Contents lists available at ScienceDirect

Ecological Informatics

journal homepage: www.elsevier.com/locate/ecolinf

Workflows and extensions to the Kepler scientific workflow system to support environmental sensor data access and analysis

Derik Barseghian [a],*, Ilkay Altintas [b], Matthew B. Jones [a], Daniel Crawl [b], Nathan Potter [c], James Gallagher [c], Peter Cornillon [d], Mark Schildhauer [a], Elizabeth T. Borer [e], Eric W. Seabloom [e], Parviez R. Hosseini [f]

[a] National Center for Ecological Analysis and Synthesis, 735 State Street, Suite 300, Santa Barbara, CA, USA
[b] San Diego Supercomputer Center, University of California San Diego, 10100 Hopkins Drive, La Jolla, CA, 92093-0505, USA
[c] OPeNDAP, Inc., 165 Dean Knauss Dr., Narragansett, RI, 02882, USA
[d] Graduate School of Oceanography, University of Rhode Island, South Ferry Road, Narragansett, RI, 02882, USA
[e] Department of Zoology, Oregon State University, Corvallis, OR, 97331, USA
[f] Wildlife Trust, 460 West 34th Street, 17th Floor, New York, NY, 10001, USA

ARTICLE INFO

Article history:
Received 9 February 2009
Received in revised form 15 August 2009
Accepted 29 August 2009

Keywords:
Scientific workflows
Sensors
Near real-time data access
Data analysis
Terrestrial ecology
Oceanography

ABSTRACT

Environmental sensor networks are now commonly being deployed within environmental observatories and as components of smaller-scale ecological and environmental experiments. Effectively using data from these sensor networks presents technical challenges that are difficult for scientists to overcome, severely limiting the adoption of automated sensing technologies in environmental science. The Realtime Environment for Analytical Processing (REAP) is an NSF-funded project to address the technical challenges related to accessing and using heterogeneous sensor data from within the Kepler scientific workflow system. Using distinct use cases in terrestrial ecology and oceanography as motivating examples, we describe workflows and extensions to Kepler to stream and analyze data from observatory networks and archives. We focus on the use of two newly integrated data sources in Kepler: DataTurbine and OPeNDAP. Integrated access to both near real-time data streams and data archives from within Kepler facilitates both simple data exploration and sophisticated analysis and modeling with these data sources.

1. Introduction and motivation

Scientific workflows are representations of the processes involved in accomplishing a scientific analysis. They combine data and computational procedures into a configurable, structured set of steps that implement semi-automated computational solutions to a scientific question. A scientific analysis, depending on its focus, can involve a number of ad-hoc processes that a scientist may use to get from raw data to publishable results.

One system for creating and using scientific workflows is Kepler. Kepler streamlines the workflow creation and execution process so that scientists can design, execute, monitor, re-run, and communicate analytical procedures with minimal effort. Kepler allows scientists to capture workflows in a format that can easily be exchanged, archived, versioned, and executed (Altintas et al., 2004a). Kepler is a free, open-source software that works on several popular operating systems, and can be used to couple disparate execution environments, e.g. providing linkages of a model written in C with network-distributed input data sources, and passing these outputs on to the R statistical environment for graphical presentation.

Scientific workflow systems have been used for accessing data from a variety of sources, including database systems (Altintas et al., 2004a), grid systems (Altintas et al., 2003, 2005; Deelman et al., 2005; Ludäscher et al., 2006; Taylor et al., 2007b), and Web Services (Altintas et al., 2004b). In addition, the Kepler workflow system has built-in tools for accessing heterogeneous environmental data by using details about data content and structure from metadata descriptions available in the Knowledge Network for Biocomplexity (KNB), a large-scale, distributed data system. Pennington et al. (2007) have shown that these technologies along with several others in Kepler can be used to solve scientific problems that require access to data in existing archives. Today, scientific workflows are widely being adopted by the scientific and engineering communities because of the advantages they provide beyond those of existing scripting and visual programming tools (Taylor et al., 2007a; Podhorszki et al., 2007; Bowers et al., 2008; Callaghan et al., 2008). However, new challenges arise when building workflows using sensor data from heterogeneous systems. Some of these challenges include: rapid exploratory analysis

* Corresponding author.
E-mail addresses: barseghian@nceas.ucsb.edu (D. Barseghian), altintas@sdsc.edu (I. Altintas), jones@nceas.ucsb.edu (M.B. Jones), crawl@sdsc.edu (D. Crawl), ndp@opendap.org (N. Potter), gallagher@opendap.org (J. Gallagher), pcornillon@gso.uri.edu (P. Cornillon), schildhauer@nceas.ucsb.edu (M. Schildhauer), borer@science.oregonstate.edu (E.T. Borer), seabloom@science.oregonstate.edu (E.W. Seabloom), hosseini@wildlifetrust.org (P.R. Hosseini).

1574-9541/$ – see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.ecoinf.2009.08.008

of the data; monitoring the quality of streaming data to ensure the "health" of a network; visualization and on-the-fly analysis of streaming data; recording, re-running, and sharing procedures that utilize sensor data; and using these data in conjunction with preexisting data housed in heterogeneous data stores.

The Realtime Environment for Analytical Processing (REAP) project addresses several of these technical challenges related to accessing and using heterogeneous sensor data from within the Kepler scientific workflow system. In this paper we describe extensions to Kepler that allow users to easily access and utilize streaming data from sensor networks and archived data from the KNB and other data networks. We develop Kepler interfaces that may be leveraged by others, augment existing Kepler infrastructure, and create workflows that use these extensions to fulfill the needs of two very different use cases and to serve as examples for future Kepler users with similar needs. We begin with a brief scientific background on the use cases, describe related work and technologies, and then describe our work that addresses the software-related needs of the use cases. We conclude with a summary and discussion of areas of future work.

2. Scientific use cases

Our initial development efforts have primarily been focused on the needs of two very different scientific use cases: 1) a terrestrial ecology use case in which near real-time data from terrestrial micrometeorological sensors will aid in a study of plant populations and their susceptibility to viral pathogens, and 2) an oceanography use case that will compare and match-up remotely sensed sea surface temperature (SST) data. While these use cases require real, multi-step analyses with specific data and computational requirements, we have attempted to address the needs of these use cases in ways that promote the re-use and extension of our work.

2.1. Terrestrial ecology

Non-native annual grasses currently dominate the west coast of the United States in areas historically dominated by perennial native bunchgrasses and forbs (Baker, 1978; Jackson, 1985). The terrestrial ecology use case focuses on the evaluation of the hypothesis that this widespread invasion by non-native annual grasses in the U.S. Pacific States is mediated by a suite of viral pathogens in the barley and cereal yellow dwarf virus group. This viral group infects both annual and perennial grasses and is carried by several common aphid species (Halbert and Voegtlin, 1995). Although mathematical models and field observations are consistent with this hypothesis (Borer et al., 2007), a thorough hypothesis test requires, in part, a detailed understanding of grass community phenology, which can be derived from a sensor network that accurately provides information about ambient meteorological conditions, soil moisture, and biomass accumulation in the grass canopy—all commonly measured factors in terrestrial ecology studies.

