IJDC | *Peer-Reviewed Paper*

Documentation and Visualisation of Workflows for Effective Communication, Collaboration and Publication @ Source

Cerys Willoughby
University of Southampton

Jeremy G. Frey
University of Southampton

Abstract

Workflows processing data from research activities and driving in silico experiments are becoming an increasingly important method for conducting scientific research. Workflows have the advantage that not only can they be automated and used to process data repeatedly, but they can also be reused – in part or whole – enabling them to be evolved for use in new experiments. A number of studies have investigated strategies for storing and sharing workflows for the benefit of reuse. These have revealed that simply storing workflows in repositories without additional context does not enable workflows to be successfully reused. These studies have investigated what additional resources are needed to facilitate users of workflows and in particular to add provenance traces and to make workflows and their resources machine-readable. These additions also include adding metadata for curation, annotations for comprehension, and including data sets to provide additional context to the workflow. Ultimately though, these mechanisms still rely on researchers having access to the software to view and run the workflows. We argue that there are situations where researchers may want to understand a workflow that goes beyond what provenance traces provide and without having to run the workflow directly; there are many situations in which it can be difficult or impossible to run the original workflow. To that end, we have investigated the creation of an interactive workflow visualization that captures the flow chart element of the workflow with additional context including annotations, descriptions, parameters, metadata and input, intermediate, and results data that can be added to the record of a workflow experiment to enhance both curation and add value to enable reuse. We have created interactive workflow visualisations for the popular workflow creation tool KNIME, which does not provide users with an in-built function to extract provenance information that can otherwise only be viewed through the tool itself. Making use of the strengths of KNIME for adding documentation and user-defined metadata we can extract and create a visualisation and curation package that encourages and enhances curation@source, facilitating effective communication, collaboration, and reuse of workflows.

Received 20 October 2016 ~ *Accepted* 23 February 2017

Correspondence should be addressed to Cerys Willoughby, School of Chemistry, University of Southampton, Southampton, SO17 1BJ. Email: cerys.willoughby@soton.ac.uk

An earlier version of this paper was presented at the 12th International Digital Curation Conference.

The *International Journal of Digital Curation* is an international journal committed to scholarly excellence and dedicated to the advancement of digital curation across a wide range of sectors. The IJDC is published by the University of Edinburgh on behalf of the Digital Curation Centre. ISSN: 1746-8256. URL: http://www.ijdc.net/

Copyright rests with the authors. This work is released under a Creative Commons Attribution (UK) Licence, version 2.0. For details please see http://creativecommons.org/licenses/by/2.0/uk/

Introduction

Computers are pervasive in science and computation is becoming central to scientific activity, with a significant amount of data now 'born digital' driving a move away from traditional scholarly publication towards electronic publication methods with computer-readable rather than human-readable formats (Bechhofer et al., 2010; Bird et al., 2013a; Donoho et al., 2009). The proliferation of digital data produced by scientific activities means that complex computational processes need to be assembled to tackle the job of analysing and understanding scientific data (Davidson and Freire, 2008). Workflow systems that not only enable the automation of repetitive tasks, but also capture complex processes and provenance information, are replacing ad-hoc approaches, such as scripting, previously used to process and manage data from scientific activities (Davidson and Freire, 2008). Many of these workflow systems are used to generate and output result datasets and other data products that are increasingly stored in data repositories separate from the resources that were used to generate them. Descriptions of how the data was produced are confined to 'materials and methods' sections of paper publications and are usually insufficient to enable practical or usable reporting of all the various settings and parameters used in the workflow (Hutton et al., 2016; Missier et al., 2012). This disconnection between the results and methods often leads to difficulties when trying to reproduce the original work, as discussed in more detail below.

Computational models and workflows have the benefit that they are reusable, not just by the authors of the workflows, but also by their colleagues, collaborators, and the wider community. Workflows can reduce the effort involved in ensuring consistency and reproducibility of computational processes (Garijo, 2013). There are many workflow systems available, and most provide the ability to share the workflows with other users who have access to the system, for example access to a cloud via a log-in, access to a corporate version of the software on an organisational server, or by sharing resources on a local software install. There are situations, however, where the workflow system does not readily enable the sharing of resources or where the author may want to deposit the workflow resources in a different location for storage, sharing or publication purposes. In these situations, anyone attempting to make use of the workflow need to be able to understand it, for example to repeat the experiment, to build a new experiment based on the original, or for producing publications.

A large amount of scientific research may be unreliable or be unable to be reproduced because of a lack of transparency leading to a growing credibility gap in computational science (Donoho et al., 2009; Hutton et al., 2016; Iqbal et al., 2016). As a result there has been a movement towards making data openly available with journal publications, including journals specifically for publishing datasets, for example Scientific Data[1], Journal of Chemical and Engineering Data[2], and Geoscience Data Journal[3]. Although some consider that data is 'self-apparent' and only needs limited documentation to support it, others argue that data and even the workflows that produced them are insufficient to enable independent assessment of research results and reproducibility of research (Belhajjame et al., 2015; Zhao et al., 2012).

