This is an Accepted Manuscript of an article published in Environmental Modelling and Software in April 2016, available online: DOI: 10.1016/ j.envsoft.2015.12.010

Using a Data Grid to Automate Data Preparation Pipelines Required for Regional-Scale Hydrologic Modeling

Mirza M. Billah[1], Jonathan L. Goodall[2, 3, *], Ujjwal Narayan[4], Bakinam T. Essawy[2], Venkat Lakshmi[5], Arcot Rajasekar[6], Reagan W. Moore[6]

[1] *Department of Biological Systems Engineering, Virginia Tech, Blacksburg, Virginia*

[2] *Department of Civil and Environmental Engineering, University of Virginia, Charlottesville, Virginia*

[3] *Department of Civil and Environmental Engineering, University of South Carolina, Columbia, SC*

[4] *CMNS-Earth System Science Interdisciplinary Center, University of Maryland, College Park, MD*

[5] *Department of Earth and Ocean Sciences, University of South Carolina, Columbia, SC*

[6] *School of Information and Library Science, University of North Carolina, Chapel Hill, NC*

[*] *To whom correspondence should be addressed (E-mail: goodall@virginia.edu; Address: University of Virginia, Department of Civil and Environmental Engineering, PO Box 400742, Charlottesville, Virginia 22904; Tel: (434) 243-5019)*

Abstract

Modeling a regional-scale hydrologic system introduces major data challenges related to the access and transformation of heterogeneous datasets into the information needed to execute a hydrologic model. These data preparation activities are difficult to automate, making the reproducibility and extensibility of model simulations conducted by others difficult or even impossible. This study addresses this challenge by demonstrating how the integrated Rule Oriented Data Management System (iRODS) can be used to support data processing pipelines needed when using data-intensive models to simulate regional-scale hydrologic systems. Focusing on the Variable Infiltration Capacity (VIC) model as a case study, data preparation steps are sequenced using rules within iRODS. VIC and iRODS are applied to study hydrologic conditions in the Carolinas, USA during the period 1998-2007 to better understand impacts of drought within the region. The application demonstrates how iRODS can support hydrologic modelers to create more reproducible and extensible model-based analyses.

Keywords: data management; workflows; hydrologic modeling, iRODS

Software availability: The software is provided open source and freely available on GitHub. Visit https://github.com/uva-hydroinformatics-lab/VIC_Pre-Processing_Rules for additional details.

INTRODUCTION

Motivation

Application of regional-scale hydrologic models presents a number of challenges associated with handling and processing large datasets. These models are data intensive and require a significant amount of time, effort, and resources in order to transform available datasets into the form required by the model (Leonard and Duffy, 2013). The information required by models is contained within multiple datasets maintained by various data providers with each dataset having unique data access protocols, file formats, and semantics (Horsburgh et al., 2014). The result is that the input datasets for a hydrologic model require specific transformations before they can be used to setup, calibrate, and validate the model (Leonard and Duffy, 2014).

Due to the level of heterogeneity across data sources and the inconsistent input file formats required by various models, these data transformation steps are difficult to automate. With the exception of a few models with robust data preparation tools, data transformation steps associated with hydrologic model data preparation often require significant manual intervention. Even for models with data preparation tools available, these tools are often tightly coupled to pre-specified data sources that may not be the best available information for a specific region or modeling objective. The end result is that modelers (i) consume significant time on tasks that could be automated, (ii) lack the ability to easily reproduce and extend past work completed by others, and (iii) are unable to take full advantage of the most recently available information when creating models.

Within the information and computer science communities, there has been work to create advanced data management and scientific workflow software (Foster, 2011; Gil

et al., 2007; Ludäscher et al., 2006; Moore and Rajasekar, 2014; Oinn et al., 2006). These tools have been applied within many scientific communities including hydrology (Fitch et al., 2011; Guru et al., 2009; Perraud et al., 2010; Piasecki and Lu, 2010). Scientific workflow environments offer many benefits for implementing data preparation tasks. For example, workflow environments coordinate and automate processing steps, as well as track the provenance of the datasets generated through the processing steps. This is important especially for reproducibility, transparency, and reuse of computational analyses.

There remain challenges, however, in applying workflow tools for hydrologic modeling support. One challenge is the lack of a common data model within hydrology (Perraud et al., 2010). It is typical for hydrologic models to require gridded data in NetCDF or GRIB formats, times series in CSV or WaterML format, and GIS layers in shapefile, geodatabase, or raster formats. These data come from NASA, NOAA, USGS, and others, with each agency adopting its own semantics and structure. Another challenge is the large number of models used in the hydrology community, each with its own data formats, structures, and semantics. There is significant heterogeneity across these hydrologic models with some models adopting a gridded data structure and others adopting a feature-oriented (e.g., watersheds and stream networks) data structure. In short, there is very little commonality between input file structures and semantics across hydrologic data and models.

For these reasons, we argue that data preparation for regional-scale hydrologic modeling is a data-centric rather than process-centric task. That is to say, due to the lack of a common data model in hydrology, the primary goal is to create data processing

pipelines that define the logic for integrating and transforming the information contained within multiple datasets into a new product required by the hydrologic model as input. While others have argued for a centralized and standardized data model for hydrology (Goodall and Maidment, 2009; Leonard and Duffy, 2014; Maidment, 2002), we argue here that the current level of heterogeneity in source data and model input requirements makes this level of standardization, while an important long-term goal, difficult to obtain in the short-term. Rather, data can remain decentralized and under the control of data providers, but there should be a means for applying server-side data transformation pipelines to reference datasets. These pipelines, which could be written and maintained by model developers and users, will allow for on-demand access to derived data products required by specific hydrologic models. In the spirit of Relational Database Management Systems (RDMS), these pipelines can be thought of as views of the underlying data, tailored for a specific model.

In this paper, we present a method for creating server-side data pre-processing pipelines using the DataNet Federation Consortium (DFC) data grid, which is powered by the Integrated Rule-Oriented Data System (iRODS). We illustrate the methodology using the Variable Infiltration Capacity (VIC) regional-scale hydrologic model. These systems are briefly introduced in the following background section. Then, in the design and implementation section, the approach for automating VIC data preparation pipelines using iRODS is presented. Finally, the pipelines are demonstrated for an example of modeling drought in the Carolinas region of the United States.

Background

The DFC grid (http://www.datafed.org) was built as part of an NSF-funded research project to provide storage and compute resources that allow for long-term access to the stored datasets. DFC is enabled by iRODS, an open source, policy-based cyberinfrastructure developed by the Data Intensive Cyber Environments (DICE) group for distributed data management (Rajasekar et al., 2010b) (http://www.irods.org). It is used by a wide variety of end users including various scientific communities (Chiang et al., 2011; Goff et al., 2011; NASA, 2015). Data management tasks and policies are implemented within iRODS as rules. Rules specify a sequence of lower-level micro-services that operate on datasets within the data grid. Users can specify sequences of data collection, transformation, curation, preservation, and processing steps within one or more rules. iRODS includes a Rule Engine (RE) that allows for remote execution of rules on iRODS resource servers, which is especially beneficial for large-datasets. iRODS, therefore, provides a means for uploading processing routines to data resource servers, whereas the typical approach used now in hydrology is to download data for local processing.