For this study we have deployed hardware that is commonly used by the ecological community to develop software for use with a realistic set of sensor equipment. A Campbell Scientific weather station was deployed at the Baskett Slough National Wildlife Refuge in Dallas, Oregon. The weather station includes a datalogger, a 900-MHz spread spectrum radio, and a battery power supply within a weatherproof enclosure. The enclosure, a directional antenna, and a solar panel that serves as power source are mounted on a six-foot tripod (Fig. 1). Eight sensors attached to the datalogger are mounted on the tripod and ground nearby. A program written in the Campbell Scientific CRBasic language runs on the datalogger, sampling data from the sensors at regular intervals, and a computer at a nearby U.S. Fish and Wildlife Service building periodically establishes radio communication to the weather station and downloads the newly collected data.

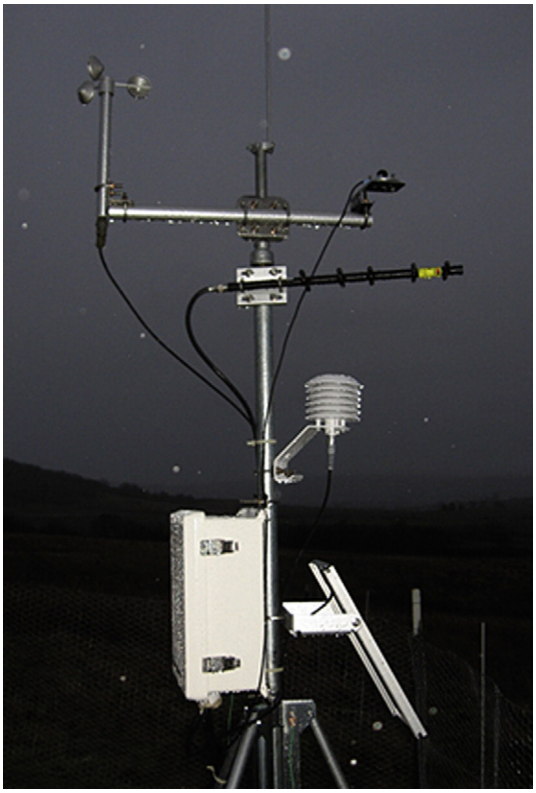


Fig. 1. Meteorological station for the terrestrial ecology use case. From left to right, starting at the top: anemometer, lightning rod, quantum point sensor, directional antenna, relative humidity and temperature sensors within gill radiation shield, enclosure and solar panel.

This use case requires easy-to-use analytical software to support issues commonly experienced in ecological research, e.g., the need for the analysis and modeling of sensor network data in near real-time to detect local thresholds (e.g., hours exceeding developmental temperature thresholds for aphids), long-term trends (e.g., within- and among-season soil moisture trends), and significant events (e.g., timing of peak plant biomass). Having access to the results of such analyses in near real-time accelerates study, and facilitates managerial tasks such as optimally planning field trips. As with many such studies, to ensure the collection of a reliable dataset, there is also a need for software that monitors and analyzes incoming data, sending alerts for events such as power or sensor malfunction. In addition, hypothesis testing for this use case, like many ecological studies, requires integration of sensor data with archived data: in this case to assess the relative impacts of disease, plant composition, rainfall, temperature, and soil nutrients on competitive interactions among grasses. With the addition of the extensions and workflows we describe in this paper, Kepler now provides a freely available software solution that fulfills these requirements.

2.2. Oceanography

Sea surface temperature (SST) fields are among the most, if not the most, broadly used observational datasets related to the ocean. They are used to delineate water masses, as indicators of the near surface density field from which geostrophic estimates are made for near surface currents (Kelly et al., 1999; Dong and Kelly, 2003), as tracers for estimating surface currents (Emery et al., 1992; Bowen et al., 2002), in support of *in situ* observations (Cornillon et al., 1988), in support of operational activities such as search and rescue, and in biological applications for a variety of species related studies (Hare et al., 2002; Baumgartner et al., 2003; Barcena et al., 2004). In addition, SST fields play an important role in air–sea interactions both at large scales (Chelton et al., 2001; O'Neill et al., 2003) and at small scales (Park and Cornillon, 2002; Song et al., 2004; Park et al., 2006) and they are beginning to be used in operational systems by the National Weather Service (Ginis, personal communication, 2006). Because of their importance in oceanography and meteorology, for both research and operational uses, significant effort has been devoted over the past twenty-five years to the development of SST fields using data obtained from satellite-borne instruments, model output, and *in situ* surveys. As a result, scientists interested in using SST data in their work face a bewildering array of SST products from which to choose, generally with little to no guidance as to which product(s) is (are) the most appropriate for their needs.

Because SST datasets can be very large, evaluations or comparisons are performed on small subsets of the data, generally scattered in space and time. The collection of these subsets is referred to as a *Match-up database*. Match-up databases may be generated from a mixture of point observations, such as those obtained from ships or buoys, and arrays, typically obtained from satellite-based observations or numerical models, or they may be generated from array datasets only. Our focus is on the latter. There are two extremes for the generation of such match-up databases: 1) they are predefined for a collection of 3-D (latitude, longitude, and time) datasets, or 2) they are built dynamically based on user input. Disadvantages of prebuilt match-up databases are that the user is constrained to datasets for which the database has been built and to the predefined density of spatial and temporal match-ups. Building the match-up database on-the-fly addresses these concerns by allowing the user to include all datasets of potential interest and to define the density of match-up locations. One disadvantage of this approach is that if a number of datasets are being compared and the user requests a relatively high density of match-up locations, building the match-up database can take a significant amount of time. Despite this drawback, we have decided to focus our use case on comparisons made using a match-up database built on-the-fly. This is especially important given the increasing number of on-line SST datasets: it is very likely that users will want to include “new” datasets in their analyses, datasets that may not been included in a prebuilt match-up database.

This use case challenged us to extend Kepler to solve the common problem of comparing heterogeneous SST datasets accessible via the web from a number of different data providers. There is widespread need among scientists for such a tool that allows them to select the suite of datasets to be included in a comparison, to build a match-up database, and then to perform and visualize statistical analyses of this database.

3. Related work

A variety of related works exist dealing with providing, accessing, analyzing and describing sensor data and remote environmental datasets. In this section we review work that is especially relevant to ours. We first describe three technologies that we leverage in our use cases to access remote datasets, and then contrast our work with the Sensor Web Enablement initiative from the Open Geospatial Consortium.

3.1. DataTurbine

In the terrestrial ecology use case, we push our sensor measurements into a DataTurbine server. DataTurbine is an open-source data streaming middleware that provides a robust and generic interface for buffering and accessing real-time and user-selected time ranges of data from a diverse set of sensors. DataTurbine also provides server mirroring capabilities and the ability to link servers together in parent–child relationships (Tilak et al., 2007).

