Proceedings of the Workshop on Education for High-Performance Computing (EduHPC), held in conjunction with the IEEE ACM International Conference on High-Performance Computing (SC) 2014, New Orleans, LA, 2014.

Teaching Parallelism Without Programming: A Data Science Curriculum for Non-CS Students

Yolanda Gil
Information Sciences Institute
University of Southern California
4676 Admiralty Way
Marina del Rey, CA 90292
gil@isi.edu

***Abstract*—The goal of our work is to develop an open and modular course for data science and big data analytics that is accessible to non-programmers. The course is designed to cover major concepts that are useful to understand the benefits of parallel and distributed programming while not relying on a programming background. These key concepts focus more on algorithmic aspects rather than architecture and performance issues. A key aspect of our work is the use of workflows to illustrate key concepts and to allow the students to practice.**

Keywords-curriculum; teaching; data science; big data; workflows; semantic workflows; WINGS; parallelism

I. INTRODUCTION

Data science has emerged as a widely desirable skill in many areas. Although courses are now available on a variety of aspects of data science and big data analytics, there is a lack of broad and accessible materials that are accessible to non-programmers. As a result, acquiring practical data science skills is out of reach for many students and professionals, posing severe limitations to our ability as a society to take advantage of our vast digital data resources. Parallel computing is an area that they would benefit from learning. However, parallel computing is traditionally taught as part of the computer science curriculum in ways that require strong programming skills [Prasad et al 2012].

In this paper, we propose a lesson plan to teach parallel computing concepts to non-programmers. The lesson plan is part of a course for teaching data science to non-CS students.

II. DATA SCIENCE FOR NON-PROGRAMMERS

We are developing educational materials for data science to provide broad and practical training in data analytics non-CS students. This includes students majoring in science and engineering who want to acquire skills to analyze data, such as biology, chemistry, and geosciences. This also includes students in the humanities that would like to pursue data-driven research, such as journalism students interested in social media analysis.

Our focus is on students that will not take programming classes. Our goal is that they learn basic concepts of data science, so they can understand how to pursue data-driven research projects in their area and be in a better position to collaborate with computer scientists in such projects.

Existing courses on data science typically require programming skills. As an example, Coursera's "Introduction to Data Science"[1] requires two college-level courses in programming. Even when targeted to non-programmers, data science curricula focus on teaching programming. For example, Columbia University's Journalism School offers a set of courses to introduce students to data practices[2] that starts out teaching basic programming skills.

Although it is always beneficial to learn programming, not every student is inclined to invest the time and effort to do so. A course that enables them to learn basic concepts of data science will be more approachable and still useful. In the spirit of computational thinking [Wing 2006], our goal is to design a curriculum that teaches computing concepts above the level of particular programming languages and implementations.

Another observation about data science curricula is that they tend to focus on databases and machine learning, with little attention to parallel and distributed computing. Although database technologies and machine learning algorithms are important, it is also important to include concepts of scalability through parallelism and distributed computation. These concepts are particularly important to include in the curriculum, as the motivation to learn about data science is often the pursuit of big data analytics and that requires understanding how to scale up computation.

Table I presents the major sections and topics of our proposed course for data science. All the topics can be introduced without requiring programming skills.

The course includes a variety of topics in parallel and distributed computing, which we describe in more detail in the next section.

The course also has more emphasis on metadata and semantics than are usually included in data science courses. There is also more emphasis on end-to-end methods for data analysis, which include data pre-processing, data post-processing, and visualization.

---

1 https://www.coursera.org/course/datasci

2 http://www.journalism.columbia.edu/page/1058-the-lede-program-an-introduction-to-data-practices/906

TABLE I. MAJOR TOPICS IN THE PROPOSED COURSE ON DATA SCIENCE FOR NON-PROGRAMMERS.