The Variable Infiltration Capacity (VIC) model is a large-scale hydrologic model that applies water and energy balances to simulate terrestrial hydrology at a regional spatial scale (Liang et al., 1996a). The scientific background of the model is summarized in the case study section, while here the model is presented from a data management perspective. The model requires several input datasets, and these datasets must be generated through a sequence of data processing steps. Figure 1 shows the data processing pipeline that is typically done manually to create the meteorological and land

surface variable inputs for a VIC model simulation. The figure illustrates the difficulties in performing this workflow, which include the fact that different datasets are required, each with a different data model, and processing steps are written in different languages and sometimes require legacy compilers due to the time that they were developed. Executing these data processing scripts currently requires significant effort, especially for new users that must learn the steps and configure often complicated legacy software to correctly execute the sequence of steps. While Figure 1 depicts the process as an integrated workflow, in practice it is instead a sequence of steps often performed independently with software developed at different times by different scientists. While this description is for VIC, we believe it describes a general problem faced by other data intensive models.

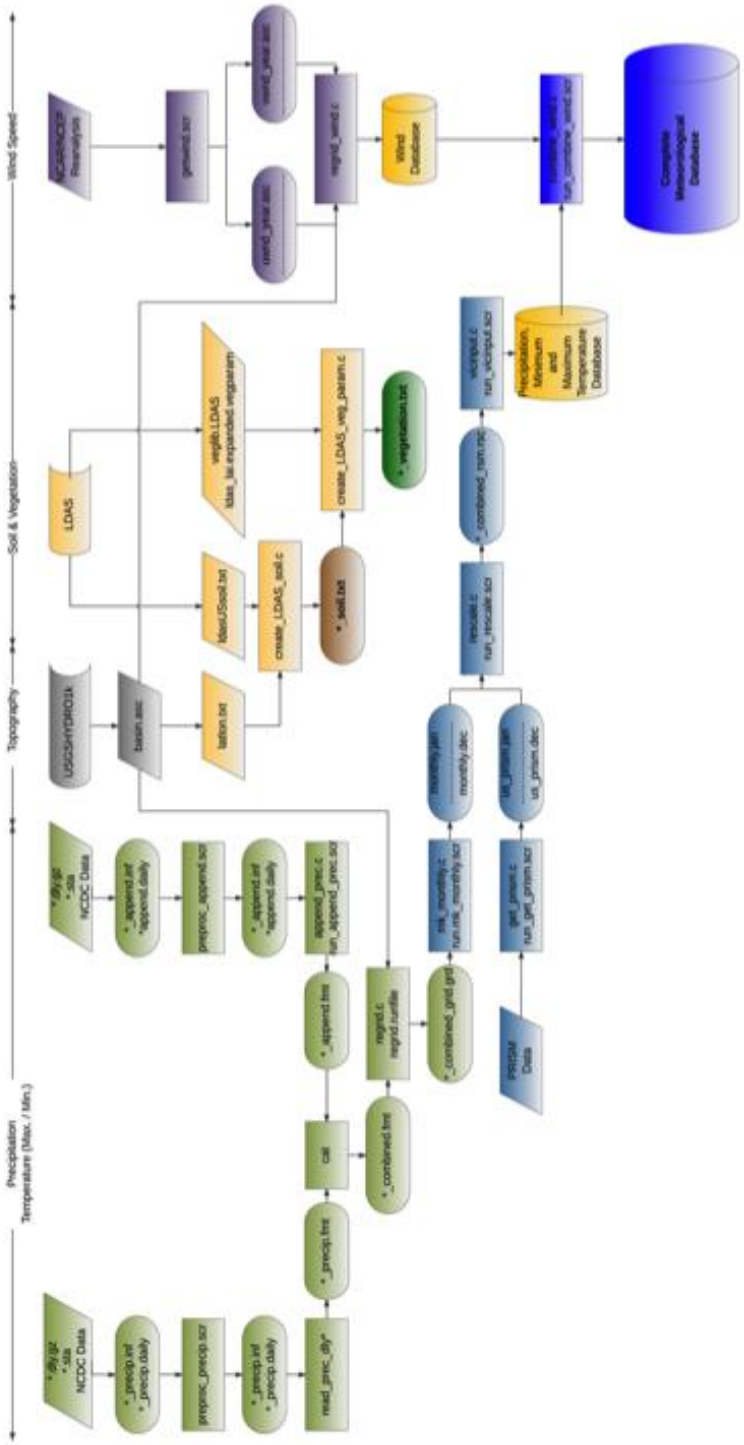

Figure 1: Data pre-processing steps for VIC model depicted as a flow chart of legacy processing scripts written in different languages, operating on different input datasets, and creating different output datasets.

Study Objective and Scope

Given these data challenges in running data-intensive hydrologic models, and given the potential advantages of advanced data management systems such as iRODS, the objective of this study is to apply iRODS to support hydrologic modeling. More specifically, the focus of this research is on the data preparation pipeline used to create the input files required for running VIC. The goal is to automate these steps as iRODS rules. The rules provide a means for server-side execution of data processing pipelines, moving closer to the long-term goal of reproducible end-to-end model simulations. The study is built on prior work where the VIC model was used for simulating a period of drought in South Carolina, USA (Billah and Goodall, 2011; Billah et al., 2015). VIC is a widely used model for investigating regional-scale hydrologic systems (Abdulla et al., 1996; Lakshmi et al., 2004; Lohmann et al., 1998; Sheffield and Wood, 2007; Sheffield et al., 2004), and the software resulting from the prototyping work can be used by this community. More generally, the outcome of this work is a methodology for creating server-side data processing pipelines that could be applied for other data-intensive hydrologic, environmental, and Earth system models.

SYSTEM DESIGN AND IMPLEMENTATION

The DFC-Hydrology Grid

The server-side data processing software shown in Figure 1 is deployed on an iRODS resource server that is part of the DFC-hydrology grid. The DFC-hydrology grid is part of the larger DFC grid (Figure 2), which is funded by the National Science Foundation (NSF) to support data collection, analysis, preservation, sharing, and

publication of scientific data and models (http://www.datafed.org). The DFC-hydrology grid consists of an iRODS resource server that communicates with an iRODS metadata catalog (iCAT) server located in the DFC grid. The iCAT server is administered by RENCI (Renaissance Computing Institute) in Chapel Hill, North Carolina. iRODS also includes a Rule Engine (RE) that interprets rules executed on the server, but initiated from a client program. The RE connects to the catalog server in the DFC grid and updates the iCAT database if new files are generated through the execution of a rule.