There are a number of factors that can hinder reproducibility of computational methods and workflows including:

1 Scientific Data: http://www.nature.com/sdata/about
2 Journal of Chemical and Engineering Data: http://pubs.acs.org/page/jceaax/about.html
3 Geoscience Data Journal: http://onlinelibrary.wiley.com/journal/10.1002/(ISSN)2049-6060

- Insufficient documentation and annotation is provided, lacking information about inputs, intermediate steps and outputs (Belhajjame et al., 2015; Fuller, 2015);
- Important computational steps missing or ambiguous (Garijo et al., 2013);
- Parameter settings and configuration files missing (Garijo et al., 2013);
- Third-party dependencies, such as Web services, are not accessible (Belhajjame et al., 2015; Garijo et al., 2013);
- Example data, input data, and intermediate data is missing (Belhajjame et al., 2015; Garijo et al., 2013);
- Software no longer runs old computations or workflows (Berthold, 2015; Garijo et al., 2013);
- Use of proprietary software and use of proprietary scripts at intermediate parts of the process or specific software infrastructures need to be installed locally (Belhajjame et al., 2015; Garijo et al., 2013).

To counter these problems, several authors have suggested that input datasets, intermediate results, and a dataflow diagram of the computation are also necessary for reproducibility (Belhajjame et al., 2015; Garijo et al., 2013) and there is a call for publications to include all primary data and data sources, together with all scripts, software source code, instructions, and step-by step protocols (Baggerly, 2010). Some authors also suggest there should be 'workflow publication' where end-to-end methods are published as workflows (Garijo et al., 2013). There are currently a limited number of journals that encourage the publication of all of these materials. GigaScience[4], for example, aims to link manuscript publications with not only associated data, but also data analysis tools and cloud-computing resources. Workflow sharing services such as myExperiment encourage researchers to package all the resources relating to a publication together so that a link can be made between the publication and the resources (De Roure, 2009). However, there is currently a lack of software tools that provide mechanisms to package computational analyses for sharing and reuse (Dudley and Butte, 2010).

Even when the source code, scripts and other data resources are included with a publication they rely on software that may not be available to another researcher who wants to understand the data and the processes that were followed due to costs associated with licensing, lack of appropriate environments, or skills. For this reason, some authors even advocate inclusion of the execution or runtime environment, so that workflows can be re-run and examined with the minimum of user interaction, for example by using virtual machines or running the workflows in the cloud (Dudley and Butte, 2010; Gorp and Manzanek, 2011; Hasham et al., 2015). However, even when the execution environment, supporting scripts, and sufficient computational resources are available, it can be difficult to replicate computations on new software and may take many months to reproduce, even for domain experts (Garijo et al., 2013). Just preserving and publishing workflows does not mean that it is possible to successfully understand, run, and reuse them (Belhajjame et al., 2015).

Research Objects (RO) are an example of enriching workflows by aggregating them with other resources that make up an experiment for the purpose of sharing or to support some research objective (Bechhofer et al., 2010; De Roure, 2009). Metadata is used to describe both the individual elements of the package and also to describe the relationships and structures within the Research Object bundle (Bechhofer et al., 2013).

4 GigaScience: http://www.gigasciencejournal.com

Research Objects are recommended to contain background to the research problem, organization context, the research design, methods (including workflows, scripts, and software), input data, results, and publications altogether (Bechhofer et al., 2013). Although no particular format is mandated there are suggested ontologies and RDF formats for representing the objects and their relationships with a drive to make the packages machine-readable (Belhajjame et al., 2015). The package or 'compendium' of resources created acts as a container for the various elements and potentially aids in storage or distribution, but also has the potential to be interactive, dynamic, and provide different views on the content (Gentleman and Lang, 2007). Such a package should also have an appropriate 'digital object identifier' together with appropriate documentation and metadata to enable good management and curation practices (Goodman et al., 2014).

These additional resources can help to provide additional context for the workflow, which can help with understandability, reproducibility, and reusability, but can still have the problems of relying on access to the execution environment and lead to a disconnection between the underlying configuration and the workflow. Creating machine-readable resources is important for findability, but does not necessarily help improve understandability for researchers once a resource has been found. Tools have been developed that extract provenance and process information from workflows and other computational resources making use of standard ontologies and models (Cuevas-Vicenttín et al., 2014; McPhillips et al., 2015). However, the generated notations and even diagrammatic representations of provenance information, although suitable for developers familiar with these models, are not a practical way to communicate information about workflows to the majority of users (Richardson and Moreau, 2016); better methods are needed to present this information to users in a way that is usable. At the very least this representation should be a simple sketch of the flow of data across the software showing how the intermediate and final data products are generated (Belhajjame et al., 2015; Garijo et al., 2013; Goodman et al., 2014). Although simple screen grabs can capture this information, they do not provide access to information about the settings and parameters or the underlying transformation in the workflow. Human and machine added annotations can assist users in understanding the meaning of data products or scientific applications (da Cruz et al., 2009). Adding in graphical representations and additional documentation can be used to bridge the gap between user-friendly notebooks and extracted provenance information to provide more detail and greater experimental insight (Wibisono et al., 2015).

One of the benefits of the workflow format – a flow chart style – is that its relative simplicity facilitates understanding of processes, even for researchers from different backgrounds (Desaulniers et al., 1988; Fuller, 2015; Ko et al., 2009; Scanlan, 1989); although this is dependent on avoiding of complex technical notation (Ungan, 2006). Much of the complexity of the workflow is hidden in the metadata and configuration of the components making up the workflow. Important context and understanding can be lost if the complex configuration is presented separately from the visual representation of the workflow.

We assert that the creation of human-readable resources, in conjunction with research object type resources, is important to enable human understanding of the workflow and its context without the user needing to know how to use, or have access to, the software that was used to create the workflow.

