available tools to include functions particularly useful for spatiotemporal hydrologic analysis. Thus, using this toolbox, it is possible for one to create a tool that makes use of both ESRI tools and custom hydrologic tools in little time without writing code.

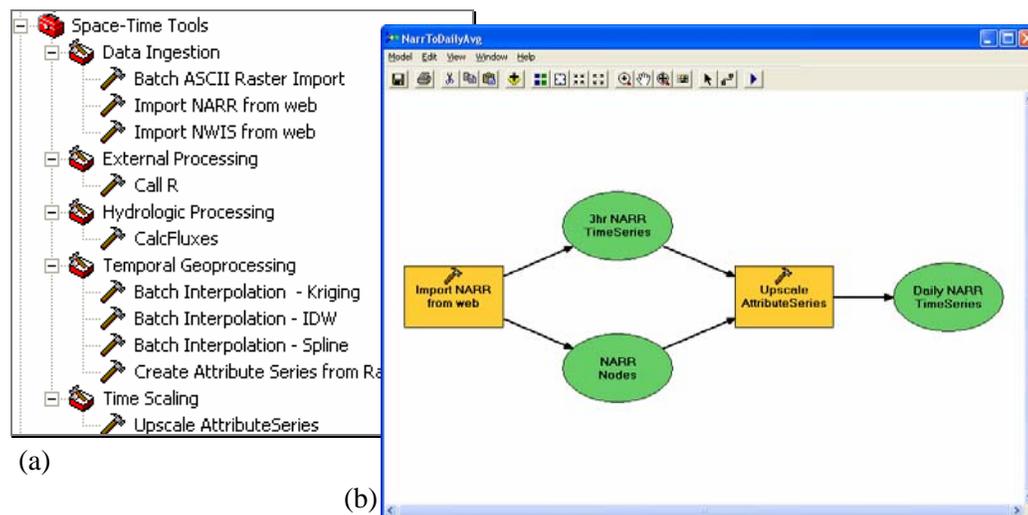


Figure 4.10: The Space-Time Toolbox is an ArcGIS Geoprocessing Toolbox which can be used as tools in Model Builder (shown on left)

The toolbox contains five toolsets: Data Ingestion, External Processing, Hydrologic Processing, Geotemporal-Processing, and Time Scaling (Figure 4.9a).

The toolsets are used to organize the individual tools into logical groupings. Within the Data Ingestion toolset are tools that allow one to query web servers for both daily stream discharge through the USGS's National Water Information System (NWIS) server and for North American Regional Reanalysis data through the NCDC's OPeNDAP server. The geospatial time series class is used to write the data returned from these servers to a feature class and time series table. There is also a tool for taking the NEXRAD ASCII grids, like those output by the NCDC NEXRAD Java Viewer, into a raster catalog indexed by TSDatetime and TSTypeID.

The ExternalProcessing toolset contains a tool for interacting with the statistical package R. The tool will open R, pass one or more commands to R, and then close R. The R commands are specified within the tool and work the same as if one is passing commands through R's user interface. Having the ability to call R from within ArcGIS provides a very powerful environment for coupled geospatial-statistical processing. It is developed for interfacing ArcGIS with existing water quality models within the Neuse River Basin. In a more general sense, however, it is an example of how software products can be coupled together by using objects from different software systems to create a new, custom tool for hydrologic analysis and modeling. This tool relies on an underlying library created by Thomas Baier for interacting with R as a COM object available on-line from <http://cran.r-project.org/doc/manuals/R-data.html#DCOM-interface>.

The Geotemporal-Processing toolset contains tools for performing batch interpolation and spatial scaling routines through time. The first process creates a

series of rasters indexed by time and time series type from a set of point geospatial time series objects. The second does the opposite; it takes a series of time indexed rasters and creates a set of geospatial time series objects. There are three spatial interpolation methods available: inverse distance weighting, spline, and kriging. These three options were chosen because they are supported in the base ESRI software. Thus, these tools make use of the same interpolation and ZonalStats routines within the ArcGIS software; they simply iterate these routines over time.

The Hydrologic Processing toolset contains a hydrologic flux coupler tool which can be used to perform mass or energy balances for a collection of hydrovolume features using the Coupling Table. This tool makes use of the hydrologic flux coupler class described in the previous sections. Because most of the mass balancing routines are encapsulated within this class, the tool itself can be created with much less code and with much less development time within this toolbox extension.

Finally, the Temporal Scaling toolset has a tool for upscaling a set of geospatial time series classes. The set of geospatial time series are created from a feature class and a time series type. For each feature within the feature class, a geospatial time series is constructed and that geospatial time series is upscaled using the Rescale method on the object. The rescaled geospatial time series is then written back to the geodatabase using the WriteToGDB method on the geospatial time series. Again, by using the hydrologic objects, the three steps in

the tool – reading the time series from Arc Hydro, up scaling the time series, and writing the time series to Arc Hydro – are accomplished with three lines of code.

4.4 ACCESSING REMOTE DATA

An exciting part of the applications built using the HydroObjects are their ability to download data directly from web servers like the National Water Information System (NWIS), Ameriflux, and the North American Regional Reanalysis (NARR) program. The web is an integral part of any information system architecture. It enables distributed data sources and processors to be fused together into an interoperable system. The result is the blurring between local and remote data where one can work with data saved to a disk as easily as remote data. It also allows greater access to large federal databases of geospatial, time series, model grids, and images. Users can request a particular portion of these large databases dynamically within analysis or model code. This section will provide more insight into how both geospatial, time series, model grids, and image data sources can be automatically ingested from the web into visualization or processing applications.

4.4.1 Geospatial Web Services

The Open Geospatial Consortium, Inc. (OGC) has established standards for sharing geospatial data through web services. The Web Feature Service (WFS), for example, is an interface for retrieving and manipulating geographic features encoded in Geographic Markup Language (GML) (<http://www.opengeospatial.org/specs/?page=specs>). When an organization creates a WFS, the geospatial data can be retrieved by client applications for

visualization and processing purposes. Client applications can be web pages or desktop applications. A web page can be designed to display data from distributed sources in a single map; a desktop application can be designed to perform geospatial analysis routines using the same remote data as input.

ESRI's ArcMap is an example of a WFS desktop client and the USGS EROS data center is an example of a WFS data provider. From ArcMap, one can view the available WFS layers provided by the EROS data center and add one or more of these to a map. Within the same map, one could add remote data from another WFS server or local data stored in an ArcGIS supported format (i.e. shapefile, geodatabase, coverage, etc.). Figure 4.11 shows ArcMap with Modis data, the NWIS streamflow stations for the United States, and subwatersheds for the Neuse River Basin. The first two layers are from EROS data center and the last is stored locally within a geodatabase.

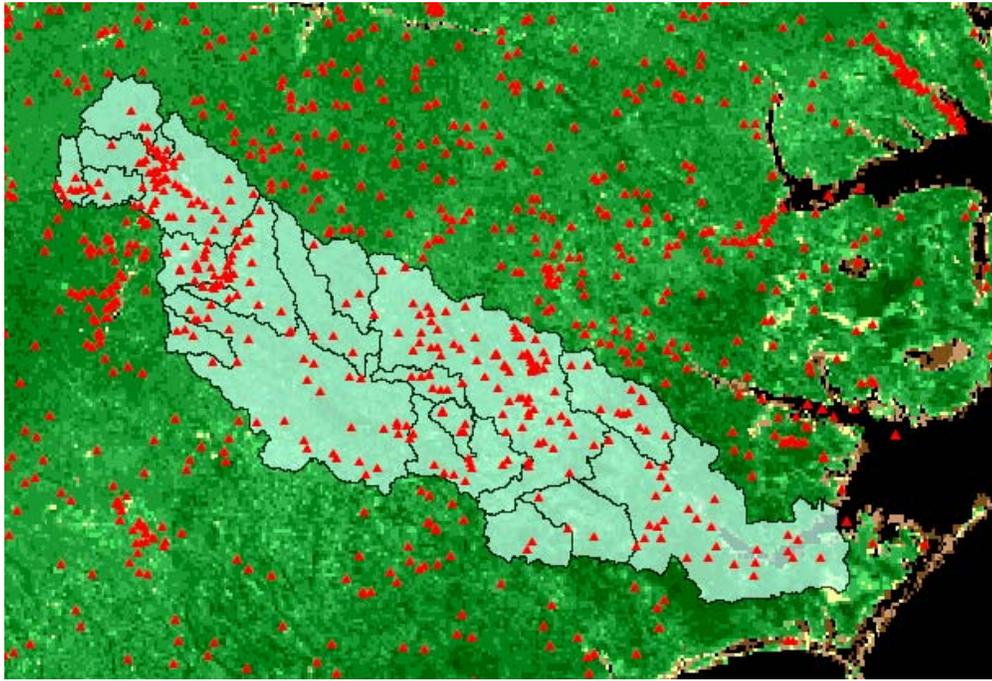


Figure 4.11: Modis data and NWIS Streamflow stations served by the USGS EROS data center as geospatial web services and viewed with ESRI's ArcMap. Neuse subwatersheds overlaying the remote data are stored locally in a geodatabase.

4.4.2 Hydrologic Time Series Web Services

The most common way of obtaining time series from a web server is to construct a URL with code that queries the remote database for a particular portion of data. The URLs of the National Water Information System (NWIS), and Ameriflux databases can be accessed through the internet using this URL manipulation method. The EPA Storet database does not use URL based queries for generating web pages, and therefore cannot be accessed in a manner similar to NWIS and Ameriflux. However, there are tools available for automatically querying information systems with an architecture similar to EPA Storet. The

San Diego Supercomputing Center developed a Java program that allows automated querying of the EPA Storet database through the internet. This Java tool is exposed as a web service, allowing it to be incorporated into client applications written in languages other than Java. Despite the ability to access EPA Storet data, it remains a less reliable solution than the NWIS data access through URL manipulation.

4.4.3 Model Grid Web Services

For grided data stored in binary, array-based data formats like netCDF, OPeNDAP (Open-source Project for a Network Data Access Protocol) provides a means for exposing the datasets to remote access through the internet (OPeNDAP was formally known as DODS and is often still referred to as DODS). OPeNDAP is software that, when loaded on a web server, allows clients to access data stored in a variety of scientific formats (netCDF, HDF, Grib, etc.) without knowledge of its actual physical storage format. A software application can be configured as an OPeNDAP client by using either the C++ or Java software libraries provided by the OPeNDAP developers.

A second method for accessing data through OPeNDAP is to create URLs that query particular files on the server for subsets of information, and return the subset of information as ASCII text. The advantage of this approach is that it allows applications not written in C++ or Java to be OPeDAP clients. The disadvantage is that the data must be transformed from binary to ASCII, slowing the overall data access process. This URL approach was used to access North American Regional Reanalysis (NARR) data in the Space-Time toolbox. There

are a number of other earth science datasets exposed for remote access using OPeNDAP (<http://www.opendap.org>, click on Data Sources), and additional tools could be added to the Space-Time toolbox to ingest these data into ArcGIS for processing.

4.4.4 Image Web Series

The Microsoft Terra Server (<http://terraserver.microsoft.com>) is an example of a relational-database and web services combination that allows access to Digital Orthophoto Quadrangles (DOQs) and Digital Raster Graphics (DRGs) images for the entire U.S. (Figure 4.12). Described as one of the largest databases in the world, the TerraServer stores images as tiles in a SQL Server. Each image tile is accompanied by a metadata file that describes that image. A client application accesses images from the SQL Server database using web services that return the metadata for a tile, or the actual tile image itself, at a given latitude, longitude point. The access to image titles is sufficient fast to allow client applications to create maps which allow users to pan throughout the United States. For each pan, the appropriate tiles are requested through the web service, and the added to the map.

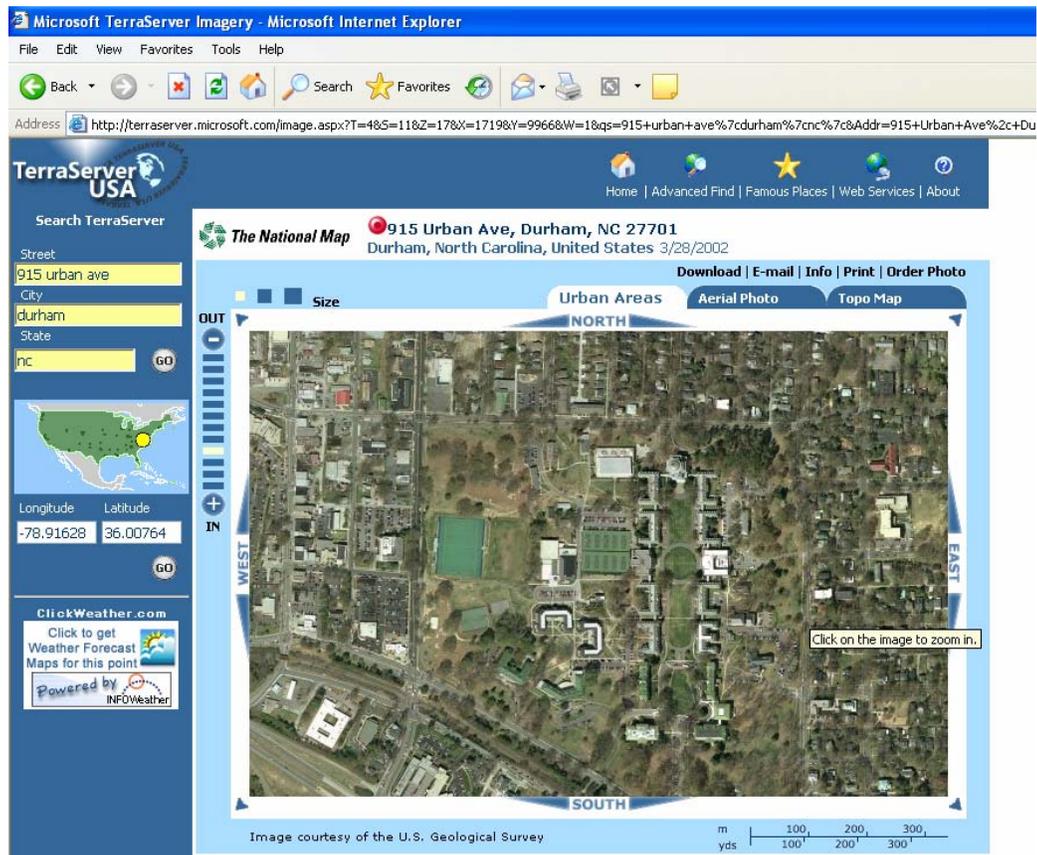


Figure 4.12: Duke University's East Campus, which is in the Neuse River Basin, viewed using the TerraServer. The underlying relational database stores image data for the entire United States and allows client applications to access the images through web services.

