

## Component-based end-user database design for ecologists

Judith Bayard Cushing · Nalini Nadkarni ·  
Michael Finch · Anne Fiala · Emerson Murphy-Hill ·  
Lois Delcambre · David Maier

© Springer Science + Business Media, LLC 2007

**Abstract** To solve today's ecological problems, scientists need well documented, validated, and coherent data archives. Historically, however, ecologists have collected and stored data idiosyncratically, making data integration even among close collaborators difficult. Further, effective ecology data warehouses and subsequent data mining require that individual databases be accurately described with metadata against which the data themselves have been validated. Using database technology would make documenting data sets for archiving, integration, and data mining easier, but few ecologists have expertise to use database technology and they cannot afford to hire programmers. In this paper, we identify the benefits that would accrue from ecologists' use of modern information technology and the obstacles that prevent that use. We describe our prototype, the *Canopy DataBank*, through which we aim to enable individual ecologists in the forest canopy research community to be their own database programmers. The key feature that makes this possible is domain-specific database components, which we call *templates*. We also show how additional tools that reuse these components, such as for visualization, could provide gains in productivity and motivate the use of new technology. Finally, we suggest ways in which communities might share database components and how components might be used to foster easier data integration to solve new ecological problems.

**Keywords** Ecosystem informatics · End-user programming · Domain-specific data structures · Spatial databases · Scientific visualization

---

J. B. Cushing (✉) · N. Nadkarni · A. Fiala  
The Evergreen State College, Olympia, WA 98505, USA  
e-mail: judyc@evergreen.edu

E. Murphy-Hill · L. Delcambre · D. Maier  
Portland State University, Portland, OR 97201, USA

M. Finch  
Department of Planetary Sciences, Lunar and Planetary Laboratory, University of Arizona,  
Tucson, AZ 85721, USA

## 1 Introduction

Because the collective analysis of data originally gathered by individuals can yield insight beyond a single data set (NRC. National Research Council, 1995, 1997), and because such analysis is needed to solve current ecological and environmental problems (<http://intranet.lternet.edu/archives/documents/foundations/WhitePaperJune2002GRS.html>), future advances in ecology will be fostered by effective information management. Database technology will become integral both to the every day data management and to the creation of new knowledge (Dunne, 2005; Nadkarni & Cushing, 1995). Unfortunately, current database technology is not adequate to meet this challenge. Workshops sponsored by the National Science Foundation, the USGS and NASA have identified the need for a new “biodiversity and ecosystem informatics (BDEI)” initiative. Noting challenges and opportunities for further research in acquisition, conversion, analysis, synthesis and dissemination of data and metadata (e.g., digital libraries, remote sensing, mobile computing, and taxonomies), participants in these workshops typically characterized ecological data and metadata as highly complex—ontologically, spatio-temporally and sociologically. Lack of harmonized protocols, resistance to depositing data and metadata in central repositories, and lack of expertise with informatics tools were also noted as contributing to the limited use of information technology (Cushing & Wilson, 2005; Maier, Landis, Frondorf, Silverschatz, Schnase, & Cushing, 2001; Schnase et al., 2003).

Although many ecology data sets are eventually integrated into database data warehouses, as in the information archives of the Long Term Ecological Research sites (<http://lternet.edu>), most ecologists maintain their own research data in spreadsheets or flat files. Appropriate information technology could not only make the data documentation for ecological archives easier and allow for data integration, but could also bring productivity gains to individual and teams of researchers. These productivity gains are required by researchers before they will adopt new technology. The benefits accruing to individual researchers would be greater the earlier the technology is adopted in a research study.

The Canopy Database Project (Cushing, Nadkarni, Delcambre, Healy, Maier, & Ordway, 2002a, b) is exploring how appropriate database technology can be integrated into the ecology research cycle. In this paper, we present a prototype database design tool that aims to enable ecologists who study forest canopies to become their own database programmers. The underlying mechanism that renders programming power to the ecologists as end users is software components drawn from commonly used domain-specific conceptual structures. The idea is that researchers would select database components that match their conceptual view of entities of interest in the research design. Once components, which we call *templates*, are selected by users, integrity rules contained within each template are used to generate a database design. We believe that such databases would be easily integrated, since there are likely to be common join points in two databases designed from templates that reference semantically similar concepts. Thus, the system would offer both physical connectivity and agreement at the semantic level, two critical aspects of integration (Maier et al., 1993). A companion visualization prototype, *CanopyView*, provides productivity benefits, i.e., incentives for ecologists to use *DataBank*. Although still a research prototype, *Canopy DataBank* shows promise for designing, documenting, archiving, and mining field databases.

This paper is organized as follows: In Section 2, we present background information and related work. Section 3 describes the Canopy Database Project. Section 4 is the heart of the paper, in which we describe *DataBank*. We first present our vision of end user database programming for ecologists, then list the functional requirements for *DataBank*, define and give examples of templates, and describe the implementation. In Section 5, we discuss

ancillary software tools that become practical and affordable because of the use of software components. We describe one such system, our visualization tool *CanopyView*. We conclude the paper with Section 6, which offers conclusions and outlines future work. Finally, Section 7 acknowledges funding sources and collaborators.

## 2 Background

In this section, we identify two major obstacles to the use of information technology by ecologists and hypothesize that database systems could help overcome these. Since our proposed solution to these problems involves end-user database programming that re-uses spatial database components, we then describe current software development practices that enhance developer capabilities or that enable programming by end users. In conclusion, we relate other ecosystem informatics work to our own, and note how some current information technology research relates to these efforts.

*Obstacles to ecologists' use of information technology* Although ecologists often consult web-accessible information, they typically enter field data into individualized, private data stores. Few of these data sets are published by ecologists, in spite of increasing pressure from funding agencies to do so.<sup>1</sup> Other motivating factors for publishing ecological data include: the availability of excellent ecological data archives (<http://www.lternet.edu>; <http://www.ecoinformatics.org>; <http://data.esa.org/>), emerging tools for recording metadata (Nottrott, Jones, & Schildhauer, 1999), and some recent evidence that applying additional data to studies might change the results (Dunne, Martinez, & Williams, 2005). The primary reason our ecology collaborators give for not publishing data sets is that adequately documenting data for archiving is a time-consuming process and that even the best documentation is inadequate for using data in new contexts or in different disciplines (Michener, Brunt, Helly, Kirchner, & Stafford, 1997; Spycher, Cushing, Henshaw, Stafford, & Nadkarni, 1996).

We realize that sociological reasons for not publishing data sets are significant, and include the following: (1) Some researchers hold data until they have gleaned all useful value from them, and thus publish certain data sets only after many years, or never. (2) Misunderstandings abound on the part of data users about how to cite referenced data, even though they clearly understand how to reference research published in academic journals. (3) Many researchers fear that data users will misinterpret their data. (4) In the case of highly processed data such as that collected with remote sensing devices, it is unclear whether all raw data should be published, or “cleaned” primary data, or only aggregate (summary) data values. Sociological issues fall outside the scope of this paper.