Section	Lesson topics
Data	What is data and what is not data; time series data; network data; geospatial data; text data; labeled and annotated data; big data
Data analysis software	Software for data analysis; inputs and outputs of programs; program parameters; programming languages; programs as black boxes
Multi-step data analysis	Pre-processing and post-processing data; building workflows by composing programs; workflows for data analysis; workflow inputs and parameters; running a workflow
Data analysis tasks	What is a data analysis task; prediction; classification; clustering; pattern detection; anomaly detection
Data pre-processing	Data cleaning; quality control; data integration; feature selection
Data post-processing	Summarization; filtering; visualization
Analyzing different types of data	Analyzing time series data; analyzing networked data; analyzing geospatial data; analyzing text; analyzing images; analyzing video
Parallel computing	Cost of computation; parallel processing; multi-core computing; distributed computing; speedup with parallel computing; dependencies across computations; limits of parallel speedup; execution failures and recovery; reduction
Semantic metadata	What is metadata; basic metadata vs semantic metadata; metadata about data collection; metadata about data processing; metadata for search and retrieval; metadata standards; domain metadata and ontologies
Provenance	What is provenance; provenance concerning data; provenance concerning agents; provenance concerning processes; provenance models; provenance standards
Semantic workflows	What is a semantic workflow; validating data analysis methods; automatically generating metadata; tracking provenance; publishing workflows; finding workflows
Visualization	Time series visualizations; geospatial visualizations; multi-dimensional spaces
Data stewardship	Data sharing; data identifiers; licenses for data; data citation and attribution
Data formats and standards	Data formats; data standards; data services; ontologies; linked open data

Learning these concepts must be supplemented with practice. But how will students with no programming skills be able to see programs in action? A major component of the course is the use of a semantic workflow system, described in Section 4, to enable students to practice complex data analysis concepts, particularly parallel and distributed computing.

III. PARALLEN AND DISTRIBUTED PROGRAMMING FOR NON-PROGRAMMERS

Table II shows in more detail the topics that we propose to cover regarding parallel and distributed computing. It also shows the learning outcomes that we target for each of the topics. These learning outcomes are in terms of the student understanding those topics by being able to determine the

applicability of relevant concepts to their own data and context.

TABLE II. LESSONS IN PARALLEL AND DISTRIBUTED COMPUTING FOR THE PROPOSED COURSE ON DATA SCIENCE FOR NON-PROGRAMMERS.

Lesson	Learning outcomes: Concepts that the student will understand
1. Cost of computation	Scaling behavior of different algorithms as data grows; limitations of sequential computation in the face of large datasets
2. Divide and conquer	Breaking down problems into smaller tasks can make problems more manageable; smaller tasks can be more amenable to a more scalable approach; parallel computing as a special case of divide and conquer
3. Parallel computing	Processing data concurrently through multiple processes; splitting large datasets into smaller ones to be processed in parallel
4. Multi-core computing	Parallel computing in a single computer with multiple processors
5. Distributed computing	Parallel computing in multiple networked computers
a. Cluster computers	Homogeneous computers accessible through a single queue
b. Cloud computing: Azure, EC3	Computing as a service; cost of computing vs cost of data uploads/downloads
c. Grid computing: Globus, Condor	Heterogeneous computers accessible through a grid
d. Virtual machines	Specifications of software requirements to be set up in a machine
e. Web services	Distributed computing through remove invocation of third-party services
6. Speedup with parallel computing	Measuring the time savings of parallel processing
7. Dependencies and message passing	Tightly-coupled computations that require communication among processors
8. Limits of speedup: Critical path	Time savings can not always be achieved; critical paths in an end-to-end data processing application
a. Amdahl's law	Measuring the time savings when only some portions of an application can be parallelized
9. Embarrassingly parallel computations	Massively parallel computing
10. When problems are not parallelizable	Not all applications lend themselves to parallel processing
11. Execution failures	Remote computers can fail; managing failures in a large distributed application
12. Reduction through MapReduce/Hadoop	Reduction as a paradigm for parallelization; MapReduce/Hadoop approach

We have noticed that the MapReduce/Hadoop paradigm is often mentioned in technical discussions on big data and data science. However, only programmers understand and appreciate the features of this paradigm. Similarly, cloud computing is a widely known term that very few understand. Making such common terms accessible and understood by non-programmers is one of our goals.

The lessons also convey notions of algorithmic complexity and computational cost. We view parallel computing as an ideal mechanism to illustrate these concepts and enable non-programmers to learn to think computationally [Wing 2006].