An iRODS client application is used to initiate data management tasks including rules within the federated grid. A commonly used iRODS client application is the i-commands utility that consists of several command-line utilities for managing datasets and executing data processing commands (Rajasekar et al., 2010a). These i-commands can be used to put and get data into/from the DFC grids, as well as execute a set of other commands for data management, analysis, and sharing. A user has the opportunity to create and execute rules to chain commands for tasks such as gaining access to heterogeneous data sources and processing datasets into the formats required by models or scientific communities.

The hydrology related data collected from various external sources are stored and preserved in the DFC-hydrology grid in their native formats. This data gathering process can be automated using iRODS functionality and different servers within the grid. For instance, the precipitation data collection can be obtained from remote servers by one iRODS resource server and data transformations could be performed by a second iRODS resource server in the DFC-hydrology grid. Because these servers are part of the same data grid, the distributed data stored across the servers appears to client applications as

part of the same logical file directory. iRODS includes services for accessing remote resources using different access protocols (e.g., HTTP or FTP). Rules and policies can be established for the grid to provide continuous data synchronization between data in the grid and data stored on remote servers. Many of these more advanced features were not included in the prototyping research described in this paper, but could be added through future work and refinement of the system.

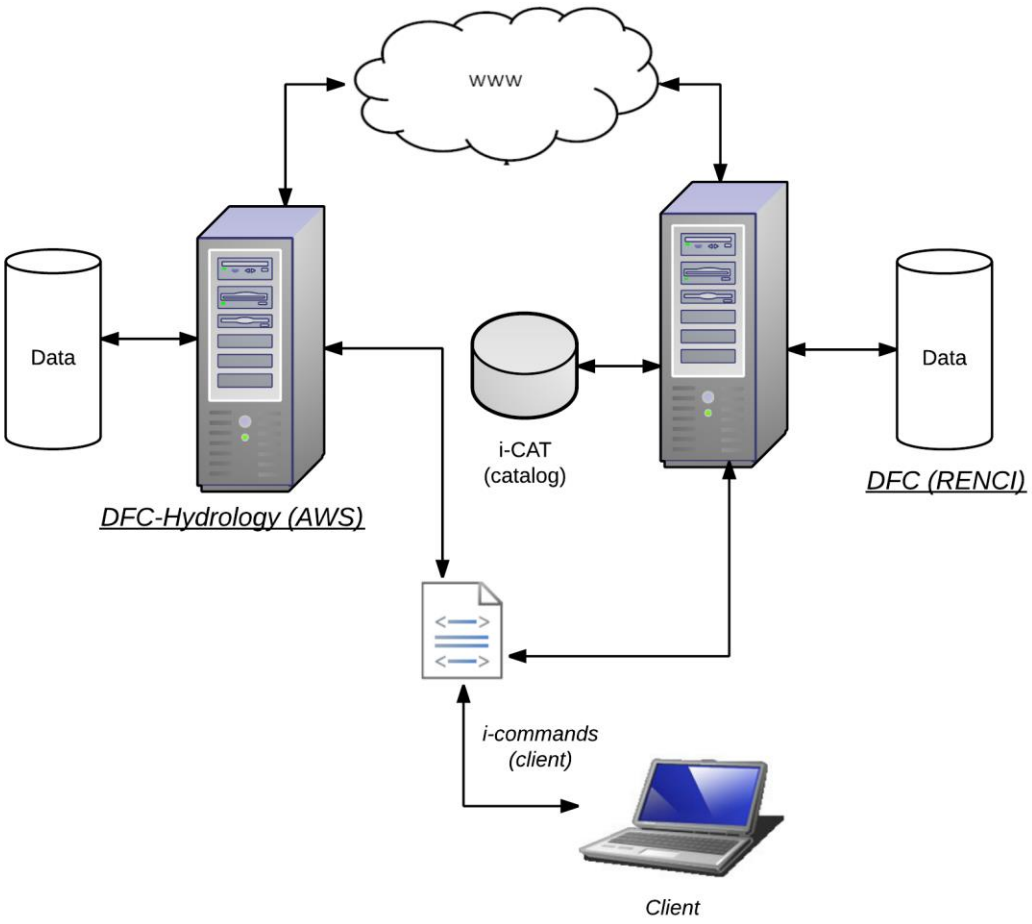

Figure 2: Schematic diagram of the NSF supported Data Federation Consortium (DFC) data management system showing the connections between the DFC-hydrology grid with the DFC grid.

Micro-Services

Micro-services are the building blocks for implementing policy-based data management within the DFC grid (Rajasekar et al., 2010a). A micro-service is a well-defined function that performs a specific task as part of a distributed workflow system. A number of micro-services are available to automate data collection, processing, and storage in the DFC federated resource servers. These micro-services are primarily developed by system or application programmers, but could also be written by scientists. The micro-services that are applied for this study are listed in Table 1. These micro-services are included in the latest iRODS release and can be chained together using rules. Although the flexibility to chain a number of micro-services provides multiple ways to complete a series of tasks, iRODS applies priorities and validation conditions to select the best micro-service to complete a given task.

Table 1: Micro-services applied for VIC model application using iRODS.

No.	Micro-service	Purpose
1	msiExecCmd	Execute commands
2	msiCollCreate	Make data collection
3	msiDataObjCreate	Create data object
4	msiDataObjWrite	Write data object
5	msiDataObjClose	Close data object
6	msiAddSelectFieldToGenQuery	Make query for data using field
7	msiAddConditionToGenQuery	Make query for data using condition
8	msiExecGenQuery	Execute Query
9	msiGetValByKey	Extract value from query result
10	msiSplitPath	Get directory path
11	msiDataObjUnlink	Delete temporary file
12	msiRmColl	Remove data collection
13	msiGetSystemTime	Get time stamp

Rules

The rule is a critical and fundamental component for iRODS. It provides a flexible mechanism to integrate external systems for specialized processing and metadata management (Hedges et al., 2009, 2007). In this system, we use rules to implement data processing pipelines on the DFC grid. Data pre-processing rules involve collecting and transforming datasets from heterogeneous sources into the inputs required by VIC. For our purposes, data were collected from the United States Geological Survey (USGS), National Climatic Data Center (NCDC), National Center For Atmospheric Research (NCAR), National Centers for Environmental Prediction (NCEP), and Land Data Assimilation System (LDAS). The DFC catalog is updated automatically when data are put into the grid. This catalog functions as an information center and enables the discovery of distributed data stored within the grid.

Data processing workflows can be implemented as rules that transform the collected datasets from the external sources into model readable inputs (Figure 3). These rules are a combination of multiple step-based routines, each of which performs a particular task. The steps are shown in Figure 3 and include tasks such as retrieving data, preparing data for gridding, adjusting observation times, and transforming gridded and rescaled datasets. For our study, the routines for executing these steps are installed on a resource server that is part of the DFC-hydrology grid and integrated into data-specific rules used to complete a series of tasks from collection and transformation of the datasets into model inputs. Separate rules were created to perform the data transformation tasks grouped into logical divisions as shown in Table 2.