Without a critical mass of archived data sets, ecological data integration and data mining cannot be effective. Database technology would help overcome technical obstacles to data archiving if applied at all stages of the research cycle. However, while the LTER Archives and other data publishers make excellent use of sophisticated database technology, individual ecologists typically lack the expertise or inclination to use current database technology (although some are good programmers and use sophisticated statistical programs and GIS), or write complex mathematical models (Michener & Brunt, 2001;

<sup>1</sup> See, for example, the NSF LTER Coweeta Site Data Policy Statement, wherein data are either published within three years of collection, or within three years of completion of a study. (<http://coweeta.ecology.uga.edu/webdocs/3/static/datapolicies.html>)

Michener, Porter, & Stafford, 1998). It is not cost effective or even practical for ecologists to hire programmers to design, implement and maintain databases for field data sets.

Even with accurately documented ecological data sets, however, data integration would be difficult without a controlled vocabulary, common data structures or schema, and database support for semantic metadata. This is because ecologists use idiosyncratic database designs and data models, and there is understandably high variability among and within the many ecological fields of research with regards to protocols, vocabulary, and conceptualization (Romanello, Beach, Bowers, Jones, Ludäscher, & Michener et al., 2005). Higher concordance among the target data sets would help, but lack of database support for integrating metadata closely with the data makes integrating such semantically variable data difficult (Jagadish & Olken, 2003).

*Related research* No one solution to the above problems will work for all ecologists, so current ecosystems informatics research runs a broad spectrum. Some consensus exists, however, that database technology might give ecologists productivity benefits similar to those the banking and airline industries enjoyed in the 1980s and 1990s. One major difference between today's ecologists and those industries, however, is that the industries were able to engage highly paid programmers to write the systems that brought about productivity increases. Because ecology data sets typically support small-scale, single- or several- researcher investigations, software tools for end users, or engineering practices that empower non-programmers is also related to our work. Because our work involves using components and enabling end-user programming, we focus on that research.

Programming with components (Szyperski, 1997), domain-specific languages (Hook & Widen, 1998; Kiebertz, 2000; Sheard, 2001; Sheard & Jones, 2002), and design patterns (Gamma, Helm, Johnson, & Vlissides, 1995) are three ways that improve programmer productivity and product reliability. Because the essence of software is a highly abstract construct of interlocking concepts (data sets, relationships among data items, algorithms, and invocations of functions), the difficult part of writing software is the specification, design, and testing of this conceptual construct, not the labor of presenting it and testing the fidelity of the representation (Brooks, 1995; Sowa, 1984). Thus, the practices on which we capitalize are promising because they attack "the conceptual essence of programming" (Brooks, 1995).

Given the above observations about software engineering, one source of software failure is that programmers are tasked with solving problems in domains outside their area of expertise. Any successful software product must capture real-world requirements and cast these into precise and unambiguous terms into an effective software design. This accounts for the plethora of advice to software engineers on how to capture requirements and produce software that actually does what the user needs (Gause & Weinberg, 1989). Use cases and the Unified Modeling Language (UML) (Fowler & Scott, 1997), and the close involvement of end users promulgated by eXtreme Programming (Beck, 2000), even as applied to scientific software development (Wood & Kleb, 2003), are cases in point.

To abrogate the need for application programmers to understand the application domain, one could empower end users to do at least some of their own programming. The near universal popularity of spreadsheet technology in the 1990s suggests that many people (including scientists) want to and can do significant programming using such a tool. The number of spreadsheet programmers far outnumbers those who program using traditional programming languages (Peyton-Jones, 2003). In spite of this, only a few efforts address

how to apply modern programming language research to improve spreadsheet programs (Burnett, Atwood, Djang, Gottfried, Reichwein, & Yang, 2001).

Several ecosystem informatics projects aim to alleviate informatics obstacles to ecological research. For example, the Science Environment for Ecological Knowledge (SEEK) project is complementary to ours, but has broader scope and focuses more on building powerful tools than on end-user programming. Further we focus on one subdiscipline of ecology (forest canopy studies), while SEEK aims to build tools for the entire discipline of ecology. Finally, their semantic integration work focuses more closely on published data sources, while we use database technology to improve individual researcher productivity and then build on that technology to improve data integration within a constrained domain.

Tisdale-Beard's work with event and process tagging is similar to ours in that she focuses on particular data structures, but her focus is on ecological events while ours is on structures of real-world ecological entities (Beard-Tisdale, Kahl, Pettigrew, Hunter, & Lutz, 2003). Henebry focuses on a slightly different ecology data structure than we do, namely temporal geospatial representations (Henebry & Merchant, 2001). Villa's efforts to formally define common ecological concepts is similar to our efforts to define structural concepts for one sub discipline of ecology, but his architecture uses a semantic layer for data integration, while ours assumes the data are composed of similar semantic components (Villa, 2001). Kennedy's data structures for taxonomic classification are tangentially related to ours in that she provides extensions to existing database technology to handle taxonomic data, and articulates what additional database research is required before ecosystem informatics objectives can be met (Raguenaud & Kennedy, 2002). Finally, much current ecosystem informatics research cited above (including ours), and the LTER research sites, use a dialect of XML (<http://www.w3.org/XML>), the ecological markup language (EML) (Nottrott et al., 1999) to represent metadata.

Mainstream database research most relevant to our own is that which explores the use of components to generate databases (Wang, Liu, & Kerridge, 2003), explorations of database primitives (Stemple & Sheard, 1991), use of structural metadata to integrate different database schema (Bernstein & Rahm, 2000), and superimposing semantic metadata on existing documents (Delcambre, Maien, Weaver, Shapiro, & Cushing, 2003; Weaver, Delcambre & Maier, 2001). Building ontologies and technology to manage those ontologies document a particular discipline's use of word; all this is relevant to how one might relate terms in different databases (Gruber, 1993; Musen et al., 2000).

### 3 The Canopy Database Project

We chose canopy science as a representative subset of ecology, but our results are applicable to the field of ecology (Lowman & Nadkarni, 1995). We initiated The Canopy Database Project in 1993, with a survey of 200+ canopy researchers to identify obstacles to forest canopy research (Nadkarni & Parker, 1994). We had expected researchers to cite physical access to the forest canopy as the greatest obstacle to research because that has traditionally been the greatest problem for them. Our survey results, however, showed that canopy researchers now see problems in management, use, and sharing of data as more critical. Partially as a result of this survey, we brought together ecologists and computer scientists in a three-day workshop to articulate why and how researchers wanted to share

data (Nadkarni & Cushing, 2001; Cushing et al., 2002a, b), and began The Canopy Database Project to help solve some of the specific problems articulated in that workshop.

The Canopy Database Project aims to develop tools to facilitate data acquisition, management, analysis and exchange relating to canopy studies at all research stages. We also seek to document and publish data sets that can be of direct use to this research community and that incidentally also demonstrate use of our tools, characterize and formalize fundamental structures of the forest canopy, and relate those structures to forest functions as appropriate for retrospective, comparative, and integrative studies. We wish to help scientists increase research productivity, simplify sharing data with close collaborators, and facilitate data archiving. We believe that metadata acquisition can become a natural byproduct of the research process.