In DataTurbine terminology, data providers are called *Sources*, and data consumers *Sinks*. The DataTurbine API provides means for developing sink and source applications to easily push and pull numeric, binary, and image data (Tilak et al., 2007). After purchasing our weather station, we developed a DataTurbine source to parse and push our numeric terrestrial sensor data into a DataTurbine server. In DataTurbine, data and their associated timestamps are stored together in an independently accessible *Channel*. The data collected from our weather station forms 13 channels, e.g., one channel for air temperature and two for volumetric water content. After being pushed into our publicly accessible DataTurbine, anyone running a sink client may access the data over the Internet. Within Kepler we have developed a DataTurbine sink, so data from DataTurbine servers may be pulled directly into Kepler workflows.

3.2. KNB Metacat

Metacat is a “network-enabled database framework that lets users store, query, and retrieve XML documents with arbitrary schemas in SQL-compliant relational database systems” (Jones et al., 2001). Metacats are an integral part of the infrastructure in the Knowledge Network for Biodiversity (KNB). Many ecological datasets are stored in Metacat systems around the world and these are directly accessible from within Kepler. A workflow author can use the Kepler Data Search panel to find and use data of interest from these systems.

Many Metacat datasets consist of ecological data described in Ecological Metadata Language (EML) (Fegraus et al., 2005). Kepler parses and presents this metadata for the end-user, enabling enhanced use of Metacat data resources in scientific workflows. For the terrestrial ecology use case, we have developed workflows that use EML data from remote Metacat servers in conjunction with near real-time sensor data from a DataTurbine server.

3.3. OPeNDAP

Many of the data sources used in the oceanography use case are accessed using the OPeNDAP DAP protocol. OPeNDAP provides “a framework that simplifies all aspects of scientific data networking”, part of which is the Data Access protocol (DAP) (Gallagher et al., 2007). The DAP is a network protocol for access to data organized as name–datatype–value tuples. It provides a way to access individual variables within data stores (e.g., files) so that only those variables requested, or parts of those variables, are transferred to the client. This is accomplished using a *Constraint Expression* which describes how to subset the data source. Because data accessed using DAP are often stored in many types of files or databases, often unique to a particular organization, all transfers take place using the DAP data model to represent the data store and its contents. The servers provide the transformations between the local storage form and the DAP network representation of data. Thus a DAP-aware client can read from any DAP server knowing that, regardless of the actual form in which data are stored, it can manipulate the contents using the DAP and its single data model. The protocol has been in use since 1995 by the Distributed Oceanographic Data System (DODS) (Gallagher and Milkowski, 1995) and subsequently by many other projects and groups.

In the oceanography use case, the DAP protocol and OPeNDAP software are used to subset and retrieve data from several DAP servers into Kepler workflows for further processing.

3.4. OGC Sensor Web Enablement

The Open Geospatial Consortium (OGC) Sensor Web Enablement (SWE) initiative is “focused on developing standards to enable the discovery, exchange, and processing of sensor observations, as well as the tasking of sensor systems" (Botts et al., 2007). OGC defines the Sensor Web as "web accessible sensor networks and archived sensor data that can be discovered and accessed using standard protocols and application program interfaces (APIs)" (Botts et al., 2007). The SWE initiative has established a number of candidate OpenGIS Specifications, including Observations and Measurements (O&M), Sensor Model Language (SensorML), Transducer Markup Language (TransducerML or TML), Sensor Observation Service (SOS), Sensor Planning Service (SPS), Sensor Alert Service (SAS), and Web Notification Service (WNS) (Botts et al., 2007).

Our work is a complementary effort to the SWE initiative; while SWE is focused on the development of standards, we are focused on providing scientists, network engineers, and laypersons the ability to access and interact with sensor data and services, possibly described by these emerging standards, from within a scientific workflow environment. We give examples of planned connections between our work and the Sensor Web Enablement initiative in section 5.1.

4. Use case related Kepler development

4.1. Introduction to Kepler workflows

First, we introduce some critical terminologies that underlie the structure of Kepler scientific workflows. In Kepler, workflow authors use a graphical user interface to implement an analytical procedure by connecting together a series of workflow components, called *Actors*, through which data are processed. Actors that may contain a hierarchy of other actors are called *Composites*. Parts of actors that receive *Tokens*, which encapsulate single or multiple data or messages, are called *Ports*. *Directors* control the execution of workflows, and in a typical, simple workflow, one director manages the execution of one set of actors. *Parameters* are settings that a user may create and configure, e.g. to serve as arguments to an actor.

4.2. Terrestrial ecology workflows in Kepler

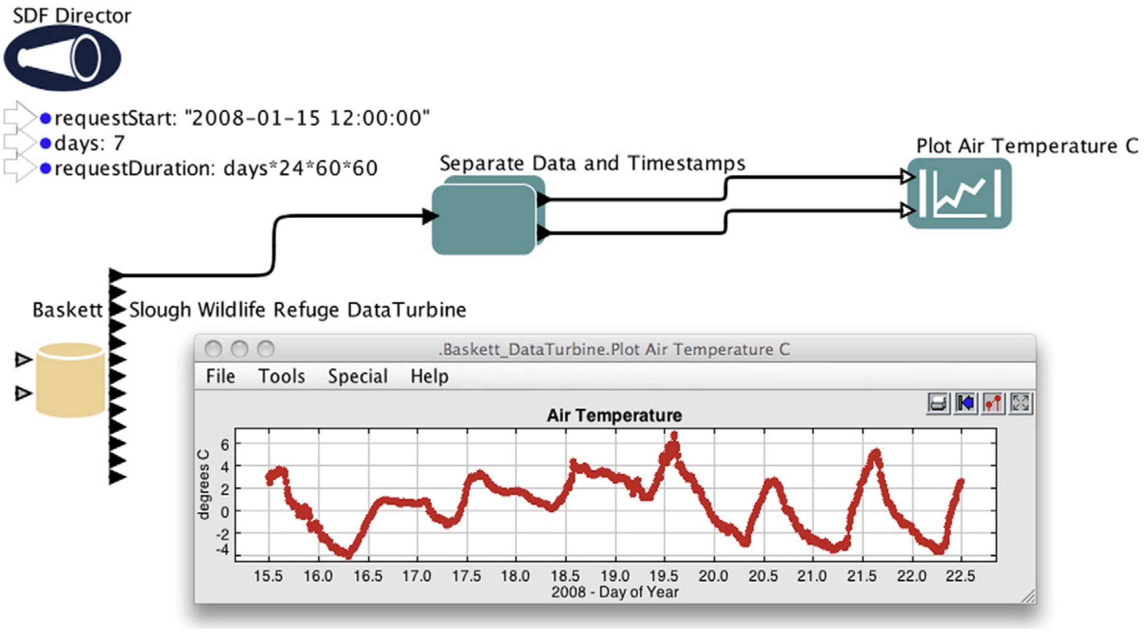
Many of the workflows for the terrestrial ecology use case require access to the sensor data that is being made available in our DataTurbine server. To facilitate access to such data from within Kepler, we have developed a Kepler DataTurbine actor that exposes data from a DataTurbine server to downstream workflow components. A Kepler workflow author configures this actor to connect to a specific DataTurbine server. The actor then automatically generates its output ports, each corresponding to a data channel that exists in the server. The user then specifies a time range of interest, connects output ports of the DataTurbine actor to other actors, and operates on the output data within their workflow. For efficiency, only data for those output ports that are connected to other actors are requested from the server. A specific time range of data may be requested, or numerous such time ranges may be requested via iteration.

