PROV-O-Viz - Understanding the Role of Activities in Provenance

Rinke Hoekstra[1,2] and Paul Groth[1]

[1] Network Institute,
VU University Amsterdam
rinke.hoekstra@vu.nl, p.t.groth@vu.nl
[2] Faculty of Law
University of Amsterdam
hoekstra@uva.nl

Abstract. This paper presents PROV-O-Viz, a Web-based visualization tool for PROV-based provenance traces coming from various sources, that leverages Sankey Diagrams to reflect the flow of information through activities. We briefly discuss the advantages of this approach compared to other provenance visualization tools. PROV-O-Viz has already been used to visualize provenance traces generated by very different applications.

Keywords: provenance, visualization, Sankey, information flow, linked data, reusability

1 Introduction

Understanding data provenance (the origin or source of data) is a critical facilitator for data quality, trust, reproducibility, compliance and debugging of complex computational systems [FBS12]. In 2013, the World Wide Web consortium released the W3C PROV standards that enable the interchange of provenance between systems [GM13]. These standards are becoming increasingly implemented [HGZ13].

Given the wealth of provenance information available, techniques are needed to help users navigate and investigate this information space. Several works have focused on the visualization or provenance using a number of presentation paradigms including networks, data flow graphs, and radial layouts [BYB+13], [MPH09], https://provenance.ecs.soton.ac.uk/vis/.

Here, we focus on a visualization approach to identify important activities within a provenance graph and link those activities together. Additionally, our aim is to show how this approach can be useful in an uncontrolled setting, i.e. for PROV coming from multiple environments, generated through the execution of diverse and potentially undefined tasks or workflows. To do so, we demonstrate a Sankey Diagram based visualization of PROV and apply that visualization to multiple provenance traces originating from multiple environments, machine learning experiments, version control systems (GitHub), and scientific workflows originating from different workflow systems. The demonstration is available at http://provoviz.org.

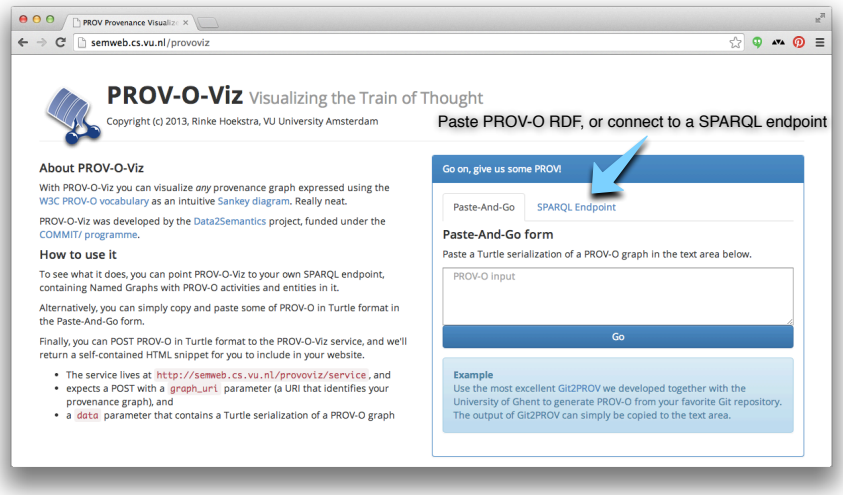

Fig. 1. Add PROV-O by pasting text, or by connecting to a SPARQL endpoint.

2 Sankey Diagrams

Our approach adopts Sankey Diagrams, which visualize the magnitude of flow within in a network. Sankey diagrams are particularly helpful in locating choke points or other places that aggregate flow. Specifically, we view a provenance graph as a network of activities where data flows through and between activities. Our aim then is to provide a view that allows us to:

1. determine important activities based on data flow; and
2. understand how data flows through a selected activity.

In a standard, directed acyclic graph (DAG) rendering, this flow gets easily lost in a large network. Other layouts, for example radial layouts, focus on the interconnectivity of data or activities. Furthermore, other layout approaches do not leverage the temporal ordering inherent in provenance graphs.