Our approach has been to develop databases for one extensive structure-function study (Van Pelt & Nadkarni, 2004), build prototype informatics tools that enable ecologists to easily design and manage field databases, and extend our work to other field studies and other subfields of ecology. We organize the development effort around three foci:

1. *Informatics Tools and Information Artifacts for Canopy Scientists.* We have three prototypes: (1) a database design and warehouse tool (*DataBank*); (2) a visualization tool (*CanopyView*); (3) an Internet reference site for canopy research, the *Big Canopy Database* (BCD). *DataBank* and *CanopyView* are described in Sections 4 and 5 of this paper. The BCD is at this point peripheral to the major functions of *DataBank*, serving other research needs by providing information of interest to forest canopy researchers during the research initiation phase and as a place to post citations and images. The BCD includes as of January 2007, 7400+ searchable scientific citations, downloadable color images with associated forest canopy locale and species identification; and other useful materials such as a glossary of canopy research terms, website links, new publications, meeting notices, canopy access techniques, climbing safety protocols, and public interest references. It was implemented in MS SQL Server, HTML and Active Server Pages (ASP), but as of January 2007, has been rewritten in .NET (<http://canopy.evergreen.edu/BCD>).
2. *Data acquisition in the field and database development for collaborating ecologists.* We both conduct our own fieldwork and collaborate with ecologists outside our team. Our fieldwork involves forest structure and function for eight sites in a 1,000-year chronosequence (ranging from 50 to 950 years) in the temperate coniferous forests of the western Cascades in Washington State. These sites represent a wide range of forest structural diversity, from simple (e.g., young monospecific coniferous plantations) to complex (e.g., diverse old-growth temperate coniferous forests). The initial study concentrated on within-tree and within-stand structural measurements (e.g., tree stems, crowns, branches, and interstitial space), but also measured functional characteristics (specifically throughfall and light) (Van Pelt & Nadkarni, 2004). We believe it is critical in our case to conduct ecology research conjointly with the eco-informatics effort, but we have also collaborated with 11 ecologists to render their data sets into *DataBank*.
3. *Conceptual and theoretical ecology.* Developing *DataBank* templates from a few ecological studies has shown that we need general conceptual structures for representing forest structure. We are thus formulating generalized spatial categories of the forest canopy and have developed a preliminary spatial characterization model, i.e., a “universe” of possible spatial models that define forest structure. We aim to use these to further refine *DataBank*'s templates.



## 4 Canopy DataBank

*DataBank* presents our vision of how future ecologists would want to design, archive, and mine field databases if database technology were integrated early into the research cycle. *DataBank* is a software prototype that uses the forest canopy structure representations that we call *templates* as domain-specific database components. We envision a tool that harnesses the power of database technology, with some of the ease of use of spreadsheets, so that an ecologist could be his or her own programmer. The idea is that a researcher selects database components that fit his or her conceptualization of the research problem, and automatically combines those components into a database that a researcher can download and populate. The resulting field database is built on components of which other tools (data-entry-form generation, data validation, visualization, analysis, etc.) are cognizant.

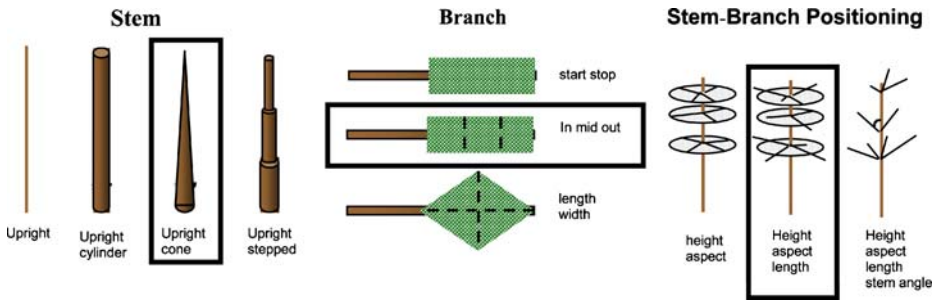
*DataBank* is a database design tool, and a data and metadata repository for canopy research projects (<http://scidb.evergreen.edu/databank/studycenter/>). Its metadata model is inspired by that at the H. J. Andrews LTER Site repository (<http://www.fsl.orst.edu/lter>). Security and privacy features allow researcher flexibility to: (1) publish metadata only (no field data), (2) make field data available only to selected colleagues, or (3) make data viewable but not downloadable. Scientists can also forbid release of personal information.

Figure 1 shows our conception of how a researcher might use templates to create a field database. The example study has three entities of interest (stem, branch, and foliage), each of which has several typical structural representations, e.g., *stem as cone with height and DBH*. The researcher first selects templates that best match the study (selected templates are enclosed in a black rectangle in the figure), and requests that a database be generated.

A key assumption underlying our work is that individual components can be extended and combined in many different ways, which means that templated databases offer greater flexibility than monolithic data models. Furthermore, ecological data are inherently spatial and most research involves making observations about structural elements, which are less likely to differ over time or between different studies, and which can be used as join points. If common and interchangeable representations of spatial data and coherent conceptualization of ecological structural elements are the components upon which databases are built, those databases can be more easily managed, provide metadata, allow the development of tools that can be used on many databases, and provide common variables over which disparate data sets can be joined. Functional data can be added in an ad hoc manner, and not affect the power of structure-based templates to provide the obvious benefits of common components.

For *DataBank* to be successful, we must provide software that supports research activities. We have begun to build software that generates data-entry forms and performs visualization, and will create other tools, building on other open source software developed by researchers in data integration (Miller, Haas, & Hernandez, 2000; Miller et al., 2001) and ecosystem informatics such as EML, Metacat, Morpho, SEEK, (<http://ecoinformatics.org/tools.html>). Metadata generation and maintenance will be addressed in a future separate tool, and is outside the scope of this paper.

*DataBank functional requirements* *DataBank* has three major functional requirements: field database design, field data repository, and data mining. We have focused our efforts to date on the field database design, temporarily setting aside issues such as metadata maintenance, analysis tools for the scientist, integration and validation of the database into an integrated archive, and cross-study data queries, as we focus on building the capability to design databases, and proof of the ability to build tools that can take as input a database built in *DataBank* (i.e., from templates).



**Fig. 1** Conceptual view of how a user might select components for database design

To design and generate a field database, a researcher briefly describes the associated ecological study, then designs the study's database. In some cases, a *DataBank* archivist might first define a larger project (a project can contain one or more studies). To describe the study, one registers its Principal Investigator (PI) and creates a new study with that researcher as PI and a second person as project archivist. The PI or archivist then adds preliminary metadata and other researchers to the study. The study's database is designed by clicking data templates and associating reference material (e.g., species, vegetation type) as source tables. The database designer can view template details, the database design, and add or modify attributes. The database is generated as SQL or an Access database. In the future, we will allow provision of attribute-level metadata such as range, and pre-populating the database with other site or study data. A populated field database can be uploaded to the repository, and checked against the templates for schema differences; this is currently done by hand. Validation against metadata will be accomplished via software and services from an LTER data center.

