Analytic Webs Support the Synthesis of Ecological Data Sets

CONCEPTS & SYNTHESIS
EMPHASIZING NEW IDEAS TO STIMULATE RESEARCH IN ECOLOGY

Ecology, 87(6), 2006, pp. 1345–1358

ANALYTIC WEBS SUPPORT THE SYNTHESIS OF ECOLOGICAL DATA SETS

Aaron M. Ellison,[1,3] Leon J. Osterweil,[2] Lori Clarke,[2] Julian L. Hadley,[1] Alexander Wise,[2] Emery Boose,[1] David R. Foster,[1] Allen Hanson,[2] David Jensen,[2] Paul Kuzeja,[1] Edward Riseman,[2] and Howard Schultz[2]

[1]*Harvard University, Harvard Forest, 324 North Main Street, Petersham, Massachusetts 01366 USA*
[2]*University of Massachusetts, Department of Computer Science, Computer Science Building, Amherst, Massachusetts 01003 USA*

Abstract. A wide variety of data sets produced by individual investigators are now synthesized to address ecological questions that span a range of spatial and temporal scales. It is important to facilitate such syntheses so that "consumers" of data sets can be confident that both input data sets and synthetic products are reliable. Necessary documentation to ensure the reliability and validation of data sets includes both familiar descriptive metadata and formal documentation of the scientific processes used (i.e., process metadata) to produce usable data sets from collections of raw data. Such documentation is complex and difficult to construct, so it is important to help "producers" create reliable data sets and to facilitate their creation of required metadata. We describe a formal representation, an "analytic web," that aids both producers and consumers of data sets by providing complete and precise definitions of scientific processes used to process raw and derived data sets. The formalisms used to define analytic webs are adaptations of those used in software engineering, and they provide a novel and effective support system for both the synthesis and the validation of ecological data sets. We illustrate the utility of an analytic web as an aid to producing synthetic data sets through a worked example: the synthesis of long-term measurements of whole-ecosystem carbon exchange. Analytic webs are also useful validation aids for consumers because they support the concurrent construction of a complete, Internet-accessible audit trail of the analytic processes used in the synthesis of the data sets. Finally we describe our early efforts to evaluate these ideas through the use of a prototype software tool, SciWalker. We indicate how this tool has been used to create analytic webs tailored to specific data-set synthesis and validation activities, and suggest extensions to it that will support additional forms of validation. The process metadata created by SciWalker is readily adapted for inclusion in Ecological Metadata Language (EML) files.

Key words: *analytic web; eddy covariance; EML; metadata; process; synthesis; SciWalker; XML.*

Introduction

Examining complex questions, integrating information from a variety of disciplines, and testing hypotheses at multiple spatial and temporal scales account for an increasing proportion of ecological and environmental research (e.g., Michener et al. 2001, Andelman et al. 2004). Such syntheses can help to identify and address the "big" ecological questions (Lubchenco et al. 1991, Belovsky et al. 2004) and contribute to the setting of local, regional, national, and global environmental policies (e.g., Schemske et al. 1994, IPCC 2001, Kareiva 2002). Synthesis is the raison d'être of the National Center for Ecological Analysis and Synthesis (NCEAS). Barely a month passes without the appearance of one or more publications by NCEAS working groups. At its heart, synthesis involves intellectual creativity: asking crosscutting questions and confronting existing paradigms from new standpoints. But rigorous syntheses cannot proceed without reliable data sets, those that are carefully documented with all of the relevant details about their content and how they were created (i.e., the *provenance* of the data sets; see Table 1 for a succinct glossary of italicized terms). An important start in this direction is being made with the many efforts underway to develop data documentation (*metadata*) tools (e.g., Michener 2000, Jones et al. 2001, Helly et al. 2002). These are improving ecologists' abilities to document precisely the structure and provenance of ecological data

Manuscript received 23 June 2005; revised 17 November 2005; accepted 28 November 2005. Corresponding Editor: B. E. Kendall.
[3] E-mail: aellison@fas.harvard.edu

TABLE 1. A glossary of key terms (italicized in text).

Term	Definition
Analytic web	Formal notation, illustrated by three coordinated graphs (data-set derivation, data-flow, and process derivation), that provides a basis for completely and precisely defining process metadata needed to produce reliable data sets.
Binding	Association of a specific instance to the placeholder designated by a type.
Data-set Derivation Graph (DDG)	Visual representation of way in which specific data-set entities (or data-set instances) have been derived through the action of specific tools or processes upon specific input data sets.
Data-flow Graph (DFG)	Visual representation of ways in which instances of data-set types can be derived by actions upon instances of input data-set types.
Digital Object Identifier (DOI)	Permanent Uniform Resource Locator (URL).
Ecological Metadata Language (EML)	Application of Extensible Markup Language (XML) used to construct metadata for ecological data sets.
Extensible Markup Language (XML)	Computer science notation used to create structured descriptions of broad classes of entities, particularly widely applied to description of structured data and creation of metadata.
Instance	Unique, individual entity. Contrast with Type.
Metadata	Data about data sets. We distinguish two kinds of metadata:
Descriptive metadata	Data about structure, content, producer, and location of a data set.
Process metadata	Data about process by which the data was derived.
Parsing	Determination of grammatical or syntactic structure of a statement.
Process Derivation Graph (PDG)	Visual representation of ways in which instances of data-set types can be derived from actions upon instances of input data-set types. Allows for more detailed description of process than can usually be described with a DFG.
Provenance	Relevant details about content and creation of an entity.
Reliability	Assurance about safety of using an entity in specific ways.
Scientific workflow	Prescription for how scientific data sets can be developed.
Searching	Examination of one or more data sets to determine if one or more instances with a particular property are present in the data sets.
Semantics	Determination of the meaning of a statement, or some part(s) of a statement.
Type	Set of entities (often called type instances) all sharing a common set of characteristics or properties. Contrast with Instance.
Uniform Resource Locator (URL)	Symbolic name used to access information (e.g., data sets, instances, or specific executable tool instances) via the Internet.

sets, from the relatively small ones in field notebooks to the terabytes generated by automated sensor networks.

This is a pressing issue. Funding agencies increasingly mandate that data sets obtained with public funds be made available with few restrictions via the Internet. In general, it is up to individual investigators to meet federal laws and directives of funding agencies. The Ecological Society of America makes allowances for researchers to archive data sets in Ecological Archives (*available online*)[4] but has no codified requirements for permanent data-set accessibility or archiving—only that the editors and publisher expect authors to make data underlying published articles available. In contrast, the Long-Term Ecological Research (LTER) network insists that the majority of data sets collected at LTER sites be archived permanently and be publicly available within two years of collection (see the LTER web site).[5]

It is a potential boon to science that collaborations among geographically distributed researchers can be facilitated by easy access to scientific data sets, and this boon creates unprecedented opportunities for participation in the active conduct of science by large groups of individuals and communities. In the Physics community, exemplars are the Globus and GriPhyN projects (*available online*).[6,7] There is a large and growing community of research groups developing tools that support such *scientific workflow* (see the Scientific Workflows Survey, *available online*).[8] Similarly rapid progress in ecological research could be achieved if ecologists could rely upon synthetic, often complex, data sets created from existing data sets, as well as real-time or near real-time massive data streams from automated sources. Examples of the latter include climate data, satellite imagery, measurements of energy, nutrient or gas fluxes, gene sequences, and data expected from the nascent National Ecological Observatory Network (NEON; information *available online*).[9]

The metadata currently associated with ecological data sets, however, are inadequate to assure that consumers of data sets can use them reliably. Documentation of the structure and content of data sets is an important start, but an equally important challenge is to assure that the analytical processing of the data sets is also well documented. Such *process metadata* protects

[4] ⟨http://esapubs.org/Archive/⟩
[5] ⟨http://www.lternet.edu/data/netpolicy.html⟩
[6] ⟨http://www.globus.org⟩
[7] ⟨http://www.griphyn.org⟩
[8] ⟨http://www.extreme.indiana.edu/swf-survey/⟩
[9] ⟨http://www.neoninc.org/⟩

subsequent consumers of documented data sets from misinterpreting the data, allows them to replicate analytical processes with the same or alternative data sets, and permits them to apply new analytical processes to the data consciously and safely.