DataTurbine also provides data request modes for streaming data in real-time, and support for these modes within Kepler has also been developed. Streaming modes are useful for "headless" (without a graphical user interface), continuously running workflows that provide notification when events of interest or problems occur. For example, a workflow author may use these modes to start a continuously running workflow that monitors a data stream for events such as sensor malfunction.

We have also developed a feature in the DataTurbine actor to fill in missing data, since it is not uncommon for real-world data to have gaps. The user may turn this option on or off; if it is on, the actor attempts to identify the sampling rate of the stream and fills any gaps with pairs of timestamps and empty data points. This yields a more uniform stream that can be easier to operate on within certain types of workflows, for example those with actors that require, in advance of execution, the number of incoming tokens.

The simple workflow and its resulting plot in Fig. 2 illustrate an easily created exploratory analysis. The workflow author has configured the DataTurbine actor to use our DataTurbine server and has specified a time range of interest (seven days starting at noon, Jan 15, 2008). The channel requested, Air Temperature, is split into its data and timestamp components and then plotted. The Synchronous Dataflow (SDF) director is used since the workflow is a simple sequential procedure that does not require dynamic scheduling.

A simple workflow for visualizing the data is often the first step a new user takes before building more complex workflows. For the



●requestStart: "2008-01-15 12:00:00"
●days: 7
●requestDuration: days*24*60*60

Fig. 2. An exploratory analysis workflow that plots sensor data from a DataTurbine server. There are three actors: a DataTurbine data source actor (with 13 output ports), a composite actor that separates a DataTurbine channel into its Data and Timestamp components, and a plotting actor that plots the time series. Three relations ("links" along which tokens flow) connect the actors. The Synchronous Dataflow (SDF) director controls execution of the workflow. For convenience, three parameters positioned beneath the director allow for easily changing the time range of data that the DataTurbine actor will output during execution. The resultant interactive plot is also shown.

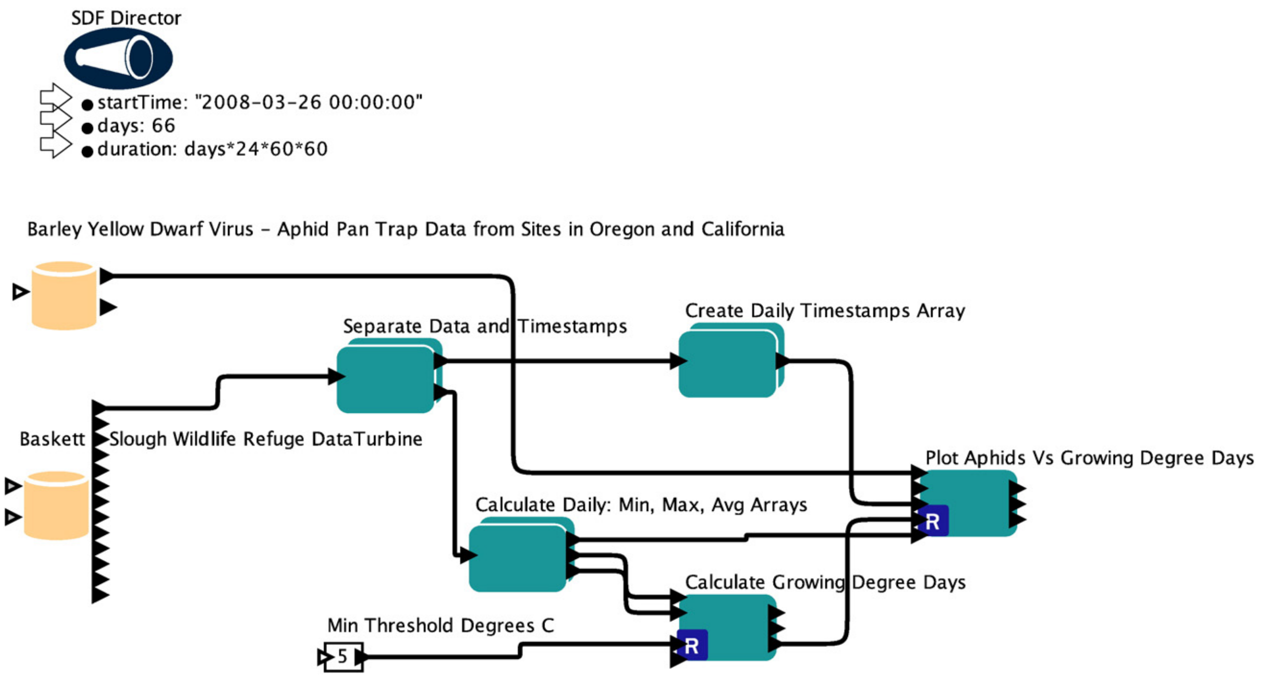

SDF Director
startTime: "2008-03-26 00:00:00"
days: 66
duration: days*24*60*60
Barley Yellow Dwarf Virus - Aphid Pan Trap Data from Sites in Oregon and California
Separate Data and Timestamps
Create Daily Timestamps Array
Baskett Slough Wildlife Refuge DataTurbine
Plot Aphids Vs Growing Degree Days
Calculate Daily: Min, Max, Avg Arrays
R
Calculate Growing Degree Days
Min Threshold Degrees C
5
R

Fig. 3. Post-hoc analysis workflow in which an EML formatted dataset found in the KNB is compared against sensor data.

terrestrial ecology use case, we have developed three categories of more complex workflows that are critical to addressing specific scientific questions. However, these workflows also demonstrate capabilities that are broadly relevant to any researcher using near real-time data. The first are event detectors, analyzing incoming streaming sensor data to detect events such as sensor malfunction, or grass emergence and initiation of growth. The second group of workflows provide quality assurance filters, processing incoming sensor data through a series of criteria to produce "higher level" derived data products that may be archived for use in post-hoc analyses. The third set of workflows is for post-hoc analysis of data, representing a series of analyses and models that combine sensor data with archived data, e.g., from experimental treatments to assess the relative effects of fertilization and disease on competitive exclusion by the annual grasses described in section 2.1.