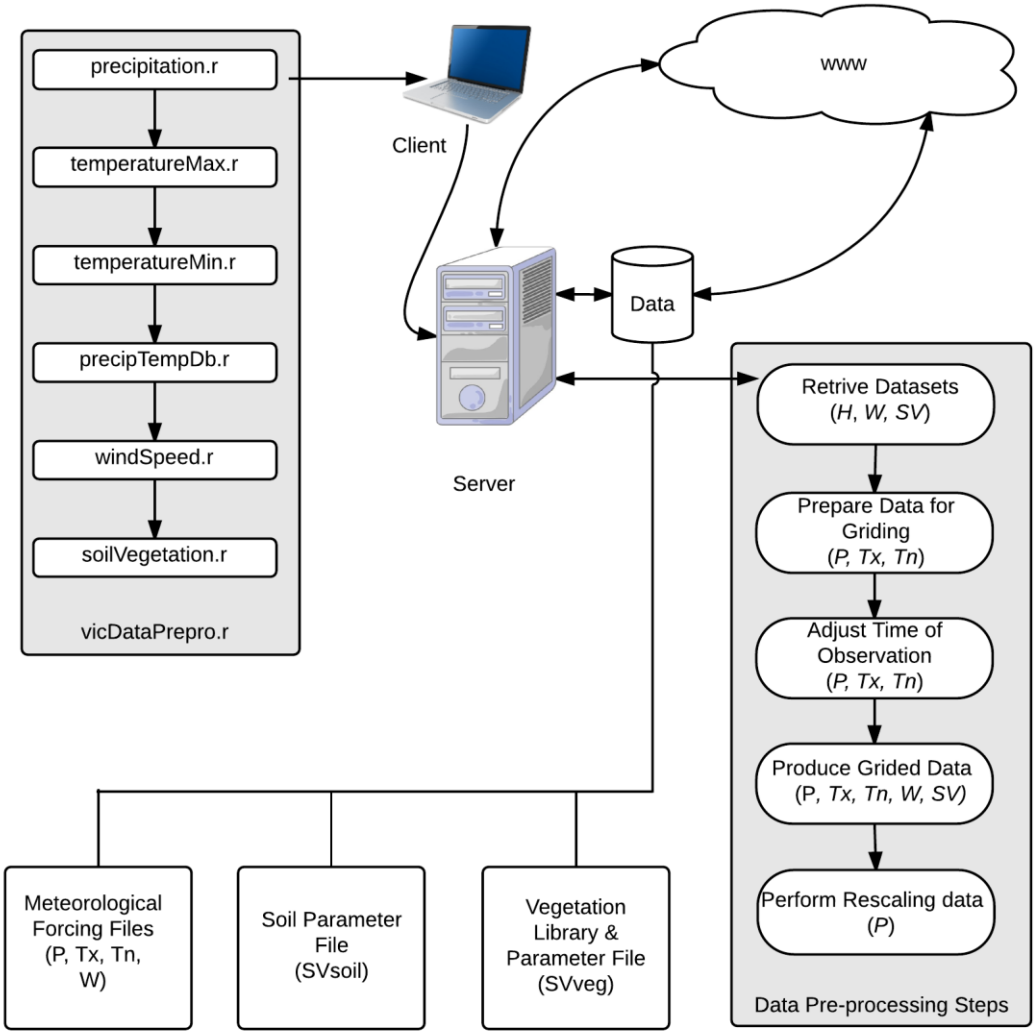

Figure 3: Model pre-processing workflows showing the major steps for transforming datasets to set up the VIC model for a specific study area. Rules are initiated from a client but executed on a server using micro-services.

Table 2: Rules created within iRODS for VIC data pre-processing.

No.	Micro-service	Purpose
1	vicDataPrepro.r	Rules all subrules for VIC data pre-processing
2	precipitation.r	Collect and process daily precipitation data
3	temeperatureMax.r	Collect and process daily maximum temperature data
4	temeperatureMin.r	Collect and process daily minimum temperature data
5	precipTempDb.r	Combines precipitation and temperature data
6	windSpeed.r	Collect and process annual wind speed data
7	soilVegetation.r	Collect and process soil and vegetation data

EXAMPLE APPLICATION

Study Area

The region used for the case study application covers all the major river basins in both North and South Carolina with a total area of 280,736 km^2 (108,393 mi^2) (Figure 5). The Pfafstetter Basin Code system was used to define this "Carolinas" study region from the HYDRO1k basin dataset. The codes were incorporated in the hydro1kCarolina.r rule to extract the area during execution of the workflow.

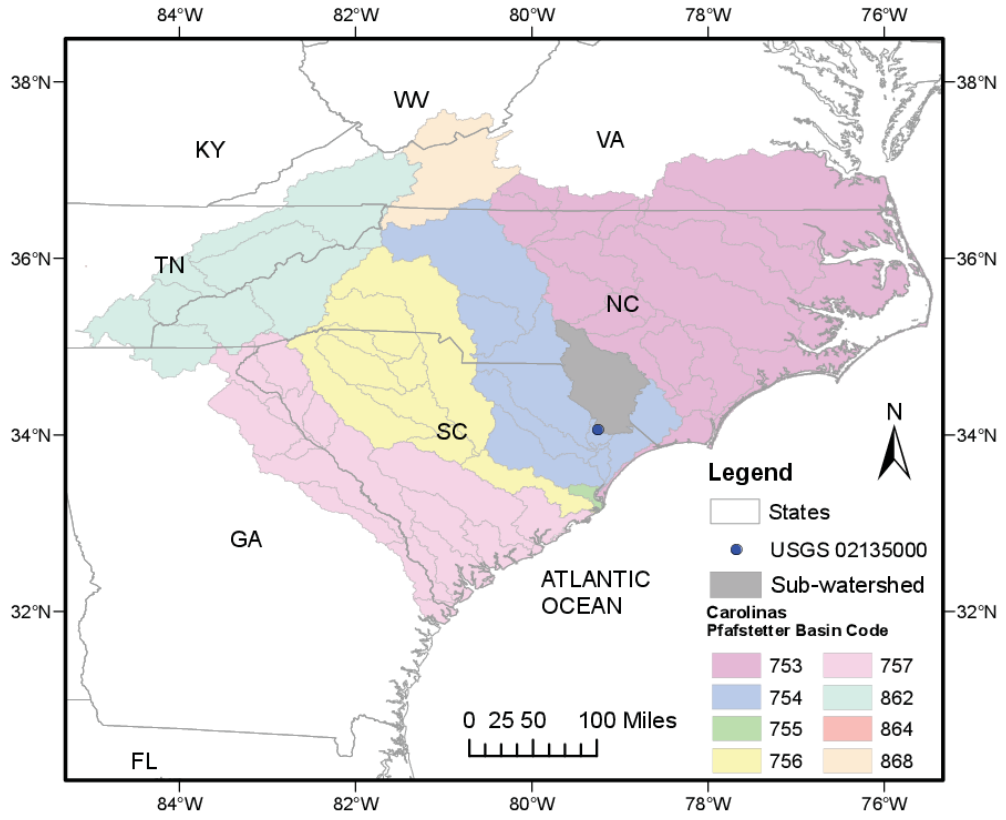

Figure 5: Study area with major river basins in North and South Carolina, USA. These subwatersheds were extracted using selected Pfafstetter Basin Code from the Hydro1K dataset. The watershed and stream gage station used in the VIC model calibration are also shown.