Hydrology is interested in many image-based file structures such as digital elevation models (DEMs), remote sensing images, and NEXRAD. The TerraServer provides a model for both storing these information sources and for exposing the images to end clients with web services. Because TerraServer is a Microsoft research venture, its hardware and software specifications are clearly

documented, making it possible to reproduce the TerraServer infrastructure for storing hydrology datasets.

The most appropriate organization for implement a image data distribution infrastructure similar to TerraServer is the USGS EROS data center. In fact, EROS already provides access to seamless elevation and hydrography data for the entire World through the Seamless Server (<http://seamless.usgs.gov/>). Seamless Sever is exposed to clients as a web application where user's select data for downloading from a map-interface. EROS also allows users to query the elevation and land cover databases for an point within the U.S.. A useful addition to the EROS web services would be to provide DEM or land use data as tiles through web services, much like Terra Server does already for other images.

4.4.5 Web Services –Centric Architecture for Hydrologic Information

This section shows that it is possible to implement web services that allow user access to national-scale geospatial, time series, model grid, and image data. The differences in how these files are stored on the server can be hidden from the client. Furthermore, some basic hydrologic data conversion routines can be implemented on the server-side, allowing the client to ingest the data that has been preprocessed remotely for ingestion into client-side model. Examples of basic hydrologic data routines would be spatial and temporal interpolation and integration – e.g. providing the user monthly averaged stream flow although the database contains daily averaged values.

One primary advantage of this system is that it allows individual researchers to utilized federal data resources without having to maintain the

datasets locally. The researchers can then concentration on building and maintaining local data sources that complement and extend the national data sources. The key to providing interoperability between the national and local data resources will be metadata that positions all data values to particular continuous or discrete locations in a geotemporal framework.

4.5 SUMMARY

This chapter has presented geospatial time series and hydrologic flux coupler which are two classes within the HydroObjects library. Using these classes as a foundation, two applications are developed to extend the ArcGIS system for handling geotemporal processing, analysis, and visualization. The first is a time series charting application called TSPlotter. TSPlotter is capable of charting both locally stored time series and time series directly from national data servers, and has basic unit and dimension conversion routines allowing one to quickly compare time series even if originally in different units. The second application is the Space-Time Toolbox which extends ArcGIS's geoprocessing environment with a collection of tools for hydrologic analysis and processing. The next chapter provides an example of how one might use these two applications to study a watershed system.

Chapter 5: Application

This section presents two custom applications built from the hydrologic classes that facilitate the visualization and processing of hydrologic time series from local and remote data sources. The first is a toolbox for ArcGIS with the ability to ingest data from federal web servers, rescale data in space and time, and perform water budget calculations for hydrovolume units. The second is a time series charting tool for ArcMap capable of charting information from a local database as well as directly from federal government web servers. An example of these two applications will demonstrate the ability to (1) obtain data directly from the National Water Information System (NWIS) and the North American Regional Reanalysis (NARR) program through OPeNDAP and (2) manipulate the spatial and temporal scales of the data for model preprocessing purposes.

The guiding scientific question for this application of the tools is posed in the Neuse Report commissioned by CUAHSI (Reckhow, 2004):

Is the water budget of the Neuse watershed and/or its sub-watersheds changing, and if so, where (the Piedmont? Coastal Plain? sub-areas of these zones? everywhere?) and by how much? (Pg 18)

Quantifying the water budget for different sub-watersheds within the Neuse Basin is important for understanding whether the present demands on water resources are being met by the supply of water via precipitation,

subsurface discharge, and upstream inflow. If water is being used at a faster rate than it is being supply by these inflows, then action is needed to return the system to a balance. In many parts of the country, particularly the American West, a significant amount of financial resources is spent attempting to ensure that water supply meets water demands.

The goal is not to fully answer this question; to do so would be a thesis within itself. Instead, the objective is to demonstrate how the tools developed through this research assist, in a more general sense, both in the formulation and testing of hypotheses regarding the storage of water within regions the Neuse Basin. Bringing together the remote data hydrologic data sources into a common geotemporal framework in which one is able couple discrete features by their exchanges of mass and energy provides the necessary infrastructure for conducting hydrologic analysis.

5.1 DEFINING THE HYDROVOLUMES

Given this overall goal, the first step in performing a water balance on any hydrologic system is to define its boundary. The boundary is the control surface through which the system exchanges mass and energy with its surrounding environment. Accounting for all exchanges through the control surface, along with internal gains or loses, allows one to estimate the internal storage or loss of material through time. The hydrovolume class described in the previous section is conceptually based on a hydrologic system and allows one to create an object with associated gain and loss time series. A hydrovolume object can be defined at any spatial scale; the entire Neuse River Basin can be a hydrovolume (Figure

5.1a) or a particular river reach channel within the basin can be a hydrovolume (Figure 5.1b).

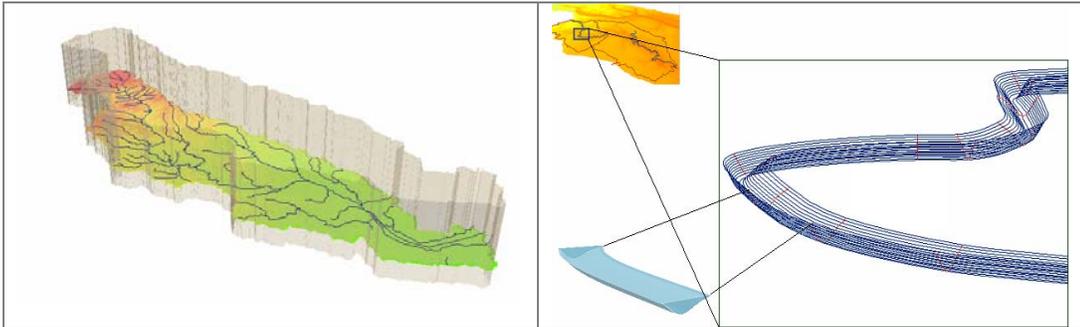


Figure 5.1: The Neuse River Basin and a river reach within the Neuse as three dimensional hydrovolumes

For this example, the Neuse River Basin was subdivided into twenty watersheds by using terrain processing techniques to derive the drainage area for the selected set of USGS discharge stations (Figure 5.2). Each of these subwatersheds is a hydrovolume with exchanges of water occurring at the gages between them. Each gage is both an inflow to one catchment and an outflow to another catchment. In addition to the horizontal exchanges of water between the catchments, there are also vertical exchanges of water between the catchment, the atmosphere, and the subsurface environment.

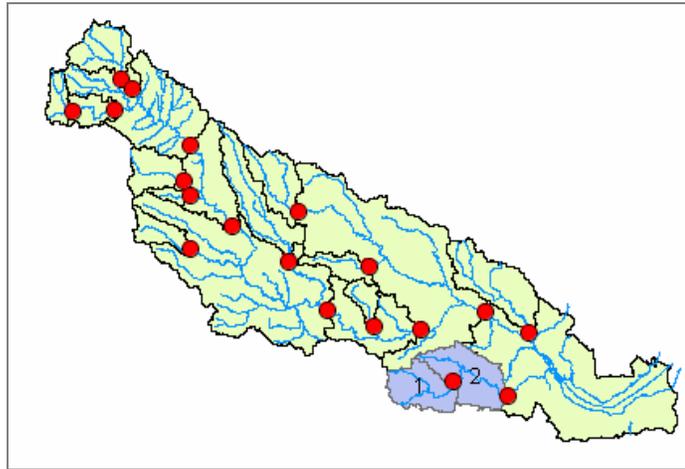


Figure 5.2: The Neuse basin divided into watersheds.

To further focus the effort, we will concentrate on the two subwatersheds highlighted in Figure 5.2 and shown with more detail in Figure 5.3. Subwatershed 1 represents the area draining to the USGS gage 02092500 on the Trent River near Trenton, North Carolina and subwatershed 2 the area draining to the USGS gage 02092554 downstream on the Trent River at Pollocksville, North Carolina. Although these watersheds are two dimensional in the geodatabase, they could be stored as three dimension volumes as shown in Figure 5.3b. This representation would be useful for determining the flux of water entering or exiting the atmosphere directly above each watershed as well as the groundwater exchanges through watershed boundary layer projected down into the subsurface. In this case, since all fluxes are perpendicular to the watershed surface, a two dimensional representation of the watersheds as hydrovolume is acceptable.

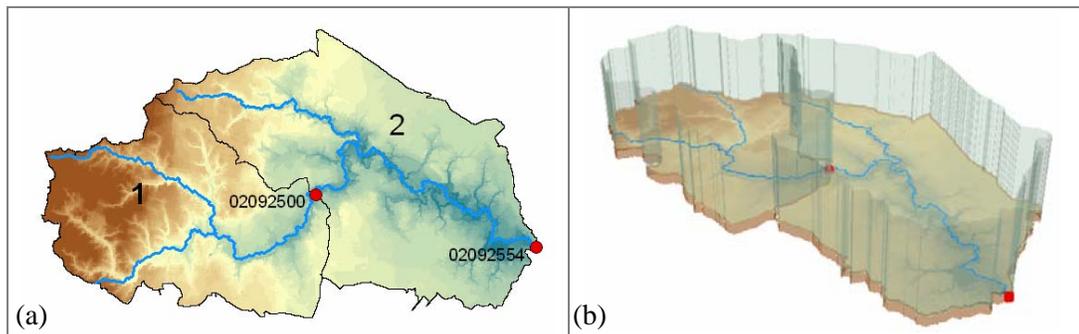


Figure 5.3: Two sub-watersheds of the Neuse River Basin.

The three dimensional image of the watersheds system (Figure 5.3b) provides a more concrete illustration of a watershed as a control volume. A control volume stores water when the inputs into the system exceed the outputs. Although a simple concept, it is difficult to observe or predict these exchanges, especially the spatially continuous fluxes such as precipitation, evaporation, and recharge. Remote sensing technologies which use ground-based radars or satellite-based sensors are producing estimates of many hydrologic fluxes at high spatial and temporal resolutions which will ultimately improve estimates of water budgets. Likewise, climate and weather models provide estimates of the complete water and energy cycles on a continental-scale and provide a valid, self-consistent data source of information when high-resolution information is unavailable.

The general algorithm for automating the water balance procedure for a collection of watersheds stored within a geodatabase is as follows:

1. identify the exchange of material occurring over the boundaries as well as internal sources or sinks of material for each watershed,

2. convert all exchanges to consistent dimensions and units using spatial and temporal interpolation and integration, and
3. sum all converted exchanges to predict the change in storage through time for each watershed.

This algorithm is generic and could apply for both mass and energy balances. Thus, one geodatabase can contain the information to perform mass and energy balances for a number of variables. Each variable would have a different coupling table which describes the exchanges of material for each hydrovolume.

A catchment water balance typically includes exchanges such as streamflow, evapotranspiration, precipitation, and subsurface infiltration (Figure 5.4). In this example, the USGS National Water Information System will provide the streamflow exchanges and the North American Regional Reanalysis (NARR) program will be used as an estimate of evaporation, evaporation, and infiltration. The Space-Time toolbox is used to automatically ingest data from NWIS and NARR, rescale the NARR data in time and space, and, finally, perform a daily water balance for a month to estimate the storage of water within each and every subwatershed of the Neuse Basin during this period of time.

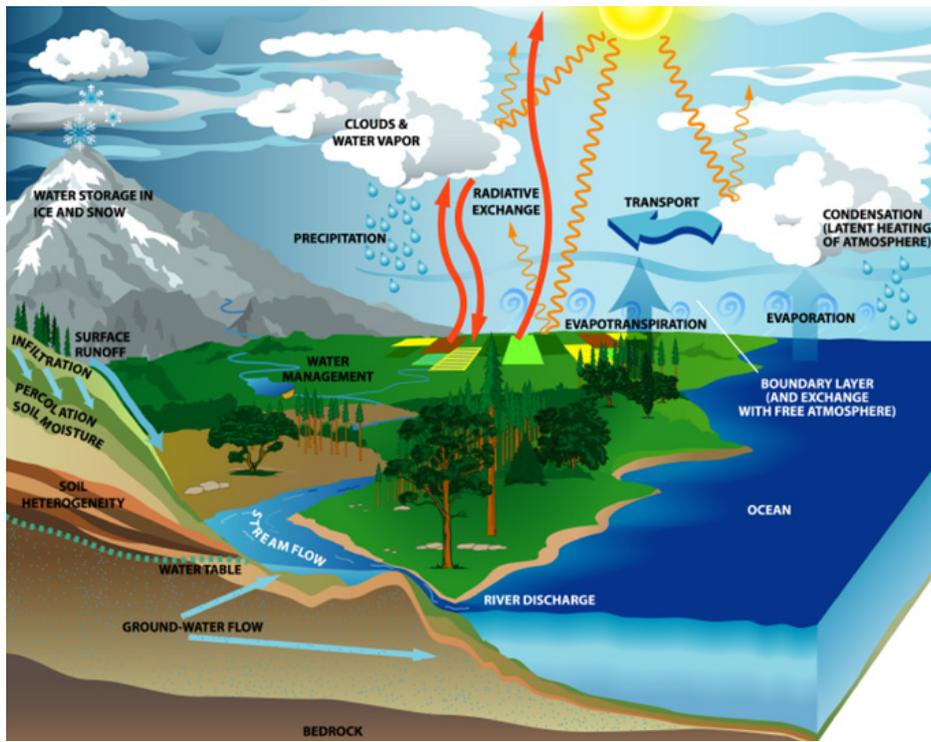


Figure 5.4: Conceptualization of the water cycle (Source: <http://www.usgcrp.gov/usgcrp/images/ocp2003/ocpfy2003-fig5-1.htm>)

5.2 DATA PREPREATION

The data preparation steps are:

1. download data from NWIS for the USGS stations and time series of interest,
2. download data from NARR for the geospatial region and time period of interest,
3. rescale the NARR data in time from 3hr accumulations to daily accumulations, and

4. rescale the NARR data in space from grid node points to watershed polygons.