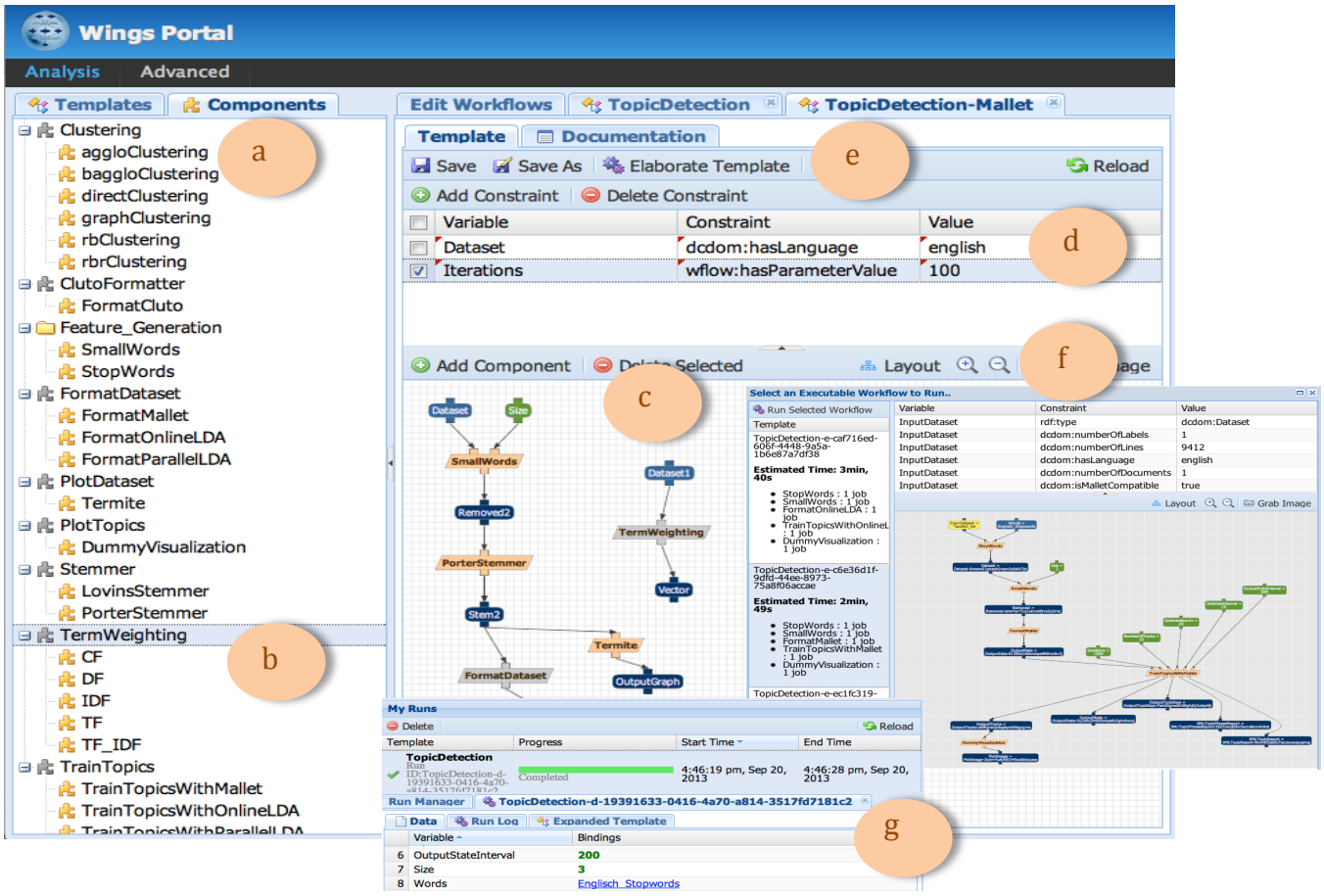

Figure 1. The WINGS user interface for composing and validating workflows: (a) a library of components is available to the user including component classes organized in a hierarchy, (b) a component or a component class can be selected and dragged and dropped into the canvas, (c) once dropped the component can be connected to express dataflow, (d) semantic constraints restrict the use of datasets and the values of parameters in the workflow, (e) the student can ask the system to validate a workflow by reasoning about the semantic constraints, (f) the student can explore and run different workflows. The examples here are for text analytics (from [Hauder et al 2011a]; Hauder et al 2011b].

IV. Semantic Workflows

To enable students to practice and experience complex data science concepts, we allow the students to interact with a workflow system that has predefined workflows that they can run and explore. The workflow system uses semantic constraints to ensure that the workflows are used properly.