In summary, expedited access to data sets creates opportunities for broadened scientific collaboration but simultaneously raises concerns regarding the reliability of these data sets and their associated results. The work we describe here illustrates that precise and complete documentation, both of data sets and of the processes used to produce them, can address these concerns. Moreover, appropriate formalisms and automated tools can ensure that the construction of descriptive and process metadata need not be an undue burden to individual producers of data sets.

The overall goals of this paper are:

1) To demonstrate that reliable ecological data sets and analytical results require not only the more familiar descriptive metadata, but also detailed process metadata.

2) To encourage the inclusion of process metadata into evolving standards for ecological metadata.

3) To demonstrate that appropriate formalisms and supporting tools can facilitate the synthesis of data sets and associated process metadata.

4) To illustrate the use of these formalisms and tools with an example.

What Are Reliable Data Sets?

Data sets used in scientific publications can be far removed from the raw data collected by their producers in the laboratory or in the field. In general, these data sets are produced when individual researchers apply a familiar sequence of scientific processes, including: sampling and making observations, data checking and cleaning (quality assurance/quality control, or QA/QC), variable transformations, statistical model construction, and statistical inference and evaluation. If these data sets are not accompanied by any documentation of the manner in which the data were gathered or subsequently processed by the producer of the data set, subsequent consumers of the data sets cannot necessarily rely on them for further analysis or synthesis. For example, data may have been collected by equipment that the producer knew to be faulty or in need of calibration, or incorrect values may have been entered into a data set when transcribing from field forms. When the producer processed these data into an analyzable data set, s/he may have removed or interpolated inaccurate values. It is important for consumers to know exactly how this was done. Similarly, construction of some data sets entails substantial processing, while others may entail little or none. In either case, consumers need to know precisely what processing has been done so that they can avoid redundant or incorrect subsequent analyses. Accurate specification in the metadata of both the source of the data and its processing yields a *reliable* data set that its consumers can use safely. Further, metadata should be adequate to allow for the reconstruction or reproduction of synthetic data sets from the original raw data. Unreliable data sets, on the other hand, lack such specification. They cannot be reconstructed or reproduced from the original raw data, and thus they may be misused, leading to unreliable results.

We can imagine a wide range of possible details that could be incorporated into process metadata, and so we offer no hard and fast requirements for them. Rather, we simply observe that the more metadata that is provided, the more it is likely to be useful in determining the accuracy and validity of subsequently derived data sets. Here, we focus on how the reliability of data sets can be increased by applying tools and technologies based upon formal notations to create and apply process metadata.

Ecological Metadata Language (EML), Scientific Workflow, and Process Metadata

Methods of data collection and descriptions of the variables in a data set currently are common elements of *descriptive metadata*: "the higher level information or instructions that describe the content, context, quality, structure, and accessibility of a specific data set" (Michener et al. 1997:331). In recent years there have been significant advances in the development of standards and tools for creating and using ecological metadata (Andelman et al. 2004). For ecologists, perhaps the most important of these has been the creation of a standard for structured metadata, *Ecological Metadata Language* (EML; information *available online*).[10] As a modular and flexible application of *Extensible Markup Language* (XML; information *available online*),[11] EML has been designed and developed by the ecological community to support data discovery, access, integration, and synthesis.

The metadata encoded by EML provides a formal description of what is inside a data set. For example, all metadata files contain the name of the person who collected the data (whom we call the "producer" in this paper), where they were collected, the types of organisms or systems sampled, a description of the structure of the data set (normally a table), the meaning of abbreviated variable names, the units of measurement, searchable key words, etc. Access to data sets is facilitated through specific information on their location on the Internet (e.g., its *uniform resource locator* [URL] or *digital object identifier* [DOI]) and the physical characteristics (e.g., file name, coding, record delimiters, and field delimiters) of data files. EML files are interpretable, *searchable*, and *parsable* by computers, which facilitate the retrieval of the metadata and the associated data sets.

EML is intended to broaden the scope of ecological research and synthesis by increasing the reliability of

[10] ⟨http://knb.ecoinformatics.org/software/eml⟩

[11] ⟨http://www.w3.org/XML/⟩

data sets that are accessible by their subsequent users (whom we refer to as "consumers"). Because EML currently lacks formal specifications for describing analytical processes, we suggest that augmenting EML with process metadata will significantly enhance the reliability of documented ecological data sets. In particular, suitably complete and precise process metadata can provide the basis for perhaps the most central type of validation in science, the reproducibility of a data set. Thus, we propose that the current structure of EML (i.e., the available XML tags) be expanded to allow for a formal and interpretable specification of the computational methods or statistical models used to derive published data sets from raw data.

Currently, EML provides two modules for the documentation of methods used to create data sets. The "Protocol" module is used to describe an established field, laboratory, or analytical procedure, essentially a standardized method such as "infrared gas analysis was used to measure concentrations of CO_2." The "Methods" module is used to describe the field, laboratory, and analytical procedures that were actually used in the creation of a particular data set: "a Li-Cor 6262 IRGA (Li-Cor Biosciences, Lincoln, Nebraska, USA) running LI-1000 software version 1.2 was used to measure concentrations of CO_2 between 9:00 and 12:00 hours on 12 August 2004." The Protocol module is prescriptive, whereas the Methods module is descriptive and may refer to one or more relevant Protocol modules, if they are available. Although the sequence of steps in a process can be captured with the "methodStep" element, the contents of both the Protocol and the Methods modules are otherwise unstructured narratives. They are neither searchable nor parsable in the same way as the descriptive metadata. Further, there is no requirement that the descriptions within either the Protocol or Methods module unambiguously define an analytical process that could be repeated so as to validate a data set by reproducing it. This is an important inadequacy that our work directly addresses.

We note that the reproduction of data sets is especially complicated when the phenomenon of interest is manifest only through data sets that have been combined and processed through complex sequences of computer-based tools and processes. If not precisely and completely stated, such complexity can create ambiguity (e.g., Thornton et al. 2005), leading to statistical or logical errors (e.g., Garcia-Berthou and Alcaraz 2004).

The processes used by producers to generate published data and results, including the tools and subprocesses employed by those processes, must be available and well documented to ensure accurate reproduction of data sets. Modifications to any of these tools or processes may be inadvertent, as when a software package is updated or the underlying operating system is modified. Lacking awareness of these modifications, attempts to reproduce data sets may proceed under the incorrect assumption that the original process is being used. If changes have been made, then the original scientific process has not been repeated, and can lead either to different results or to the false conclusion that confirmation of prior results has occurred (e.g., Dominici et al. 2004).

To ensure that data sets attain the desired degree of reliability that comes from being reproducible, we propose that every data set generated by an ecological research project should have attached to it not only descriptive metadata, but also structured process metadata that formally describes the processes by which the producer generated the data set, including the sequence of tools, techniques, and intermediate data sets used. Such process metadata is a critical complement to existing descriptive metadata provided by the EML standard, and with suitable XML extensions, could be incorporated directly into EML files.

We are acutely aware of the considerable burden that the need to create both descriptive and process metadata places upon a data set's producer. We note in particular that our community gives far greater rewards for publications and grants than for the generation of archival data sets and associated metadata. Thus, we propose that tools and technologies be used by data-set producers to create concurrently both data sets and the desired process metadata that future consumers will need. This seems especially important as our work has indicated that producers often have considerable difficulty in defining completely and precisely the processes used to turn raw data into usable data sets.