These steps will be accomplished using tools within the Space-Time toolbox and these individual tools will be linked into a single model which automates the data preparation processes.

5.2.1 Reading the NWIS and OPeNDAP Servers

The first tool will ingest the time series from federal datasets stored at the United States Geological Survey (USGS) and the National Climate Data Center (NCDC) for the study region and time period. The USGS maintains the National Water Information System (NWIS) which houses streamflow, groundwater, and water quality samples. The NCDC maintains a variety of weather and atmospheric datasets including both historical and real-time observations and model output. The modeling program we will use is the North American Regional Reanalysis (NARR). This program uses current weather model technology along with historical observations to predict atmospheric conditions for 1979-2003.

The streamflow data are stored in a relational database, and is averaged on a daily time scale. The NARR data, on the other hand, are stored an array-based file format similar to netCDF called GRIB and are served over the web using OPeNDAP (<http://opendap.org/>). The temporal resolution is three hours and the spatial resolution is 32-km. Thus, before performing the water balance, these datasets must be transformed in space and time to estimate the exchanges for each watershed on the same temporal scale.

Both datasets are exposed over the Internet and subsets of the data bases can be accessed by constructing a URL that queries the underlying information. Visual Basic .Net, C# .Net, Java, and Python all allow one to read websites in a manner similar to reading a local text file (i.e. constructing a string that represents the location of the file – the URL – and parsing the returned webpage text. Microsoft's .Net was used to code the Space-Time toolbox and it includes a series of classes which make the process of requesting and reading web pages straightforward. Thus, while the underlying data structure of each of these data sets is quite different, both can be queried by the same technique: (1) build a URL for a station or region, a variable, and a time interval and (2) parse the returned webpage to extract the time series properties.

The Space-Time toolbox includes a tool for ingesting stream discharge from NWIS into Arc Hydro time series format (Figure 5.5). It also includes a tool for ingesting NARR data from OPeNDAP into Arc Hydro time series (Figure 5.6). Using these two tools allows us to populate our model from national datasets “on-the fly”. Both of these tools can deliver the station or model grid node as a feature and load that feature into a feature class of the geodatabase.

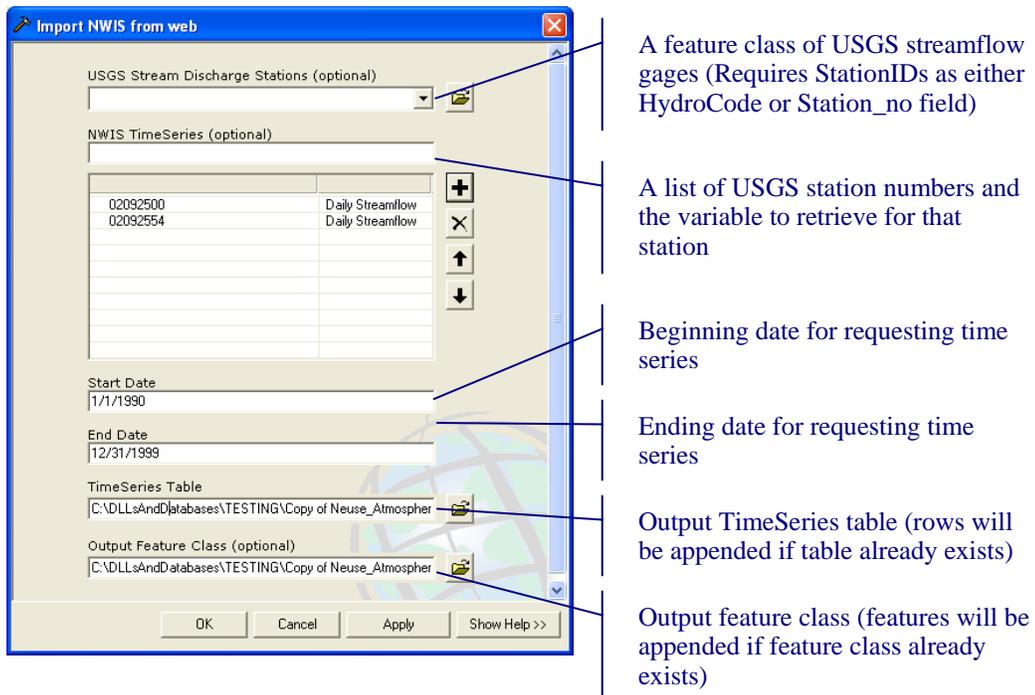


Figure 5.5: Interface for NWIS reader tool

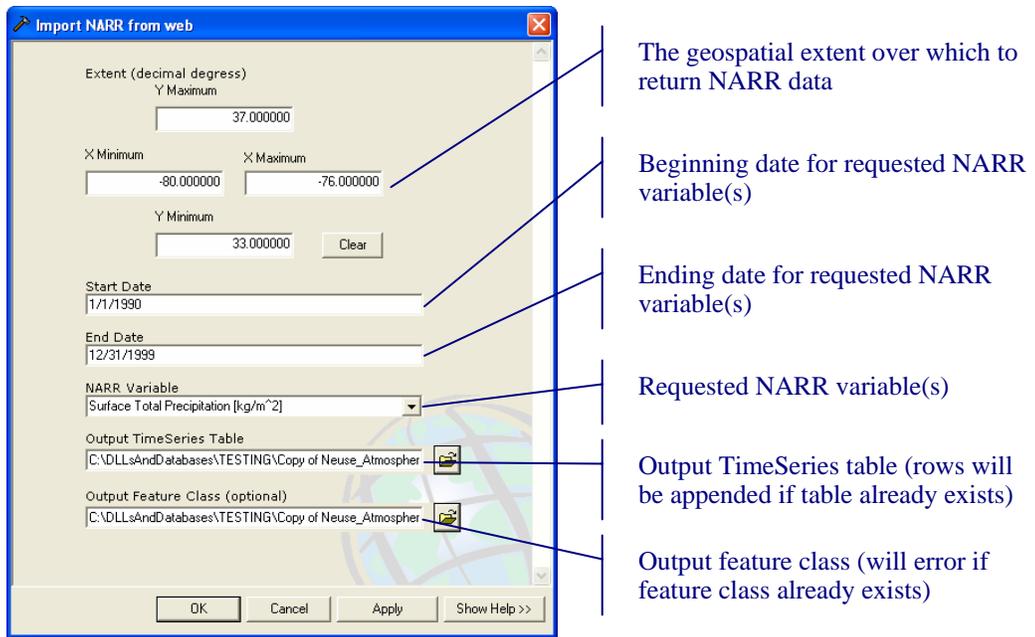


Figure 5.6: Interface for NARR reader tool

5.2.2 Temporally Rescaling the NARR Data

Once the data has been downloaded from these two servers, the next step is to rescale the NARR data from a 3 hour to daily time interval to be consistent with the stream discharge data. To do this, we will use the ASUpscale tool (Figure 5.7). AS is short for Attribute Series which we defined earlier as a feature class where every feature has an associated time series. This tool will build a geospatial time series object for each feature in the feature class, upscale each geospatial time series object, and, finally, write each geospatial time series object to the Arc Hydro geodatabase.

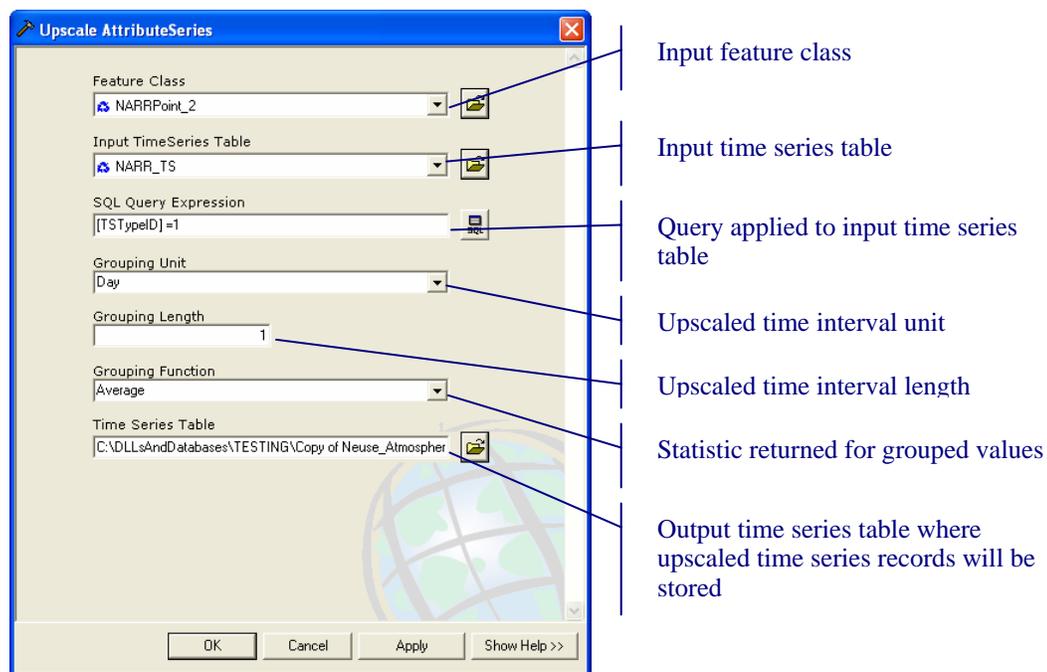


Figure 5.7: Interface for Up-scale Attribute Series Tool

5.2.3 Geospatially Rescaling the NARR Data

After rescaling the NARR in time, the next step is to geospatially rescale the NARR data from grid node points to the watersheds. This is a two step process: (1) attribute series to raster series (Figure 5.8) and (2) raster series to attribute series (Figure 5.9). The average of the raster pixels which fall over a particular watershed feature is taken as the flux value for that watershed feature for that time step. Going through rasters for scaling from points to watersheds is not absolutely necessary, but having this process as two distinct tools allows more flexibility in how one generates time series for the watershed features. For example, NEXRAD data is delivered as a series of rasters from the NCDC Java Viewer, requiring only the second tool and not the first.

If one wishes to extend this tool ingest data from the NEXRAD Java Viewer, then one would first use the Viewer to export NEXRAD data into a ASCII grid format. Under the present architecture of the Viewer, this must be done ahead of time; it is not possible to dynamically assess the information over the web. Once these ASCII grids are stored locally, the Space-Time toolbox provides a tool which will ingest these grids into a raster series catalog. The raster series catalog is an ESRI Raster Catalog class with the fields TSDatetime and TSTypeID. Then, using this raster series catalog, a second tool in the Space-Time toolbox will summarize the NEXRAD grids over the watershed polygons. Finally, instead of coupling rainfall from NARR (as will be shown in the following section), the coupling table can be altered to couple rainfall from NEXRAD averaged over the watershed entities.

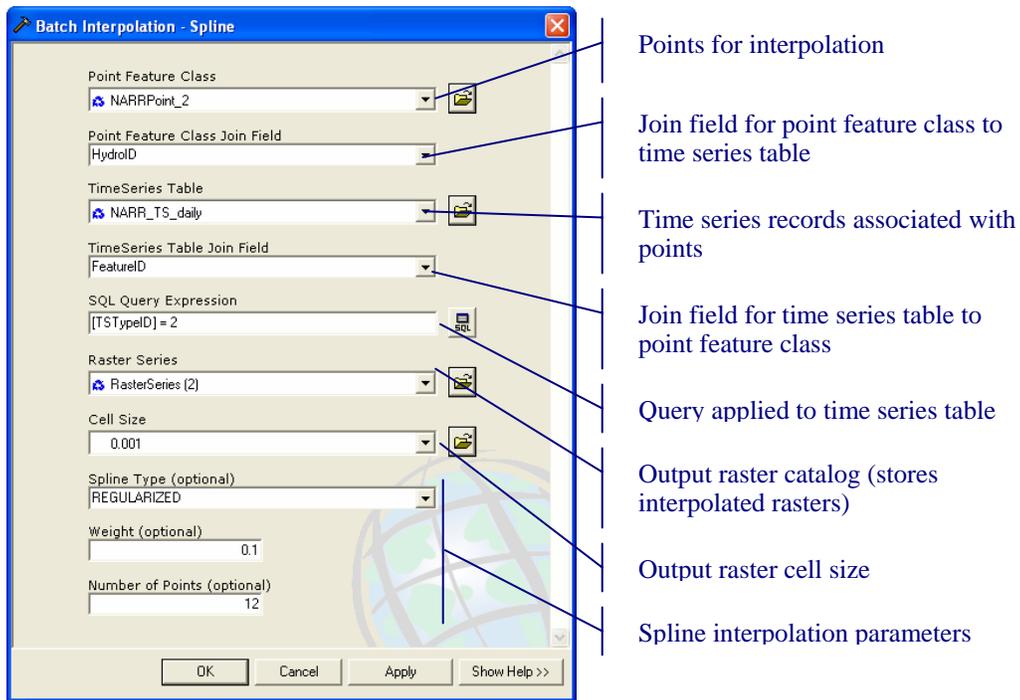


Figure 5.8: Interface for Batch Interpolation Tool

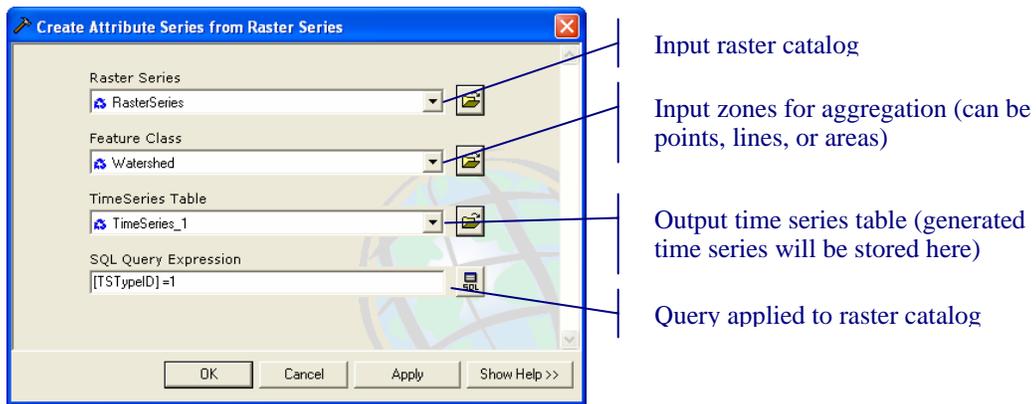


Figure 5.9: Interface for Batch Zonal Statistics Tool

5.3 PERFORMING A WATER BALANCE

Now that the data was been retrieved from the web servers and scaled in space and time to estimate the exchanges for each watershed feature, the final step is to perform a water balance. To do this, it is necessary to couple these exchanges to hydrovolume features by building a coupling table. The table allows one to associate any number of sources or sinks of material for a feature. The actual time series values do not need to be directly related to that feature (although they can be, as is the case in the example below); they can be related to a second feature, but coupled to the control volume feature. For example, although streamflow is related to a monitoring point feature, it can be coupled to a watershed feature as an exchange between that watershed and its surrounding environment. Thus, the coupling table provides hydrologic connectivity between features and allows one to estimate the movement and material between discrete entities within the watershed environment.