In this paper we describe our project to generate an interactive visualization of a workflow that enables a user without access to the original workflow system to be able to view and understand a workflow from a single user-friendly representation that provides an image of the workflow together with additional context, such as design information, parameters and data. This visualisation can be a personal record of the

 doi:10.2218/ijdc.v12i1.532

workflow creation and deployment, used for collaboration for discussing the workflow processes and results, for reuse and validation to understand the construction and function of the workflow, for publication and for archive. The visualisation extracts and presents curation and provenance metadata in a user-readable format that can also be output in a machine-readable form from the same source to enable a workflow package to be produced that can be used by both humans and applications, for example by creating a machine-readable representation of the metadata associated with a Workflow Research Object bundle as recommended by Belhajjame et al. (2015).

The project builds upon our previous research considering capture at source and adequate documentation as a mechanism for the 'preservation of memory' for research (Bird et al., 2013b).

We considered a number of different workflow systems for the project, including E-science central, Galaxy, Taverna, Kepler, Pipeline Pilot, Vistrails, and InforSense, but ultimately chose KNIME (Berthold et al., 2008) as the platform for the project. We selected KNIME because it is easy to use, and has well developed multidisciplinary communities in which it is well used (Bodkin, 2012). KNIME is also well supported through good documentation and community forums, is reliable, and facilitates a broad range of data processing and modelling. These features of KNIME provide a large potential user base for a visualization, documentation and packaging tool to be integrated in the product. In this paper we present KNIME as an example for the benefits of creating an interactive visualisation, documentation and packaging tool for workflows.

KNIME Workflows

KNIME is an extendable, open-source workflow development environment based on the Eclipse IDE. Workflows are constructed visually using standardised building blocks in the form of nodes that are connected with pipes that pass along data or models to the next node (Berthold et al., 2008). A simple example workflow from KNIME is shown in Figure 1; in this workflow data is read from a CSV file and undergoes some calculations and transformations, finally producing a results table and scatter plot to represent the data.

KNIME includes over one thousand built-in nodes with a variety of input, output, connectivity, transformation, and analysis capabilities. These enable data from a variety of local and online sources to be accessed, processed, analysed, and output within a single workflow. The open nature of KNIME also enables the product to be extended; in addition to the built-in nodes, there are also hundreds more nodes available as third-party extensions from community developers and partner organisations. Facilities exist for individuals and organisations to create their own nodes for private use or to contribute to community extensions. Existing workflows can also be utilised as if they were a single node through the use of meta-nodes or wrapped nodes.

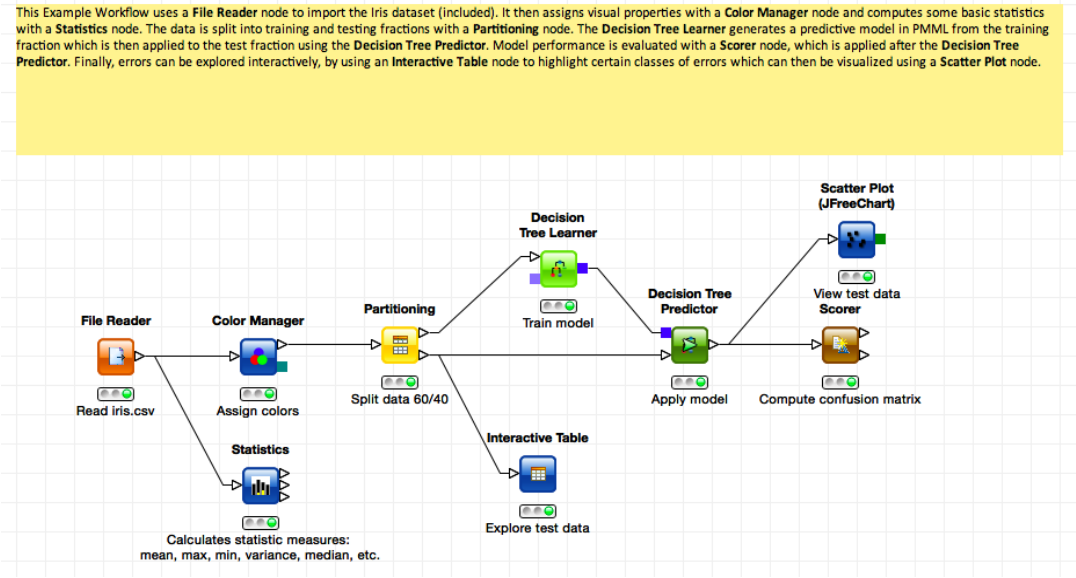

Figure 1. Example workflow in KNIME.

KNIME enables sharing of workflows, primarily for organisations through the team collaboration version KNIME Teamspace or the cloud-based KNIME Server edition, enabling workflows to be reused and or processes to be linked together (Fuller, 2015). This enables members of the same organisation that use KNIME to view and run the workflows, but sharing resources outside of this environment is more difficult, although KNIME workflows are shared on the KNIME community forums and on the myExperiment[5] collaborative environment.

Curation @ Source in KNIME

KNIME automatically acquires both system and context metadata that can then be accessed from workflows. The system metadata includes information about the underlying Eclipse instance and Java version, but also information about the system user, directories and system locale, language, and time zone settings. The context metadata is specific to the workflow and includes filenames and paths, author, editor, and creation and modification date-time stamps. This metadata – together with the details of the nodes, connections, settings, and data in the workflow – enables supporting provenance traces to be extracted from a workflow and exposed in a human-readable (and machine-readable way) as part of the visualisation and workflow package.