We use the WINGS semantic workflow system (http://www.wings-workflows.org). WINGS is an intelligent workflow system that can assist users, and therefore students, to create valid workflows [Gil et al 2010] and automate the elaboration of high-level workflows [Gil et al 2011a; Gil et al 2011b]. Users find pre-defined workflows and workflow components that they can reuse and extend to create their own workflows. As users select and configure workflows to be executed, WINGS ensures that workflows are correctly composed by checking that the data is consistent with the semantic constraints defined for the workflow and its components. Users can track execution progress and view results.

Workflows offer a visual programming language for complex multi-step data analytics. We have reported on non-programmers easily using complex data analysis workflows [Hauder et al 2011].

Workflows have been used in courses to teach visualization [Silva et al 2011]. We believe that they can be a powerful paradigm to teach other concepts in data science.

WINGS has been used for physics-based seismic hazard analysis [Gil et al 2007b; Maechlin et al 2005], climate model comparison, water quality [Gil et al 2011c; Villamizar et al 11], biomedical image analysis [Kumar et al 2010; Kurc et al 2009], text analytics [Hauder et al 2011a; Gil et al 2013a], image and video analysis [Sethi et al 2013a; Sethi et al 2013b], population genomics [Gil et al 2012; Gil et al 2013b], and clinical cancer omics [Gil et al 2013b]. These workflows can be used to illustrate different topics in the course.

Figure 1 shows a snapshot of the WINGS user interface for composing and validating workflows, in this case using workflows for text analytics [Hauder et al 2011a; Hauder et al 2011b]. WINGS can validate the workflows that are

created by the user by reasoning about the semantic constraints that have been defined for the workflow, its components, and all the associated input and output object