One example from the first category, event detection, is a workflow that analyzes our meteorological station's battery power level and sensor data outputs and sends email warnings if data points

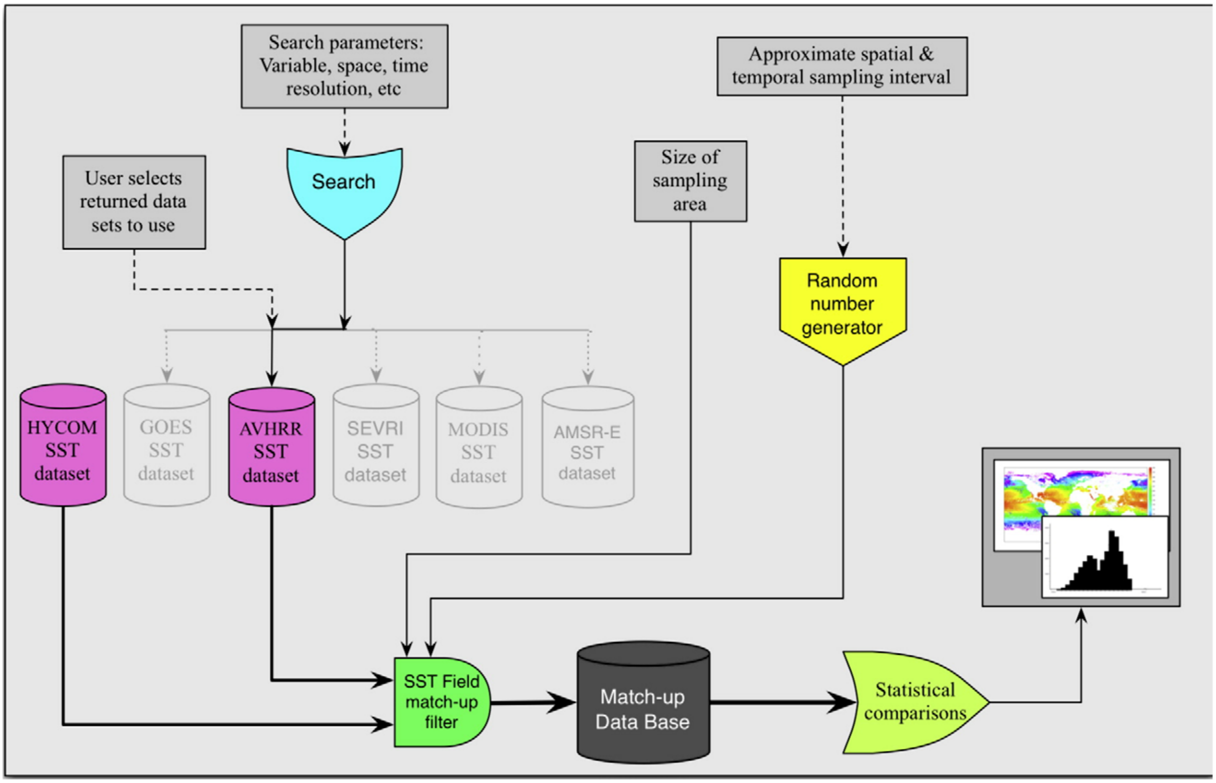


Fig. 4. Schematic of a sample application of the SST Comparison procedure.

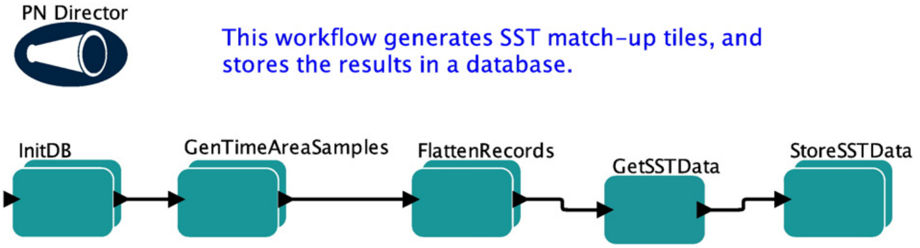

PN Director
This workflow generates SST match-up tiles, and stores the results in a database.
InitDB
GenTimeAreaSamples
FlattenRecords
GetSSTData
StoreSSTData

Fig. 5. The "Build Tiles" workflow. The workflow first chooses the set of times and tile locations to build the match-up dataset. Next, it retrieves the SST measurements from OPeNDAP servers and stores them in a local SQL database. The PN director used in this workflow runs each actor in parallel, thereby decreasing the overall workflow execution time.

fall outside specified thresholds. On receiving such an alert, a user may look at the data more closely to determine if any action is necessary (e.g., a trip to the site for repair), thus minimizing periods of data loss or the collection of poor quality data.

An example from the second category—workflows that generate higher-level derived data products—is a workflow that operates on the photosynthetically active radiation data collected from our meteorological station's light sensors. This workflow requests light sensor data from our DataTurbine server, carries out a series of quality checks (e.g., checking if the data are within accepted ranges and if the ambient, "above canopy" data values are greater than those from the sensors along the ground), and then outputs "cleaned" and "error" datasets. This workflow uses a set of RExpression actors (the RExpression actor provides integrated access to the R language and environment for statistical computing and graphics (R Development Core Team, 2009)), leveraging R's statistical and data-manipulation functions.

By adding the ability to access streaming data from DataTurbine servers from within Kepler, a new set of options is available to users, not only for using these data on their own, but in novel combination with preexisting datasets. Kepler now provides a unified environment within which a user may analyze streaming data in near real-time against preexisting datasets. An example from the terrestrial ecology use case for such post-hoc analysis is the workflow shown in Fig. 3, in which archived, on-line data (an EML formatted dataset from the KNB Metacat: the number of aphids caught in pan traps at Baskett Slough Wildlife Refuge and other sites), is plotted against streamed sensor data (Baskett Slough Wildlife Refuge meteorological sensor data pulled from our DataTurbine server).

4.3. Oceanography workflows in Kepler

To meet the needs of the oceanography use case, we have developed workflows to statistically compare a suite of SST datasets accessible via OPeNDAP. The steps—selection, acquisition, and analysis—of this suite of workflows are outlined below in the context of a simple example. Although the workflows have been designed for SST datasets, there is nothing that constrains the measurements to SST; the procedure will work for any collection of geospatial time series datasets representing a given variable.

Fig. 4 is a rendering of the various components required for a simple SST comparison scenario. In this scenario, the user uses the oceanographic workflows to compare SST fields output from the Hybrid Coordinate Ocean Model (HYCOM), an ocean general circulation model (OGCM), with those available from the Pathfinder v.5 dataset, based on Advanced Very High Resolution Radiometer (AVHRR) retrievals.

Initially the user specifies the HYCOM SST dataset (this is the primary dataset in this example), the parameter of interest (SST in this case), the spatial and temporal ranges from which to build the match-up database, the fraction of SST fields to be sampled and the fraction of the area of each field to be sampled, and the number of tiles (subareas) that are to be used for this area. Known SST datasets are checked for those that meet the search parameters and the results are presented to the user who then specifies which ones are to be used, such as the AVHRR dataset in this case.