Model Application

VIC is able to simulate the land surface portion of the hydrologic cycle by solving the full water and surface energy balance equations (Liang and Lettenmaier, 1994; Liang et al., 1996b). VIC was calibrated for the study region and period using the following seven parameters: variable infiltration curve (b), maximum base flow (D_{smax}), fraction of base flow where base flow occurs (D_s), fraction of maximum soil moisture content above which nonlinear base flow occurs (W_s), mid (d_2) and deep (d_3) soil layer depth,

and minimum stomatal resistance (r_0) (Abdulla and Lettenmaier, 1997a, 1997b; Crow, 2003; Troy et al., 2008). The range investigated and the final values of the parameters applied in this study are described in more detail in Billah e al. (2015).

A comparison of the monthly average streamflow predicted by the calibrated VIC model and streamflow observations are provided in Figure 6. The streamflow observations are for the Little Pee Dee River at Galivants Ferry station that is part of the USGS National Water Information System (NWIS) network (USGS 02135000; Figure 5) for the period of 1998 to 2007. This streamflow station has a drainage area of 7,257 km^2 and includes portions of both North and South Carolina. This station was selected based on its available time series record and because it is on an unmanaged portion of a river network. The Nash-Sutcliffe Efficiency (NSE) index of the final calibration is 0.6 at this and other stations used for calibration and validation of the model but not shown here. This NSE value is considered to be a satisfactory calibration by watershed-scale hydrologic modelers (Moriasi et al., 2007). Further details on the model calibration and validation are provided in Billah et al. (2015).

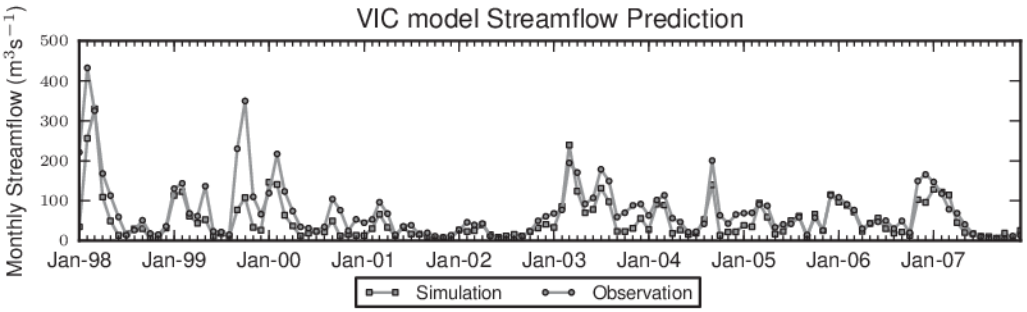

Figure 6: Streamflow comparison between the VIC model predictions and USGS observations. The comparison is performed at the USGS station Little Pee Dee River at Galivants Ferry, SC (Station Number 02135000).

Data Pre-Processing Pipeline

When executing the data pre-processing rules for a specific area of interest, the first step is to retrieve climate data (precipitation, maximum and minimum temperature data) from the DFC federated grid by initiating the hydro1kCarolinas.r rule. This rule uses a series of micro-service calls initiated on the DFC federated grid to extract and register data for the area of interst from natinoal-scale datasets (Figure 4). We use the Pfafstetter basin numbering system as described in Furans and Olivera (2001) for defining study area from HYDRO1k basin and DEM datasets. The climate datasets for the defined study area were downloaded via FTP from the NCDC Global Historical Climatology Network (GHCND) database. While downloading the climate data, a buffer of 0.25° was considered around the defined study area to collect sufficient climate data. The climate data contained precipitation, maximum and minimum temperature, and wind speed datasets.

The precipitation data is then processed using the precipitation.r rule, which uses the GHCND data and convertes the station specific datasets into gridded datasets with a spatial resolution of 1/8°. Similarly, temperatureMax.r and temperatureMin.r rules convert station specific temperature values into gridded datasets with 1/8° spatial resolution. Rules are also used to process wind speed, soil, and vegetation data from their respective sources. The annual wind data were collected from NCAR/NCEP and processed to generate gridded datasets of 1/8° resolution for the study area using windSpeed.r. The soilVegetation.r rule was applied to transform the LDAS soil and vegetation information into information required for the model. The result is a collection

of VIC input files derived from national-scale reference datasets using iRODS rules that largely wrap legacy data processing routines.

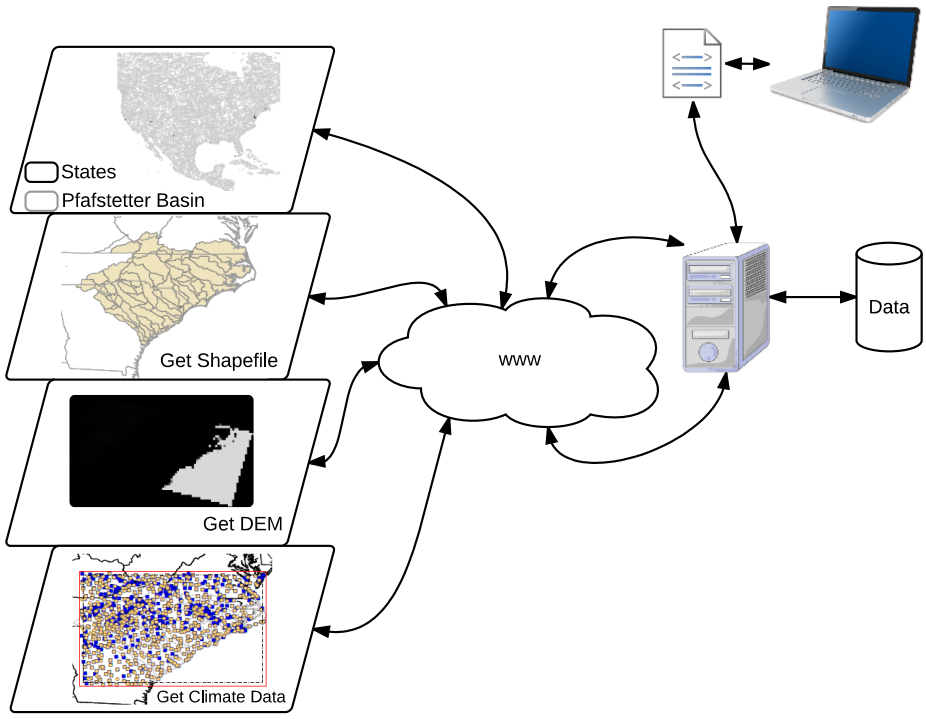

Figure 4: Data flow in the hydro1kCarolinas.r rule that extracts climate data from NCDC GHCND using HYDRO1K basin/DEM datasets to define a study region.