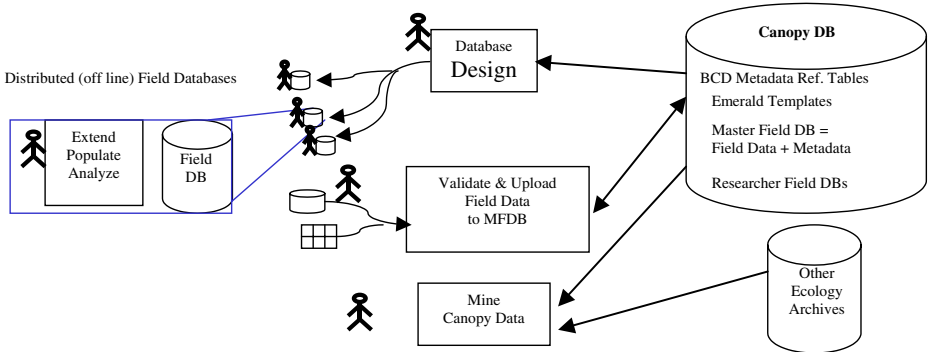
The *DataBank Warehouse, Study Center* integrates individual field studies into a single warehouse. Data in the warehouse can currently be queried using only simple study-level metadata (e.g., “find all studies at the Ohanepecosh site, undertaken after 1995” or “find all studies conducted at the Wind River Canopy Crane Research Facility by Robert Van Pelt”). The identified field databases can be viewed and downloaded, if the user has appropriate permission.

To integrate databases created with known components, we envision a user first identifying templates of interest, and then running queries generated from that schema. Because templates are organized hierarchically, and transformations between templated databases can be preprogrammed, *DataBank* will be able to identify databases generated with more specific templates and populate the working data set. Because templates are metadata “aware,” metadata could also be made relevant to a query. An example of a cross-study query is: “find the average diameter for trees with height greater than 20 m,” or “find the average diameter for trees with height greater than 20 m for studies in the US Pacific Northwest.”

Figure 2 shows how we envision individual researchers interacting with our software. A user first designs a database, then uses that database in the field (perhaps extending the design), populating the database and analyzing the data. Once ready for publication, the data and its documentation are validated and uploaded to a common repository (*Study Center*), where it can be mined, potentially in conjunction with other ecological archives.

*DataBank* complements, but does not replace, existing archives such as those of the Wind River Canopy Crane Research Facility and the LTER. It differs from the former database in that it spans several sites. It differs from the latter in that it specializes services





**Fig. 2** How ecologists would use *DataBank*'s software suite. Note: As this article goes to press in 2007, the *Master Field DB (MFDB)* has been renamed *Study Center*, and *Emerald Templates* are known as *templates*

for one community and provides help in research design, and that it is not meant as a long-term ecology archive. Thus, for example, citations are not limited to projects whose data are stored in *DataBank*, but are meant as community-wide references. Because we comply with metadata requirements for data deposition at LTER sites, those who archive in *DataBank* could easily archive at an LTER site.

*DataBank templates* A template represents data collected when measuring a particular physical object in the real world, e.g., a tree or branch, and appears to users as a conceptual database primitive—a domain-specific data type. Templates usually have absolute or relative spatial attributes. To a computer scientist, templates are collections of variables, each grouped as one or more relational tables, that can be composed into an end-user defined database. When more than one database table is generated, appropriate relationships between the tables are induced. Templates now carry some metadata, transparently to the end user, that can later be exploited for data visualization. In future implementations, we plan to expand template metadata so that it can be exploited for data validation, form building, validation and data mining. We have designed templates for research place (e.g., site, plot), stem (tree), branch, ground cover, and various functional observations (e.g., rainfall, light). Figure 3 shows how information about a particular template, in this case “branch with foliage measurement” is shown to a user as he or she designs a database.

Figure 4 gives a conceptual view of how a user might use several templates to create a database design. In this example, a researcher is collecting field data about epiphytes (canopy-dwelling plants) as percent cover per branch-quadrat for each of several epiphyte species. To build a database for this study, he or she would use a branch-quadrat template. Data collected for each quadrat are epiphyte species and percent cover on the quadrat for that species, as well as the date of the observation. Since the data of interest are located on a tree branch, the researcher must also collect information about branches and trees, and hence would include a branch- template and a tree- template. Because each tree must be located in space and that space must be described, the researcher must include plot- and site- (location) templates.

Only attributes lettered in non-italicized black are part of the template that the user sees; i.e., the user sees only information relevant to the conceptual idea of the study, not to database design. Thus, studyID appears in every table, but is shown in italics because, though the system carries information about what study the table “belongs to,” this information is hidden. Similarly, foreign keys that reflect the full location hierarchy (shown in grey) will appear in the interface, but do not appear in the template conceptual view or in

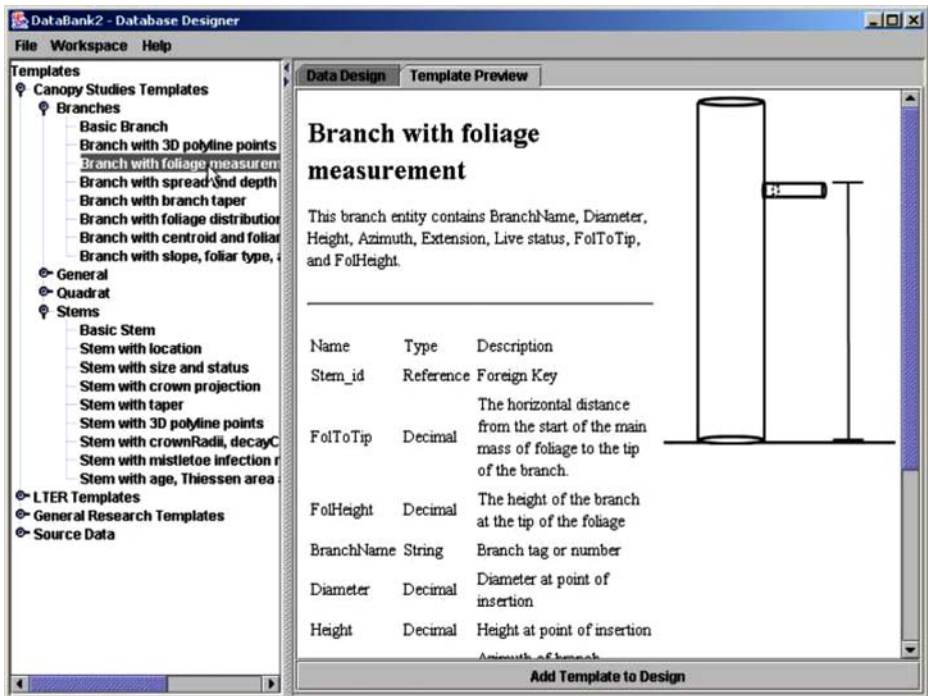


Fig. 3 Template information available to user during the design process

the normalized database. Only the foreign key of the location immediately “above” a specific table is included in that table. Primary keys (underlined) are shown in the template conceptual view. The tree\_Obs entity is bordered with dashed lines because it is not a template, but an observation (Thiessen Area) explicitly defined by the user. Such simple data elements are easy to add to a database design, and if such measurements are measured only once, they are included as variables in the parent table. If they are taken more than once, then a separate table is generated and a date attribute added; in this case one row in each such table represents a separate measurement.

*DataBank implementation* The database design part of our system has been implemented, and we have some experience now with developing and using templates. The prototype is currently implemented in Microsoft SQL Server, Java, JDBC, and Enhydra. The system currently allows creation of a database from a few prototype templates and the download of that database. Any SQL database, or even a spreadsheet, could be generated from the database design, but we currently generate MS Access databases because MS Access is widely available to our users. Figure 5 shows an MS Access database created from a *DataBank* template design similar to the example used above. We have separate metadata documentation tools (in Access and Excel) and a simple web-accessible browser for viewing study data that have been uploaded.