There is growing interest in using formal notations, such as data-flow graphs, to document the processes used to generate scientific data sets (e.g., Ailamaki et al. 1998, Altintas et al. 2004*a*, *b*, Ludäscher et al. 2006). Data-flow graph definitions can be used to define many processes used by ecologists and other scientists, but we argue that they may be incapable of capturing important subtleties and complexities in these processes. Thus, we have developed the concept of an *analytic web*, a more powerful formal notation that provides a basis for completely and precisely defining the process metadata needed to produce more reliable data sets. The process metadata of an analytic web not only increases the reliability of a data set by enabling its production and reproduction, but also serves as a rigorous basis for the development of automated tools that can support additional analysis and synthesis. Before we discuss in detail the components of an analytic web and the ways that it supports both producers and consumers, we first present an example of an ecological data set whose reliability can be increased by an analytic web.

An Analytic Web for Ecosystem Carbon Flux

Measuring ecosystem C flux

The increasing atmospheric concentration of carbon dioxide (CO_2) and its relationship to global temperature are well known (IPCC 2001). General circulation models (Cramer et al. 2004, Meehl et al. 2004) and direct

measurements of ecosystem–atmosphere exchange using eddy covariance methods (Baldocchi et al. 1988, Hollinger et al. 1994, Barford et al. 2001) provide data that are used to estimate sources or sinks for C; predict how changes to ecosystems will alter atmospheric CO_2 levels; and forecast how climate change will affect C storage. The data sets are processed with statistical and mathematical models that provide continuous estimates of CO_2 exchange rates.

The accuracy of carbon flux estimates from eddy covariance data varies with micrometeorological conditions. During daylight, forest canopies rarely present a serious barrier to accurate measurements because solar radiation heats the air near the surface and creates adequate vertical convection. At night, convection is much weaker and the ground may become as cold as (or colder than) the air above it, creating a layer of stable or sinking air above the surface. Due to these problems, accurate eddy covariance data may be unavailable for from 40% to 75% of nighttime hours (Barford et al. 2001, Saleska et al. 2003, Hollinger et al. 2004). Data are also lost during equipment calibrations, maintenance, or malfunctions.

To estimate carbon flux over long periods, data gaps must be filled by interpolation. In publications with major policy implications (e.g., Wofsy et al. 1993, Goulden et al. 1996, Barford et al. 2001, Saleska et al. 2003), these processed data are often reduced to a single graph. Although averaged and processed data are available online, interpolated data are rarely identified and thus cannot be distinguished from measured values in online data sets (see e.g., the National Institute for Global Environmental Change [NIGEC] North East Regional Center Data Archive/Exchange, *available online*).[12] Further, the procedures for interpolation and gap filling usually are not readily available. Uncertainty about the processes used to fill in the data contributes to uncertainty about the reliability of the data and all estimates, predictions, and forecasts derived from them.

Valid syntheses of carbon exchange data require unambiguous knowledge of the processes used for data acquisition and manipulation, but processes for filling data gaps vary widely. Differences in data processing can impede reliable forecasts; for example, eddy covariance measurements in the Amazon basin in the 1990s were used to conclude that Amazon forests were storing 1–5 Mg $C{\cdot}ha^{-1}{\cdot}yr^{-1}$ (Grace et al. 1995, 1996, Malhi et al. 1998, Carswell et al. 2002). Saleska et al. (2003), using different criteria for discarding and interpolating data, found an annual net carbon *loss* of 0–2 Mg $C{\cdot}ha^{-1}{\cdot}y^{-1}$ from two Amazonian forests. The earlier estimates of substantial C storage likely were caused by inclusion of data from nighttime periods when low turbulence was reducing the measured carbon efflux.

[12] ⟨http://www-as.harvard.edu/data/nigec-data.html⟩

PLATE 1. Eddy covariance tower in a hemlock stand at the Harvard Forest. Photo credit: J. Hadley.

Collecting, excluding, and interpolating eddy covariance data

We use carbon flux measurements and their subsequent processing to illustrate how to create formal process metadata. Vertical and horizontal wind vectors and CO_2 mixing ratio at 5 Hz are measured with a sonic anemometer mounted 5 m above the forest canopy (see Plate 1), beside an intake port from which air is pumped to a closed-path infrared gas analyzer (Hadley and Schedlbauer 2002). A laptop computer collects these data and computes 10-minute running means of each of these "tower variables." Every 30 minutes, the computer calculates and stores a mean value for each variable, and the covariances of all other variables, with deviations from the running mean of the vertical wind velocity. A datalogger at the same site measures other environmental variables affecting carbon flux, including photosynthetically active radiation (PAR), air and soil temperature, atmospheric humidity, and soil moisture. These environmental variables are measured every 30 seconds, and 30-minute averages are stored.

Before carbon flux is estimated, we first discard measured CO_2 fluxes if the wind direction is unsuitable for flux measurements, because local topography creates

unpredictable turbulence patterns, or the forest of interest is absent in that direction. Next, the 30-minute means and covariances of the tower variables are checked to identify atmospheric conditions in which measured C flux is not limited by turbulence or weak vertical convection. These atmospheric conditions usually hold during daylight hours, but not at night. Here, we describe the process by which valid nighttime carbon flux data are identified and used to estimate C flux during nighttime periods of inadequate vertical mixing or turbulence.

We examine the relationship between friction velocity, u^* (a measure of turbulence), and CO_2 flux to identify a u^* value (u^*_{thr}, the threshold value) above which estimated CO_2 flux does not increase significantly. Observed values of CO_2 flux are discarded if $u^* < u^*_{thr}$. When the u^*_{thr} and wind direction criteria are both applied, >75% of the nighttime observations may be rejected.

Next, we fill the resulting gaps in the CO_2 flux data by estimating the values that would have been observed if $u^* \geq u^*_{thr}$, using regression models derived from reliable observations (CO_2 flux $\mid u^* \geq u^*_{thr}$) and measured environmental variables. For nighttime observations, the predictor variables, identified using stepwise multiple regression, are soil temperature, air temperature, and, occasionally, soil moisture. These relationships change over time, so a new regression model is created for each 1–3 months of data.

This prose description of the process by which raw eddy flux data are collected (at 5 Hz), postprocessed in real time (30-minute running average), checked for accuracy by the investigator (excluding inaccurate data), and estimated values interpolated (using regression models on independent data) could be reported within the Protocol or Methods section of an EML metadata file, but it would be difficult to exactly reproduce these steps from the prose description. For example, there is locally modified custom software used for the 30-minute averaging of the 5Hz data. The data are checked for accuracy by eye, and by using heuristic searches in Excel spreadsheets. Critical values of u^*_{thr} vary among investigators and change as more data accrue. Best-fit regression models differ depending on time of year or location, and values and precision of regression coefficients differ among software packages, and versions of individual packages. An analytic web can be constructed that contains the formal, structured metadata for these processes.

An Analytic Web Defined

An analytic web is a formal definition of a scientific process. Here we propose that an analytic web be represented by three coordinated types of graphs: a *data-flow graph*, a *data-set derivation graph*, and a *process derivation graph* (Fig. 1), all of which were originally developed for use in defining and controlling software development projects (e.g., Ghezzi et al. 2003). The three different graphs are intended to support data-set producers and consumers in different ways. Here we provide only a sketch of the features of each type of graph, emphasizing its role and value.

The most familiar of the three types of graphs is the data-flow graph (DFG; Fig. 1A), which is similar in form and *semantics* to similar graphs in systems such as Kepler (Ludäscher et al. 2006). A DFG deals with *types*, rather than specific *instances*, of data sets, tools, and processes. It defines the sequence of tools and processes that a producer applies to raw data and intermediate data sets when creating a final data set. In a DFG, different icons are used to differentiate tools or processes (ovals in Fig. 1A) from data sets (boxes in Fig. 1A). Arrows (or edges) in Fig. 1A connecting these icons (or nodes) represent the flow of data sets into and out of these tools or processes. For example a DFG might specify that a process type of "interpolate via linear regression" could be applied to a data-set type of "eddy flux data" and the DFG would then also indicate the type(s) of data sets that would be produced. The DFG specifies only that a process for interpolating via linear regression must be used, not which specific tool is used to execute the process. This is analogous to the way in which a cookbook provides clear instructions to chefs so that they can produce specific dishes, but does not mandate any specific cookware, implement, or brand of food. However, when the general description of the type of each data set in a DFG is associated with, or "bound to," a specific data set and the general description of the type of each tool or process is bound to a specific tool or process, the DFG along with this *binding* information describes precisely what sequence of activities must be performed in order to produce the new resultant data sets of the types specified by the DFG.