5.3.1 Coupling Geospatial Time Series to Hydrovolumes

Coupling establishes the connectivity between hydrovolume features and exchanges represented by geospatial time series. Each hydrovolume is treated as a separate discrete space entity with its own fluxes and flows, linked to the hydrovolume using a coupling table. Figure 5.10 shows the coupling table for hydrovolumes 9614 and 9623. Each of these hydrovolume features is coupled to vertical fluxes and horizontal flows. The hydrovolume class has the ability to automatically convert the flux time series to flow dimensions in the water

balancing procedure by using the intersection of area of the watershed and the area of the flux time series as the conversion factor.

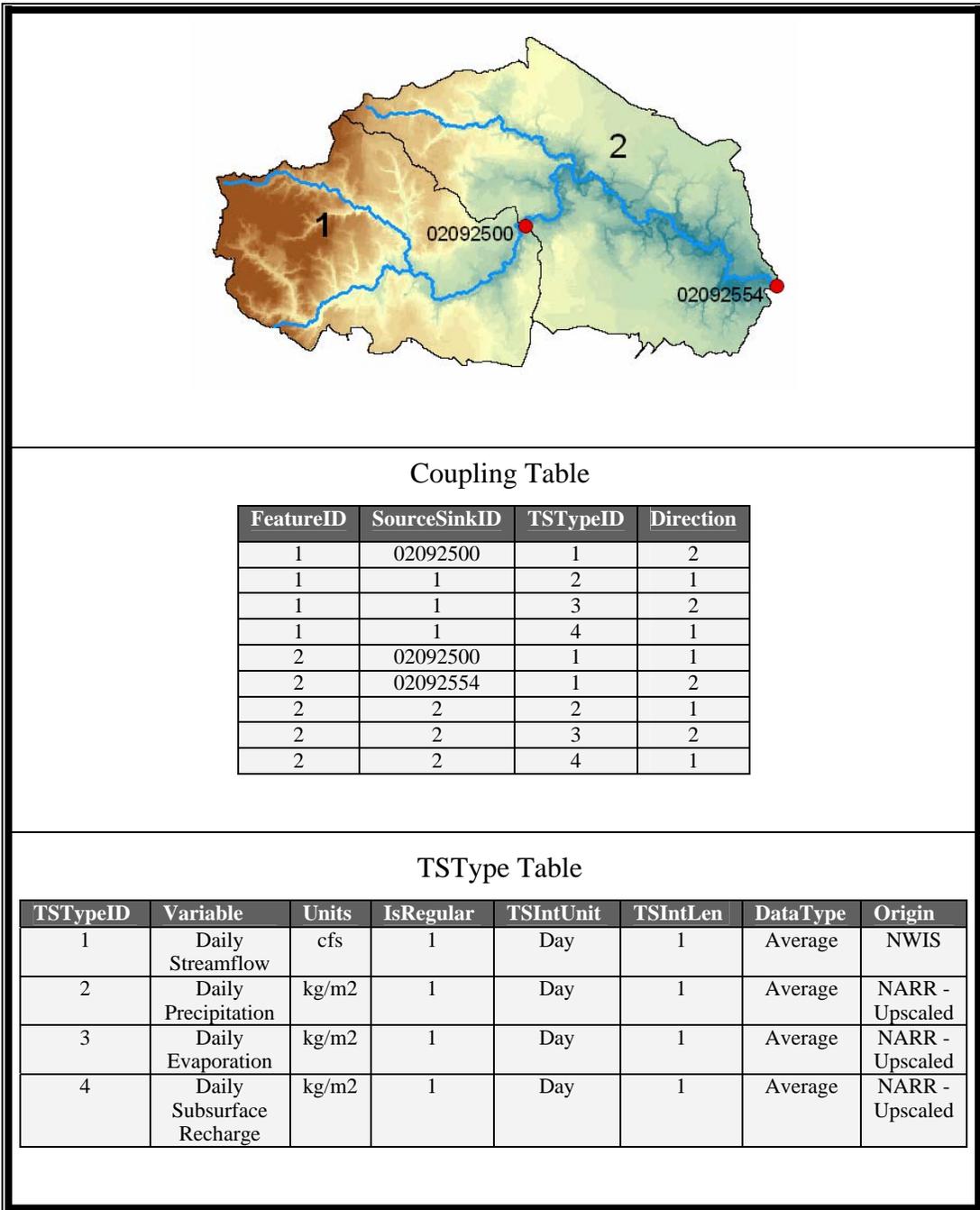


Figure 5.10: The hydrologic flux coupling table and the TSType table for two watersheds within the Neuse Rive Basin

Figure 5.11 describes the coupling table in terms of the direct meaning of each field and a useful way of thinking of groups of the fields in terms of the hydrovolume and geospatial time series classes. The FeatureID field of the coupling table is the HydroID of the hydrovolume feature; the SourceSinkID is the HydroID of the feature with the associated exchange time series; and the TSTypeID describes the time series exchange. Together these two fields represent a geospatial time series. The direction field specifies whether the exchange is either into or out of the control volume. Thus, these three fields, SourceSinkID, TSTypeID, and Direction, together represent a vector time series of an exchange occurring through the control surface of the hydrovolume feature.

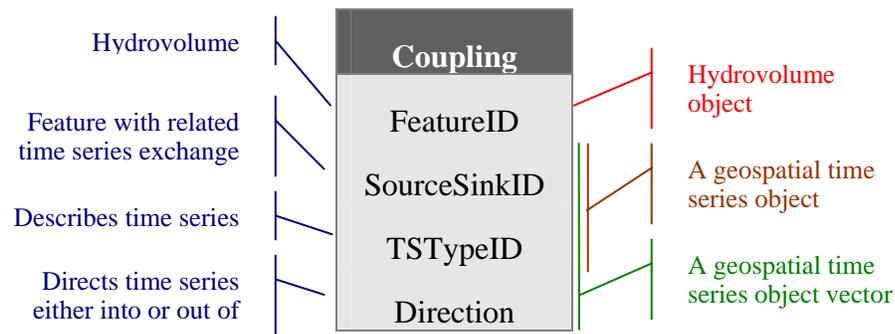


Figure 5.11: The Coupling table describes the relationship between a hydrovolume feature and its coupled exchanges of material represent by a geospatial time series vector.

5.3.2 The Hydrovolume Tool

After temporally and spatially rescaling the NARR fluxes and coupling the features, the final step is to execute the hydrovolume tool that calculates the change in storage for each catchment feature based on its coupled geospatial time

series (Figure 5.12). The change in storage for each watershed feature is written to the geodatabase upon completion.

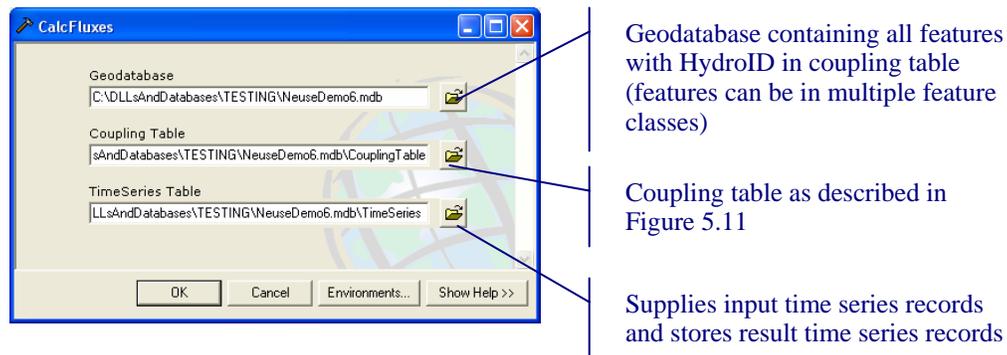


Figure 5.12: Interface for flux coupler tool

5.4 AUTOMATING THE PROCESS USING ARCGIS MODEL BUILDER

As mentioned at the beginning of this chapter, one distinct advantage of using the ArcGIS geoprocessing environment for hydrologic tool develop is the ability to link individual processes into a workflow where the output from one tool becomes the input for a second tool. Therefore, all of the steps discussed in the data preprocessing section can be linked into a single model that downloads data from federal data sources, rescales the information both temporally and spatially, and performs a water balance for twenty watersheds within the Neuse (Figure 5.13).

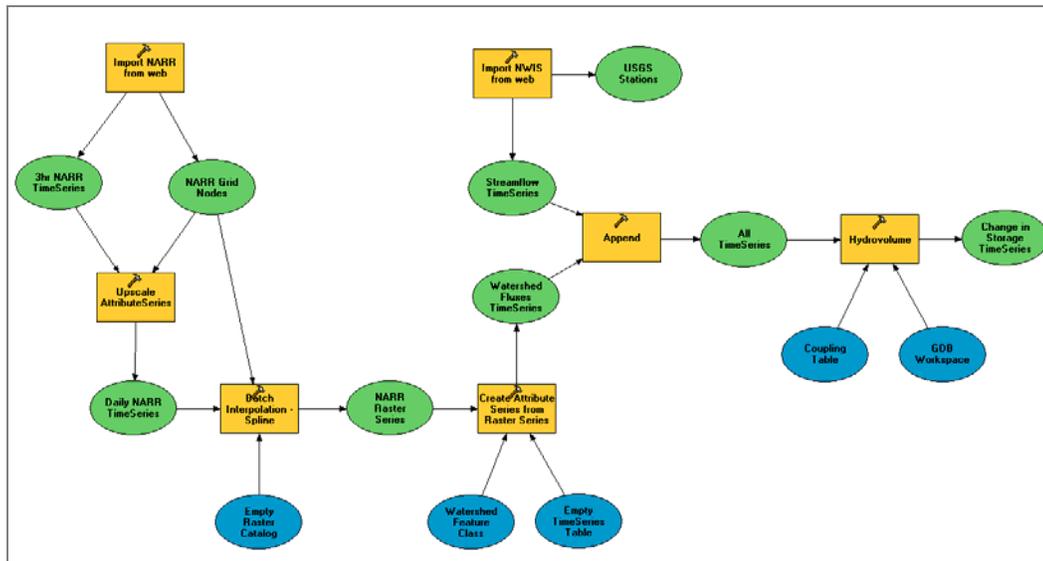


Figure 5.13: An ArcGIS Model Builder model for performing a water budget analysis on the Neuse River Basin including automatic ingestion of web data, spatial and temporal rescaling, and coupling exchanges to watershed features for estimation of water storage through time.

Because the time series data is downloaded automatically from remote data servers, the only inputs for this model are the hydrovolume features, the coupling table, and the time interval for analysis. A second advantage is that, because these national-scale data servers, it is possible to move this analysis technique to any location within the United States by simply providing a new watershed feature class and coupling table. This example demonstrates that it is possible to use of large federated datasets as if they were local sources of information. Because these data servers are optimized for quick searches on data, it is possible to drive hydrologic models with data coming directly from federal databases without having to manually query the data source through a website.

5.5 VIEWING HYDROLOGIC DATA IN SPACE AND TIME

In some cases, what is most needed by the researcher is simply the ability to view hydrologic observations and model output in geospatial and temporal context to address questions such as:

1. what was measured when?
2. is the data useful for my model?
3. how does my model output correspond with observations?
4. how does the streamflow at location x related to the watershed properties of that location?

In such a case, the most useful means of examining the system is to have access to remote databases and the ability to overlay these remote data sources along with locally stored information within a common geotemporal framework. Not only is this a useful setup for visualizing the complete system, but also for locating data required to parameterize, calibrate, or validate a model. Once the necessary data is located, it can either be downloaded to the user's machine or, if possible, the visualization tool should provide a URL to link to the remote information.

5.5.1 Introduction to TSPlotter: An ArcGIS Extension for Viewing Local and Remote Time Series Related to Geospatial Features.

For this reason, a second ArcGIS extension named TSPlotter was created for the visualization of remote and local time series related to geospatial features. The current version of the extension is able to read local Arc Hydro geodatabases as well as remote data sources including the National Water Information System and Ameriflux and work is underway to incorporate OPeNDAP data sources, such as the North American Regional Reanalysis dataset, and water quality data stored

by the Environmental Protection Agency in STORET. Many federal data sources are exposed over the web and can be automatically harvested through URL manipulation, so this list will likely continue to grow as the tool evolves.