The second step in this scenario generates the match-up database, and is implemented by the workflow shown in Fig. 5. Using the specified temporal sampling fraction, the appropriate number of HYCOM SST fields is randomly selected and the AVHRR field nearest in time to each of the selected HYCOM fields is identified. The workflow also calculates the area of each tile and randomly selects the spatial center of each tile. As currently configured, the location of tiles is randomly selected for each instance in time; i.e., the tiles are located at different places from one temporal sample to the next. This could be changed so that the spatial location of tiles is the same at all times or this could be a user specified option. Once the times, tile sizes, and tile locations are known, the workflow acquires

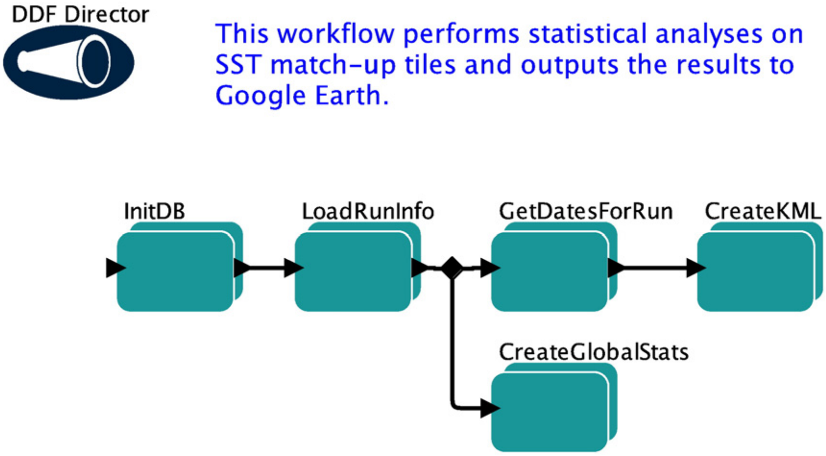
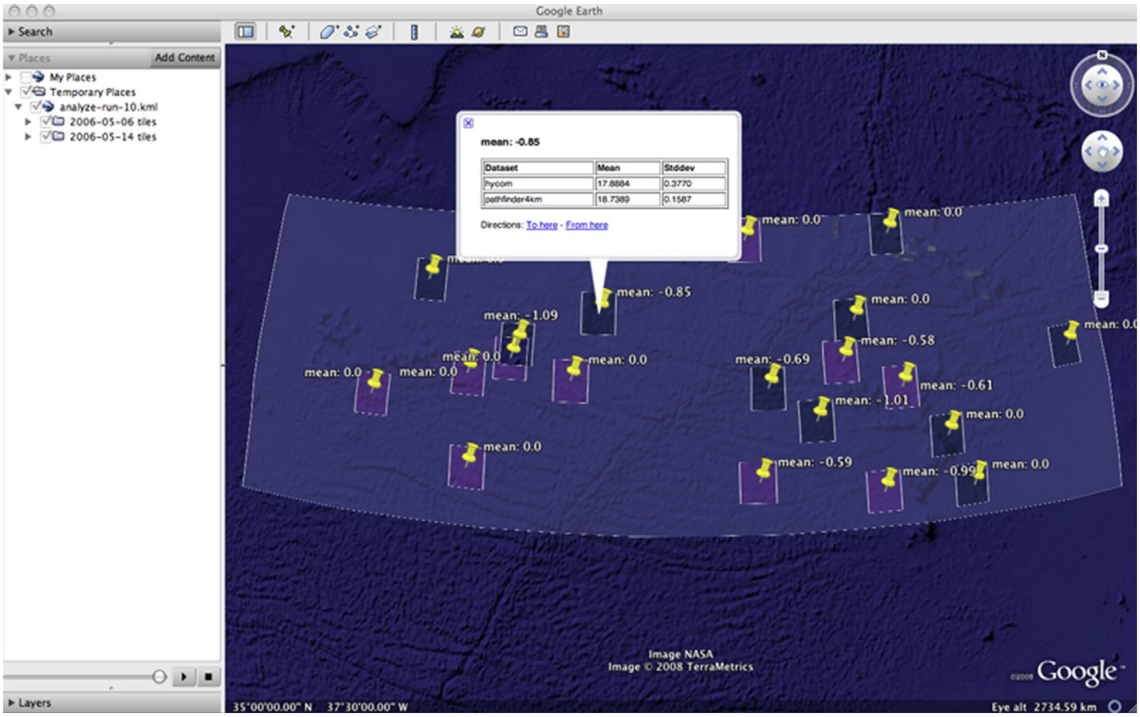

DDF Director
This workflow performs statistical analyses on SST match-up tiles and outputs the results to Google Earth.
InitDB
LoadRunInfo
GetDatesForRun
CreateKML
CreateGlobalStats

Fig. 6. The Analyze Tiles workflow analyzes a single run of the Build Tiles workflow. The sizes, locations and times of the match-up tiles are written to a KML file so that they may be viewed in Google Earth. This workflow uses the DDF director since it allows actors to write and read variable amounts of data.



mean: -0.85

Dataset	Mean	Stddev
hycom	17.8884	0.3770
pathfinder4km	18.7389	0.1587

Directions: To here - From here

Fig. 7. The result of an execution of the workflows in Figs. 5 and 6 is shown above, displayed in Google Earth. Specific points display the difference in mean temperatures between two datasets.

first the data from one dataset and then from the other. As currently configured the datasets are both assumed to be remote and accessed via OPeNDAP, but the workflow could be modified to build the match-up database from a local dataset(s) and/or a remote dataset(s). The match-up dataset is written to a relational database.

In the third and final step, the "Analyze Tiles" workflow analyzes the data stored in the match-up database (Fig. 6). Currently the analyses are very simple calculations of the mean and standard deviations of the SST values in each tile for the two datasets as well as the difference between the means. This workflow generates a KML file based on the results of these analyses, which can be displayed in Google Earth (Fig. 7).

A number of enhancements to Kepler were made to implement these workflows that are valuable beyond this use case. For example, to import satellite derived sea surface temperature data into Kepler, we developed an OPeNDAP actor that provides access to data served by any Data Access Protocol (DAP) 2.0 compatible data source. The OPeNDAP actor reads data from a single DAP data server and provides that data as either a matrix (1×N, or N×M) or an array of more complex dimensionality for processing by downstream actors in a Kepler workflow.

Each DAP server provides (serves) many data sources and each of those data sources can be uniquely identified using a URL in a way that is similar to how pages are provided by a web server. The OPeNDAP actor takes as configuration parameters the URL to a specific data granule available on a server and an optional constraint expression (CE). Based on the URL and optional CE, the actor configures its output ports to match the variables to be read from the data source.

Additionally, a new Kepler data type for timestamps was created, along with actors to create timestamps, calculate the difference between timestamps, choose a random timestamp within a time span, and convert timestamps to formatted strings. The oceanography workflows use the timestamp data type and related actors to retrieve randomly selected time-slices of SST measurements.

4.4. Data source handling in Kepler

We have now demonstrated access to two new types of data systems within Kepler: DataTurbine data streams and data stores available through OPeNDAP. This significantly enhances the data systems that scientists can utilize in Kepler (Table 1), but emphasizes one of the primary remaining shortcomings: customized actors for each data source.