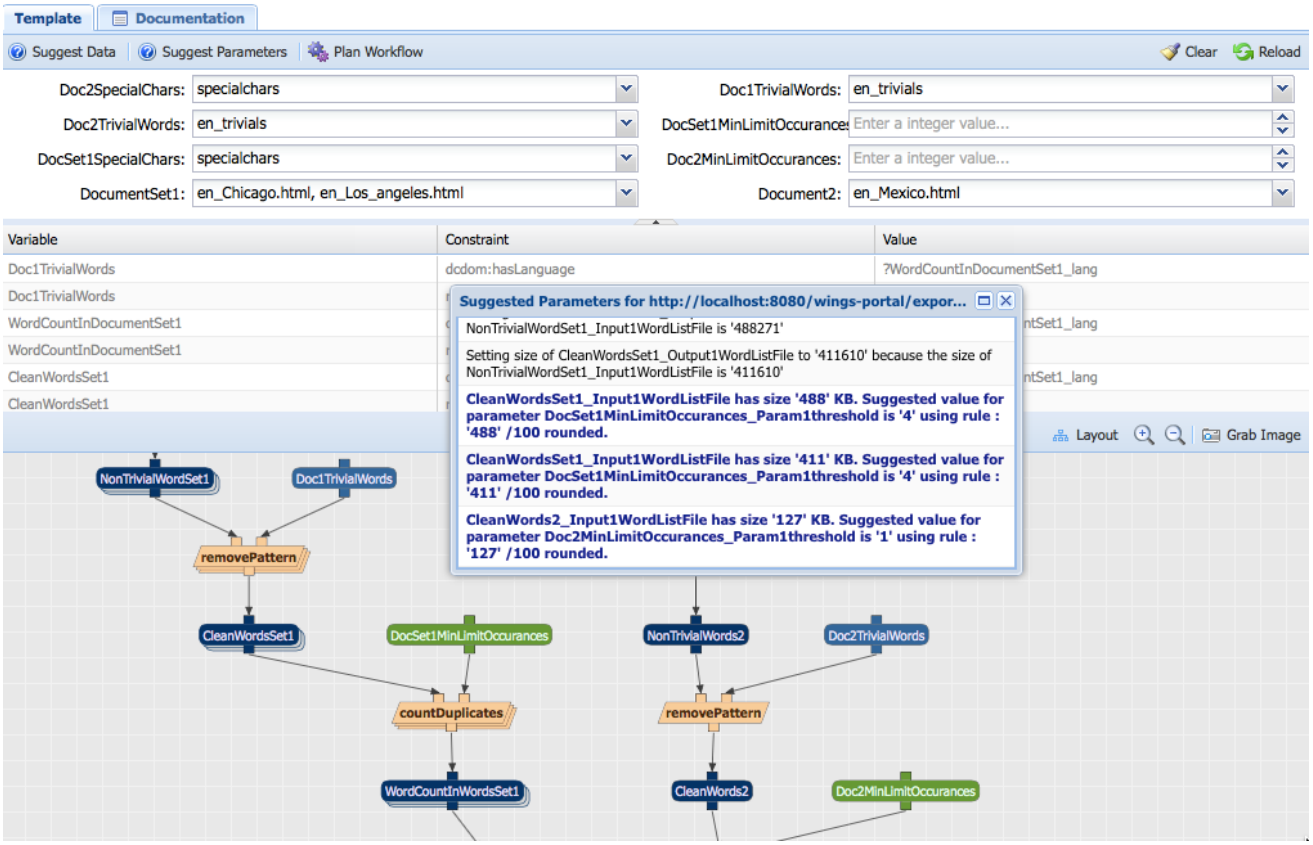

Figure 2. The student selects the data for the workflow, and can ask WINGS to suggest values for the parameters.

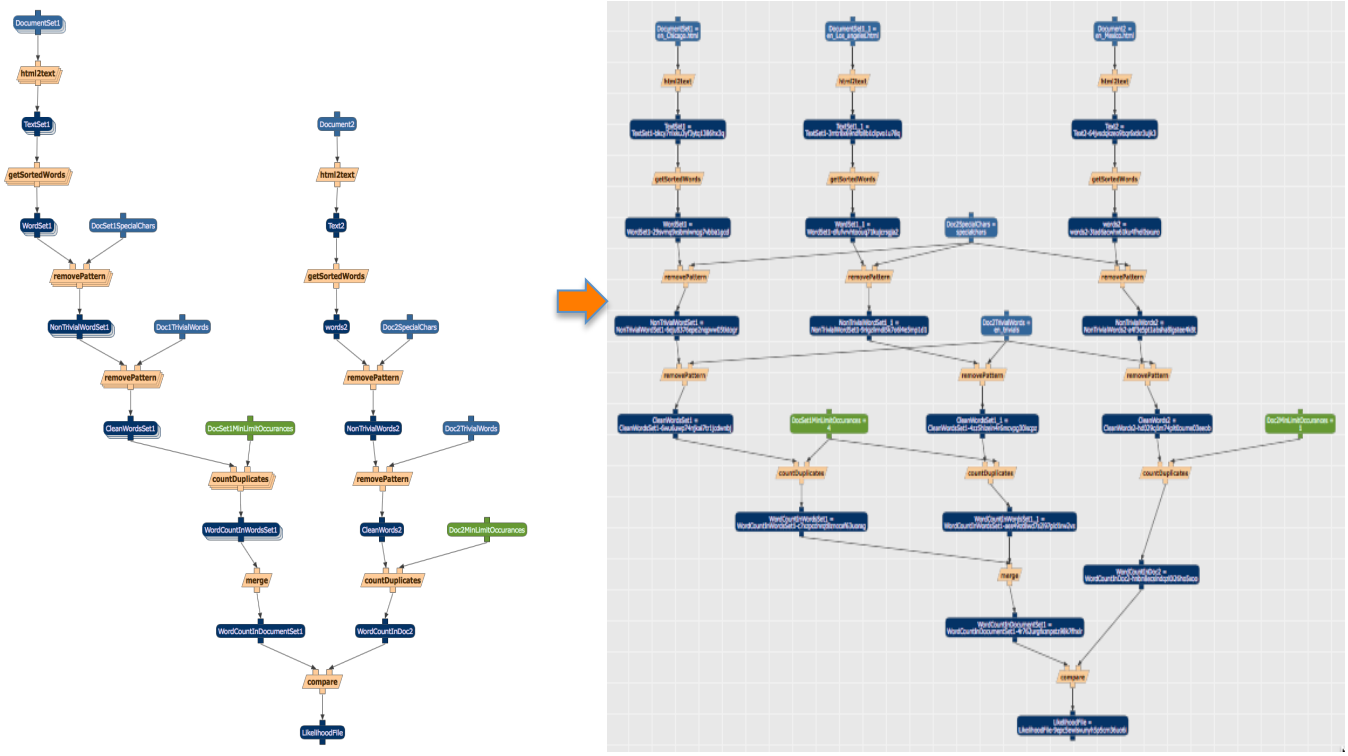

Figure 3. The student selects the workflow on the left and 3 datasets, and can ask WINGS to generate the workflow on the right which will process the 3 datasets in parallel.

variables. WINGS can also elaborate workflows by adding details about additional parameters as well as constraints for the object variables. A high-level introduction to WINGS can be found in [Gil et al 2011a], a formal description of the workflow representation language and reasoning algorithms is given in [Gil et al 2011b].