TSPlotter provides a user-friendly and intuitive means for viewing time series related to a spatial feature by simply clicking on that feature within the map (Figure 5.14). The underlying database or web server is then queried for the time series related to the selected feature. The advantage of this system opposed to other time series plotting systems is that it allows one to point-and-click map features instead of the more typical approach of provide a station number or name for plotting a feature's time series.

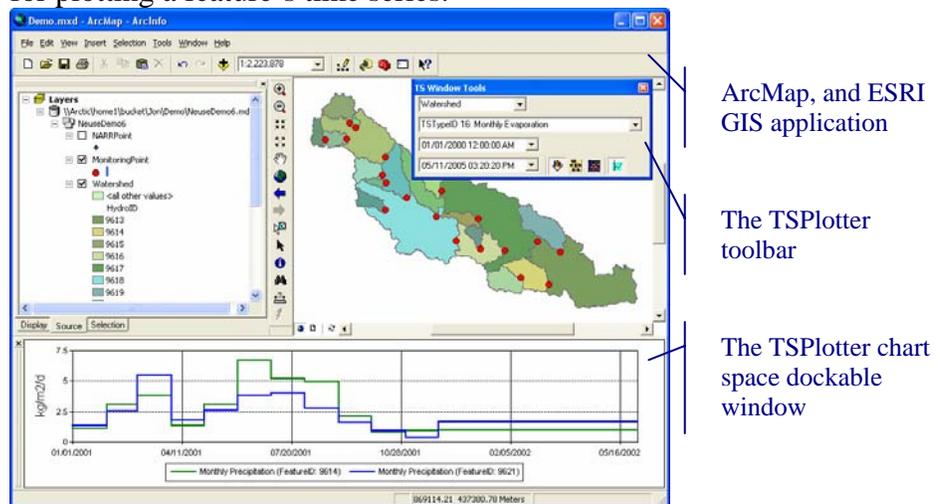


Figure 5.14: Visualizing monthly evaporation time series related to watershed features using the TSPlotter within ArcGIS.

5.5.2 The Graphical User Interface

The toolbar includes drop-down boxes for selecting a layer within the map, a variable, and a time interval (Figure 5.15). The layer drop-down box is

populated from the list of layers within the map's table of context. It is automatically updated when the user adds and removes layers. The variable drop-down box is set depending on the data source selected in the tool options (this will be discussed later). For Arc Hydro data, it lists only the time series types available for the selected layer. For remote data sources, it lists the available variables for that particular source of information (daily streamflow for NWIS, evaporation for Ameriflux, etc.).

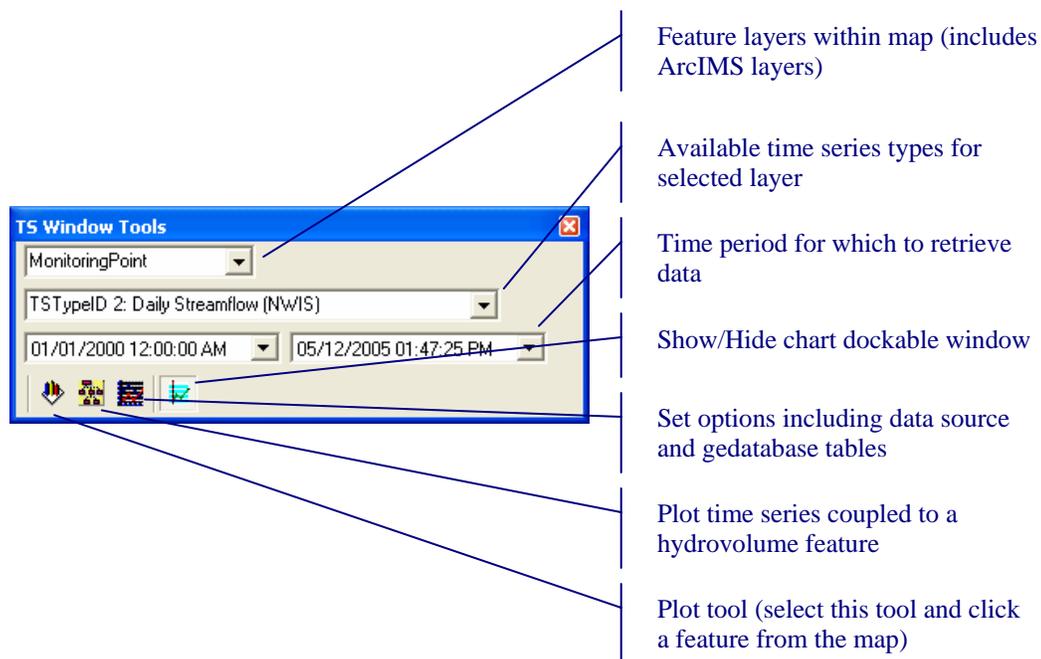


Figure 5.15: Explanation of the TSPlotter toolbar

One exception to this rule is water quality data from NWIS. Water quality data presents the challenge that, while there is a long list of available variables over the

entire database, the list of available variables is highly dependent on which site is requested. One site may have twenty variables and another site have none. For this reason, when the user selects water quality data from NWIS, the variable drop-down box simply states “NWIS: Water Quality” and not a list of available variables. Then, when the user selects a USGS station from the map, the NWIS server is queried for all water quality parameters available for that station. This list is then presented to the user and the user may select one or more parameters to add to the plot (Figure 5.16).

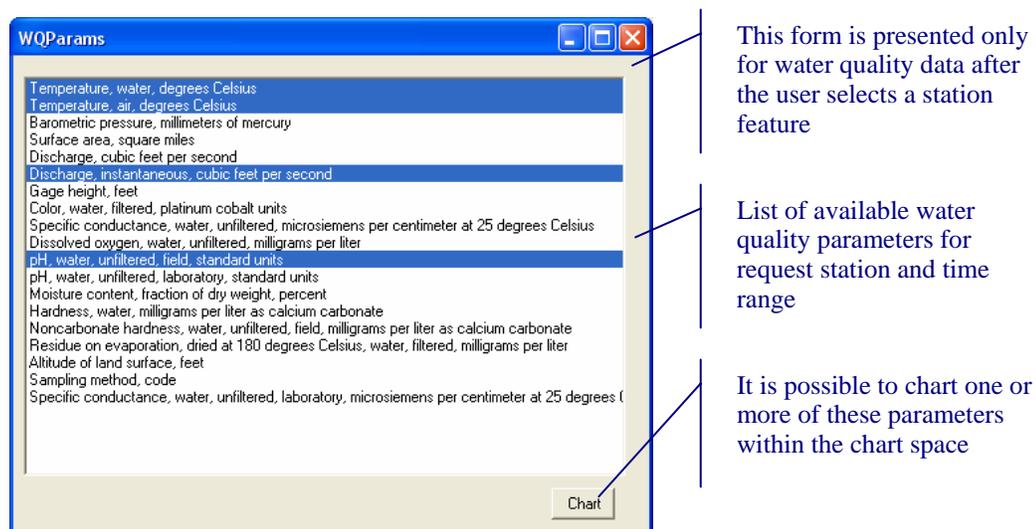


Figure 5.16: The available water quality samples for USGS station 02091000, Nahunta Swamp near Shine, NC. For water quality data, an intermediate set is required because the list of available variables varies widely between sites.

5.5.3 Changing Between Data Sources

As was mentioned in the previous section, the user can change the data source from a local Arc Hydro geodatabase to a remote data source under the

options window of the tool (Figure 5.17). When one uses TSPlotter to visualize time series directly from remote data servers, the application will not automatically save the time series to disk as the Space-Time toolbox will. The time series is kept in memory as a geospatial time series object and is plotted directly from this object. Thus, one can use the TSPlotter to visualize hydrologic observations directly from the web within a desktop application instead of within a web browser.

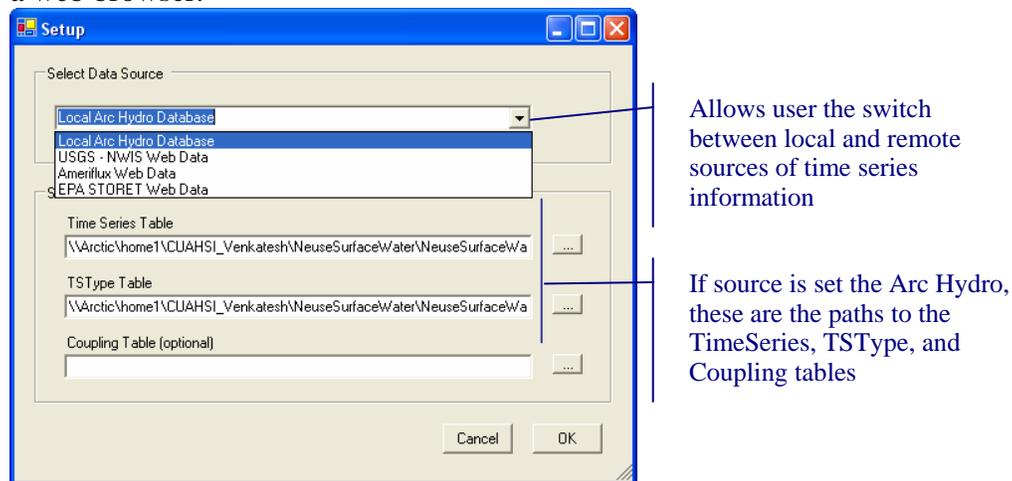


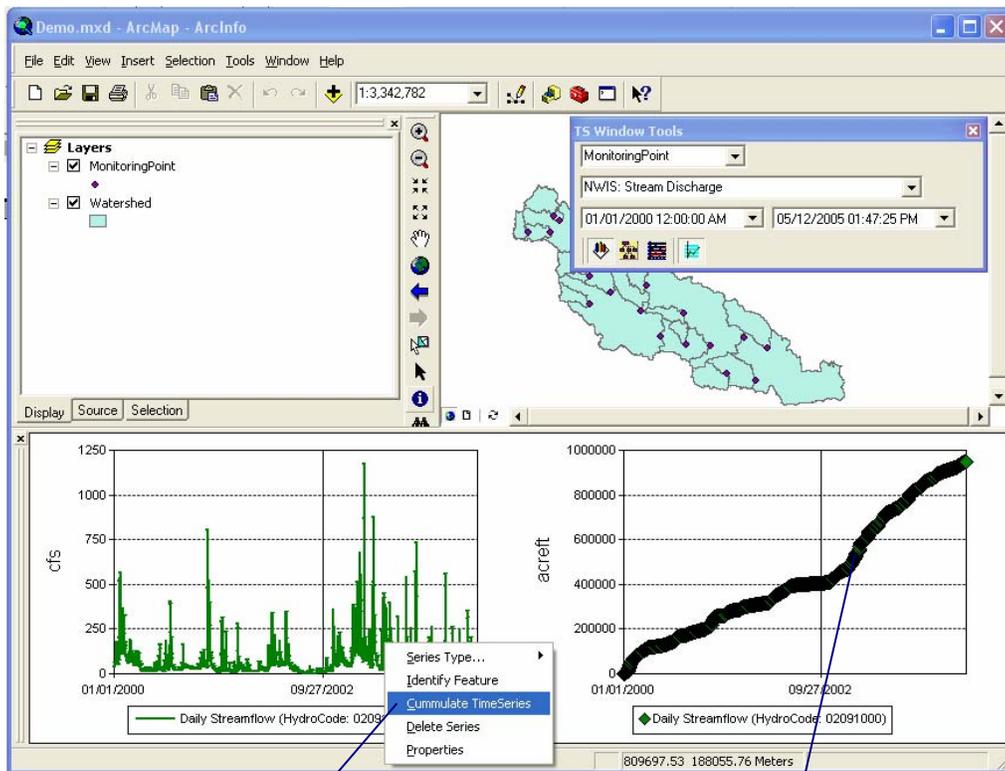
Figure 5.17: Changing the data source from a local Arc Hydro database to a web server.

The primarily downside of visualizing remote data is the amount of time required (1) for the web server to process the query and serve the requested data and (2) to retrieve the data from the web server over the Internet. The time required to execute either step could make this data access technique useless for general visualization and analysis purposes. Preliminary tests have shown that the time to plot data from the National Water Information System database compared to a local Arc Hydro database with records for only the Neuse River Basin are

comparable in execution time. Both take around a second from the time the data is request to the time the data is plotted.

5.5.4 Manipulating Time Series

Because the TSPlotter is built on-top-of the hydrologic classes, it is able to convert time series between dimensions as well as between units. For example, one could plot the streamflow for a particular USGS station and, by simply right-clicking on the time series plot and selecting to cumulate the time series, plot the volume of water passing through that station over time (Figure 5.18). Likewise, if one is plotting precipitation data from two data sources, one with inch units and the other in millimeters, the user can simply drag-and-drop a time series from one chart to the other and the geospatial time series object will be automatically converted the that chart's units.



By right-clicking on a series, user is presented option to

Cumulated series is plotted on a second chart within the chart space because it has different units.

Figure 5.18: Cumulating streamflow to visualize the volume of water passing through the station over time.

The chart space can be used to plot data from a number of different sources. For example, one could plot local time series from Arc Hydro along with web-based data from NWIS or Ameriflux within the same chart space (Figure 5.19). This is possible because the time series are all objects of the same class within the code, thus no matter what their source is one can interact with each in a similar matter. Thus, once the properties of a geospatial time series object have

been populated from a data source, it can be plotted, rescaled in time, or written to an Arc Hydro geodatabase by simply calling the appropriate method of the object.