3 PROV-O-Viz

PROV-O-Viz is a web-based PROV visualization tool that leverages Sankey Diagrams and adds a number of provenance specific features. PROV-O-Viz uses the PROV-O RDF serialization of PROV. Figures 1 and 2 show a screenshots of PROV-O-Viz where we highlight these features:

1. Import of PROV data from both plain text and published data (i.e. available at a URL)

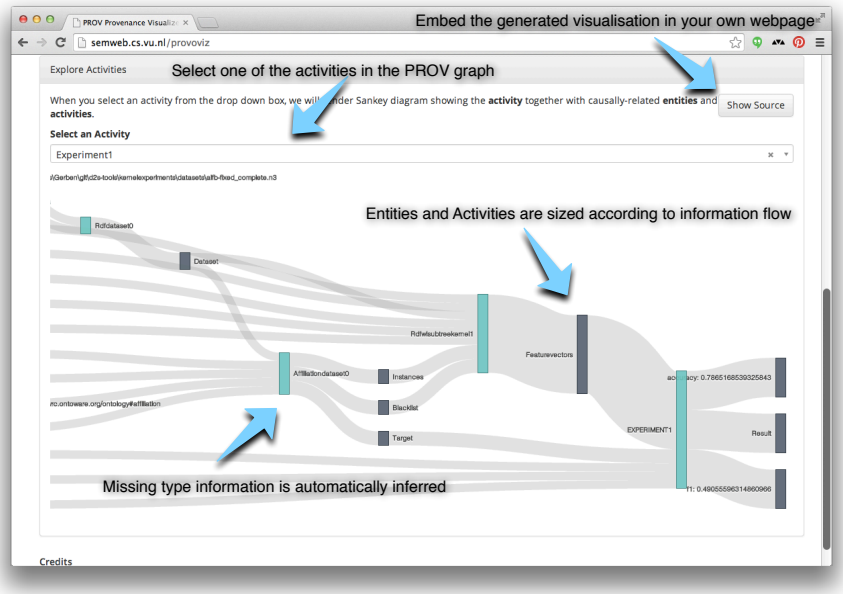

Fig. 2. Visualization of a provenance trace generated by Ducktape.

2. Focus on particular activities within a provenance diagram, by selecting them from a dropdown box.
3. Highlight data flows in and out of activities within the diagram,;the width of the box indicates the amount of information flowing through the activity.
4. Leverage reasoning to fill out missing information within a provenance graph.

Additionally, we allow provenance graphs to be embedded directly within web pages. This allows provenance visualizations to be included directly with other web applications. Furthermore, this visualization is self contained. Once the provenance is rendered there is no need to call to the server. For example, in LinkItUp ([HG13], `http://linkitup.data2semantics.org`), an application to enrich the content of data with metadata, PROV-O-Viz is used to display the provenance of how the application enriches data with this extra data. Thus, users understand how the application makes its suggestions. (We will also demonstrate this capability.)

3.1 Evaluation

We evaluated the visualization capabilities of PROV-O-Viz by using it to inspect PROV data coming from four different sources. First of all, the provenance traces of scientific workflows executed through the Taverna and WINGS workflow sys-

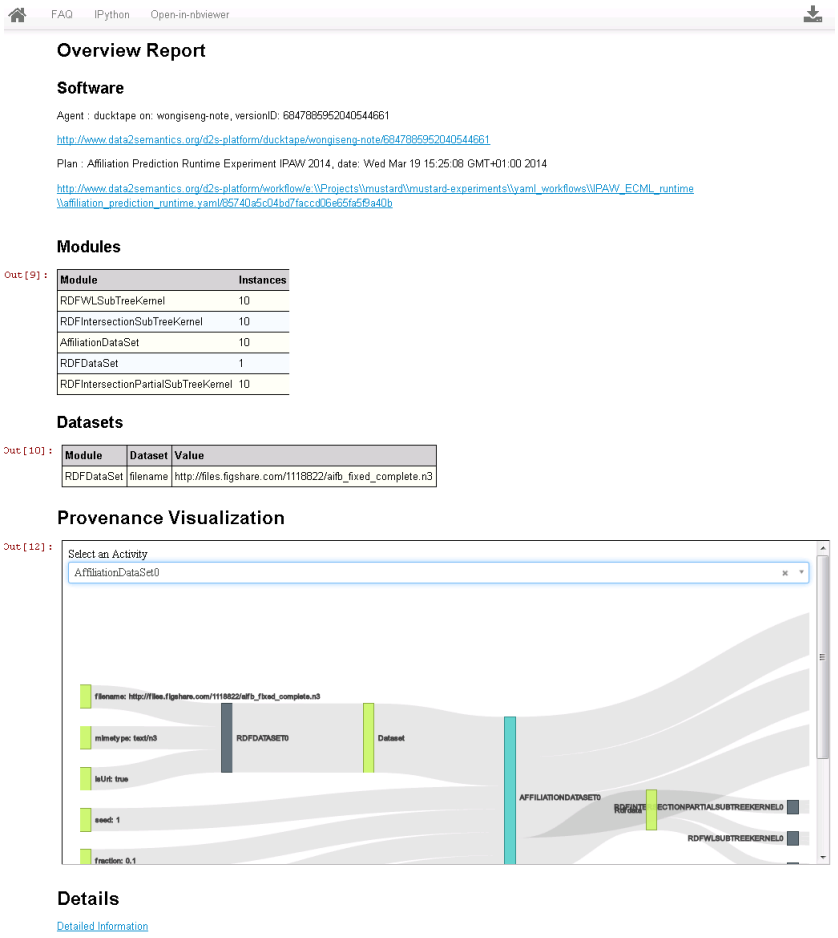

Fig. 3. Overview report of a runtime experiment, generated by Ducktape [WBdV+14].