Thus, the DFG is particularly useful to producers because it illustrates a recipe that can be used to produce new data sets. This recipe is precise enough that the DFG, along with the binding information described, can be automatically executed by a DFG interpreter (*available online* in SciWalker).[13] The DFG interpreter must send the appropriate input data sets to the appropriate tools and processes, initiate the execution of those tools and processes on those data sets, and then store and transmit the derived data sets according to the iconic description of the DFG. Now that data sets are readily accessible across the Internet, such an automatic tool, driven by an iconic representation such as a DFG, could lead to rapid scientific syntheses. But consumers of data sets produced by others must be guided by accurate metadata to avoid inappropriate uses of data sets. A particularly useful guide is the information contained in an analytic web's second graph, the data-set derivation graph (DDG, Fig. 1B).

[13] ⟨http://laser.cs.umass.edu/tools/sciwalker.shtml⟩

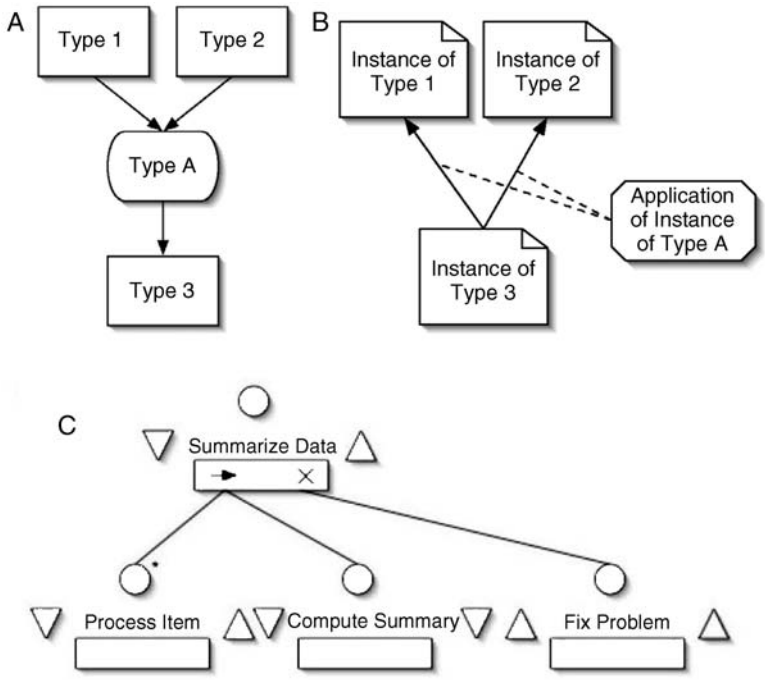

FIG. 1. Examples of a data-flow graph (DFG), data-set derivation graph (DDG), and process derivation graph (PDG). (A) In a DFG, data-set types are indicated by rectangles, and process types are indicated by ovals. Arrows (edges) indicate the direction of flow. In this DFG, data sets of Type 1 and Type 2 are used by a tool of type A to create a data set of Type 3. (B) A DDG illustrates the particular outcomes that result from executing the DFG. Instances of data sets are indicated by rectangles with one corner clipped off, and instances of tools are indicated by ovals with all four corners clipped off. Arrows (edges) indicate how a given data set was derived from a previous data set. These arrows are annotated (dotted lines) to indicate which process was applied. In this DDG, a particular data set of Type 3 was derived from two particular data sets of Type 1 and Type 2, respectively, by applying a particular instance of Tool A. (C) A process derivation graph is a symbolic representation of the procedural details needed to produce the DDG. Each tool or process type is represented by a name over a rectangle, called a "step." Associated with each step is a set of icons that indicate its substeps and their execution order, pre- and post-requisites, and exception handling directives.

A DDG documents the specific data sets created when a producer applies the processes defined by the DFG, using specific tools and processes on specific input data or data sets, and contains the precise details of the processes by which they were created. The DDG thus contains the detailed process metadata required to support reproduction of a given data set and seems to us to be the minimal information needed to reliably use a producer's data set. Just as in the DFG (Fig. 1A), the DDG (Fig. 1B) uses different icons to differentiate specific instances of a given data set (clipped boxes, or nodes) from specific instances of processes used (e.g., application of a particular version of a statistical routine; clipped ovals). Each node is connected by an arrow (edge) to the data set(s) from which it was derived. Each time that a DFG is executed, a new set of instances, organized as the nodes of a DDG, is created. Each data set represented by a DDG node can then be stored independently with a unique Internet-accessible address (a URL or DOI).

Data set instances, such as those represented by DDG nodes, are the usual focus of attention and thus are the objects that are normally documented with metadata specified by EML. For example, "eddy flux data collected at the Harvard Forest on 1 June 2004 at hourly intervals" is an example of a data set that would be incorporated as a node in the DDG component of an analytic web. It is a specific instance of the type of data called "eddy flux data" that would be incorporated into the corresponding DFG component of the analytic web. Consumers who must validate the reliability of a data set, especially if they want to do so by reproducing it, require the documentation provided by the DDG, namely the specific data sets and tools that were actually used. The quantity and intricacy of this documentation is indeed considerable, but it can be produced automatically with the DFG interpreter in SciWalker. We illustrate this automation with an analytic web that organizes processes, data sets, and their associated metadata for the eddy flux data.

We emphasize that the DFG and DDG provide different kinds of information, and that both are of interest and use, both to producers and to consumers. The DDG provides the exact specific details of precisely which data sets were processed by precisely which tools in order to produce exactly which product data sets. Consumers require such specifics in order to reproduce data sets of interest, and producers need these specifics