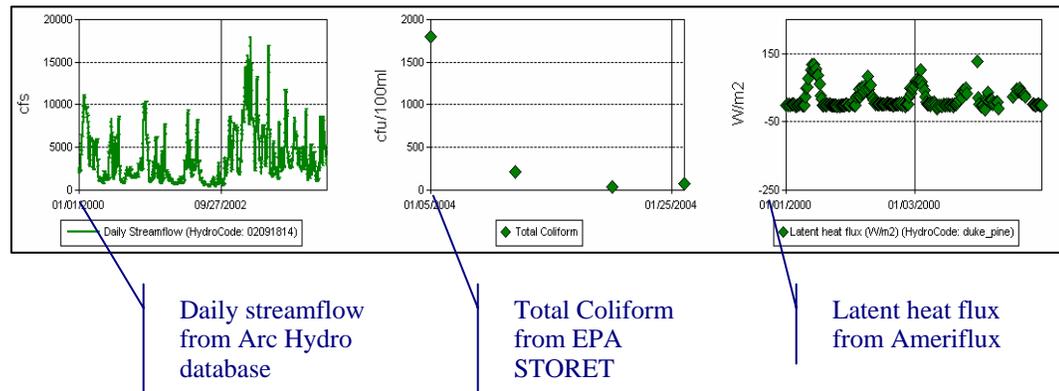


Figure 5.19: Plotting data from multiple sources, both local and remote, within the same chart space.

This disconnection between a data's physical storage format and how it is represented within the software is important because it provides interoperability between data from disparate sources. The Unidata Integrated Data Viewer (IDV) has a similar means for separating the physical data storage format from how data is represented within the software. The IDV is built on a class library called VisAD which provides most of the mapping and charting abilities seen in IDV. VisAD takes the approach of defining a set of classes that can be used to represent numerical data structures, instead of providing support specific support for storage formats like images, grids, and tables. Thus, VisAD can support nearly any type of numerical data by building a bridge between that data format and a class for representing that format within the software (Murray et al. 2001).

5.5.5 Case Study Results Viewed With TSPlotter

Returning to the Neuse River Basin water balance case study, the results calculated using the Space-Time toolbox can be viewed within ArcMap by using the TSPlotter. The user can create charts of the raw data, the pre-processed model data, and the water balance results to better understand the distribution, movement, and storage of water within the Neuse during January, 1990. Figure 5.20 shows the change in storage for three subwatersheds within the basin. The blue Subwatershed, feature 9621, lost nearly 160 million cubic meters during January of 1999 according to this water balance while the red subwatershed, feature 9625, lost 40 million cubic meters and the green subwatershed, feature 9614, gained approximately ten million gallons.

Why did the green subwatershed gain water over the month while the other two watersheds lost water? Why is the storage within the blue watershed more variable over the month compared with the other watersheds? Above anything else, the tools developed through this research contribute to hydrologic sciences by enabling hydrologists to ask these types of questions. Furthermore, because the hydrologist has access to a variety of hydrologic flux and state variables through the internet, it is possible to test hypotheses regarding the questions. For example, snow is not accounted for in this water balance and, although snow is infrequent in North Carolina, it is possible that apparent loss of water within the blue and red subwatershed could be accounted by including snow fall as an input to the subwatershed systems. The NARR has a snow fall component which could be used to test this hypothesis.

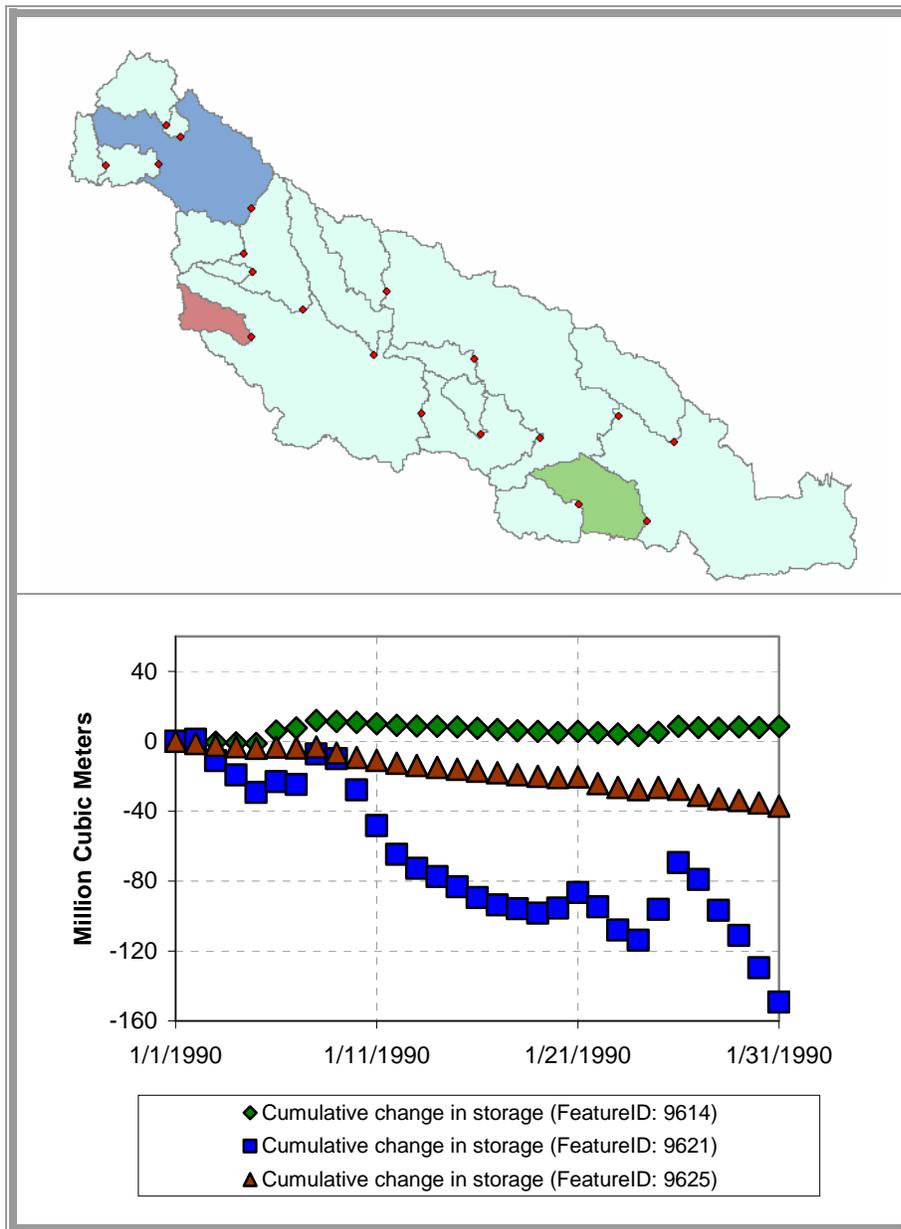


Figure 5.20: Water balance for three subwatersheds within the Neuse River Basin calculated from precipitation and evaporation from the North American Regional Reanalysis (NARR) program and streamflow from USGS Gages.

5.6 SUMMARY

This application demonstrates how one could use software built from using HydroObjects to visualize and process hydrologic data in space and time. As an example of analysis of hydrologic systems, a toolset was developed within the ArcGIS geoprocessing environment which allows one to automatically ingest data the National Water Information System and the North American Regional Reanalysis datasets and to transform these raw data in both space and time in preparation for hydrologic modeling. A simple but generic water balance tool was used to understand the how hydrovolumes within the river basin landscape store and exchange water through time.

As an example of visualization of hydrologic systems, a second tool, TSPlotter, was demonstrated that allows one to plot time series related to geospatial features from within ArcMap. This extension is capable of reading both geodatabase structured according to the Arc Hydro data model as well as remote data sources like the National Water Information System and Ameriflux. TSPlotter provides an interoperable environment capable of reading multiple data formats and working with the data in the same manner, irregardless of its original format.

Chapter 6: Conclusions

The aim of this research was to prototype a geotemporal framework for integrating hydrologic data of various formats and sources. Hydrologists spend a significant amount of time not only with basic information technology tasks (e.g. file conversion, web queries, software packages, etc.), but also with basic hydrologic data conversion tasks (e.g. unit manipulations, spatial and temporal integration and interpolation, etc.). It is possible to build a basic set of hydrologic classes that can help hydrologists to quickly gain access to local and remote data sources for a variety of data formats, and to perform basic conversions and processing of hydrologic data in space and time.

6.1 SUMMARY

To illustrate these ideas, two applications were developed as extensions to ArcGIS. The first is a toolbox for performing basic temporal geoprocessing and hydrologic analysis. The toolbox's functionality is demonstrated by performing a water balance for subwatersheds within the Neuse River Basin. This model automatically ingests data from National Water Information System (NWIS) and the North American Regional Reanalysis (NARR) program directly from the web, rescales the NARR field data in both space and time, and lastly performs a water balance for twenty subwatersheds within the Neuse that represent the drainage area between USGS stations. The water balance portion of the model accounts for horizontal exchanges of water recorded at USGS streamflow stations and

vertical exchanges of water predicted by NARR to estimate the change in storage through time.

The second application is a toolbar and dockable window within ArcMap named TSPlotter that is capable of plotting time series by simply point-and-clicking features within the map. The data source for the time series can either be a local database formatted according to the Arc Hydro schema or a remote web server. The supported web servers at this point are NWIS and Ameriflux; work is underlay to include EPA STORET as well. If a hydrologist adds the NWIS Web Feature Service (WFS) to ArcMap for displaying the locations of USGS streamflow gages, then TSPlotter can be used, in conjunction with the streamflow gages layer, to plot time series directly from the web for any location within the United States – without having any of this information stored locally.

Both of these applications make use of the same underlying hydrologic class library. The library consists of two classes for spatiotemporal modeling and data management: geospatial time series and hydrologic flux coupler. The geospatial time series class is used to store and operate on a hydrologic time series with a georeferenced geometry as a property. The hydrologic flux coupler class allows one to group time series into a system and then, from that system, derive new time series such as the change in storage over time. A flux coupler object is capable of converting time series between dimensions (e.g. a flux to a flow) by intersecting the shape of the hydrovolume feature with the shape of the geospatial time series feature representing the flux to find the area over which the mass is transferred.

6.2 CONCLUSIONS

In the first chapter of this dissertation, the following question was posed:

How can hydrologists integrate observed and modeled data from various sources into a single description of the environment?

Answering this question provides an appropriate framework for summarizing the research conclusions.

Interoperability between hydrologic datasets should come from a common geotemporal referencing system. Each value within each dataset is representative of some point, line, area, or volume of space and some point or interval of time. If each value is properly referenced to a geospatial and temporal coordinate system and the details of that coordinate system are clearly specified within the dataset's metadata, then it is possible to build software which combines data into a common geotemporal picture of the environment. Furthermore, this geotemporal view should maintain two representations of space – fields and entities – and these two representations must be capable of interacting with one another (i.e. interpolate a space-time field from observation point entities, or summaries of a space-time field within watershed polygons).

To populate the geotemporal environment, a collection of hydrologic classes is required for digitally representing hydrologic concepts. This research presents two of these classes, geospatial time series and hydrologic flux coupler, but there are certainly others. It is important to understand that these classes are not tailored for a specific data source or data format, they are generalized data models meant to represent hydrologic concepts within space and time. Thus, a

geospatial time series can be populated with data from NWIS, Arc Hydro, EPA STORET, etc. Each class has a set of properties and methods which a software developer can use to more quickly gain access to information and perform basic hydrologic transformations. The end result is a set of basic building blocks, or what is common referred to in computer science as an Application Programming Interface (API), for accessing and manipulating hydrologic information that hydrologists can use to aid in the development of new and the extension of existing hydrologic software.

Many federal databases, as well as the model output from NARR and ETA, are accessible online both through human-guided web queries and through machine-to-machine web services. Human-guided web queries are where the user visits a website, fills out one or more web pages, and then, based on parameters on the website, an underlying database is queried for a subset of information. The data resulting from the query are typically returned as a webpage with ASCII text. For these types of information systems, it is often possible to bypass the web interface by constructing a URL to directly query the underlying database from a client application. Visualization and processing applications can be configured to directly ingest data from federal databases.

Thus, the first method for accessing and querying federal databases is through URL manipulation on the client side. This approach works well for both relational database information sources as well as model grid information sources served with OPeNDAP. Both data sources can be queried for particular variables observed at particular stations over a defined time interval. The TSPlotter and

Space-Time toolbox are examples of applications which use URL manipulation to automatically ingest data from these two sources.

A second method for obtaining data is demonstrated by Microsoft's Terra Server and the USGS EROS data center web services. These methods follow a defined protocol, Simple Object Access Protocol (SOAP), for passing information between a server and a client. The TerraServer delivers Digital Orthophoto Quadrangle (DOQ) and Digital Raster Graphic (DRG) image titles stored in "one of the largest online databases in the world" (<http://terraserver.microsoft.com/About.aspx?n=AboutWhatIs>) to a client application over the internet. The USGS EROS web services use the same protocols to allow a client application to query for the elevation or the land cover at any location within the United States. When the client application passes the location to the EROS web server, the web server searches a collection of digital elevation models, some of which overlap in space, and returns to the client the elevation from the highest resolution grid available at that location.

The primary advantage of SOAP web services compared with the URL-based queries is that the URL-based queries will break if the data server ever changes the process for constructing URLs. The web service method is also superior in how it is implemented within both the server and client code. Many software languages have embraced SOAP standards for web services and have toolkits which ease the process of developing SOAP web services on both a server and a client. The Microsoft .Net Studio, for example, allows one to add a

web service to an application nearly as easily as adding a local resource stored as a dynamic link library.

Therefore, it is possible to construct objects using a variety of data formats and data models both directly from remote data sources as well as from local data sources. This research has demonstrated that hydrologists can gain access to subsets of large national and continental-scale datasets directly over the internet. Furthermore, in many cases, these data servers can process the request and return the data in sufficient time for both real-time visualization and processing on the client's machine. This is extremely important because it means one does not need to reproduce federal databases on a local machine for hydrologic visualization and processing purposes. The moment when a copy of these datasets is made, the data are the responsibility of the researchers and not the federal agency. Interacting with the database to retrieve data only when needed for a hydrologic model or visualization application keeps the data management responsibilities, such as quality assurance, hardware investments, and personal support with the data providers and not the data users.