WINGS is released as open source software, and uses open standards. In particular, Wings uses the W3C RDF standard [Brickley and Guha 2004] to represent semantic constraints, and uses other W3C semantic web standards such as SPARQL for queries and PROV for provenance [Gil and Miles 2013].

V. Semantic Workflows for Student Practice

Figure 2 illustrates how WINGS helps students to run valid workflows that exemplify complex data analyses. In this case, the workflow classifies text into categories, and has two parameters to be set. The user can ask WINGS to suggest values for those parameters, which WINGS will do based on the data selected by the user.

Figure 3 shows how WINGS helps users understand parallel programming concepts. Given the workflow template on the left, which indicates parallel processing through stacked boxes, WINGS can expand it to generate an executable workflow. Once the user selects input datasets, in this case 3 different ones, WINGS generates the workflow on the right, which shows how may processes will be run for the data selected.

WINGS also enables the students to see the intermediate and final results of the workflow execution. This helps them understand what is happening in each of the branches of the computation, and how the results are put together to generate a single output of the workflow.

As a pre-test, we used workflows to teach core concepts of parallel programming to two students. Neither one had programming knowledge. Both understood the concepts, and found the materials accessible. Both found the concepts taught to be potentially useful. More thorough tests will be required in order to ensure reasonable confidence that the material is accessible.

VI. Conclusions

We have proposed a course for non-programmers to learn about data science, and in particular concepts of parallel and distributed computing. The course allows the students to practice by using semantic workflows. The workflows capture complex multi-step data analysis methods, which include semantic constraints about their use. This enables the workflow system to validate the workflows and assist the students to set up the analysis properly.

Acknowledgment

We gratefully acknowledge support from the National Science Foundation (NSF) with award ACI-1355475.

References

[1] [Callahan et al 2006] "Managing the Evolution of Dataflows with VisTrails." Steven P. Callahan, Juliana Freire, Emanuele Santos, Carlos E. Scheidegger, Claudio T. Silva and Huy T. Vo. Proceedings of IEEE Workshop on Workflow and Data Flow for Scientific Applications (SciFlow), 2006.

[2] [De Roure et al 2009] De Roure, D; Goble, C.; Stevens, R. "The design and realizations of the myExperiment Virtual Research Environment for social sharing of workflows". Future Generation Computer Systems, 25 (561-567), 2009.

[3] [Deelman et al 2005] "Pegasus: a Framework for Mapping Complex Scientific Workflows onto Distributed Systems." Deelman, E., Singh, G., Su, M. H., Blythe, J., Gil, Y., et al. Scientific Programming Journal, vol. 13, 2005.

[4] [Dinov et al 2009] Dinov I, Van Horn JD, Lozev KM, Magsipoc R, Petrosyan P, Liu Z, MacKenzie-Graham A, Eggert P, Parker DS and Toga AW (2009) Efficient, Distributed and Interactive Neuroimaging Data Analysis using the LONI Pipeline. Front. Neuroinform., 3(22):1-10.

[5] [Dooley et al 2006] "From Proposal to Production: Lessons Learned Developing the Computational Chemistry Grid Cyberinfrastructure," R. Dooley, K. Milfeld, C. Guiang, S. Pamidighantam and G. Allen, 4(2), 2006.

[6] [Garijo and Gil 2011] "A New Approach for Publishing Workflows: Abstractions, Standards, and Linked Data." Daniel Garijo and Yolanda Gil. Proceedings of the 6th Workshop on Workflows in Support of Large-Scale Science (WORKS-11), held in conjunction with SC-11, 2011.

[7] [Garijo et al 2013] "Quantifying Reproducibility in Computational Biology: The Case of the Tuberculosis Drugome." Daniel Garijo, Sarah Kinnings, Li Xie, Lei Xie, Yinlinag Zhang, Philip E. Bourne, and Yolanda Gil. 2013.