Model Results

VIC model runs generate a number of grid-based hydrologic flux and state variable outputs. One of these outputs is soil moisture estimated for each grid cell in the simulation domain and for each soil layer within the model. Soil moisture is an important indicator of drought, so it is used as an example model output. It is possible to create data post-processing rules similar to what was done for data pre-processing to extract model results from the various output files generated by the model. For example, a rule could be created to summarize the soil moisture over the study region for each soil layer and to

display this information as a time-series plot of monthly soil moisture within the three VIC soil layers (Figure 7). The plot of soil moisture provides a way of depicting the impact of the drought on soil moisture in 2003 and its recovery in 2005, particularly the deep soil layer which is more sensitive to long term trends in water availability compared to the middle and upper soil layers.

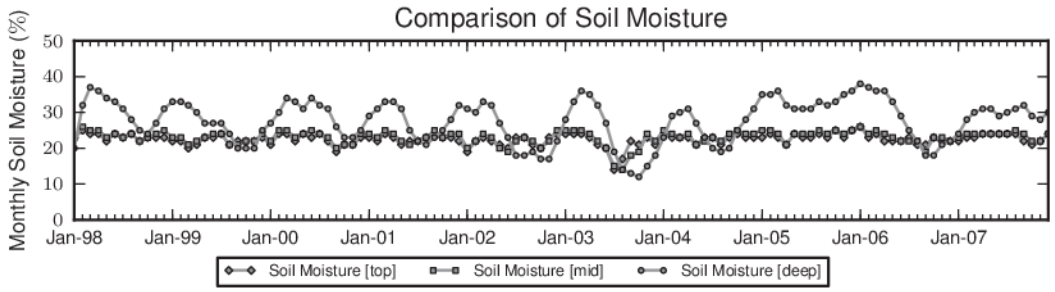

Figure 7: Comparison of monthly averaged soil moisture in the three soil layers predicted by the VIC model in the Carolinas for the periods of 1998 to 2007.

Benchmarking

We performed a benchmarking experiment to quantify the time required to execute the data pre-processing pipeline and the overhead introduced through remote execution using iRODS. The experiment was conducted using virtual machines (VMs) in the Amazon Web Service (AWS) cloud. VMs were created using the AWS Elastic Compute Cloud (EC-2) service. We used a m3.medium instance type, which at the time of this research had a high frequency Intel Xeon E5-2670 v2 (Ivy Bridge) 64-bit processor, a Solid State Drive (SDD), and 3.75 GB of RAM for the iRODS resource server. For the iRODS client machine we used a t2.micro instance type, which at the time of this research had a high frequency Intel Xeon processer with turbo and up to 3.3 GHz and 1 GB of RAM.

We ran each rule on the resource server and initiated execution of the rule from the client machine. We also ran the shell scripts that the rules wrap directly on the m3.medium instance in order to measure the time lag introduced by iRODS and the Rule Engine. We ran the processing script for the Carolinas study region for the period 1997-2008 at a 1/8-degree resolution. We tracked the wall time required to complete each of the processing steps and executed each process three times to measure variability in the execution time. The data processing pipeline makes use of data from five different federal data providers, it touches 35 different files and file directories, and it generates a 1.6GB file directory that represents the input files for VIC for the study region and time period of interest.

Results of this benchmarking experiment show that the overall rule takes approximately 10 minutes to run for the area and period of interest (Table 3). This assumes reference data have already been gathered and loaded into the data grid. The vast majority of the processing time is spent on meteorological data transformations compared to the soil and vegetation data transformations. This is because the rule processes precipitation, minimum and maximum air temperature, and wind speed directly from data providers (NCDC and NCAR). For soil and vegetation data, the rule makes use of pre-processed data created specifically for VIC models provided through the North American Land Data Assimilation System (NLDAS).

The iRODS Rule Engine (RE) does appear to introduce some overhead compared to directly calling the process using shell scripts on the server. Considering the overall rule processing times and averaging across the three runs for each case, the RE case was 11 seconds (2%) slower. Given that the RE allows for remote execution of the data

processing pipeline, we feel this is a fairly minimal cost to pay for the added benefit. Because the focus of this research was on creating the data processing pipeline rather than optimizing its execution time, there are likely opportunities to reduce these execution times through modifications to the rule structure and underlying source code.

Table 3: Wall time to complete for each data processing step when the logic internal to the rule is initiated directly on the server using a shell script and when the rule is initiated from a client machine and executed on the server using the iRODS Rule Engine (RE). For each case, the rule was executed three runs to capture the execution time variability.

	Execution Time (sec)					
	On Server/ Shell Script			**From Client / Rule Engine**		
Rule	Run 1	Run 2	Run 3	Run 1	Run 2	Run 3
precipitation.r	82	75	76	71	71	69
temeperatureMax.r	162	162	162	161	162	162
temeperatureMin.r	161	161	156	161	160	159
precipTempDb.r	39	38	38	38	39	38
windSpeed.r	115	122	116	144	132	125
soilVegetation.r	27	20	27	27	27	27
vicDataPrepro.r (parent rule)	586	578	575	602	591	580

SUMMARY, DISCUSSION, AND CONCLUSIONS

The hydrologic modeling process involves many steps from data access and transformation, to model setup, calibration, and validation, to analysis and visualization of model outputs. This entire “end-to-end’’ process involves some steps that are easily automated and others that require intervention by expert modelers. The goal in this and related work is to automate those steps that are straightforward but tedious, while still allowing experts to guide the process and intervene when needed. Currently too many steps that can be automated are not. As a result, modelers are unable to focus on the important tasks that require their expertise and insights because time must be spent on

more basic data gathering and transformation steps. Furthermore, the steps that could be automated are typically not thoroughly documented and, even if they are thoroughly documented, are time consuming to repeat. This makes independent reproducibility of model results, a requirement for scientific progress and water resource management objectives, difficult or even impossible.

It is important to note the specific data challenges required for a hydrologic model application. First, VIC requires a large amount of data when applied to a region the size of the Carolinas, and of course even more data would be required for Continental scale model executions, which are not uncommon when applying VIC. The data used in the VIC model pre-processing steps included meteorological datasets at point stations and on grids, topography datasets available as grids, and soil and vegetation datasets also available as grids. Transformation of these raw input data resulted in intermediate datasets with different spatial projections, filled gaps, and other modifications required before initiating the model simulation. Over the years, researchers have created scripts for completing many of these data pre-processing steps, but less effort has been devoted to schemes for integrating these scripts into reproducible workflows. As data volumes continue to increase, more sophisticated ways of handling these data are needed.

This work addresses these challenges by leveraging the iRODS technology and the DataNet Federation Consortium (DFC) cyberinfrastructure to create data processing pipelines that automate pre-processing steps for VIC. The workflows developed for VIC include sufficient information to allow others to independently reproduce the model results from well-known reference data products. The workflows, therefore, act as a means for forced documentation of the steps used to create model input files including

the provenance of data as it is transformed from the form provided by federal and academic data repositories into the form required by models.