KNIME provides a number of ways that users can add documentation, annotations, and user-defined metadata to their workflows. This metadata can be used to make the workflows easier to understand, in the same way as comments make code more maintainable and easier to follow, for example users can choose to add the following elements that can be incorporated in a visualisation that is created to help support understanding and reuse:

- A custom name for the node (in addition to the name indicating the type of node) – acting a short description for the node;
- Adding annotations to the workflow – typically to describe the function of a group of nodes;

5 MyExperiment: http://www.myexperiment.org/

 doi:10.2218/ijdc.v12i1.532

- Custom node descriptions – these are only visible from the context menu on each node, but they do provide a place where a long description can be provided for each node that can be captured for a visualisation or workflow description.

Users can also add their own user-defined variables in the form of key-value pairs that can be accessed within the workflows. These can be used to dynamically alter settings without updating the workflow code, or purposed for adding information such as version numbers, URIs, or any other information that might be useful for curation.

Curating the Workflow

Our primary objective in this project was to create a proof-of-concept interactive graphical visualisation of the workflow to expose the configuration and metadata associated with the workflow, to create links to the data generated by the workflow, and to package these resources together such that they could easily be shared through everyday mechanisms such as email or Dropbox[6], or alternatively deposited in a workflow sharing repository, data repository, within an ELN or other archive service. The visualisation could then be accessed and used by collaborators and members of the community to be able to gain an understanding of the purpose, function, and technical implementation of the workflow without needing to access the execution environment. The visualisation would be able to provide answers for user questions about the workflow and computations including prospective provenance, retrospective provenance, and details of the execution context. For example, exposing the reasoning behind the experiment or series of experiments, documenting the sources used, the function and structure of processing steps and scripts, variables used in the run, number of rows of data before and after a transformation, and paths and results from a specific sample.

In addition to creating a visualisation, we also wanted to enable any additional annotations, descriptions, and background narrative of the workflow, together with selected data such as input, intermediate, and results data, to be displayed and accessible from the relevant parts of the workflow visualization. The following requirements were defined for the project:

1. Generate a visual representation of the workflow including the nodes, node names, and connections between nodes;
2. Capture and display any annotations the user made to the workflow;
3. Provide a way to capture and display the metadata and configuration of properties for the nodes;
4. Provide a way to capture and display metadata relating to the workflow itself, including user-defined variables;
5. Provide a way to capture intermediate and results data from the workflow and create links to this data associated with the originating node within the visualisation;
6. Provide a way to specify where to store the files on the local machine;
7. Provide a way to include context information or background relating to the development of the workflow;
8. Not to include any third-party dependencies.

6 Dropbox: www.dropbox.com

As mentioned, key to understanding and reusability of workflows is the inclusion of adequate documentation that explains the process that the workflow represents and additional context. We also wanted to make use of both user-defined and machine-captured metadata about the workflow, data, nodes and settings to include in the visualisation, but also generate machine readable content that could be uploaded to an appropriate repository for curation purposes, together with the visualisation and selected data. KNIME has facilities that enable files to be zipped and also uploaded to a repository or other location if an appropriate Web service API or connectivity node is available for transferring the content. These facilities enable a user-friendly visualisation, machine-readable content, data and computation resources to all be packaged into a single bundle for sharing, publishing or storing.

Creating the Visualisation

The generated visualisation is a HTML file containing SVG content to describe the graphical elements of the workflow, such as the node types, connections between them, and textual elements. The HTML file also contains a number of JavaScript functions that handle displaying the dynamic data, such as tables for the node configuration information, a mouse-over on node output ports to show the data output information at that point in the workflow, and to remove and replace the dynamic elements depending on the actions of the user. We chose to implement the visualisation without the use of any supporting libraries to keep the implementation simple and reduce reliance on third-party resources. Our approach was to make use of a workflow within KNIME collapsed into a single metanode that could then be used within each workflow after the workflow has been developed and run.

In common with other Eclipse-based tools, KNIME stores configuration information for user-defined resources within a Workspace directory. Within this directory are sub-folders and xml file resources that describe the workflow as a whole, including the nodes present, their connections, and metadata such as author, timestamps, and user-defined variables. Within each sub-folder resources that describe the configuration and description of each node are defined in XML files. This information can be parsed within a workflow to build up the SVG representing the nodes, connections and annotations in the workflow. Figure 2 shows an example of a generated visualisation that contains metadata, imported design information, the workflow, and settings for a selected node.

The SVG for each node contains information about the node type, node icon, input and output port types, and the properties that are configured for each node, together with metadata about the node such as the developing organisation and version. The information about each specific node can be viewed when the node is selected.

Machine-generated provenance metadata, such as the creator of the workflow, the author of the workflow, and timestamps, are included in the header section of the workflow visualisation so that as much context information is captured and available as possible. Additional information, such as operating system version, could also be extracted and used in the visualisation in the future. KNIME also provides a number of ways that user-defined variables can be set for a workflow and if these are set for the workflow they are also included in the visualisation in this header section.

User-generated custom names for nodes and annotations are displayed within the SVG of the workflow, and the custom node descriptions are visible as part of the provenance and properties information for the node when the node is selected by the user. Ideally custom node descriptions are completed by the user as they develop the workflow and contain information that explains the choice of settings or source of input data. In order to provide an additional way of capturing context information or

background to the workflow we have added an option for the user to include a Word file in their workflow visualisations as discussed in the following section.