Table 1
Some Kepler data source actors.

Actor Name	Data System	Metadata Format	Data Format
EML200DataSource	EarthGrid/Metacat	Ecological Metadata Language (EML)	Various (CSV, raster images, vector images)
DarwinCoreDataSource	DiGIR	–	DarwinCore
DatabaseQuery	JDBC	–	Relational
OrbPacketObjectSource	AntelopeORB	–	Orb packets
DataTurbine	DataTurbine	–	Named timestamp/data channels
OpendapDataSource	OPeNDAP	OPeNDAP DDS	OPeNDAP data model
SRBSGet	Storage Resource Broker (SRB)	Uncontrolled name–value pairs	Various
FTPClient	FTP	–	Various
URLToLocalFile	HTTP	–	Various
GridFTP	GridFTP	–	Various

No or user-defined metadata format is denoted by a dash.

Because each data system has its own actor with parameters customized for use with that particular data system, it is difficult for scientists to utilize these actors without prior familiarity with each of the corresponding data systems. For example, to use the OPeNDAP actor, one must know and understand the OPeNDAP URL syntax, while using the DataTurbine actor requires understanding how to constrain incoming data streams based on a timestamp and duration value. Neither of these is easy to handle by a person unfamiliar with that system.

5. Discussion

Kepler workflows can now be used to accomplish the full suite of analysis and modeling procedures employed in our two use cases, including accessing both streaming sensor data and archived historical data via many of the data access protocols in use today. As a consequence, Kepler represents one of the few analytical environments in which effective data access is combined with formal specification of an analysis to allow one to completely and accurately archive an analysis in an executable format. In contrast, many analytical systems such as R and Matlab currently provide only limited integrated support for accessing scientific data networks, typically via various relational database access clients. By developing actors that can read data directly from sources such as DataTurbine and OPeNDAP, Kepler provides tools useful to a broad set of data providers and consumers, and addresses needs beyond those specific to our example use cases. Although we have designed workflows that use the DataTurbine actor to retrieve data from a server hosting terrestrial meteorological data, and the OPeNDAP actor to retrieve data from OPeNDAP servers hosting SST data, a user may easily configure these actors to access different servers with different types of data. These new data source actors may be used in conjunction with the many other data systems supported, for the construction of workflows that can, as our work demonstrates, vary widely in purpose and complexity.

Our workflows also serve as useful starting points for a scientist or network manager using sensor network data. We provide examples of exploratory analyses, monitoring and alerting, visualization, and more complex analyses creating higher-level data products. After creating such workflows, the user reaps the benefits of Kepler: the workflow is easily shared, archived, modified, or re-run. When opened in Kepler, a workflow itself provides a visual depiction of an analysis, which can be flexibly annotated and organized to appeal and be comprehendible to a wide audience.

5.1. Future work

We envision several areas of future work that will continue to streamline and improve the analytical process for scientists using workflows. Based on our work with DataTurbine and OPeNDAP, we plan to develop more general inspection, monitoring, and control interfaces that work with other common sensor middleware software such as Boulder Real Time Technologies' Antelope software. With these interfaces, for example, scientists will be able to more easily browse data and be alerted when events of interest occur, and network engineers will be able to monitor the performance of deployed sensors or adjust data sampling rates.

Another area of work will focus on effective management of sensor stream data by using workflows for quality assurance to create high-quality datasets that are deposited in long-term archives. We plan to build workflows that format streaming sensor data into EML and then archive these data as they progress through quality assurance and processing steps into a Metacat. For example, it will be possible for a network engineer to request “level 0” or “raw” data from a Metacat, while a scientist might request a “cleaned” data product.

Additionally, we will investigate integrating Kepler with the Sensor Web Enablement initiative so that scientists can access data that are exposed via these emerging standards from within Kepler. For example, we could enable workflow authors to easily obtain and use data from a Sensor Observations Service, and provide mechanisms for using data that are organized using the Observations and Measurements specification.

As discussed in section 4.4, the existing Kepler data source actors present workflow developers with many different types of interfaces. To address this complication, as we develop new data source actors, we will unify and generalize those that already exist, shielding workflow authors whenever possible from underlying technological details, and leaving them with simpler sets of options from which to choose. This work will be critical to improving Kepler's usability for scientists that may not be familiar with the various data access protocols that are in use. In the meantime, simply having access to the many datasets available through various data access protocols from within a single workflow system should result in significantly increased capabilities to efficiently access, monitor, analyze and integrate diverse environmental information.

Acknowledgements

This material is based upon work supported by the National Science Foundation under award 0619060 for the REAP project, and by the National Center for Ecological Analysis and Synthesis, a Center funded by NSF (Grant #DEB-0553768), the University of California, Santa Barbara, and the State of California.

References

Altintas, I., Bhagwanani, S., Buttler, D., Chandra, S., Cheng, Z., Coleman, M., Critchlow, T., Gupta, A., Han, W., Liu, L., Ludäscher, B., Pu, C., Moore, R., Shoshani, A., Vouk, M., 2003. A modeling and execution environment for distributed scientific workflows. 15th Intl. Conference on Scientific and Statistical Database Management (SSDBM). Boston, Massachusetts.

Altintas, I., Berkley, C., Jaeger, E., Jones, M., Ludäscher, B., Mock, S., 2004a. Kepler: an extensible system for design and execution of scientific workflows. 16th International Conference on Scientific and Statistical Database Management. IEEE publication number P2146.

Altintas, I., Jaeger, E., Lin, K., Ludäscher, B., Memon, A., 2004b. Abstract. 2nd Intl. Conference on Web Services (ICWS), San Diego, California, July 2004.

Altintas, I., Birnbaum, A., Baldridge, K., Sudholt, W., Miller, M., Amoreira, C., Potier, Y., Ludäscher, B., 2005. A framework for the design and reuse of grid workflows. Intl. Workshop on Scientific Applications on Grid Computing (SAG'04). LNCS, 3458. Springer.

Baker, H.G., 1978. Invasion and replacement in Californian and neotropical grasslands. In: Wilson, J.R. (Ed.), Plant Relations in Pastures. CSIRO, East Melbourne, pp. 368–384.

Barcena, M.A., Flores, J.A., Sierro, F.J., Perez-Folgado, M., Fabres, J., Calafat, A., Canals, M., 2004. Planktonic response to main oceanographic changes in the Alboran Sea (Western Mediterranean) as documented in sediment traps and surface sediments. Mar. Micropaleontol. 53, 423–445.

Baumgartner, M.F., Cole, T.V.N., Clapham, P.J., Mate, B.R., 2003. North Atlantic right whale habitat in the lower Bay of Fundy and on the SW Scotian Shelf during 1999–2001. Mar. Ecol. Prog. Ser. 264, 137–154.

Borer, E.T., Hosseini, P.R., Seabloom, E.W., Dobson, A.P., 2007. Pathogen-induced reversal of native dominance in a grassland community. Proc. Natl. Acad. Sci. U. S. A. 104, 5473–5478.