[8] [Gil and Miles 2013] "A Primer for the PROV Provenance Model." Gil, Y. and Miles, S. (Eds). World Wide Web Consortium (W3C) Working Group Note, 2013.

[9] [Gil et al 2009] Gil, Y., Groth, P., Ratnakar, V., and C. Fritz. "Expressive Reusable Workflow Templates." Proceedings of the IEEE e-Science Conference, Oxford, UK, 2009.

[10] [Gil et al 2007a] Gil, Y.; Cheung, W. K.; Ratnakar, V.; and Chan, K. "Privacy Enforcement in Data Analysis Workflows." Proceedings of the AAAI Workshop on Privacy Enforcement and Accountability with Semantics (PEAS), held in conjunction with the Sixth International Semantic Web Conference (ISWC) and the Second Asian Semantic Web Conference (ASWC), Busan, Korea, 2007.

[11] [Gil et al 2007b] "Wings for Pegasus: Creating Large-Scale Scientific Applications Using Semantic Representations of Computational Workflows." Yolanda Gil, Varun Ratnakar, Ewa Deelman, Gaurang Mehta, and Jihie Kim. Proceedings of the 19th Annual Conference on Innovative Applications of Artificial Intelligence (IAAI), Vancouver, British Columbia, Canada, July 22-26, 2007.

[12]

[13] [Gil et al 2011a] Gil, Y.; Ratnakar, V.; Kim, J.; Gonzalez-Calero, P. A.; Groth, P.; Moody, J.; and Deelman, E. "Wings: Intelligent Workflow-Based Design of Computational Experiments." IEEE Intelligent Systems, 26(1). 2011.

[14] [Gil et al 2011b] Gil, Y.; Gonzalez-Calero, P. A.; Kim, J.; Moody, J.; and Ratnakar, V. "A Semantic Framework for Automatic Generation of Computational Workflows Using Distributed Data and Component

Catalogs." Journal of Experimental and Theoretical Artificial Intelligence, 23(4), 2011.

[15] [Gil et al 2011c] "Mind Your Metadata: Exploiting Semantics for Configuration, Adaptation, and Provenance in Scientific Workflows." Gil, Y.; Szekely, P.; Villamizar, S.; Harmon, T.; Ratnakar, V.; Gupta, S.; Muslea, M.; Silva, F.; and Knoblock, C. (2011). Proceedings of the Tenth International Semantic Web Conference (ISWC), Bonn, Germany.

[16] [Gil et al 2012] "Using Semantic Workflows for Genome-Scale Analysis." Gil, Y., Deelman, E. and Mason, C. Proceedings of the Tenth Annual International Conference on Intelligent Systems for Molecular Biology (ISMB), Long Beach, CA, 2012.

[17] [Gil et al 2013a] "Time-Bound Analytic Tasks on Large Datasets through Dynamic Configuration of Workflows", Gil, Y., Ratnakar, V., et al. Proceedings of the 8th Workshop on Workflows in Support of Large-Scale Science (WORKS-13), held in conjunction with SC-13, 2013.

[18] [Gil et al 2013b] "Using Semantic Workflows to Disseminate Best Practices and Accelerate Discoveries in Multi-Omic Data Analysis." Gil, Y.; McWeeney, S.; and Mason, C. E. Proceedings of the AAAI Workshop on Expanding the Boundaries of Health Informatics using AI (HIAI), held in conjunction with the Conference of the Association for the Advancement of Artificial Intelligence (AAAI), Bellevue, WA, 2013.

[19] [Hauder et al 2011a] Hauder, M., Gil, Y. and Liu, Y. "A Framework for Efficient Text Analytics through Automatic Configuration and Customization of Scientific Workflows". Proceedings of the Seventh IEEE International Conference on e-Science, Stockholm, Sweden, December 5-8, 2011.

[20] [Hauder et al 2011b] Hauder, M.; Gil, Y.; Sethi, R.; Liu, Y.; and Jo, H. "Making Data Analysis Expertise Broadly Accessible through Workflows." Proceedings of the Sixth Workshop on Workflows in Support of Large-Scale Science (WORKS'11), held in conjunction with SC 2011, Seattle, WA, 2011.