6.3 RECOMMENDATIONS

There are two primary recommendations for future research and development. First, the HydroObjects library will be critical for building and maintaining hydrologic software on both server and client machines. The current version of the HydroObjects library is dependent on both ESRI and Microsoft software (ESRI software is used for georeferencing and geoprocessing while Microsoft software is used for charting). It is possible, however, to isolate and

remove the dependences on third party software to provide a more generic hydrologic class library. Then, application-specific libraries for geoprocessing, charting, etc. could utilize the base hydrologic class library for accessing remote and local information, and for performing basic hydrologic processing.

For this reason, the proposed overall structure of the hydrology software library is comprised of two levels: a base level which includes libraries for data access and hydrologic processing and an application level for incorporation within popular commercial and open-sources software (Figure 6.1). This structure provides a separation between commercial and open-source software. It is not realistic to completely ignore commercial software products because they are so widely used by hydrologists and provide a software base which the hydrologic community will never be able to maintain themselves. However, the amount of code development within a commercial software system should be minimized to protect against changes to the commercial software system.

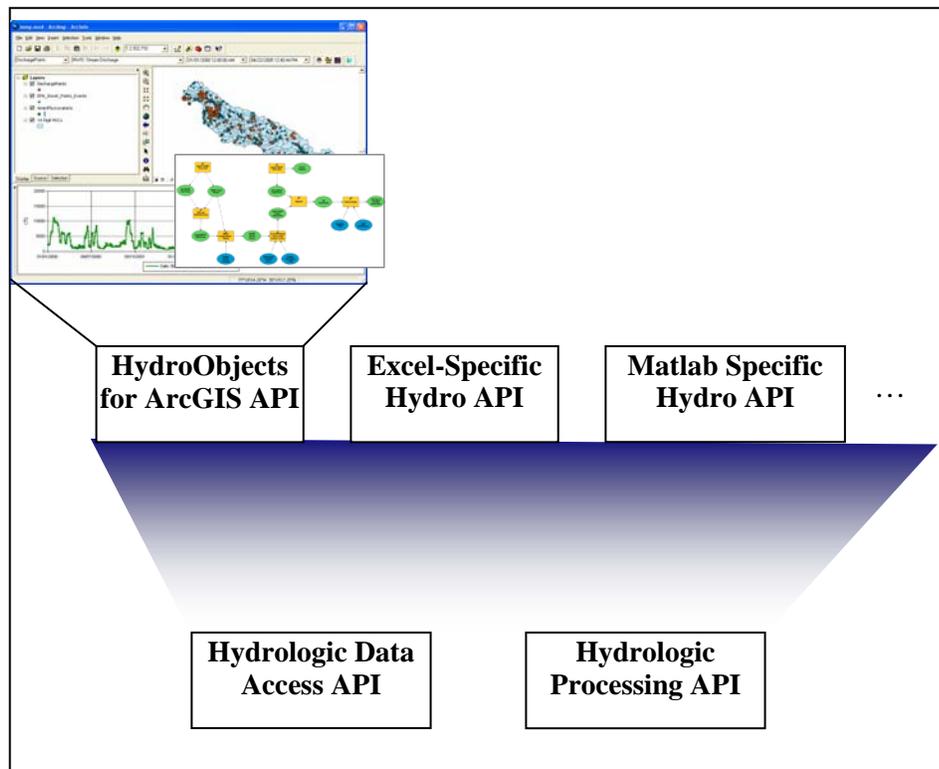


Figure 6.1: Hierarchy of hydrology software libraries or APIs (Application Programming Interface). Base level includes core data access and hydrologic processing routines. Application level incorporates base libraries and third party software libraries to customize commercial software for hydrologic analysis.

Second, web services must play a central role in allowing the integration of distributed and heterogeneous datasets and models. This research suggests that there should be three types of services: (1) data services, (2) processing services, and (3) modeling services. Data services simply supply raw information from data suppliers to data consumers. Processing services manipulate raw data to provide information required for modeling or analysis. Modeling services are hydrologic or hydraulic models built for particular regions of the watershed.

Users can manipulate the model's basic input parameters and then run the model to simulate the movement and distribution of water.

The end goal of a hydrologic information system should be to provide a flexible, distributed cyberinfrastructure for sharing hydrologic data and models. This will allow community members to participate by adding services or consuming services freely and easily. Adopting a service-oriented infrastructure will also insure that researchers can communicate despite differences in computer operating systems and preferred software languages. Researchers can build and maintain services that expose their data, analysis tools, and models to other community members, alleviating the burden on the central CUAHSI HIS organization, and encouraging open and free exchange of data, information, and models.

Appendix A

Unit types and conversions factors in Conversions.xml. Many, but not all, of conversion abbreviations and factors are from Maidment (1993).

TSUnitType		
Name	Conversion factor	Abbreviation(s)
Length		
Meter	1	m
Inch	2.54E-2	in
Foot	3.048E-1	ft, feet
Mile	1.6093E3	mi
Millimeter	1.0E-3	mm
Centimeter	1.0E-2	cm
Area		
Square meters	1	m ²
Square centimeters	1.0E-4	cm ²
Hectare	1.0E4	ha
Square kilometer	1.0E6	km ²
Square feet	9.2903E-2	ft ²
Acre	4.046856E3	acre
Square mile	2.59E6	mi ²
Volume		

Cubic meter	1	m ³
Cubic centimeter	1.0E-5	cm ³
Liter	1.0E4	L
Cubic kilometer	1.0E9	km ³
Hectare-meter	1.0E4	ham
Cubic foot	2.8317E-2	ft ³
US gallon	3.7854	gal
Acre-foot	1.2335E3	acreft
Time		
Second	1	s
Minute	6.0E1	min
Hour	3.66E3	h
Day	8.64E4	day
Year	3.1536E7	year
Volume Flowrate		
Cubic meter per second	1	m ³ /s
Liter per second	1.0E-3	L/s
Cubic meter per day	1.1574E-5	m ³ /d
Cubic feet per second	2.8317E-2	ft ³ /s, cfs
Gallon per minute	6.30902E-5	gpm
Million gallons per day	4.3813E-1	MGD
Acre-foot per year	3.9113E-5	acreft/year
Centimeter per second	1.0E-2	cm/s

Volume Area Flux		
Meter per second	1	m/s
Millimeter per hour	2.77778E-7	mm/h
Millimeter per day	1.15741E-8	mm/d, mm/day
Centimeter per hour	2.77778E-6	cm/hr
Meter per day	1.15741E-5	m/d
Inch per day	2.93981E-7	in/day, in/d
Inch per hour	7.05556E-6	in/h
Feet per second	3.048E-1	ft/s
Mass Area Flux		
Kilogram per square meter per second	1	kg/(m ² s), kg/m ² /s
Kilogram per square meter per day	1.15741E-5	kg/(m ² d), kg/m ² /d
Energy Area Flux		
Watt per square meter	1	W/m ²
Watt per square centimeter	1.0E4	W/cm ²
Megajoules per day per square meter	1.1574E1	MJ/day/m ²
Langley per second	4.1868E4	ly/s
Langley per minute	6.978E2	ly/min
Langley per day	4.8458E-1	ly/day

Energy		
Watt per second	1	W/s
Joule	1.0E3	J
Kilojoule	1.0E6	kJ
Kilowatt per hour	3.6E9	kWh
Calorie	4.1868E3	cal
Kilocalorie	4.1868E6	kcal
British thermal unit	1.0551E6	BTU
Mass per area		
Mass per area	1	kg/m ²

Appendix B

HydroObjects class documentation.

NAMESPACE LIST

The namespaces specified in this document are:

Namespace	Assembly
CRWR.Utils	HydroObjects
CRWR.HydroObjects	HydroObjects

Namespace : CRWR.Utils

CRWR.Utils Type List

ENUMERATIONS

Type	Summary
TimeSeriesOp.MathOp	Supported mathematical operations

CLASSES

Type	Summary
GenUtils	General functions and routines used throughout the classes.
TimeSeriesOp	The TimeSeriesOp class facilitates the addition and subtraction of geospatial time series objects.

CRWR.Utils Enumerations

TIMESERIESOP.MATHOP ENUMERATION

Summary

`nestedPublic enumeration TimeSeriesOp.MathOp`

Supported mathematical operations

Remarks

Supported mathematical operations

Enumeration Members

Field	Summary
Add	Adds each value of the TSValues array with the same TSDatetime value

Subtract	Subtracts each value of the TSValues array with the same TSDatetime value
----------	---

CRWR.Utils Classes

TIMESERIESOP CLASS

Summary

```
public class TimeSeriesOp
```

The TimeSeriesOp class facilitates the addition and subtraction of geospatial time series objects.

Remarks

Constructor Members

Name	Access	Summary
TimeSeriesOp()	public	Creates a new instance of the TimeSeriesOp class and sets the temporal domain from the properties of the GeospatialTimeSeries

Property Members

Name	Access	Summary
ResultTimeSeries : <i>GeospatialTimeSeries</i>	public	Returns the geospatial time series resulting from the operation

TimeSeries : <i>GeospatialTimeSeries</i>	public	Returns a particular geospatial time series within the operators collection of geospatial time series
---	--------	---

Method Members

Name	Access	Summary
AddTimeSeriesToList() : <i>Void</i>	public	Adds a geospatial time series to the operator
CalcResultTSValues() : <i>Array</i>	public	Provides only the resulting array of TSValues

GENUTILS CLASS

Summary

```
public class GenUtils
```

Method Members

Name	Access	Summary
CheckField() : <i>Void</i>	public	Checks for a field within a table of a particular name and type.
ColumnExists() : <i>Boolean</i>	public	Checks if a column exists on a table within a DB using

		ADO and not ArcObjects
GetFeatureArcObjects() : <i>IFeature</i>	public	Gets a feature from a geodatabase by its HydroID using ArcObjects
GetFeatureArcObjects() : <i>IFeature</i>	public	Gets a feature from a geodatabase by its HydroCode using ArcObjects
GetFeatureClass() : <i>String</i>	public	Gets a feature class within a geodatabase using ADO and not ArcObjects
GetFeatureClassArcObjects() : <i>IFeatureClass</i>	public	Gets a feature class from a geodatabase using ArcObjects
GetHydroIDFromHydroCode() : <i>Int64</i>	public	Gets the HydroCode for a feature with a given HydroID. Looks in the TimeSeries Table. Uses ADO and not ArcObjects.
GetTSUnitTypeFromTSTypeID() : <i>UnitTypeEnum</i>	public	Gets the unit type of a given TSTypeID. Uses ArcObjects.
GetUnitType() : <i>UnitTypeEnum</i>	public	Gets the unit types based on a unit string. Looks up unit type using the

		Conversion.xml document within the application folder.
IncreaseDateByInterval() : <i>Object</i>	public	Increases a date by a given TSInterval
IsRegularAsBoolean() : <i>Boolean</i>	public	Converts a IsRegular from a long type to a Boolean type (1 = True, 0 = False)
IsRegularAsLong() : <i>Int64</i>	public	Converts a IsRegular from a Boolean type to a long type (1 = True, 0 = False)
TableExists() : <i>Boolean</i>	public	Checks if a table exists within a DB using ADO and not ArcObjects
ToSeconds() : <i>Double</i>	public	Returns the number of seconds in a TSInterval
ValidateTSTable() : <i>Void</i>	public	Checks that schema of a table matches the Arc Hydro TimeSeries table schema. Will add required fields if they do not currently exist.
ValidateTSTypeTable() : <i>Void</i>	public	Checks that schema of a table matches the Arc Hydro TSType table schema. Will add required

		fields if they do not currently exist.
WriteTSColToGDB() : <i>Void</i>	public	Writes a collection of GTS with the same TSType properties to a GDB

NAMESPACE : CRWR.HYDROOBJECTS

CRWR.HydroObjects Type List

ENUMERATIONS

Type	Summary
GeospatialTimeSeries.DataTypeEnum	Time series types as defined by Arc Hydro
GeospatialTimeSeries.StatisticEnum	Statistic types
GeospatialTimeSeries.TSIntervalEnum	Time interval unit types
GeospatialTimeSeries.UnitTypeEnum	Basic dimensions for measurements

CLASSES

Type	Summary
GeospatialTimeSeries	The GeospatialTimeSeries class is a discrete space-time representation of hydrologic variables. It is a generic design that can be populated with data stored in any physical format. It has the inherent ability, however, to be constructed directly from an Arc Hydro geodatabase.
HydrologicFluxCoupler	Represents a volume of space which has associate inputs and outputs through time

CRWR.HydroObjects Enumerations

GEOSPATIALTIMESERIES.STATISTICENUM ENUMERATION

Summary

nestedPublic enumeration
GeospatialTimeSeries.StatisticEnum

Statistic types

Remarks

Statistics types enum

Enumeration Members

Field	Summary
-------	---------

Count	Number of TS values in object
Maximum	Maximum TS value in object
Median	Median TS value in object
Minimum	Minimum TS value in object

GEOSPATIALTIMESERIES.TSINTERVALENUM ENUMERATION

Summary

nestedPublic enumeration
 GeospatialTimeSeries.TSIntervalEnum

Time interval unit types

Remarks

Enumeration Members

Field	Summary
Day	unit of time
Hour	unit of time
Minute	unit of time

Month	unit of time
Other	Used for irregular time series
Second	unit of time
Week	unit of time
Year	unit of time

GEOSPATIALTIMESERIES.DATATYPEENUM ENUMERATION

Summary

nestedPublic enumeration
 GeospatialTimeSeries.DataTypeEnum

Time series types as defined by Arc Hydro

Remarks

Enumeration Members

Field	Summary
Average	The average rate over a time interval, calculated as the incremental value divided by the duration of the data interval

Cumulative	The accumulated value since the beginning of the record
Incremental	The difference in the cumulative values at the beginning and end of a time interval
Instantaneous	A condition at a given instant of time
Maximum	The maximum value of a variable in a time interval
Minimum	The minimum value of a variable in a time interval

GEOSPATIALTIMESERIES.UNITTYPEENUM ENUMERATION

Summary

nestedPublic enumeration
 GeospatialTimeSeries.UnitTypeEnum

Basic dimensions for measurements

Remarks

Enumeration Members

Field	Summary
Area	[L ²]
Concentration	[M/L ³]

Energy	A base dimension [E]
EnergyAreaFlux	$[E/L^2/T]$
EnergyLineFlux	$[E/L/T]$
EnergyPerArea	$[E/L^2]$
EnergyPerLength	$[E/L]$
EnergyPerVolume	$[E/L^3]$
Length	A base dimension [L]
Mass	A base dimension [M]
MassAreaFlux	$[M/L^2/T]$
MassFlowrate	$[M/T]$
MassLineFlux	$[M/L/T]$
MassPerArea	$[M/L^2]$

MassPerLength	[M/L]
Power	[E/T]
Time	A base dimension [T]
Volume	[L ³]
VolumeAreaFlux	[L/T]
VolumeFlowrate	[L ³ /T]
VolumeLineFlux	[L ² /T]

CRWR.HydroObjects Classes

GEOSPATIALTIMESERIES CLASS

Summary

```
public class GeospatialTimeSeries
```

The GeospatialTimeSeries class is a discrete space-time representation of hydrologic variables. It is a generic design that can be populated with data stored in any physical format. It has the inherent ability, however, to be constructed directly from an Arc Hydro geodatabase.