Figure 6 shows the *DataBank* architecture and how the above system components relate to one another. A template is represented as an XML document (template.xml), with small and large icons representing the template (pic.gif, bigpc.gif). As a user designs a database, templates are placed into a database design (using a shopping cart metaphor). The system

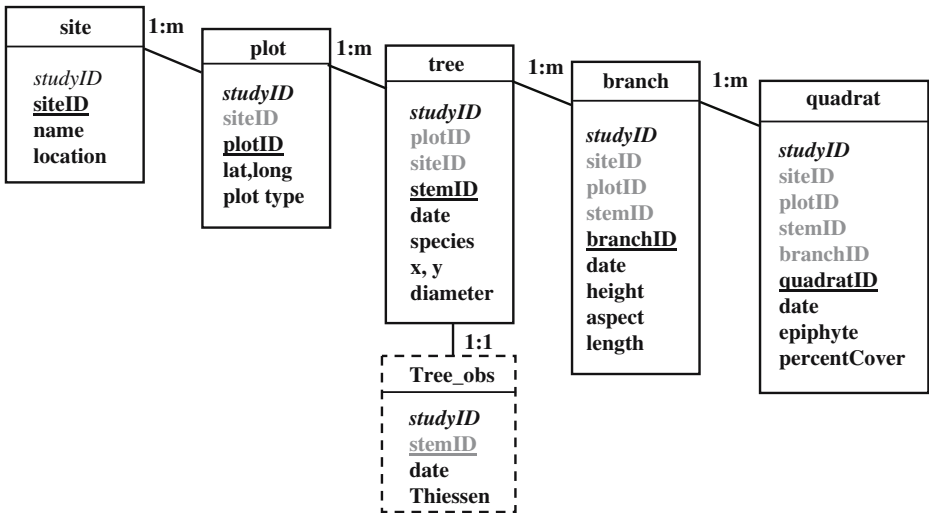
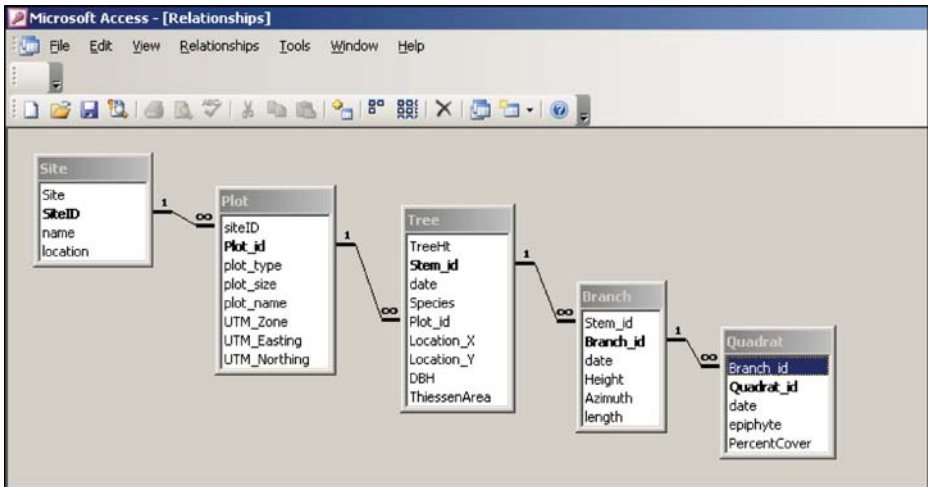


Fig. 4 Templates for plot, site, stem, branch and quadrat

component *Template Entity Observation Framework (TEOF)* combines templates into an internal representation that checks dependencies for a particular template, e.g., a database that includes a tree should also have a species table. The TEOF representation is made persistent (DB Design) so that a user’s design session can span several sessions and designs can be stored for later reference or reuse. The *Template Database Model (TDM)* converts a TEOF design into an SQL dialect and generates a database that can be downloaded to the user.

We identify three implementation issues: template representation, development platform, and how to represent normalized database designs and databases to end users.

- (1) Templates are currently represented as XML documents, which include annotations about how elements are grouped into tables and how those tables are related. A collection of XML elements is mapped to Java objects and then to SQL tables and relationships. As we articulate templates, we also specify rules on how to compose them into database designs for field databases. We seek a more comprehensive representation of templates that will include integrity constraints, data validation rules, and parameterized scripts.
- (2) In our first prototype, we found HTML/ASP/SQL Server technology too brittle for a flexible user interface. We chose HTML/Java/Enhydra/SQL Server for the second version, but the web-accessibility requirement still limits user interface functionality. Our third implementation (completed in 2005) separates the data generation and data warehousing functions, with the former implemented as a downloadable Java application. While this facilitates database design changes in the field, a web-accessible interface would be preferable so we could capture database and template designs and metadata as they are created by users, and so that users would not have to download new versions of the software and new templates as the libraries are updated.
- (3) The appropriate level of abstraction for presenting templates and the subsequent database design to the end user remains an open question. Database entities are more abstract, with greater normalization, than most researchers prefer. Thus, for example, while recursive entities offer great flexibility, they are difficult for end users to understand.



**Fig. 5** A database designed from plot, site, stem, branch and quadrat templates

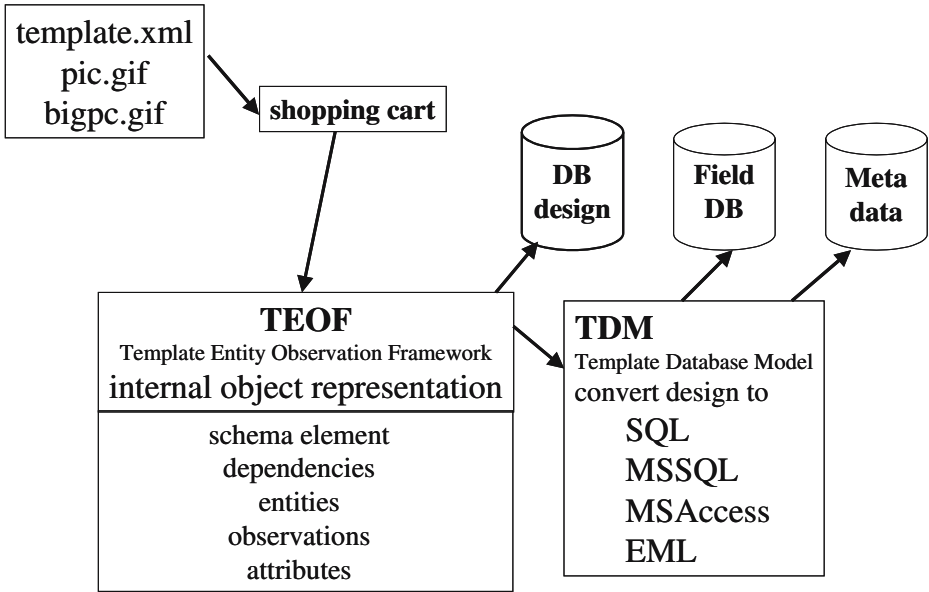
The above implementation issues are complicated by conceptual issues that have arisen during implementation. For example, we ask whether templates should essentially foster top-down or bottom-up database design, or design by example. Efforts to define sufficiently general database templates that would be useful to a significant number of scientists has led to a theoretical ecology effort to categorize the ecological structures with which canopy researchers work. We also realize that ecologists will need to modify database designs in the field, after the database itself has been at least partially populated, and so are faced with issues of how the tools should deal with such on the fly changes.