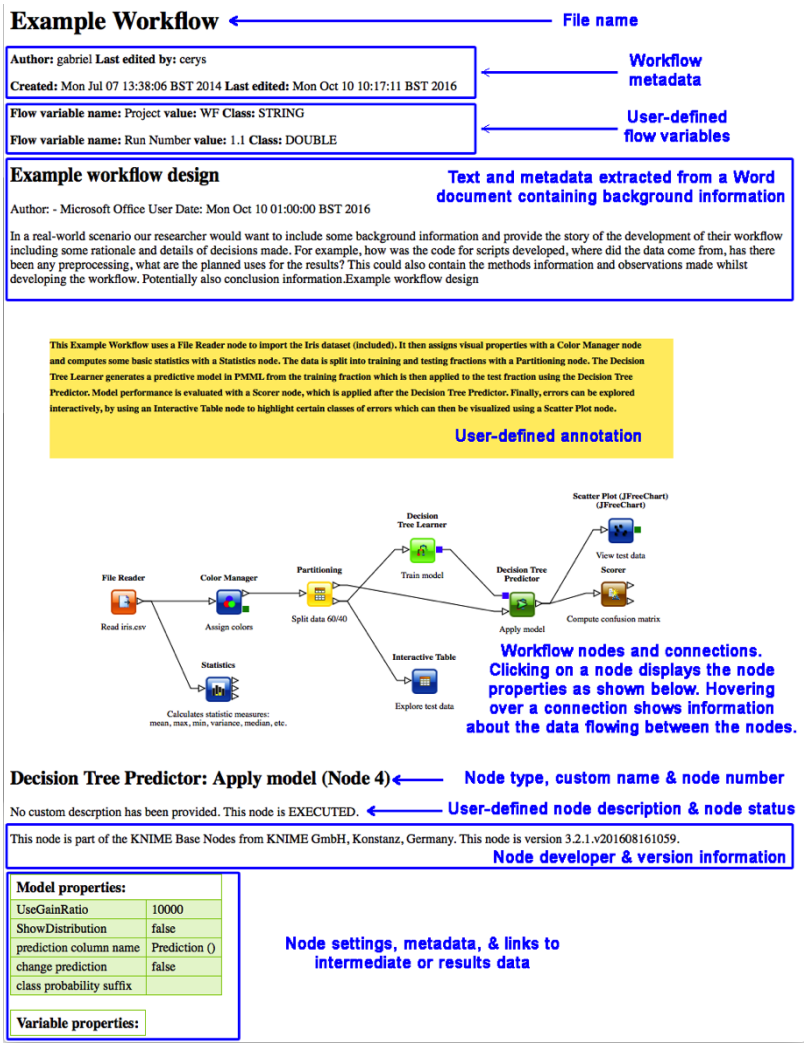

Figure 2. Example of the visualisation file showing metadata, imported design information, the workflow, and settings for a selected node.

Incorporating Conceptual Context

An important element of workflow context that we wanted to capture as part of the curation process is information about the design, development, and rationale of the workflow. Before the workflow is even created there is a 'composition' stage where decisions are made related to the hypothesis of the experiment, how and where to perform the experiment, and the technologies, resources and datasets to use (da Cruz et al., 2009), as shown in the Workflow Lifecycle in Figure 3. Whether the design and other planning elements need to be incorporated in the visualisation is dependant to some extent on where the visualisation is to be presented, for example one option for a repository to store the workflow and visualisation resources might be an Electronic Laboratory Notebook (ELN) where the notes for the design and development context for the workflow may be already be documented and so additional information may not be necessary. However, in many cases this information may be captured within a word-processed document or the sharing mechanism may not include any mechanism for

capturing this kind of documentation. We therefore wanted to provide an optional way to add context to the workflow visualisation by importing the contents of a text or Word file to be displayed in the HTML file above the visual representation of the workflow, as shown in Figure 2, enabling this valuable documentation to be readily available within the same file.

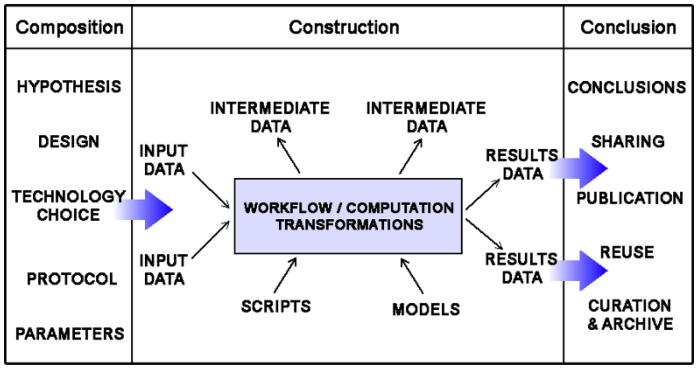

Figure 3. Workflow Lifecycle.

Workflows can also contain input forms to enable users to add parameters into the flow on both the desktop or online versions. Users can choose to edit the value of the node directly, but they are also exposed on workflows that have been converted to individual nodes (such as wrapped-nodes or meta-nodes). We have been able to use these fields for the node that generates the visualisation in order to set details such as the location store files, a string for a URI for the workflow, and the location of a Word file containing background information as described above.

Capturing and Linking Data

The majority of the time, data that is flowing through KNIME workflows is not stored on the local system, and where it is, it is not possible to access the content in a usable format to include in a workflow package or bundle. In order to capture this data so that input, intermediate and results data can be stored and included in the visualisation, it is necessary to output and write it as a specific file type. Although these file types could include formats such as CSV or a PMML model, for this proof of concept we have chosen to save table data as HTML files and images as PNG files, so that the format of the data and examples of the data produced can be read by a human user of the workflow. Unfortunately, for the proof of concept this requires that the user must explicitly select the node from which they want to capture data and attach a second 'Get Data' meta-node to extract data for inclusion. Ideally there would be an-built mechanism to do this for each output within each node or by selecting from a list of available – but such a mechanism does not yet exist within KNIME from within the workflow. Once the data is saved the visualisation flow is able to identify the data and create a hyperlink to the correct file within the properties of the node, so that the data can be viewed directly from the visualisation.