In the case of VIC, and likely in the case of other hydrology applications as this work is extended to include other hydrologic models, the workflows were created by leveraging existing scripts written to complete specific data pre-processing tasks. The approach used provides a means for placing these scripts in larger data processing pipelines and removes the need to access and understand the details of the original scripts to reproduce model results or to reuse the tools for a new study. In the latter case, modification of the rules is only required when selecting a new area of interest. The Pfafstetter Basin codes are replaced in the hydro1kCarolinas.r rule. However, all other workflows are used without modification to automate the remaining data processing steps, which effectively build the input files required for a VIC model.

A primary focus of this work is demonstrating a methodological approach to assist in data-intensive hydrologic modeling. Using iRODS has advantages that include workflow automation, access control, data transfers, and data synchronization. iRODS is flexible and robust; it was possible to extend the software by developing rules specific for the data processing pipelines associated with running the VIC model. Because of the data grid concept used by iRODS, it was possible to design workflows in such a way that distributed computers could be leveraged to perform the data gathering and preparation steps. It was also possible to make use of legacy and new software tools for server-side processing of reference datasets. For instance, we applied ecohydroworkflow (Miles, B., Band, 2013), a shareable workflow for data management for hydrologic models, to

collect and register GHCND and HYDRO1k datasets from NCDC and USGS, respectively, in the DFC grid.

The rules created within iRODS to enable data pre-processing steps are a key step to achieving reproducible hydrologic model runs. It was possible to chain the data pre-processing routines by creating a rule named vicDataPreprocessing.r in iRODS. This overall rule was designed to call a series of sub-rules, with each sub-rule performing a series of steps required to transform the reference datasets into the specific form required by the model. Doing so provided a level of granularity so that the subrules could later be re-used within other applications in the DFC grid. Having this capability will reduce human errors introduced by manual data transforms. It will also free researchers to devote more time to enhancing, calibrating, and validating models, rather than on tedious steps required to set-up first iterations of the model.

A longer-term goal is to allow researchers a means for publishing post-processing workflows that can be used to recreate publication figures using reference datasets stored in a data grid. Creating data post-processing workflows would supplement this work by providing reproducible ways of visualizing large collections of model results. Post-processing rules could result in publication-ready figures that could be reproduced by other researchers. Various stakeholder groups have unique needs, so creating general visualization tools will not always be possible. Creating rules that leverage lower-level micro-services to visualize model results in customized ways could provide a powerful tool provided through iRODS and the DFC cyberinfrastructure.

While our approach was to use nested subrules when creating the overall data pre-processing rule to foster reuse, questions remain as to the level of reuse that will be

practical across hydrologic simulation models. Hydrologic models can be grouped into classes based on the use cases considered when developing the model. VIC falls into the "macro-scale" class of hydrologic models, meaning it is typically applied to regional, continental, or even global scale hydrologic systems. Other hydrologic models focus on more local scale systems such as a single catchment. We know that different classes of hydrologic models will require different schemes for pre-processing tasks such as discretizing the landscape, and will make use of different reference datasets to set up and parameterize the model. A key challenge moving forward, however, will be to determine the correct level of granularity of rules for data pre-processing that will provide a flexible environment able to support the wide variety of reference datasets and models used by the hydrologic community.

ACKNOWLEDGMENTS

The authors wish to acknowledge support from the National Science Foundation (NSF) under the project DataNet Full Proposal: DataNet Federation Consortium (Award Number:094084 and from Amazon through an Amazon Web Services (AWS) in Education Research grant.

REFERENCES

Abdulla, F. a., Lettenmaier, D.P., 1997a. Application of regional parameter estimation schemes to simulate the water balance of a large continental river. J. Hydrol. 197, 258–285. doi:10.1016/S0022-1694(96)03263-5

Abdulla, F. a., Lettenmaier, D.P., 1997b. Development of regional parameter estimation equations for a macroscale hydrologic model. J. Hydrol. 197, 230–257. doi:10.1016/S0022-1694(96)03262-3

Abdulla, F. a., Lettenmaier, D.P., Wood, E.F., Smith, J. a., 1996. Application of a macroscale hydrologic model to estimate the water balance of the Arkansas-Red River Basin. J. Geophys. Res. 101, 7449. doi:10.1029/95JD02416

Billah, M.M., Goodall, J.L., 2011. Annual and interannual variations in terrestrial water storage during and following a period of drought in South Carolina, USA. J. Hydrol. 409, 472–482. doi:10.1016/j.jhydrol.2011.08.045

Billah, M.M., Goodall, J.L., Narayan, U., Reager, J.T., Lakshmi, V., Famiglietti, J.S., 2015. A methodology for evaluating evapotranspiration estimates at the watershed-scale using GRACE. J. Hydrol. 523, 574–586. doi:10.1016/j.jhydrol.2015.01.066

Chiang, G.-T., Clapham, P., Qi, G., Sale, K., Coates, G., 2011. Implementing a genomic data management system using iRODS in the Wellcome Trust Sanger Institute. BMC Bioinformatics 12, 361. doi:10.1186/1471-2105-12-361

Crow, W.T., 2003. Multiobjective calibration of land surface model evapotranspiration predictions using streamflow observations and spaceborne surface radiometric temperature retrievals. J. Geophys. Res. doi:10.1029/2002JD003292

Fitch, P., Perraud, J.-M., Cuddy, S., Seaton, S., Bai, Q., Hehir, D., Sims, J., Merrin, L., Ackland, R., Herron, N., 2011. The Hydrologists Workbench: more than a scientific workflow tool, in: Proceedings, Water Information Research and Development Alliance Science Symposium.

Foster, I., 2011. Globus Online: Accelerating and Democratizing Science through Cloud-Based Services. IEEE Comput. Soc. 15, 70–73. doi:doi:10.1109/MIC.2011.64

Furnans, J., Olivera, F., 2001. Watershed Topology - The Pfafstetter System. Esri User Conf. Vol. 21.