The *DataBank* concept and software has been tested by producing templates and subsequent databases for the 12 ecological studies as described in Section 3, with several publicly available from *DataBank*. The current system implementation can be seen at <http://canopy.evergreen.edu/databank> and a new user interface design can be seen at [http://scidb.evergreen.edu/databank/databank\\_docs\\_02062005.pdf](http://scidb.evergreen.edu/databank/databank_docs_02062005.pdf).

## 5 Ancillary software tools

To increase the utility of *DataBank* and demonstrate the effectiveness of using templated databases, we developed a visualization application called *CanopyView* (Cushing, Nadkarni, Finch, & Kim, 2003; Finch, 2003). *CanopyView* is implemented in Java, using the Visualization Toolkit (VTK) (Schroeder, Martin & Lorensen, 1998). With *CanopyView*, we can build useful visualizations of templated databases by carrying information from templates into the application, with the application reading templated databases as input (Fig. 7).

Similar to *DataBank*'s reuse of common data structures (templates), *CanopyView* implements reusable visualization modules. These modules typically represent physical entities (e.g., stems, plots) and the functional observations on those entities (e.g., percent cover, infection ratings). Certain templates in *DataBank* have visualization modules associated with them. *CanopyView* reads a *DataBank* generated Access database as input and then loads the associated visualization components, including a user interface



**Fig. 6** The *Canopy DataBank* software architecture

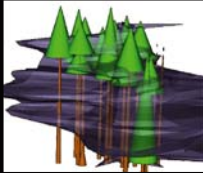
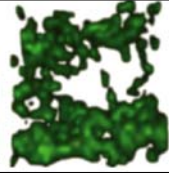
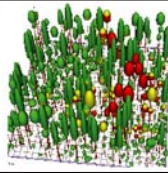
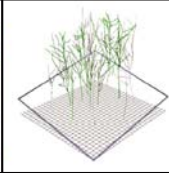
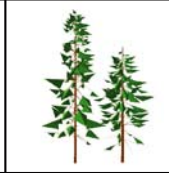
component to manage the visualization components. When the user requests a particular visualization, the system builds a 3D scene. The entities to be added to the visualization scene can be selected from the GUI or via a SQL query. *CanopyView* then extracts the selected data from the database and inserts them into a 3D scene. Using this method, multiple databases can be opened at once and visually combined into the same scene. *CanopyView* has been tested with several field data sets.

As it stands now, visualization modules are implemented specifically for certain template “families.” Because the templates are hierarchically organized, and the application generally works at higher level of abstraction, changes to templates rarely affect the tool. In spite of this, the tool is susceptible to changes in templates, and more work is needed to express the mapping from *DataBank* templates to *CanopyView* modules and further develop the path between these two applications.

Just as *CanopyView* capitalizes on the fact that it “knows” the components of a database used as input to it, other tools could similarly and perhaps even more easily be produced that “write programs” for the ecologist and increase his or her productivity. For example, because field data intake forms for databases created using the same templates will be very similar, parameterized SQL queries or (in the case of MS Access) Visual Basic Code, embedded in templates, can easily be included in *DataBank* to generate intake forms. Such forms would look like those created by an ecologist using spreadsheets, and we have produced a proof of concept of this feature.

## 6 Conclusions and future work

Our work to date has shown the technical promise of using a relatively small number of domain-specific data structures (templates) to easily construct individualized field data-

				
Canopy airspace (grey) overlain with stems at Martha Creek in southern Washington.	Surface area density map of the canopy of an eastern deciduous forest at Smithsonian Environmental Resource Center.	Dwarf mistletoe infection ratings in an old-growth Douglas-fir forest, Washington.	Polyline representations of <i>Castanea crenata</i> , Japan.	Full stem reconstructions at the Trout Creek site in southern Washington state.

**Fig. 7** Sample visualizations of *DataBank* data sets, built with *CanopyView*

bases. Several such databases would be more easily comparable than idiosyncratically designed databases, and such a system is likely more practical than using global schemas. If end users can design effective and comparable databases using templates, then productivity gains in their research process, as well as easier data archiving and data mining, should ensue. We have shown that templated databases also facilitate tools for visualization and we believe that other tools for field data management and data analysis would significantly contribute to increased research productivity. Future development includes more general and easier-to-manage templates, refining the current implementation, adding productivity features, and implementing a data warehouse and data mining facility.

The existing software is a research prototype not a tool, and databases are currently created by trained informatics staff, not end users (ecologists). Now that we have created a proof of concept, we aim to refine the prototype to include a more usable interface and tighter integration of associated tools.

Templates are currently hand-crafted in XML, which is prone to error even when written by a specialist trained in XML syntax. We have built a template editor and manager, but it has not yet been user-tested so that we are confident that domain researchers and information managers can build, test and publish their own templates. Also, the templates we have devised are reverse-engineered from existing data sets. Although they cover stable structural concepts, we cannot demonstrate that the templates we have now will cover future studies. While the articulation of templates is a domain-centered task, we recognize that this must now be accomplished jointly between ecologists and computer (or information) scientists, and we have mounted a project to build theoretically justifiable canopy conceptual structures. A general observation template would define generic forest functional observations such as light, temperature, and percent interception of rainfall at a particular location or on a forest entity. Our refinement of new and existing tools fall into four categories:

- (1) Our visualization tool *CanopyView* needs to be extended to include more generalized visualizations so that data for the new templates can be visualized and data from several studies can be more easily queried and viewed on one canvas. We also aim to make this tool more robust in the face of changes to data templates.
- (2) For field data entry, we have devised queries and forms in an ad hoc manner that allow researchers to more efficiently use templated databases - as they now use



spreadsheets. These queries and forms display enough generality that we believe we can build parameterized scripts that could be specialized for particular researcher databases. Similarly, we could generate databases specifically for palm top computer, or remote or in situ sensors.

- (3) To provide metadata, we have built a rudimentary tool, and in the future prefer to use one of the emerging standards (EML, Morpho). We also aim to associate templates with an ontology that could later be used with templated databases and their associated metadata for data interpretation and integration. We also aim to maintain some metadata automatically.
- (4) There are currently no built-in statistical scripts or pre-programmed and parameterized aggregate computations. Most researchers import data into statistical packages for such calculations, but we could provide some help to them by providing automatic export to a commonly used statistical package, along with parameterized scripts to run the most common statistical calculations. We hope to build these, working with a statistical consultant and our ecologist collaborators on the analysis of their data sets and subsequent generalization of those analyses into parameterized scripts embedded in particular templates.

Other more distant objectives are to support ecological synthesis. This would require better support for data warehousing, transforming databases from one form to another (e.g., 3-D to 2-D), or populating new databases from existing ones, and data integration. We currently have no declarative programming capability to accomplish this, but a schema mapping tool such as CLIO (Miller et al., 2001) would allow users to link data elements in separate databases and automatically load one database from another. For domain-specific warehouse and data mining, we will use existing tools (Metacat & Morpho, 2003) and share tools with LTER Information Management colleagues.