Limitations and Futures

There are a number of limitations with creating an interactive visualisation and a package for curating workflows within KNIME. The deliberate choice of using HTML, SVG and JavaScript without the use of any third-party libraries has made the creation of

 doi:10.2218/ijdc.v12i1.532

interactive elements, pretty-formatting, and browser compatibility non-trivial, although improvements due in SVG 2.0 and making use of third-party libraries will help to improve both aesthetics and behaviour of the display functions in the visualisation.

One of the more difficult aspects of visualising the workflow has been creating links between data produced by the workflow and the visualisation itself. This is partly because data flowing through the nodes and the identity of specific nodes are held as internal data in KNIME and therefore inaccessible within a flow. Although we have managed to find workarounds, these are not ideal and we have been in dialogue with KNIME in the hope that the issues can be more elegantly resolved.

A more robust solution would be the development of a stand-alone node, which could be submitted as a community extension to KNIME, or an Eclipse plugin to the menu using Java, to perform the parsing tasks and creation of the interactive workflow visualisation and workflow package. While we could extend our research to develop such a function, an ideal solution would be for KNIME to develop such a built-in tool to do the job of capture, visualise, and package an individual workflow run as a Research Object – creating both the human and machine-readable elements for storage and sharing outside of KNIME itself.

In a future incarnation of the project it would be possible to add elements that would match the recommendations for Research Object packages, for example:

- Capturing input data files;
- Capturing information about the run version without user intervention;
- Building up a structured description of the process steps by extracting the node descriptions and creating a narrative from the parameters for each node;
- Using ontologies, controlled vocabularies, and performing semantic analyses of the workflow text with user-specified dictionaries;
- Providing an aggregation file using identifiers with structured information about the resources included in the package and their relationships;
- Extracting individual scripts and models into files for back-up and viewing;
- Generating a provenance trace using a recognised standard, such as Open Provenance Model (OPM) or ProvONE[7] to enable effective curation (Cuevas-Vicenttín et al., 2014);
- Compressing and uploading the workflow package to a specified ELN, repository, or other location using available APIs.

Figure 4 reflects the Workflow Lifecycle from Figure 3 and how the user-defined information and resources, together with machine-generated metadata can be used to generate a workflow package containing not only data files and provenance information, but also human-readable resources. These human-readable resources can enable users to more easily understand and re-use workflows without needing to run them directly or for which they do not have access to the workflow software.

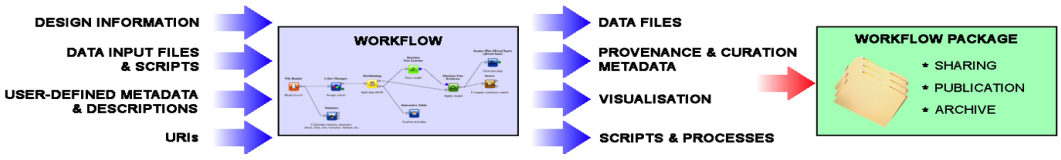

Figure 4. Generating a workflow package from workflow resources and curation metadata.

7 ProvONE: http://vcvcomputing.com/provone/provone.html

Visualisation and Research Object Creation on Other Workflow Platforms

There are numerous workflow tools available, each with slightly differing behaviours, facilities for adding documentation and metadata, and ways of exporting both data and information about the workflow. In VisTrails[8], for example, metadata can be added to the workflow, and nodes within the message flow also contain a variety of metadata and parameters. VisTrails also captures metadata about the history of changes to the workflow and provenance of the workflow execution. Graphical representations of the workflow, history, and provenance can be exported as PDF files. Output data can be manually exported for inclusion in a package. Another example, Kepler[9], enables the creation of annotations and documentation, in addition to metadata describing individual nodes in workflows. Data can be manually exported as text or images. Kepler enables users to export the workflow in Modeling Markup Language (MoML), XML, and as a static image. The user can also choose to export an interactive graphical representation that displays node parameters on mouse-over, similar to the representation we created in KNIME. The Kepler version does not display additional metadata or documentation about the workflow, or include any links or information about the data from the workflow.

It would be possible to create a visualisation and packaging tool that provide the kinds of functionality discussed for KNIME for each individual workflow platform. However, a better solution would be to encourage workflow developers to create tools that export workflow resources in standard formats including both machine and human-readable resources. Adoption of Research Object formats, and standards such as MoML, may encourage workflow developers to create functions within software that enable the automatic generation of a Research Object package containing computer and human readable representations of the workflow, including design documentation, data, and metadata to describe it.

Conclusions

We have demonstrated that it is possible to create a non-expert human-readable visualisation of a workflow that provides the benefits of the ‘flow chart style’ representation of workflows coupled with links to the data resources, user-authored design information, and with user-friendly metadata information about the workflow provenance, execution environment, and parameters. Our approach to curating workflows combines the visualisation with machine-readable provenance traces, computation resources, and data files that can all be stored and shared for the benefit of collaborators and the community.