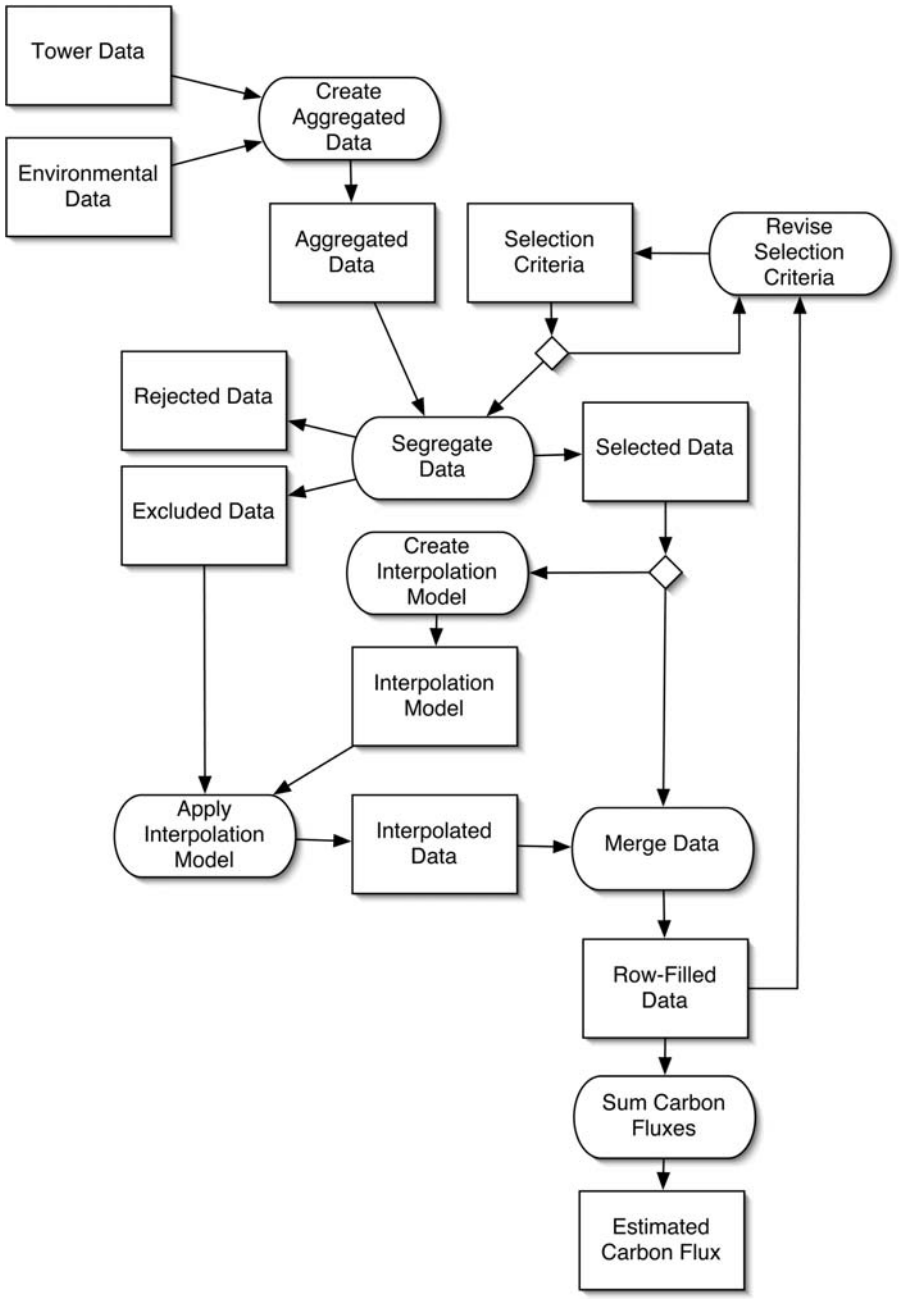

FIG. 2. The data-flow graph (DFG) of the processes used for the analysis of the eddy covariance data and for tracking the effects of changes in the output resulting from changes in u^*_{thr} (one of the selection criteria).

in order to be sure that they can correctly keep track of the exact provenance of the ever-proliferating products of their research. The DDG has clear value in documenting the forward flow of computations and it also documents equally clearly a retrospective view of where various data sets came from. This seems particularly important for consumers who are perhaps curious or skeptical about results, or who might be interested in reusing a process in which one or more of the data sets or tools that had been used now differs. In this case, the DFG becomes essential, as this graph documents the types of the data sets and tools that must be used in any such substitution. If a consumer wishes to apply the process to different input data sets, or to employ different processing tools, the DFG specifies the types of such data sets and tools. Through a tool such as SciWalker, the process can be reapplied easily using these alternative data sets or tools. The DFG is also

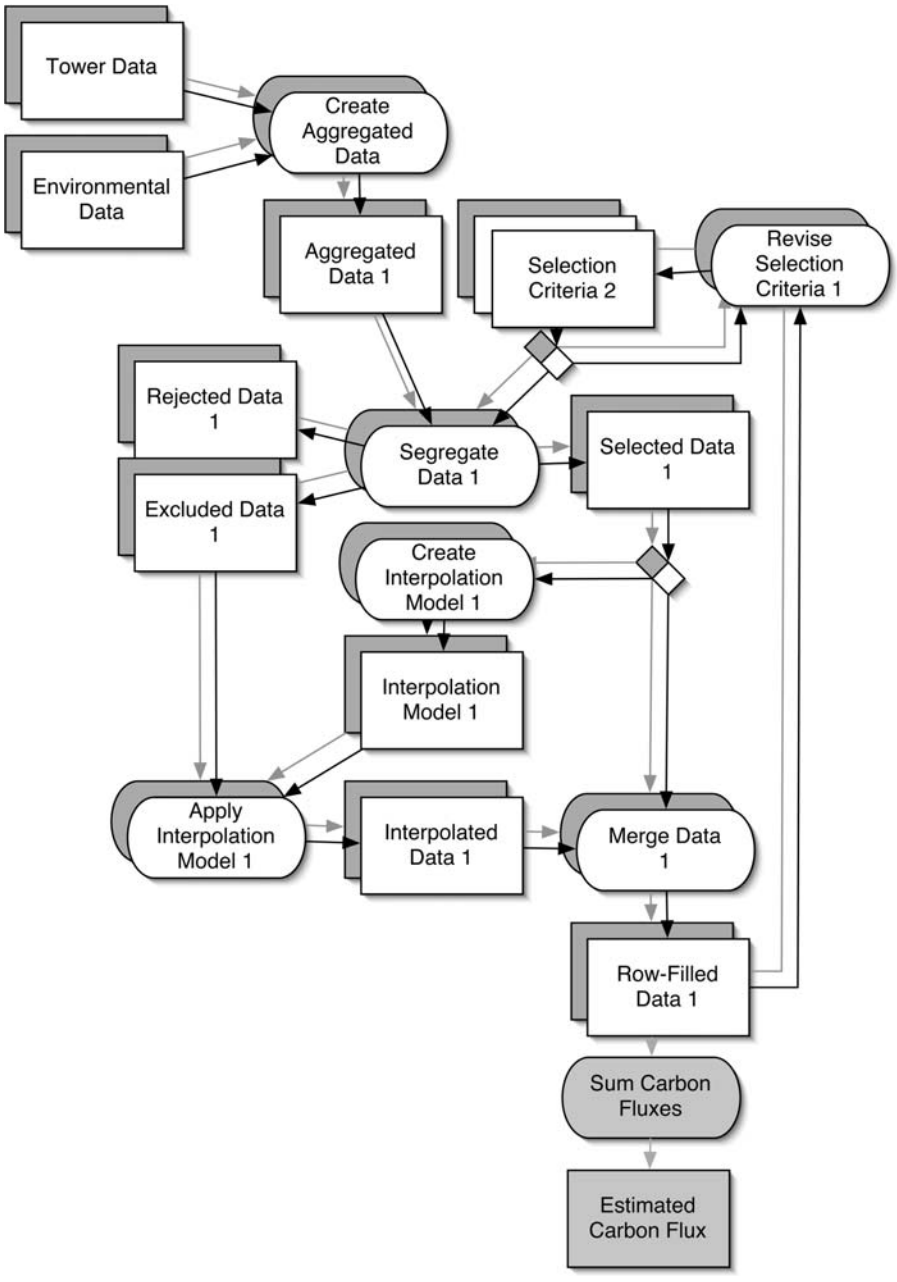

FIG. 3. A stacked view of data-set derivation graphs (DDG) that illustrates different instances using stacked icons.

valuable to a producer who might want to define and explore a variant of an existing process and thus uses the original process as a blueprint for an alternative process definition. In this case, it is not just the tools and data sets that may vary, but the actual process definition itself may vary, and this change is represented in the newly defined DFG.

An analytic web for the eddy covariance data

We used SciWalker to create the DFG that formally describes our processing of eddy covariance data and estimation of carbon flux (Fig. 2; see Appendix A for an animation of how the tool was used to create the DFG). Note that this figure does not describe the specifics of generating any specific data sets. Rather, it is a description of the general process by which desired types of data sets have been, and can be, developed. The first step of the process is to combine tower data (raw data sets coming directly from a flux tower) and environmental data (readings taken directly from environmental sensors) into aggregated data. This is done by matching rows based on time and date stamps,

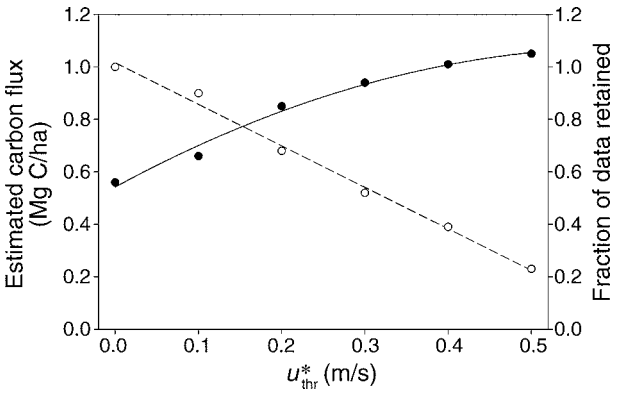

FIG. 4. Estimated nighttime CO_2 flux (solid circles and solid line) and the fraction of data used in the analysis (open circles and dashed line) from April to June 2001 above a central Massachusetts hemlock forest, for southwesterly winds and for different ranges of friction velocity (u^*) (Hadley and Schedlbauer 2002).

using a "create aggregated data" process. The DFG does not specify any particular tool or system to carry out the create aggregated data process, but rather indicates that any such tool or system that effects the needed matching could suffice.