At least as daunting as the technical challenges we face are the sociological issues that arise as one aims to introduce new technology to end users. We identify three such issues: establishing a critical mass of users and data (including templates), recruiting and training volunteer curators, and finding long term funding for the operational system.

Although increasing researcher productivity is a necessary condition for ecologists to use database tools, it may not be sufficient. Integrating systems such as those we propose into the ecological research cycle will involve changes in the way ecology is practiced, and rewards for archiving data sets are not yet generally perceived. Although such sociological changes are beyond the scope of this project, our work has suggested that both ecologists and computer scientists will be change agents and will have to articulate and work together towards establishing rewards for data archiving and integrative ecology, as technical advances are introduced into the scientific arena.

**Acknowledgments** We acknowledge many contributors, including former staff members Erik Ordway, Steven Rentmeester and Abraham Svoboda, and students Youngmi Kim, James Tucker, Brook Hatch, Neil Honomichl, and Peter Boonekamp. We are thankful for the participation of LTER information experts James Brunt, Don Henshaw, Nicole Kaplan, Eda Melendez, Ken Ramsey, Gody Spycher, Susan Stafford, and Kristin Vanderbilt; consultants Bonnie Moonchild and Jay Turner, and computer scientists Eric Simon, Dennis Shasha, and Phil Bernstein. Many field researchers contributed data and advice, including Barbara Bond, Roman Dial, Hiroaki Ishii, Elizabeth Lyons, David Shaw, Steve Sillett, Akihiro Sumida, and Robert Van Pelt.

This work has been supported by the National Science Foundation grants and Research Experience for Undergraduate Supplements: BDI 04-17311, 03-019309, 99-75510; BIR 96-30316, 93-00771; INT 99-81531.

## References

- Beard-Tisdale, K., Kahl, J. S., Pettigrew, N., Hunter, M., & Lutz, M. (2003). BDEI: Event and process tagging for information integration for the international gulf of maine watershed. In *NSF Workshop on Biodiversity & Ecosystem Informatics*. Olympia, WA.
- Beck, K. (2000). *Extreme programming explained*. Boston, MA: Addison Wesley.
- Bernstein, P. A., & Rahm, E. (2000). Data warehouse scenarios for model management. In *ER2000 conference proceedings* (pp. 1–15). Salt Lake City, UT: Springer.
- Brooks, F. P. J. (1995). No silver bullet—essence and accident in software engineering. In F. P. Jr. Brooks (Ed.), *The mythical man-month anniversary edition*. Reading, MA: Addison Wesley.
- Burnett, M., Atwood, J., Djang, R. W., Gottfried, H., Reichwein, J., & Yang, S. (2001). Forms/3: A first-order visual language to explore the boundaries of the spreadsheet paradigm. *Journal of Functional Programming*, *11*, 155–206.
- Cushing, J. B., Nadkarni, N. M., Delcambre, L., Healy, K., Maier, D., & Ordway, E. (2002a). The development of databases and database tools for forest canopy researchers: a model for database enhancement in the ecological sciences. In *SSGRR2002W*, L'Aquila, Italy.
- Cushing, J. B., Nadkarni, N. M., Delcambre, L., Healy, K., Maier, D., & Ordway, E. (2002b). Template-driven end-user ecological database design. In *SCI2002*. Orlando, FL.
- Cushing, J. B., Nadkarni, N. M., Finch, M., & Kim, Y. (2003). The canopy database project: Component-driven database design and visualization for ecologists. In Poster. *VIS 2003*. Seattle, WA.
- Cushing, J. B., & Wilson, T. (July 2005). Eco-Informatics for Decision Makers—Advancing a Research Agenda. *Invited paper, 2nd international workshop on data integration in the life sciences*. In L. Raschid, & B. Ludaescher (Eds.). San Diego, CA.
- Delcambre, L., Maier, D., Weaver, M., Shapiro, L., & Cushing, J. B. (2003). Superimposing spatial enrichments in traditional information. In *International workshop on next generation geospatial information*. Cambridge (Boston), MA.
- Dunne, J. (2005). Emerging ecoinformatic tools and accomplishments for synthetic ecological research across scales. *Ecological Society of America Annual Meeting*, August 7–12. Session presenters: J. Cushing, M. Weiser, J. Alroy, M. Jones, J. Quinn, N. Martinez, J. Dunne, and U. Brose.
- Dunne, J., Martinez, N., & Williams, R. (2005). Webs on the web: Ecoinformatic approaches to synthetic food-web research from cambrian to contemporary ecosystems. In emerging ecoinformatic tools and accomplishments for synthetic ecological research across scales. *Ecological Society of America Annual Meeting*, August 7–12.
- Finch, M. *The canopy database project: Component-driven database design and visualization for ecologists*. In Demonstration. *VIS 2003*. Seattle, WA.
- Fowler, M., & Scott, K. (1997). *UML distilled*. Reading, MA: Addison-Wesley.
- Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995). *Design patterns*. Boston, MA: Addison Wesley.
- Gause, D. C., & Weinberg, G. M. (1989). *Exploring requirements*. New York: Dorset House.
- Gruber, T. R. (1993). A translation approach to portable ontologies. *Knowledge Acquisition*, *5*, 199–220.
- Henebry, G. M., & Merchant, J. W. (2001). Geospatial data in time: limits and prospects for predicting species occurrences. In J. M. Scott, P. J. Heglund, & M. Morrison (Eds.), *Predicting species occurrences: issues of scale and accuracy*. Covello, CA: Island.
- Hook, J., & Widen, T. (1998). Software design automation: Language design in the context of domain engineering. In *Proceedings of SEE '98*. San Francisco, CA.
- Jagadish, H. V., Olken, F., et al. (2003). NSF/NLM workshop on data management for molecular and cell biology, report data management for the biosciences. *OMICS: A Journal of Integrative Biology* *7*, 1.
- Kieburz, R. (2000). Defining and implementing closed domain-specific languages. OGI Technical Report [http://www-internal.cse.ogi.edu/PacSoft/publications/phaseiiiq13papers/design\\_and\\_impl.pdf](http://www-internal.cse.ogi.edu/PacSoft/publications/phaseiiiq13papers/design_and_impl.pdf).
- Lowman, M. D., & Nadkarni, N. M. (1995). *Forest canopies*. San Diego, CA: Academic.
- Maier, D., Cushing, J. B., Hansen, D. M., Purvis III, G. D., Bair, R. A., DeVaney, D. M., et al. (1993). Object data models for shared molecular structures. In R. Lysakowski (Ed.), *First international symposium on computerized chemical data standards: databases, data interchange, and information systems*. Atlanta, GA: ASTM.
- Maier, D., Landis, E., Frondorf, A., Silverschatz, A., Schnase, J., & Cushing, J. B. (2001). Report of an NSF, USGS, NASA workshop on biodiversity and ecosystem informatics. <http://www.evergreen.edu/bdei/2001/>
- Metacat, & Morpho (2003). <http://knb.ecoinformatics.org/software/>.
- Michener, W., & Brunt, J. (Eds.) (2001). *Ecological data-design, management and processing*. Blackwell Science Methods in Ecology Series.