The facilities provided by KNIME are particularly beneficial for the creation of supporting documentation for workflows, but we can only make use of this rich documentation potential within the visualisation if the information is provided in the first place. We know that it is difficult to encourage researchers to generate appropriate metadata and documentation for their data, but developing complex workflows in KNIME benefits from adding custom names to distinguish the purpose of individual nodes and adding annotations to help group particular processes or sections of the workflow. Our hope is that researchers can appreciate the value of documentation in KNIME for their own benefit; providing the ability to translate this effort into a meaningful resource that can be shared and utilised outside the software may help further encourage the generation of quality documentation and the use of meaningful

8 VisTrails: https://www.vistrails.org
9 Kepler: https://kepler-project.org

user-defined metadata. This documentation and metadata can enrich the workflow for communication, collaboration, publication, reuse, and encourage curation at source. Awareness that the work being done is going to be shared, published and reproduced can encourage better behaviour and can also benefit the author in the future (Donoho et al., 2009). Current initiatives to encourage the publication and citation of data and complete reporting of methodology such as recognition and 'badges' are helping to reinforce the values of transparency, openness, and reproducibility (Nosek et al., 2015). Perhaps these initiatives can be broadened to include generating 'user-friendly' resources for workflows including context, documentation, visualisations, and recognition for creating quality curation information.

Acknowledgements

This research was partially supported by the University of Southampton Institute for Life Science (IfLS) Research Stimulus fund, and the IT as a Utility Digital Economy Network (EPSRC EP/K003569) based on work undertaken as part of the EPRSC e-Science programme grants (EP/G026238/, GR/R67729/ and EP/C008863). We also thank Jon Fuller and Alexander Fillbrunn from KNIME for their advice.

References

Baggerly, K. (2010). Disclose all data in publications. *Nature, 467(7314)*, 401. doi:10.1038/467401b

Bechhofer, S., De Roure, D., Gamble, M., Goble, C., & Buchan, I. (2010). Research objects: Towards exchange and reuse of digital knowledge. *Nature Precedings*. doi:10.1038/npre.2010.4626.1

Bechhofer, S., Buchan, I., De Roure, D., Missier, P., Ainsworth, J., Bhagat, J., Couch, P., Cruickshank, D., Delderfield, M., Dunlop, I., Gamble, M., Michaelides, D., Owen, S., Newman, D., Sufi, S., & Goble, C. (2013). Why linked data is not enough for scientists. *Future Generation Computer Systems, 29(2)*, 599–611. doi:10.1016/j.future.2011.08.004

Belhajjame, K., Zhao, J., Garijo, D., Gamble, M., Hettne, K., Palma, R., Mina, E., Corcho, O., Gomex-Perez, J.M., Bechhofer, S., Klyne, G., & Goble, C. (2015). Using a suite of ontologies for preserving workflow-centric research objects. *Web Semantics: Science, Services and Agents on the World Wide Web, 32*, 16–42. doi:10.1016/j.websem.2015.01.003

Berthold, M.R. (2015). Reproducibility and KNIME. Retrieved from https://www.KNIME.org/blog/reproducibility-and-KNIME

Berthold, M.R., Cebron, N., Dill, F., Gabriel, T.R., Kötter, T., Meinl, T., Ohl, P., Thiel, K., & Wiswedel, B. (2008). KNIME: The Konstanz Information Miner. In C. Preisach, H. Burkhardt, L. Schmidt-Thieme, & R. Decker (Eds.), Data Analysis, Machine Learning and Applications, pp. 319–326. Springer Berlin Heidelberg. doi:10.1007/978-3-540-78246-9_38

Bird, C.L., Willoughby, C., Coles, S.J., & Frey, J.G. (2013a). Data curation issues in the chemical sciences. *Information Standards Quarterly, 25(3)*, 4–12. doi:10.3789/isqv25no3.2013.02

Bird, C.L., Willoughby, C., & Frey, J.G. (2013b). Laboratory notebooks in the digital era: The role of ELNs in record keeping for chemistry and other sciences. *Chemical Society Reviews, 42(20)*, 8157–75. doi:10.1039/c3cs60122f

Bodkin, M.J. (2012). Why don't we see a greater uptake of computational chemistry approaches by the medicinal chemistry community? *Future Medicinal Chemistry, 4(15)*, 1889-1891. doi:10.4155/fmc.12.154

Cuevas-Vicenttín, V., Kianmajd, P., Ludäscher, B., Missier, P., Chirigati, F., Wei, Y., Koop, D., & Dey, S. (2014). The PBase scientific workflow provenance repository. *International Journal of Digital Curation, 9(2)*, 28-38. doi:10.2218/ijdc.v9i2.332

da Cruz, S.M., Campos, M.L., & Mattoso, M. (2009). Towards a taxonomy of provenance in scientific workflow management systems. In Congress on Services - I, Los Angeles, CA, 2009, pp. 259-266. doi:10.1109/SERVICES-I.2009.18

Davidson, S.B. & Freire, J. (2008). Provenance and scientific workflows: challenges and opportunities. In Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data (pp. 1345-1350). ACM.

De Roure, D. (2009). Research objects. Retrieved from http://wiki.myexperiment.org/index.php/Research_Objects

Desaulniers, D.R., Gillan, D.J., & Rudisill, M. (1988). The effects of format in computer-based procedure displays. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 32(5)*, 291-295.

Donoho, D.L., Maleki, A., Rahman, I.U., Shahram, M. & Stodden, V. (2009). Reproducible research in computational harmonic analysis. *Computing in Science & Engineering, 11*(1), 8-18.