Once the aggregated data set has been created, the DFG specifies that this intermediate data set is to be split into three parts: rejected data (in this example, all the daytime data, as our goal is a model of nighttime carbon flux), excluded data (data collected when the wind was blowing from any direction other than the southwest [i.e., 0–179° E of N and 271–359° E of N], and when $u^* \leq 0.4$), and selected data (all the rest). Each of these data sets results from an application of a segregate data process. Once the DFG had been built and all the bindings specified, the DFG interpreter in SciWalker can access the input data sets and tools needed for derivation of the intermediate and output data sets. It uses the DFG to specify the sequence of application of the tools, the data sets needed as their inputs, and the created output, synthesized data sets.

Concurrently SciWalker creates the DDGs (Fig. 3; see Appendix B for an animation of how the DDGs were created) describing the development of these output data sets (see Appendix C for a simplified excerpt). Unlike the DFG in Fig. 2, which describes a general process applied to general types of data sets, the DDG in Fig. 3 is a description of how to take specific data sets and apply specific tools and systems to them. In particular, we used this DDG to document the analyses we used to assess the effect of varying u^*_{thr} on estimates of nighttime carbon flux. We began by taking data sets collected in 2000 and 2001 in a hemlock forest in central Massachusetts (Hadley and Schedlbauer 2002) as the instances of tower data and environmental data. We used a linear regression model, with data from independently measured environmental variables (e.g., soil temperature, soil moisture), to estimate carbon flux from the environmental data and to fill in the gaps left in the total data set when we removed the excluded data (Hadley and Schedlbauer 2002). This linear regression model is an analytical process applied to the selected data. In this case, we used the "lm" function in R (version 1.9.0) for regression analysis. All of these data sets and tools are organized into DDGs that comprise a record of the statistical functions and software versions used to create the various instances of row-filled data. If the initial input data sets have been stored, and are Internet-accessible via URLs, then SciWalker can retrieve them and precisely execute the documented sequences of applications, tools, and systems. SciWalker can then capture the resulting data sets and store them wherever the user specifies, returning the URLs to facilitate access by consumers. The DFG and DDG serve as the desired process metadata.

Using SciWalker to compare instances of analyses of eddy covariance data

Different input data sets have been, and might still be, processed in this way. If each is Internet-accessible, then tools such as SciWalker could re-execute the derivation process, with each execution producing different output data sets. Fig. 3 illustrates this conceptually, using a "stacked view" to show the relationship between a data-set type in the DFG and the various data-set instances in different DDGs created by iteratively applying the DFG's description to different input data sets. We note that the stacked view is intended to be functionally useful, in addition to being visually suggestive. A tool such as our SciWalker prototype (see footnote 13) can support the automatic access of a DDG data set through its URL by pointing to and clicking on the visual icon representing it. Thus, a data-set stack is a convenient way to gain easy access to related data sets.

Thus, in the eddy flux example, we varied the value of u^*_{thr} in successive applications of the DFG, and obtained a collection of output data sets and DDGs that could be depicted in this stacked fashion. Using SciWalker and clicking on the stacked icons made it easy to access the various output data sets. In doing so, we found that estimated carbon flux (the result of applying the processes "aggregate data," "segregate data," and "interpolate data" to the tower data and the environmental data) increased nonlinearly with the value of u^*_{thr}. It appeared that C flux approached an asymptote, whereas the fraction of data retained decreased linearly

with u^*_{thr} (Fig. 4). Others have examined effects of u^*_{thr} on estimates of carbon flux (Barford et al. 2001, Saleska et al. 2003, Hollinger et al. 2004), but neither the sets of accepted and excluded data nor the gap-filling models used in these studies are readily accessible. The SciWalker tool, and the resulting analytic webs, provide these items while simultaneously improving the ease and speed of data processing and analysis for us as data set producers. Moreover, the analytic web is a substantial improvement for consumers who would like to validate the reliability of derived results and the data sets on which they are based.

It is important to reiterate that although the DFG and DDG provide essential process metadata, descriptive metadata, such as the EML file for the eddy covariance data in data set HF103 in the Harvard Forest Data Archive (*available online*),[14] are still needed to explain the origins and particulars of the tower data, the environmental data, and the interpolation model. Both process metadata and descriptive metadata are essential components of reliable data sets.

Do We Need More Than DFGs and DDGs?

While the DFG and DDG, and SciWalker, add considerable value for both producers and consumers of data sets, we are not satisfied that they go far enough in protecting consumers from misuse of data sets. To illustrate one of our concerns, note that there is nothing in the DFG specification of Fig. 2 that would prevent a careless (or malicious) consumer from mismatching selected data and excluded data sets with each other. Once a given selection criterion tool or process has been created, only the selected data and excluded data sets derived from it, and a single tower data set, should be used with each other in the subsequent evolution of the DDG. Yet the DFG simply mandates that an instance of selected data and an instance of excluded data are all that is needed to drive forward the derivation of the DDG. While this confusion may seem unlikely, we note that as scientific data processing becomes increasingly complex, the opportunities for such confusions increase. Moreover, the very ease with which users may access processes, tools, and data sets in SciWalker via a single mouse click in itself creates risks. It is desirable to constrain process execution and synthesis of data sets to those data sets for which the execution and synthesis are meaningful. We provide such constraints through the use of a third type of graph, which we refer to as a *process derivation graph* (PDG).

The PDG incorporates a stronger and broader set of semantic features that enables the producer to specify constraints for which data set and process combinations are acceptable and which are not. The PDG's stronger semantics also can specify the full range of possible activities that might need to be undertaken during data-set development, specifically including those that must happen in response to unexpected, or undesired, actions or occurrences that are unwelcome, but not unexpected (e.g., transient failures of equipment or sensors that require resorting to backup systems). A PDG is necessary because our experience has demonstrated that it is difficult to represent processes that must deal with such exceptional conditions in DFGs. Although DFGs are more visually familiar to users, PDGs are often a better basis for data-set production because of their stronger constraints on invalid combinations of input data sets, and their greater ability to deal with processes that incorporate exceptional conditions. Space does not permit a full treatment of PDGs in this paper, but Fig. 1C illustrates the iconography of a language, Little-JIL, that incorporates the desired semantic features required for a PDG. The interested reader is encouraged to learn more about this language and its features from Wise (1998) and Osterweil et al. (2005). Subsequent versions of SciWalker are expected to incorporate facilities for building and using PDGs in concert with DFGs and DDGs.

Discussion

It is our goal that analyses and syntheses of all ecological data, including synthetic data sets, be accompanied by formal process metadata. Developing analytic webs, and a toolkit to produce them, are the first steps in that direction. An essential continuation of the work of this project will be the evaluation and evolution of both the analytic web concepts and the SciWalker prototype through application of both to a range of ecological data sets and synthetic questions. At present we have used SciWalker to create and use only a very small number of analytic webs. The results are encouraging, but point towards needed modifications and enhancements to SciWalker, including the addition of new forms of process metadata.