[21] [Hull et al 2006] "Taverna: A Tool for Building and Running Workflows of Services", Hull D, Wolstencroft K, Stevens R, Goble C, Pocock M, Li P, Oinn T. Nucleic Acids Research, Vol 34, 2006.

[22] [Kale et al 2013] "Capturing Data Analysis Expertise with Visualization in Workflows." Kale, D.; Di, S.; Liu, Y.; and Gil, Y. In AAAI Fall Symposium on Discovery Informatics: AI Takes a Science-Centered View on Big Data, 2013.

[23] [Kurc et al 2009] "High Performance Computing and Grid Computing for Integrative Biomedical Research." Kurc, T. M.; Hastings, S.; Kumar, V. S.; Langella, S.; Sharma, A.; Pan, T.; Oster, S.; Ervin, D.; Permar, J.; Narayanan, S.; Gil, Y.; Deelman, E.; Hall, M. W.; and Saltz, J. H. 2009. Journal of High Performance Computing Applications, 23(3):252-264.

[24] [Kumar et al 2010] "Parameterized Specification, Configuration, and Execution of Data-Intensive Scientific Workflows." Vijay Kumar, Tahsin Kurc, Varun Ratnakar, Jihie Kim, Gaurang Mehta, Karan Vahi, Yoonju Lee Nelson, P. Sadayappan, Ewa Deelman, Yolanda Gil, Mary Hall, Joel Saltz. Cluster Computing Journal, Vol 13, 2010.

[25] [Ludäscher et al 2006] "Scientific workflow management and the Kepler system." LudLudäscher, B. Concurrency and Computation: Practice and Experience. 18, 2006.

[26] [Maechlin et al 2005] "Simplifying construction of complex workflows for non-expert users of the Southern California Earthquake Center Community Modeling Environment," P. Maechling, H. Chalupsky, M. Dougherty, E. Deelman, Y. Gil, S. Gullapalli, V. Gupta, C. Kesselman, J. Kim, G. Mehta, B. Mendenhall, T. A. Russ, G. Singh, M. Spraragen, G. Staples, and K. Vahi, SIGMOD Record, vol. 34, pp. 24-30, 2005.

[27] [Prasad et al 2012] Prasad Sushil K., Gupta Anshul, Kant Krishna, Lumsdaine Andrew, Padua David, Robert Yves, Rosenberg Arnold, Sussman Alan, Weems Charles, Literacy for All in Parallel and Distributed Computing: Guidelines for an Undergraduate Core Curriculum, CSI Journal of Computing, Vol 1. 2, 2012.

[28] [Reich et al 2006] "GenePattern 2.0". Reich M, Liefeld T, Gould J, Lerner J, Tamayo P, Mesirov JP. Nature Genetics. 38(5):500-501, 2006.

[29] [Sendlinger et al 2008] "Transforming Chemistry Education through Computational Science," Sendlinger, S.C.; DeCoste, D.J.; Dunning, T.H.; Dummitt, D.A.; Jakobsson, E.; Mattson, D.R.; Wiziecki, E.N. Computing in Science & Engineering, 2008.

[30] [Sethi et al 2013a] "Structured Analysis of the ISI Atomic Pair Actions Dataset Using Workflows." Sethi, R. J.; Jo, H.; and Gil, Y. Pattern Recognition Letters, 34(15). 2013.

[31] [Sethi et al 2013b] "Large-Scale Multimedia Content Analysis Using Scientific Workflows." Sethi, R. J.; Gil, Y.; Jo, H.; and Philpot, A. In Twenty-First ACM International Conference on Multimedia, Barcelona, Spain, 2013.

[32] [Silva et al 2011] "Using VisTrails and Provenance for Teaching Scientific Visualization." C. Silva, E. Anderson, E. Santos, and J. Freire. Computer Graphics Forum, 30(1), 2011.

[33] [Villamizar et al 2011] "Scientific workflows to assess the response of the Californian San Joaquin River to flow restoration efforts." Villamizar, Sandra; Gil, Yolanda; Szekely, Pedro; Ratnakar, Varun; Gupta, Shubham; Muslea, Maria; Silva, Fabio; and Harmon, Thomas. Fall Meeting of the American Geophysical Union, 2011.

[34] [Wing 2006] Jeannette Wing. "Computational Thinking". Communications of the ACM vol. 49, no. 3, March 2006.