Dudley, J.T., & Butte, A.J. (2010). In silico research in the era of cloud computing. *Nature Biotechnology, 28(11)*, 1181–1185. doi:10.1038/nbt1110-1181

Fuller, J. (2015). KNIME in the data-driven lab of the future! Retrieved from https://www.KNIME.org/blog/KNIME-in-the-data-driven-lab-of-the-future

Garijo D., Kinnings, S., Xie, L., Zhang, Y., Bourne, P.E., & Gil, Y. (2013). Quantifying reproducibility in computational biology: The case of the Tuberculosis Drugome. *PLoS ONE 8(11)*:e80278. doi:10.1371/journal.pone.0080278

Gentleman, R. & Lang, D.T. (2007). Statistical analyses and reproducible research. *Journal Of Computational And Graphical Statistics, 16(1)*, 1-23. doi:10.1198/106186007X178663

Goodman A., Pepe A., Blocker A.W., Borgman C.L., Cranmer K., Crosas M., Di Stefano, R., Gil, Y., Groth, P., Hedstrom, M., Hogg, D.W., Kashyap, V., Mahabal, A., Siemiginowska, A., & Slavkovic, A. (2014). Ten simple rules for the care and feeding of scientific data. *PLoS Computational Biology 10(4)*: e1003542. doi:10.1371/journal.pcbi.1003542

Gorp, P. Van, & Mazanek, S. (2011). SHARE : A web portal for creating and sharing executable research papers. In M. Sato, S. Matsuoka, P.M. Sloot, G.D. van Albada & J. Dongarra (Eds.), *Proceedings of the International Conference on Computational Science, ICCS 2011, 4*, (pp. 589–597). doi:10.1016/j.procs.2011.04.062

Hasham, K., Munir, K., & McClatchey, R. (2015). Using cloud-aware provenance to reproduce scientific workflow execution on cloud. In M. Helfert, V. Méndez Muñoz, & D. Ferguson (Eds.), *Proceedings of the 5th International Conference on Cloud Computing and Services Science* (pp. 49–59). doi:10.5220/0005452800490059

Hutton, C., Wagener, T., Freer, J., Han, D., Duffy, C. & Arheimer, B. (2016). Most computational hydrology is not reproducible, so is it really science? *Water Resources Research*. Accepted Author Manuscript. doi:10.1002/2016WR019285

Iqbal, S.A., Wallach, J.D., Khoury, M.J., Schully, S.D., John, P., & Ioannidis, A. (2016). Reproducible research practices and transparency across the biomedical literature. *PLoS Biology, 14(1)*, 1–13. doi:10.1371/journal.pbio.1002333

Ko, R.K.L., Lee, S.S.G., & Lee, E.W. (2009). Business process management (BPM) standards: A survey. *Business Process Management Journal, 15(5)*, 744 - 791. doi:10.1108/14637150910987937

McPhillips, T., Song, T., Kolisnik, T., Aulenbach, S., Belhajjame, K., Bocinsky, R. K., Cao, Y., Cheney, J., Chirigati, F., Dey, S., Freire, J., Jones, C., Hanken, J., Kintigh, K.W., Kohler, T.A., Koop, D., Macklin, J.A., Missier, P., Schildhauer, M., Schwalm, C., Wei, Y., Bieda, M., & Ludäscher B. (2015). YesWorkflow: A user-oriented, language-independent tool for recovering workflow information from scripts. *International Journal of Digital Curation, 10*(1), 298-313. doi:10.2218/ijdc.v10i1.370

Missier, P., Ludäscher, B., Dey, S., Wang, M., McPhillips, T., Bowers, S., Agun, M., Altintas, I. (2012). Golden trail: Retrieving the data history that matters from a comprehensive provenance repository. *International Journal of Digital Curation, 7(1)*, 139-150. doi:10.2218/ijdc.v7i1.221

Nosek, B.A., Alter, G., Banks, G.C., Borsboom, D., Bowman, S.D., Breckler, S.J., Buck, S., Chambers, C.D., Chin, G., Christensen, G., & Contestabile, M. (2015). Promoting an open research culture. *Science, 348(6242)*, 1422-1425.

Pimentel, J.P., Dey, S., McPhillips, T., Belhajjame, K., Koop, D., Murta, L., Braganholo, V., & Ludäscher, B. (2016). Yin and yang: Demonstrating complementary provenance from noWorkflow and YesWorkflow. In M. Mattoso & B. Glavic (Eds.), *Provenance and Annotation of Data and Processes*, pp 161-165. Springer International Publishing. doi:10.1007/978-3-319-40593-3_13

Richardson, D.P., & Moreau, L. (2016). Towards the domain agnostic generation of natural language explanations from provenance graphs for casual users. In M. Mattoso and B. Glavic (Eds.), *Provenance and Annotation of Data and Processes*, pp. 95-106. Springer International Publishing. doi:10.1007/978-3-319-40593-3_8

Scanlan, D.A. (1989). Structured flowcharts outperform pseudocode: an experimental comparison. *IEEE Software, 6(5)*, 28-36. doi:10.1109/52.35587

Ungan, M. (2006). Towards a better understanding of process documentation. *The TQM Magazine, 18(4)*, 400–409. doi:10.1108/09544780610671066

Wibisono, A., Bloem, P., de Vries, G.K., Groth, P., Belloum, A., & Bubak, M. (2015). Generating scientific documentation for computational experiments using provenance. *Provenance and Annotation of Data and Processes*, pp. 168-179. Springer International Publishing. doi:10.1007/978-3-319-16462-5_13

Zhao, J., Gomez-perez, J. M., Belhajjame, K., Klyne, G., & Garcia-Cuesta, E. (2012). Why workflows break – Understanding and combating decay in Taverna workflows. In IEEE 8th International Conference on E-Science (pp. 1-9).