In particular, we intend to focus more attention upon the use of the analytic web graphs to support consumers. Much of this paper has emphasized ways in which analytic webs can be used to help producers create new scientific data sets, but we believe that there is equal value and motivation for analytic webs to be used to support consumers; for example, the specific details captured in a DDG document, the specific tools that were applied to particular data sets, and in what order the tools were applied. Although this is useful information to a consumer, there currently is no support for using this information to automatically drive the re-execution of a DFG. The DDG contains all the necessary information to replicate this execution, or some variant of this execution, but the consumer, now a producer, would need to manually enter the URLs for each data set and tool employed in the new execution. Since all this information is present in the DDG, SciWalker could provide automated support to facilitate such re-execution.

[14] ⟨http://harvardforest.fas.harvard.edu⟩

Similarly, consumers may want to create a new process definition that is similar to an existing process definition. Currently, SciWalker does not provide capabilities for deriving one process definition from another. Instead the consumer, again turned producer, would have to create a new DFG by copying and modifying an existing DFG. Software engineering approaches, such as version control and configuration management, as well as programming language approaches, such as inheritance, should provide useful capabilities to reuse of process definitions.

We are acutely aware of the fact that there are important pragmatic considerations that could interfere with the potential adoption of an analytic web approach, and we will explore the feasibility of the enhancements described above. For example, even the smallest typographical error in a URL prevents successful access to data sets or tools, and thwarts reproduction of data sets and reuse of processes. Default naming conventions must be employed to help users differentiate between different versions of data sets and processes. In addition, attempts to evaluate the use of alternative tools will require correct understandings of the capabilities of both the original tools and the proposed variants.

In addition to the enhancements described above, we intend to demonstrate that analytic web graphs can be used to determine if inappropriate or unsafe sequences of actions can be performed. For example, Oates and Jensen (1998, 1999) have shown that certain sequences of actions, such as smoothing and interpolation, can lead to statistical results that can be unreliable. Techniques used in software analysis (Dwyer et al. 2004) seem applicable to the determination of whether or not such sequences of activities might be performed during any possible execution of a scientific process. This type of analysis could be performed on a DDG to detect problems after the fact. Alternatively, when applied to a DFG or PDG, this analysis could determine if there is any potential execution that could cause such an erroneous sequence of events to occur. A PDG's ability to capture the exceptional processing steps taken in response to unwelcome contingencies, all too common in real scientific investigation, makes it a suitable subject for rigorous determination of potential risks to validity arising from nonstandard processing activities during exception handling.

Finally, we note that an essential aspect of our future work will be to continue to compare this work to complementary approaches. Current implementations of scientific workflow structures, such as Kepler (Altintas et al. 2004*a*, *b*, Ludäscher et al. 2006), emphasize support for data-set producers, and most seem to base their support upon data-flow graphs. Kepler's DFG structure allows for great generality and flexibility in the specification of the ways in which data sets can be moved between processing nodes, and it is especially effective in supporting the processing of streaming data, such as data produced by sensors and intended for real-time processing. Kepler also incorporates powerful features for support of consumers, such as the production of detailed textual documentation about the derivation of data sets.

But Kepler has a number of drawbacks. The DFG used in Kepler seems to complicate the representation of some iterative processing. It appears intended to provide dynamic support for producers, allowing real-time binding of processing capabilities to nodes even as the process is executing. This allows producers to deal with exceptional conditions of the kind that we addressed above. But since the precise processes followed in these exceptional cases are only documented post hoc, consumers of data sets produced in this way will not know which sequences of tools and processes have been applied in developing data sets until the data sets have actually been produced. Thus, data consumers may find that the data sets coming from a Kepler process will have varying provenances, and some may be incompatible with subsequent processing contemplated by the consumer. In contrast, an analytic web contains a PDG that supports specification of how exceptional conditions are handled, enabling analyzers to document all possible tool execution sequences so that consumers can safely plan their subsequent processing steps. Thus, the additional expressiveness of the PDG, and its potential for supporting more definitive validation of reliability, distinguishes our approach from that taken by Kepler and similar DFG-based systems.

Other systems for scientific workflow use Petri Nets (Peterson 1977) to represent scientific processes (Hoheisel et al. 2004, Zhang 2004). Petri Nets are adept at representing parallelism, and thus seem well adapted to the specification of scientific processes where more than one investigator or team will work concurrently. Petri Nets do not scale well and, like all representations of concurrent behavior, can easily introduce unintended consequences. This is another example where analysis would be useful. Petri Nets, like DFGs, have difficulty representing exception handling, whereas the PDG supports concurrency and exception handling relatively naturally.

In general, an analytic web differs from other scientific workflow systems in that it utilizes multiple graph representations, each of which sheds light on a different aspect of the process. The DDG elaborates on the provenance of each individual data set instance. The PDG is adept at capturing all of the nuances of potentially complex processes. The DFG offers a straightforward view of the process that may be intuitively appealing in its simplicity. Future research will help us to determine whether these three graphs are indeed well adapted to the clear, precise, and complete representation of scientific processes in ways that facilitate the work of both producers and consumers of ecological data.

Acknowledgments

This work was supported by NSF grant CCR-0205575, and is a contribution from the Harvard Forest Long Term Ecological Research (LTER) program. We thank Aimée Classen, Elizabeth Farnsworth, Nick Gotelli, Julia Jones, Matt Jones, Bill Shipley, Kristin Vanderbilt, and four anonymous referees for critical comments that significantly improved the clarity of the manuscript.

Literature Cited

Ailamaki, A., Y. E. Ioannidis, and M. Livny. 1998. Scientific workflow management by database management. Pages 1–10 *in* Proceedings of the 10th International Conference on Scientific and Statistical Database Management. Capri, Italy.

Altintas, I., C. Berkeley, E. Jaeger, M. Jones, B. Ludäscher, and S. Mock. 2004*a*. Kepler: an extensible system for design and execution of scientific workflows. Pages 423–424 *in* 16th International Conference on Scientific and Statistical Database Management. Santorini Island, Greece.

Altintas, I., C. Berkeley, E. Jaeger, M. Jones, B. Ludäscher, and S. Mock. 2004*b*. Kepler: towards a grid-enabled system for scientific workflows. *In* The Tenth Global Grid Forum—The Workflow in Grid Systems Workshop. Berlin, Germany.

Andelman, S. J., C. M. Bowles, M. R. Willig, and R. B. Waide. 2004. Understanding environmental complexity through a distributed knowledge network. BioScience **54**:240–246.

Baldocchi, D. D., B. B. Hicks, and T. P. Meyers. 1988. Measuring biosphere-atmosphere exchanges of biologically related gases with micrometeorological methods. Ecology **69**: 1331–1340.

Barford, C. C., S. C. Wofsy, M. L. Goulden, J. W. Munger, E. H. Pyle, S. P. Urbanski, L. Hutyra, S. R. Saleska, D. Fitzjarrald, and K. Moore. 2001. Factors controlling long- and short-term sequestration of atmospheric CO_2 in a mid-latitude forest. Science **294**:1688–1691.

Belovsky, G. E., D. B. Botkin, T. A. Crowl, K. W. Cummins, J. F. Franklin, M. L. Hunter, Jr., A. Joern, D. B. Lindenmayer, J. A. MacMahon, C. R. Margules, and J. M. Scott. 2004. Ten suggestions to strengthen the science of ecology. BioScience **54**:345–351.

Carswell, F. E., et al. 2002. Seasonality in CO_2 and H_2O flux at an eastern Amazonian rain forest. Journal of Geophysical Research-Atmospheres **107**(D20):8076.

Cramer, W., A. Bondeau, S. Schaphoff, W. Lucht, B. Smith, and S. Sitch. 2004. Tropical forests and the global carbon cycle: impacts of atmospheric carbon dioxide, climate change and rate of deforestation. Philosophical Transactions of the Royal Society of London Series B-Biological Sciences **359**: 331–343.

Dominici, F., A. McDermott, and T. J. Hastie. 2004. Improved semiparametric time series models of air pollution and mortality. Journal of the American Statistical Association **99**:938–948.

Dwyer, M. B., L. A. Clarke, J. M. Cobleigh, and G. Naumovich. 2004. Flow analysis for verifying properties of concurrent software systems. Association for Computing Machinery—Transactions on Software Engineering and Methodology **13**:359–430.