Remarks

Using this class, one could create an `GeospatialTimeSeries` object and use methods of the object to do such things as transform its units, write it to a disk, or add it to a Microsoft Office Web Components (OWC) chartspace. The properties of the object are used in each of these methods to properly manipulate the object.

Constructor Members

Name	Access	Summary
<code>GeospatialTimeSeries()</code>	public	Creates a new instance of the <code>GeospatialTimeSeries</code> class
<code>GeospatialTimeSeries()</code>	public	Creates a new instance of the <code>GeospatialTimeSeries</code> class from an Arc Hydro geodatabase

Property Members

Name	Access	Summary
<code>DataType</code> : <i><code>DataTypeEnum</code></i>	public	Type of time series data e.g. instantaneous, cumulative, averaged, etc.
<code>FeatureID</code> : <i><code>Int64</code></i>	public	HydroID of the feature described by the time series (HydroID is a unique ID used in the Arc Hydro data model).
<code>GeneratedDescription</code> :	public	Description of how a time series was

<i>String</i>		generated
GeodatabasePath : <i>String</i>	public	Gives the geodatabase path, if the object was constructed. from Arc Hydro
HydroCode : <i>Object</i>	public	The public identifier of the feature described by the time series
InGDB : <i>Boolean</i>	public	Read only property that indicates if the object was constructed from an Arc Hydro geodatabase and has not since been modified.
IsRegular : <i>Boolean</i>	public	Whether data are regularly or irregularly measured in time
Origin : <i>String</i>	public	Description of the source for the time series
Shape : <i>IGeometry</i>	public	The shape of the feature associated to the time series
TSDateTimes : <i>Array</i>	public	The date/times of measurement
TSIntervalLength : <i>Double</i>	public	The length of time represented by each measurement. This is used with TSIntervalUnit to represent the TSInterval.

TSIntervalUnit : <i>TSIntervalEnum</i>	public	The unit of time (second, minute, hour, etc.) represented by each measurement. This is used with TSIntervalLength to represent the TSInterval.
TSTypeID : <i>Int64</i>	public	Identifier for the type of time series
TSUnitType : <i>UnitTypeEnum</i>	public	Gives the dimensions of the measurement unit
TSValues : <i>Array</i>	public	The time series values
UID : <i>Guid</i>	public	Read only GUID for the object.
Units : <i>String</i>	public	Units of measurement
Variable : <i>String</i>	public	Description of the time series being measured, e.g. Daily Streamflow

Method Members

Name	Access	Summary
AddToChartSpace() : <i>Void</i>	public	Adds the time series to a Microsoft Office Web Components (OWC10) ChartSpace object.

AddToChartSpace() : <i>Void</i>	public	Adds the time series to a Microsoft Office Web Components (OWC10) ChartSpace object.
ChangeUnits() : <i>GeospatialTimeSeries</i>	public	Changes to Units property and updates the TSValues to a new measurement unit
CloneTS() : <i>GeospatialTimeSeries</i>	public	Clones the object with the exception of the TSUID.
GetTSDatetimeMax() : <i>DateTime</i>	public	Returns the maximum date
GetTSDatetimeMin() : <i>DateTime</i>	public	Returns the minimum date
GetTSStatistic() : <i>Object</i>	public	Returns a statistic of the time series
RescaleTime() : <i>GeospatialTimeSeries</i>	public	Produces a new time series by upscaling the original time series.
ResetTSValues() : <i>Object</i>	public	Reset the date and value arrays according to the current min and max dates
SetTSValues() : <i>Void</i>	public	Sets the TSValues from the Arc Hydro database.

WriteToGDB() : <i>Void</i>	public	Writes the TS object to an Arc Hydro geodatabase
----------------------------	--------	--

HYDROLOGICFLUXCOUPLER CLASS

Summary

```
public class HydrologicFluxCoupler
```

Represents a volume of space which has associate inputs and outputs through time

Remarks

Created for a particular feature within an Arc Hydro geodatabase. Requires an additional table named CouplingTable which relates a volume feature to its associated flux and flow features.

Constructor Members

Name	Access	Summary
HydrologicFluxCoupler()	public	Creates a new instance of the HydroSystem class for a particular feature in an Arc Hydro geodatabase.

Property Members

Name	Access	Summary
AreaFluxes : <i>Collection</i>	public	A collection of the related geospatial time series with area flux dimensions

Flows : <i>Collection</i>	public	A collection of related geospatial time series with flow dimensions
LineFluxes : <i>Collection</i>	public	A collection of the related geospatial time series with line flux dimensions
NumOfAreaFluxes : <i>Int64</i>	public	The number of geospatial time series with area flux dimensions related to this HydroSystem
NumOfFlows : <i>Int64</i>	public	The number of geospatial time series with flow dimensions related to this HydroSystem
NumOfLineFluxes : <i>Int64</i>	public	The number of geospatial time series with line flux dimensions related to this HydroSystem

Method Members

Name	Access	Summary
GetChangeInStorage() : <i>GeospatialTimeSeries</i>	public	Returns the change for the HydroSystem taking into account all in and out flows and fluxes
NetAreaFlux() : <i>GeospatialTimeSeries</i>	public	The net area flux into volume

NetFlow() : <i>GeospatialTimeSeries</i>	public	The net flow into volume
NetLineFlux() : <i>GeospatialTimeSeries</i>	public	The net line flux into volume

Bibliography

- Arctur, D., and M. Zeiler. 2004. *Designing Geodatabases: Case Studies in GIS Data Modeling*. ESRI Press, Redlands, CA.
- Baker, K. S., B. J. Benson, D. L. Henshaw, D. Blodgett, J. H. Porter, and S. G. Stafford. 2000. Evolution of a multisite network information system: The LTER information management paradigm. *Bioscience* **50**:963-978.
- Band, L., M. Moss, and F. Ogden. 2003. The CUAHSI Plan for a Network of Hydrologic Observatories. Pages 19-24 *in* First Interagency Conference on Research in the Watersheds, Benson, Arizona.
- Betancourt, T. L., and O. Wilhelmi. 2003. Final Report on the GIS Demonstration Project. National Center for Atmospheric Research, Bolder, CO.
- Chow, V. T., D. R. Maidment, and L. Mays. 1989. *Applied Hydrology*. McGraw-Hill, New York.
- Council, W. S. a. T. B. o. t. N. R. 2001. *Envisioning the Agenda for Water Resources Research in the Twenty-first Century*. National Academy Press, Washington, D.C.
- Goodall, J. L., D. R. Maidment, and J. Sorenson. 2004. Representation of Spatial and Temporal Data in ArcGIS. *in* GIS and Water Resources III. AWRA, Nashville, TN.
- Graham, W., A. Kruger, P. Kumar, V. Lakshmi, U. Lall, D. Lettenmaier, D. Maidment, M. Piasecki, and C. Zheng. 2002. Consortium of Universities for the Advancement of Hydrologic Science Inc. (CUAHSI) Hydrologic Information Systems White Paper. Consortium of Universities for the Advancement of Hydrologic Science Inc., Washington, DC.
- Habermann, T., J. Cartwright, R. Schweitzer, I. Barrodale, and E. Davies. 2004. Integrating science data into geographic information systems. *in* 20th International Conference on Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology. American Meteorological Society, Seattle, WA.

- Ho, Y., J. Weber, and J. Caron. 2005. A comparison of common geospatial data model between AS and GIS community. *in* 21th International Conference on Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology. American Meteorological Society, San Diego, CA.
- Hoffer, J. A., and F. R. McFadden. 2002. Modern database management, 6th edition. Prentice Hall, Upper Saddle River, NJ.
- Hooper, R. P., K. H. Reckhow, and L. E. Band. 2004. Designing a Network of Hydrologic Observatories. *in* 2004 Joint Asia Oceania Geosciences Society 1st Annual Meeting & APHW 2nd Conference, Singapore.
- Hornberger, G. M., J. Aber, J. Bahr, R. Bales, K. Beven, E. Foufoula-Georgiou, G. Katul, J. L. K. III, R. Koster, D. Lettenmaier, D. McKnight, K. Miller, K. Mitchell, J. Roads, B. R. Scanlon, and E. Smith. 2001. A Plan for a New Science Initiative on the Global Water Cycle. Water Cycle Study Group of the U.S. Global Change Research Program (SGCRP-WCSG).
- Keller, G. R. 2003. GEON (GEOscience Network) -- A first step in creating cyberinfrastructure for the geosciences. *in* Electronic Seismologist.
- Langran, G. 1992. Time in geographic information systems, pbk. edition. Taylor and Francis, London ; New York.
- Maidment, D. R. 1993. Handbook of hydrology. McGraw-Hill, New York.
- Maidment, D. R., editor. 2002. Arc Hydro: GIS for Water Resources. ESRI Press, Redlands, CA.
- Maidment, D. R. 2004. Creating Hydrologic Information Systems.
<http://www.ce.utexas.edu/prof/maidment/visual/meetings/utahstate9feb2004.ppt>. Accessed January 15, 2004.
- Maidment, D. R. 2005. A Data Model for Hydrologic Observations. Pages 23 *in* CUAHIS Hydrologic Information System Symposium, Austin, Texas.

- Murray, D., B. Hibbard, T. Wittaker, and J. Kelly. 2001. Using VisAD to Build Tools for Visualizing and Analyzing Remotely Sensed Data. *in* IEEE 2001 International Geoscience and Remote Sensing Symposium, Sydney, Australia.
- Murray, D., T. Whittaker, J. McWhirter, and S. Wier. 2004. Integrating GIS data with Geoscience data in Unidata's IDV. *in* 20th International Conference on Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology. American Meteorological Society, Seattle, WA.
- Nativi, S., M. B. Blumenthal, J. Caron, B. Domenico, T. Habermann, D. Hertzmann, Y. Ho, R. Raskin, and J. Weber. 2004. Differences among the data models used by the Geographic Information Systems and Atmospheric Science communities. *in* 20th International Conference on Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology. American Meteorological Society, Seattle, WA.
- NSF Advisory Panel on Cyberinfrastructure. 2003. Revolutionizing Science and Engineering Through Cyberinfrastructure: Report of the National Science Foundation Blue-Ribbon Advisory Panel on Cyberinfrastructure. National Science Foundation, Arlington, VA 2203.
- Peuquet, D. J. 2001. Making space for time: Issues in space-time data representation. *Geoinformatica* 5:11-32.
- Peuquet, D. J. 2002. Representations of space and time. The Guilford Press, New York, NY.
- Reckhow, K., L. Band, C. Duffy, J. Famiglietti, D. Genereux, J. Helly, R. Hooper, W. Krajewski, D. McKnight, F. Ogden, B. Scanlon, and L. Shabman. 2004. Designing Hydrologic Observatories: A Paper Prototype of the Neuse Watershed. CUAHSI Technical Report Number 6, Consortium of Universities for the Advancement of Hydrologic Science, Inc. (CUASHI), Washington, D.C.
- Rew, R., G. Davis, S. Emmerson, and H. Davies. 1997. NetCDF User's Guide for C. <http://www.unidata.ucar.edu/packages/netcdf/guidec/>. Accessed January 15, 2004.

- Salas, J. D. 1993. Analysis and Modeling of Hydrologic Time Series. *in* D. R. Maidment, editor. Handbook of Hydrology. McGraw-Hill, New York.
- Singh, V. P., and D. A. Woolhiser. 2002. Mathematical modeling of watershed hydrology. *Journal of Hydrologic Engineering* 7:270-292.
- Sumrada, R. 2003. Temporal Data and Temporal Reference System. *in* International Federation of Surveyors (FIG) Working Week 2003, Paris, France.
- Tooby, P. 2003. GEON Overview: Cyberinfrastructure for the Geosciences. <http://www.geongrid.org/about.html>. Accessed January 15, 2004.
- Whiteaker, T. L., O. Robayo, D. R. Maidment, and D. Obenour. 2005. From a NEXRAD Rainfall Map to a Flood Inundation Map. Accepted to the *Journal of Hydrologic Engineering*.
- Wilhelmi, O. V., and J. C. Brunskill. 2003. Geographic information systems in weather, climate, and impacts. *Bulletin of the American Meteorological Society* 84:1409–1414.

Vita

Jonathan Lee Goodall was born in Harrisonburg, Virginia on March 14, 1979. He is the son of Paul and Jane Goodall. Jonathan attended his first two years of high school at North Hunterdon High School in Annandale, New Jersey and then transferred to Spotswood High School in Penn Laird, Virginia. Upon graduation from Spotswood High School, Jonathan attended the University of Virginia earning the degree Bachelor of Science in Civil Engineering in May of 2001. He began graduate study in Civil Engineering at the University of Texas at Austin in August of 2001 and earned the degree of Master of Science in Engineering in May of 2003.

Permanent address: 915 Urban Avenue, Durham, North Carolina 27701

This dissertation was typed by the author.