Gil, Y., Deelman, E., Ellisman, M., Fahringer, T., Fox, G., Gannon, D., Goble, C., Livny, M., Moreau, L., Myers, J., 2007. Examining the Challenges of Scientific Workflows. Computer (Long. Beach. Calif). 40, 24–32. doi:10.1109/MC.2007.421

Goff, S.A., Vaughn, M., McKay, S., Lyons, E., Stapleton, A.E., Gessler, D., Matasci, N., Wang, L., Hanlon, M., Lenards, A., Muir, A., Merchant, N., Lowry, S., Mock, S., Helmke, M., Kubach, A., Narro, M., Hopkins, N., Micklos, D., Hilgert, U., Gonzales, M., Jordan, C., Skidmore, E., Dooley, R., Cazes, J., McLay, R., Lu, Z., Pasternak, S., Koesterke, L., Piel, W.H., Grene, R., Noutsos, C., Gendler, K., Feng, X., Tang, C., Lent, M., Kim, S.-J., Kvilekval, K., Manjunath, B.S., Tannen, V., Stamatakis, A., Sanderson, M., Welch, S.M., Cranston, K.A., Soltis, P., Soltis, D., O'Meara, B., Ane, C., Brutnell, T., Kleibenstein, D.J., White, J.W., Leebens-Mack, J., Donoghue, M.J., Spalding, E.P., Vision, T.J., Myers, C.R., Lowenthal, D., Enquist, B.J., Boyle, B., Akoglu, A., Andrews, G., Ram, S.,

Ware, D., Stein, L., Stanzione, D., 2011. The iPlant Collaborative: Cyberinfrastructure for Plant Biology. Front. Plant Sci. 2, 34. doi:10.3389/fpls.2011.00034

Goodall, J.L., Maidment, D.R., 2009. A spatiotemporal data model for river basin- scale hydrologic systems. Int. J. Geogr. Inf. Sci. 23, 233–247. doi:10.1080/13658810802032193

Guru, S.M., Kearney, M., Fitch, P., Peters, C., 2009. Challenges in using scientific workflow tools in the hydrology domain, in: 18th World IMACS Congress and MODSIM09 International Congress on Modelling and Simulation. Cairns, Qld., pp. 3514–3520.

Hedges, M., Blanke, T., Hasan, A., 2009. Rule-based curation and preservation of data: A data grid approach using iRODS. Futur. Gener. Comput. Syst. 25, 446–452. doi:10.1016/j.future.2008.10.003

Hedges, M., Hasan, A., Blanke, T., 2007. Management and preservation of research data with iRODS. Proc. ACM first Work. CyberInfrastructure Inf. Manag. eScience CIMS 07 17–22. doi:10.1145/1317353.1317358

Horsburgh, J.S., Tarboton, D.G., Hooper, R.P., Zaslavsky, I., 2014. Managing a community shared vocabulary for hydrologic observations. Environ. Model. Softw. 52, 62–73. doi:10.1016/j.envsoft.2013.10.012

Lakshmi, V., Piechota, T., Narayan, U., Tang, C., 2004. Soil moisture as an indicator of weather extremes. Geophys. Res. Lett. 31, 2–5. doi:10.1029/2004GL019930

Leonard, L., Duffy, C.J., 2013. Essential Terrestrial Variable data workflows for distributed water resources modeling. Environ. Model. Softw. 50, 85–96. doi:10.1016/j.envsoft.2013.09.003

Leonard, L., Duffy, C.J., 2014. Automating data-model workflows at a level 12 HUC scale: Watershed modeling in a distributed computing environment. Environ. Model. Softw. 61, 174–190. doi:10.1016/j.envsoft.2014.07.015

Liang, X., Lettenmaier, D.P., 1994. A simple hydrologically based model of land surface water and energy fluxes for general circulation models. J. Geophys. Res. 99, 14,415–14,428. doi:10.1029/94JD00483

Liang, X., Lettenmaier, D.P., Wood, E.F., 1996a. One-dimensional statistical dynamic representation of subgrid spatial variability of precipitation in the two-layer variable infiltration capacity model. J. Geophys. Res. 101, 21403. doi:10.1029/96JD01448

Liang, X., Wood, E.F., Lettenmaier, D.P., 1996b. Surface soil moisture parameterization of the VIC-2L model: Evaluation and modification. Glob. Planet. Change 13, 195–206. doi:10.1016/0921-8181(95)00046-1

Lohmann, D., Raschke, E., Nijssen, B., Lettenmaier, D.P., 1998. Regional scale hydrology: I. Formulation of the VIC-2L model coupled to a routing model. Hydrol. Sci. J. 43, 131–141. doi:10.1080/02626669809492107

Ludäscher, B., Altintas, I., Berkley, C., Higgins, D., Jaeger, E., Jones, M., Lee, E.A., Tao, J., Zhao, Y., 2006. Scientific workflow management and the Kepler system. Concurr. Comput. Pract. Exp. 18, 1039–1065. doi:10.1002/cpe.994

Maidment, D.R., 2002. Arc Hydro: GIS for Water Resources. ESRI Press.

Miles, B., Band, L., 2013. EcohydroWorkflowLib. <http://pythonhosted. org/ecohydroworkflowlib/index.html> (verified 05.29.2013).

Moore, R., Rajasekar, A., 2014. Reproducible Research within the DataNet Federation Consortium, in: International Environmental Modelling and Software Society 7th International Congress on Environmental Modelling and Software. San Diego, CA.

Moriasi, D.N., Arnold, J.G., Liew, M.W. Van, Bingner, R.L., Harmel, R.D., Veith, T.L., 2007. M e g s q a w s 50, 885–900.

NASA, 2015. NASA Climate Data Service (NCDS) [WWW Document]. URL http://cds.nccs.nasa.gov (accessed 10.16.15).

Oinn, T., Greenwood, M., Addis, M., Alpdemir, M.N., Ferris, J., Glover, K., Goble, C., Goderis, A., Hull, D., Marvin, D., Li, P., Lord, P., Pocock, M.R., Senger, M., Stevens, R., Wipat, A., Wroe, C., 2006. Taverna: lessons in creating a workflow environment for the life sciences. Concurr. Comput. Pract. Exp. 18, 1067–1100. doi:10.1002/cpe.993

Perraud, J., Fitch, P.G., Bai, Q., 2010. Challenges and Solutions in Implementing Hydrological Models within Scientific Workflow Software. AGU Fall Meet. Abstr. -1, 06.

Piasecki, M., Lu, B., 2010. Development of a Hydrologic Modeling Platform Using a Workflow Engine, in: AGU Fall Meeting Abstracts. p. 1239.

Rajasekar, A., Moore, R., Hou, C.-Y., Lee, C. a., Marciano, R., de Torcy, A., Wan, M., Schroeder, W., Chen, S.-Y., Gilbert, L., Tooby, P., Zhu, B., 2010a. iRODS Primer: Integrated Rule-Oriented Data System, Synthesis Lectures on Information Concepts, Retrieval, and Services. doi:10.2200/S00233ED1V01Y200912ICR012

Rajasekar, A., Moore, R., Wan, M., Schroeder, W., 2010b. Policy-based Distributed Data Management Systems. JODI J. Digit. Inf. 11, 1–16.

Sheffield, J., Goteti, G., Wen, F., Wood, E.F., 2004. A simulated soil moisture based drought analysis for the United States. J. Geophys. Res. D Atmos. 109, 1–19. doi:10.1029/2004JD005182

Sheffield, J., Wood, E.F., 2007. Characteristics of global and regional drought, 1950-2000: Analysis of soil moisture data from off-line simulation of the terrestrial hydrologic cycle. J. Geophys. Res. Atmos. 112, 1–21. doi:10.1029/2006JD008288

Troy, T.J., Wood, E.F., Sheffield, J., 2008. An efficient calibration method for continental-scale land surface modeling. Water Resour. Res. 44, 1–13. doi:10.1029/2007WR006513