Garcia-Berthou, E., and C. Alcaraz. 2004. Incongruence between test statistics and P values in medical papers. BioMed Central Medical Research Methodology **4**:13.

Ghezzi, C., M. Jazayeri, and D. Mandrioli. 2003. Fundamentals of software engineering, Second edition. Pearson Education, Upper Saddle River, New Jersey, USA.

Goulden, M. L., J. W. Munger, S.-M. Fan, B. C. Daube, and S. C. Wofsy. 1996. Exchange of carbon dioxide by a deciduous forest: response to interannual climate variability. Science **271**:1576–1578.

Grace, J., J. Lloyd, J. McIntyre, A. C. Miranda, P. Meir, H. S. Miranda, C. Nobre, J. Moncrieff, J. Massneder, Y. Malhi, I. Wright, and J. Gash. 1995. Carbon dioxide uptake by an undisturbed tropical rainforest in western Amazonia, 1992 to 1993. Science **270**:778–780.

Grace, J., Y. Malhi, J. Lloyd, J. McIntyre, A. C. Miranda, P. Meir, and H. S. Miranda. 1996. The use of eddy covariance to infer the net carbon dioxide uptake of the Brazilian rainforest. Global Change Biology **2**:209–218.

Hadley, J. L., and J. L. Schedlbauer. 2002. Carbon exchange of an old-growth eastern hemlock (*Tsuga canadensis*) forest in central New England. Tree Physiology **22**:1079–1092.

Helly, J. J., T. T. Elvins, D. Sutton, D. Martinez, S. E. Miller, S. Pickett, and A. M. Ellison. 2002. Controlled publication of digital scientific data. Communications of the Association for Computing Machinery **45**:97–101.

Hoheisel, A. 2004. User tools and languages for graph-based grid workflows. Pages 1–9 *in* Proceedings of Workflow in Grid Systems Workshop in GGF10. Berlin, Germany.

Hollinger, D. Y., J. D. Aber, B. Dail, E. A. Davidson, S. M. Goltz, H. Hughes, M. Y. Leclerc, J. T. Lee, A. D. Richardson, C. Rodrigues, N. A. Scott, D. Achuatavarier, and J. Walsh. 2004. Spatial and temporal variability in forest-atmosphere CO_2 exchange. Global Change Biology **10**:1689–1706.

Hollinger, D. Y., F. M. Kelliher, J. N. Byers, J. E. Hunt, T. M. McSeveny, and P. L. Weir. 1994. Carbon dioxide exchange between an undisturbed old-growth temperate forest and the atmosphere. Ecology **56**:134–150.

IPCC (Intergovernmental Panel on Climate Change). 2001. Climate change 2001: synthesis report. IPCC Secretariat, Geneva, Switzerland.

Jones, M. B., C. Berkeley, J. Bojilova, and M. Schildhauer. 2001. Managing scientific metadata. IEEE Internet Computing, **5**(5):59–68.

Kareiva, P. M. 2002. Applying ecological science to recovery planning. Ecological Applications **12**:629.

Lubchenco, J., A. M. Olson, L. B. Brubaker, S. R. Carpenter, M. M. Holland, S. P. Hubbell, S. A. Levin, J. A. MacMahon, P. A. Matson, J. M. Melillo, H. A. Mooney, C. H. Peterson, H. R. Pulliam, L. A. Real, P. J. Regal, and P. G. Risser. 1991. The sustainable biosphere initiative: an ecological research agenda. Ecology **72**:371–412.

Ludäscher, B., I. Altintas, C. Berkley, D. Higgins, E. Jaeger, M. Jones, E. A. Lee, J. Tao, and Y. Zhao. 2006. Scientific workflow management and the Kepler system. Concurrency and Computation: Practice and Experience. *In press*.

Malhi, Y., A. D. Nobre, J. Grace, B. Krujit, M. G. P. Pereira, A. Culf, and S. Scott. 1998. Carbon dioxide transfer over a central Amazonian rain forest. Journal of Geophysical Research **D24**:31593–31612.

Meehl, G. A., W. M. Washington, J. M. Arblaster, and A. X. Hu. 2004. Factors affecting climate sensitivity in global coupled models. Journal of Climate **17**:1584–1596.

Michener, W. K. 2000. Ecological knowledge and future data challenges. Pages 162–174 *in* W. K. Michener and J. W. Brunt, editors. Ecological data: design, management and processing. Blackwell Science, Oxford, UK.

Michener, W. K., T. J. Baerwald, P. Firth, M. A. Palmer, J. L. Rosenberger, E. A. Sandlin, and H. Zimmerman. 2001. Defining and unraveling biocomplexity. BioScience **51**:1018–1023.

Michener, W. K., J. W. Brunt, J. J. Helly, T. B. Kirchner, and S. G. Stafford. 1997. Nongeospatial metadata for the ecological sciences. Ecological Applications **7**:330–342.

Oates, T., and D. Jensen. 1998. Large data sets lead to overly complex models: an explanation and a solution. Pages 294–298 *in* Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD98), New York, New York, USA, 27–31 August 1998.

Oates, T., and D. Jensen. 1999. Toward a theoretical understanding of why and when decision tree pruning algorithms fail. Pages 372–378 *in* Proceedings of the Sixteenth National

Conference on Artificial Intelligence (AAAI99). Orlando, Florida, USA, 18–22 July 1999.

Osterweil, L. J., A. Wise, L. A. Clarke, A. M. Ellison, J. L. Hadley, E. Boose, and D. R. Foster. 2005. Process technology to facilitate the conduct of science. Pages 401–415 *in* M. Li, B. Boehm, and L. J. Osterweil, editors. Unifying the software process spectrum: international software process workshop, SPW 2005, Beijing, China, revised selected papers. Lecture notes in Computer Science, Springer-Verlag, Berlin, Germany.

Peterson, J. L. 1977. Petri Nets. Association for Computing Machinery Computing Surveys **9**:223–252.

Saleska, S. R., S. D. Miller, D. M. Matross, M. L. Goulden, S. C. Wofsy, H. R. de Rocha, P. B. de Camargo, P. Crill, B. C. Daube, H. C. de Freitas, L. Hutyra, M. Keller, V. Kirchnoff, M. Menton, J. W. Munger, E. Hammond-Pyle, A. H. Rice, and H. Silva. 2003. Carbon in Amazon forests: unexpected seasonal fluxes and disturbance-induced losses. Science **302**: 1554–1557.

Schemske, D. W., B. C. Husband, M. H. Ruckelshaus, C. Goodwillie, I. M. Parker, and J. G. Bishop. 1994. Evaluating approaches to the conservation of rare and endangered plants. Ecology **75**:584–606.

Thornton, P. E., R. B. Cook, B. H. Braswell, B. E. Law, W. M. Post, H. H. Shugart, B. T. Rhyne, and L. A. Hook. 2005. Archiving numerical models of biogeochemical dynamics. EOS **86**:431.

Wise, A. 1998. Little-JIL 1.0 language report. Computer Science Technical Report, University of Massachusetts, Amherst, Massachusetts, USA.

Wofsy, S. C., M. L. Goulden, J. W. Munger, S. M. Fan, P. S. Bakwin, B. C. Daube, S. L. Bassow, and F. A. Bazzaz. 1993. Net CO_2 exchange in a mid-latitude forest. Science **260**:1314–1317.

Zhang, S., N. Gu, and S. Li. 2004. Grid workflow based on dynamic modeling and scheduling, Pages 35–39 *in* Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC2004), Vol. 2. Las Vegas, Nevada, USA.

APPENDIX A

An animation that illustrates the construction by SciWalker of the eddy flux analytic web (*Ecological Archives* E087-079-A1).

APPENDIX B

An animation that illustrates the execution by SciWalker of the eddy flux analytic web (*Ecological Archives* E087-079-A2).

APPENDIX C

Simplified XML for the "create interpolation model" sequence of the analytic web shown in Figs. 2 and 3 (*Ecological Archives* E087-079-A3).