- Michener, W., Brunt, J., Helly, J., Kirchner, T., & Stafford, S. (1997). Non-spatial metadata for the ecological sciences. *Ecological Applications*, 7, 330–342.
- Michener, W., Porter, J. H., & Stafford, S. (Eds.) (1998). *Data and information management in the ecological sciences: a resource guide*. Albuquerque, NM: LTER Network Office, University of New Mexico.
- Miller, R. J., Haas, L. M., & Hernandez, M. (2000). Schema mapping as query discovery. In *Proceedings of the international conference on very large Data bases (VLDB)* (pp. 77–88). Cairo, Egypt.
- Miller, R. J., Hernandez, M. A., Haas, L. M., Yan, L., Ho, C. T. H., Fagin, R., et al. (2001). The clio project: Managing heterogeneity. *SIGMOD Record*, 30, 78–83.
- Musen, M. A., Ferguson, R. W., Grosso, W. E., Noy, N. F., Crubezy, M., & Gennari, J. H. (2000). Component-based support for building knowledge-acquisition systems. In *Conference on intelligent information processing (IIP 2000) of the international federation for information processing world computer congress (WCC 2000)*. Beijing, China.
- Nadkarni, N. M., & Cushing, J. B. (1995). *Final report: Designing the forest canopy researcher's workbench: computer tools for the 21st century*. Olympia, WA: International Canopy Network.
- Nadkarni, N. M., & Cushing, J. B. (2001). Lasers in the jungle: The forest canopy database project. *Bulletin of the Ecological Society of America*, 82, 200–201.
- Nadkarni, N. M., & Parker, G. G. (1994). A profile of forest canopy science and scientists—who we are, what we want to know, and obstacles we face: Results of an international survey. *Selbyana*, 15, 38–50.
- Nottrott, R., Jones, M. B., & Schildhauer, M. (1999). Using Xml-structured metadata to automate quality assurance processing for ecological data. In *Third IEEE computer society metadata conference, Bethesda, MD*: IEEE Computer Society.
- NRC. National Research Council. (1995). *Finding the forest for the trees: The challenge of combining diverse environmental data-selected case studies*. Washington, DC: National Academy.
- NRC. National Research Council. (1997). *Bits of power: issues in global access to scientific data*. Washington, DC: National Academy.
- Peyton-Jones, S. (2003). *Spreadsheets—functional programming for the masses*. Invited talk. Technical symposium on software, science & society. Oregon Graduate Institute of the Oregon Health and Science University, Friday, December 5, 2003. <http://web.cecs.pdx.edu/~black/S3S/speakers.html> and <http://web.cecs.pdx.edu/~black/S3S/PJ.html>.
- Raguenaud, C., & Kennedy, J. (2002). Multiple overlapping classifications: issues and solutions. In *14th international conference on scientific and statistical database management—SSDBM 2002* (pp. 77–86). Edinburgh, Scotland: IEEE Computer Society.
- Romanello, S., Beach, J., Bowers, S., Jones, M., Ludäscher, B., Michener, W., et al. (2005). Creating and providing data management services for the biological and ecological sciences: science environment for ecological knowledge. In *17th International Conference on Scientific and Statistical Database Management-SSDBM 2005*.
- Schnase, J. L., Cushing, J., Frame, M., Frondorf, A., Landis, E., Maier, D., et al. (2003). Information technology challenges of biodiversity and ecosystems informatics, special issue on data management in bioinformatics, *Information Systems*. In: M. J. Zaki, & J. T. L. Wang (Eds.) Volume 28, 4., June 2003. (pp 241–367). Elsevier Science.
- Schroeder, W., Martin, K., & Lorensen, B. (1998). *The visualization toolkit*. Upper Saddle River, NJ: Prentice Hall.
- Sheard, T. (2001). Accomplishments and research challenges in meta-programming. Invited talk. In *Semantics, applications, and implementation of program generation 2001. LNCS*, Volume 2196. (pp. 2–44). Florence, Italy: Springer.
- Sheard, T., & Jones, S. P. (2002). *Templatemetaprogrammingforhaskell. Haskell workshop*. Pittsburg, PA: ACM.
- Sowa, J. F. (1984). *Conceptual structures: information processing in mind and machine*. Reading, MA: Addison Wesley.
- Spycher, G., Cushing, J. B., Henshaw, D. L., Stafford, S. G., & Nadkarni, N. M. (1996). Solving problems for validation, federation, and migration of ecological databases. Global networks for environmental information. In *Proceedings of Eco-Informa '96* (pp. 695–700). Lake Buena Vista, FL.: Ann Arbor, MI: Environmental Research Institute of Michigan (ERIM).
- Stemple, D., & Sheard, T. (1991). A recursive base for database programming primitives. In *Proceedings of next generation information system technology, LNCS*, (pp. 311–332). Springer.
- Szyperski, C. A. (1997). *Component software*. Addison-Wesley.
- Van Pelt, R., & Nadkarni, N. M. (2004). Horizontal and vertical distribution of canopy structural elements of pseudotsuga menziesii forests in the pacific northwest, *Forest Science*, 50: 326–341.

- Villa, F. (2001). Integrating modelling architecture: A declarative framework for multi-paradigm, multi-scale ecological modeling. *Ecological Modelling*, 137, 23–42.
- Wang, B., Liu, X., & Kerridge, J. (2003). Agenerative and component based approach to reuse in database applications. In *5th generative programming and component engineering young researcher workshop*. (September)
- Weaver, M., Delcambre, L., & Maier, D. (2001). A superimposed architecture for enhanced metadata. In *DELOS workshop on interoperability in digital libraries, held in conjunction with European Conference on Digital Libraries (ECDL 2001)*. Darmstadt, Germany.
- Wood, W. A., & Kleb, W. L. (2003). Exploring XP for scientific research. *IEEE Software*, 20, 30–36.

### URL's referenced in the paper

- [canopydb] At <http://canopy.evergreen.edu>, you will find general information about our project, including links to our software prototypes, *DataBank*, *CanopyView* and the *BCD*.
- [IterSyn] <http://intranet.lternet.edu/archives/documents/foundations/WhitePaperJune2002GRS.html>. LTER 2000–2010: A DECADE OF SYNTHESIS, June 2002.

### Other sites about ecosystem informatics or software cited in this paper follow

- Biodiversity and Ecosystem Informatics Workshops (sponsored by NSF, USGS and NASA) and <http://canopy.evergreen.edu/bdeipi>
- Ecoinformatics: <http://www.ecoinformatics.org> and <http://ecoinformatics.org/tools.html>
- Ecological Markup Language (EML) <http://cvs.ecoinformatics.org/cvs/cvsweb.cgi/eml/>
- Ecological Society of America (ESA) archives: <http://data.esa.org/>
- Knowledge Network for BioComplexity (KNB) <http://knb.ecoinformatics.org>
- Long Term Ecological Network (LTER): <http://lternet.edu> and the H.J. Andrews LTER Data Repository: <http://www.fsl.orst.edu/lter>
- Science Environment for Ecological Knowledge (SEEK): <http://seek.ecoinformatics.org>
- Protégé and Ontology Management Systems: <http://protege.stanford.edu> and [http://protege.stanford.edu/publications/ontology\\_development/ontology101-noy-mcguinness.html](http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html)
- VTK: <http://www.kitware.com>.
- XML: <http://www.w3.org/XML>