tems, that are made available as part of the Wf4Ever ProvBench benchmark.[3] The Taverna PROV traces do not explicitly provide the *type* of events and activities that many visualizations rely on. PROV-O-Viz automatically infers these types by applying reasoning over the PROV-O schema definitions. Even though some of these datasets are relatively large, focusing on the ego graph of information dependencies flowing through the selected activity allows the visualization to remain manageable. At the moment, however, PROV-O-Viz generates a visualization for the ego graph centered around every activity. This means that for provenance traces that contain very many connected activities, the process of generating the Sankey diagram may take a long time. After the diagrams have

[3] See https://github.com/provbench/Wf4Ever-PROV/.

been built, the visualization will be very responsive. Embedded PROV-O-Viz diagrams are already generated, and therefore do not suffer from this potential performance hit. The next version will feature a more responsive user interface, that keeps users up-to-date as to the progress made in generating the visualizations.

The Ducktape platform[4] is another such scientific workflow system that is focused on Machine Learning tasks. The visualization in Figure 2 is based on the provenance of one of the steps in a Machine Learning pipeline. Ducktape can generate interactive reports of workflow execution that embeds a visualization of its provenance trace [WBdV+14]. See Figure 3 for a screenshot of such a report.

The LinkItUp system for enriching metadata for datasets stored in the Figshare.com scientific data publishing platform, stores all enrichment activities performed by users as part of a provenance trace. This provenance trace can be inspected from within the application through a call to the PROV-O-Viz API.

Git2PROV is a web service that can convert Git version histories to a provenance trace expressed in various PROV compliant syntaxes.[5] Every commit is represented as a PROV activity. Visualizing these graphs can be even more challenging than those of the workflow systems because version commit histories are tree-shaped, and highly connected: they all originate from the same initial commit. Workflow systems can produce large graphs, but oftentimes these are in fact multiple separate graphs for runs against multiple files.

4 Conclusion

In this demonstration, we show how generic visualization tools can be used to interrogate provenance coming from multiple different applications. This provides evidence that provenance can provide added value without domain specific extensions. In future work we will focus on the ability to generate entity-centric diagrams, a browsing feature, allowing users to click through the various parts of the provenance graph. We are furthermore considering the implementation a more efficient method for calculating the information flow, e.g. based on centrality measures based on current flow in an electrical network [BF05].

Acknowledgements

This work was funded under the Dutch national programme COMMIT.

References

BF05. Ulrik Brandes and Daniel Fleischer. Centrality measures based on current flow. In *Proceedings of STACS 2005*, volume 3404 of *Lecture Notes in Computer Science*, pages 533–544. Springer, 2005.

[4] See https://github.com/Data2Semantics/ducktape
[5] See http://git2prov.org.

BYB+13. Michelle A. Borkin, Chelsea S. Yeh, Madelaine Boyd, Peter Macko, Krzysztof Z. Gajos, Margo Seltzer, and Hanspeter Pfister. Evaluation of filesystem provenance visualization tools. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2476–2485, 2013.

FBS12. Juliana Freire, Philippe Bonnet, and Dennis Shasha. Computational reproducibility: State-of-the-art, challenges, and database research opportunities. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, SIGMOD '12, pages 593–596, New York, NY, USA, 2012. ACM.

GM13. Paul Groth and Luc Moreau. PROV overview: An overview of the prov family of documents. Technical report, W3C, 2013.

HG13. Rinke Hoekstra and Paul Groth. Linkitup: Link discovery for research data. In *Discovery Informatics: AI Takes a Science-Centered View on Big Data, AAAI Fall Symposium Series*, 2013.

HGZ13. Trung Dong Huynh, Paul Groth, and Stephan Zednik. Prov implementation report. Technical report, W3C, 2013.

MPH09. Björn Meyer, Steffen Prohaska, and Hans-Christian Hege. Provenance visualization and usage. Technical report, 2009.

WBdV+14. Adianto Wibisono, Peter Bloem, Gerben K.D. de Vries, Paul Groth, Adam Belloum, and Marian Bubak. Generating scientific documentation for computational experiments using provenance. In *Proceedings of IPAW 2014*, 2014.