Botts, M., Percivall, G., Reed, C., Davidson, J., 2007. OGC sensor web enablement: overview and high level architecture. OGC White Paper (OGC Document 07-165).

Bowen, M.M., Emery, W.J., Wilkin, J.L., Tildesley, P.C., Barton, I.J, Knewtson, R., 2002. Extracting multiyear surface currents from sequential thermal imagery using the maximum cross-correlation technique. J. Atmos. Ocean. Technol. 19, 1665–1676.

Bowers, S., McPhillips, T.M., Riddle, S., Anand, M.K., Ludäscher, B., 2008. Kepler/pPOD: scientific workflow and provenance support for assembling the tree of life. IPAW 5272, 70–77.

Callaghan, S., Maechling, P., Deelman, E., Vahi, K., Mehta, G., Juve, G., Milner, K., Graves, R., Field, E., Okaya, D., Gunter, D., Beattie, K., Jordan, T., 2008. Reducing time-to-solution using distributed high-throughput mega-workflows—experiences from SCEC CyberShake. 2008 Fourth IEEE International Conference on eScience, pp. 151–158.

Chelton, D.B., Esbensen, S.K., Schlax, M.G., Thum, N., Freilich, M.H., Wentz, F.J., Gentemann, C.L., McPhaden, M.J., Schopf, P.S., 2001. Observations of coupling between surface wind stress and sea surface temperature in the eastern tropical Pacific. J. Climate 14, 1479–1498.

Cornillon, P., Evans, D., Brown, O.B., Evans, R., Eden, P., Brow, J., 1988. Processing, compression and transmission of satellite IR data for near-real time use at sea. J. Atmos. Oceanic Technol. 5, 320–327.

Deelman, E., Singh, G., Su, M.-H., Blythe, J., Gil, Y., Kesselman, C., Mehta, G., Vahi, K., Berriman, G.B., Good, J., Laity, A., Jacob, J.C., Katz, D.S., 2005. Pegasus: a framework for mapping complex scientific workflows onto distributed systems. Sci. Prog. J. 13 (3), 219–237.

Dong, S., Kelly, K.A., 2003. Seasonal and interannual variations in geostrophic velocity in the Middle Atlantic Bight. J. Geophys. Res. 108 (C6), 3172.

Emery, W.J., Fowler, C., Clayson, C.A., 1992. Satellite-image-derived Gulf Stream currents compared with numerical model results. J. Atmos. Ocean. Technol. 9, 286–304.

Fegraus, E.H., Andelman, S., Jones, M.B., Schildhauer, M., 2005. Maximizing the value of ecological data with structured metadata: an introduction to ecological metadata language (EML) and principles for metadata creation. Bull. Ecol. Soc. Am. 86 (3), 158–168.

Gallagher, J., Milkowski, G., 1995. Data transport within the distributed oceanographic data system. World Wide Web Journal: Fourth International World Wide Web Conference Proceedings, pp. 691–702.

Gallagher, J., Potter, N., Sgouros, T., Hankin, S., Flierl, G., 2007. The Data Access Protocol—DAP 2.0, NASA ESE RFC 004, Revision 1.1, 10 October 2007. http://www.esdswg.org/spg/rfc/ese-rfc-004.

Ginis, Personal communication, 2006.

Halbert, S., Voegtlin, D., 1995. Biology and taxonomy of vectors of barley yellow dwarf viruses. In: D'Arcy, C.J., Burnett, P.A. (Eds.), Barley Yellow Dwarf: 40 Years of Progress. Minnesota, The American Phytopathological Society, St. Paul, pp. 217–258.

Hare, J.A, Churchill, J.H., Cowen, R.K., Berger, T.K., Cornillon, P.C., Dragos, P., Glenn, S.A., Govoni, J.J., Lee, T.N., 2002. Routes and rates of larval fish transport from the southeastern to the northeastern United States Continental Shelf. Limnol. Oceanogr. 47, 774–1789.

Jackson, L.E., 1985. Ecological origins of California's Mediterranean grasses. J. Biogeogr. 12, 349–361.

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

Kelly, K.A., Singh, S., Huang, R.-X., 1999. Seasonal variations of sea surface height in the Gulf Stream region. J. Phys. Oceanogr. 29, 313–327.

Ludäscher, B., Altintas, I., Berkley, C., Higgins, D., Jaeger-Frank, E., Jones, M., Lee, E., Tao, J., Zhao, Y., 2006. Scientific workflow management and the Kepler system. Special issue: workflow in grid systems. Concurr. Comput.: Pract. Exp. 18 (10), 1039–1065.

O'Neill, L.W., Chelton, D.B., Esbensen, S.K., 2003. Observations of SST-induced perturbations of the wind stress field over the Southern Ocean on seasonal timescales. J. Clim. 16, 2340–2354.

Park, K.-A., Cornillon, P., 2002. Stability-induced modification of sea surface winds over Gulf Stream rings. Geophys. Res. Lett. 29.

Park, K.-A, Cornillon, P., Codiga, D., 2006. Modification of surface winds near ocean fronts: effects of Gulf Stream rings on Scatterometer (QuikSCAT, NSCAT) wind observations. J. Geophys. Res. 111 (C3).

Pennington, D., Higgins, D., Peterson, A.T., Jones, M.B., Ludaescher, B., Bowers, S., 2007. Ecological niche modeling using the Kepler workflow system. In: Taylor, I., Gannon, D., Deelman, E., Shields, M. (Eds.), Workflows for eScience: Scientific Workflows for Grids, Chapter 8. Springer.

Podhorszki, N., Ludäscher, B., Klasky, S.A., 2007. Workflow automation for processing plasma fusion simulation data. High Performance Distributed Computing. WORKS '07: Proceedings of the 2nd Workshop on Workflows in Support of Large-Scale Science, pp. 35–44.

R Development Core Team, 2009. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. http://www.R-project.org.

Song, Q., Hara, T., Cornillon, P., Friehe, C.A., 2004. A comparison between observations and MM5 simulations of the marine atmospheric boundary layer across a temperature front. J. Atmos. Ocean. Technol. 21, 170–178.

Taylor, I., Gannon, D., Deelman, E., Shields, M. (Eds.), 2007a. Workflows for e-Science: Scientific Workflows for Grids. Springer, New York. 530 pp.

Taylor, I., Shields, M., Wang, I., Harrison, A., 2007b. The Triana workflow environment: architecture and applications. In: Taylor, I., Deelman, E., Gannon, D., Shields, M. (Eds.), Workflows for e-Science. Springer, New York. 320–339 pp.

Tilak, S., Hubbard, P., Miller, M., Fountain, T., 2007. The Ring Buffer Network Bus (RBNB) DataTurbine streaming data middleware for environmental observing systems. Proceedings of the Third IEEE International Conference on E-Science and Grid Computing (December 10–13, 2007). E-Science. IEEE Computer Society, Washington, DC, pp. 125–133.