# Towards Unified Provenance Granularities

Timothy Lebo, Ping Wang, Alvaro Graves, and Deborah McGuinness

Tetherless World Constellation
Rensselaer Polytechnic Institute
Troy, NY, USA
lebot@rpi.edu,wangp5@cs.rpi.edu,gravea3@rpi.edu,dlm@cs.rpi.edu
http://tw.rpi.edu

**Abstract.** As Open Data becomes commonplace, methods are needed to integrate disparate data from a variety of sources. Although Linked Data design has promise for integrating world wide data, integrators often struggle to provide appropriate transparency for their sources and transformations. Without this transparency, cautious consumers are unlikely to find enough information to allow them to trust third party content. While capturing provenance in RPI's Linking Open Government Data project, we were faced with the common problem that only a portion of provenance that is *captured* is effectively *used.* Using our water quality portal's use case as an example, we argue that one key to enabling provenance use is a better treatment of provenance granularity. To address this challenge, we have designed an approach that supports deriving abstracted provenance from granular provenance in an open environment. We describe the approach, show how it addresses the naturally occurring unmet provenance needs in a family of applications, and describe how the approach addresses similar problems in open provenance and open data environments.

**Keywords:** Data Integration, Transparency, Provenance Granularity, Derived Abstractions, Provenance of Provenance, Linked Data

## 1 Introduction

Open Data is growing in popularity and is freely available for anyone to use and republish as they wish, with few or no restrictions from copyright, patents or other mechanisms of control. Open Government Data (OGD) is one rapidly growing portion of Open Data. Catalyzed in 2009 by the United States and the United Kingdom, governments from local to national levels are publishing their data for public use [14, 5]. These data are available for personal or commercial use and offer the potential to increase government transparency and accountability and create many opportunities for businesses and communities. These data have the potential to help citizens understand important topics such as pollutants near their home [18], crimes in their neighborhood [8], public works[1], the economy [3], natural disasters[2] [9], and political activities [14].

[1] https://recollect.net

[2] http://purl.org/twc/lebo/ipaw/2012/od-natural-disasters

Although individual datasets may be interesting on their own, there is a hope and expectation that combining disparate datasets will lead to even more insight and value – the whole should be greater than the sum of its parts. Linked Open Data is becoming a popular method to connect and publish data on the web [10]. One highly cited view[3] has grown from twelve to 295 datasets between 2007 and 2011. Each of those 295 datasets ranges in size and comprises many more subsets of data. For example, the TWC-LOGD dataset[4] that our group publishes contains almost 10 billion RDF triples created from thousands of datasets. In addition, we have cataloged[5] more than 710,000 other datasets that can be added. The Linked Open Data cloud is continuing to grow and already provides information about a range of topics including Life Sciences, Government, Scholarly Publications, Social Media, and E-Commerce.

Unfortunately, current approaches for creating Linked Data present both implicit and explicit challenges around trust of the Linked Data itself. Because many primary data sources do not publish their material as Linked Data, third parties are left to independently transform and republish it. As illustrated in Figure 1, this forces application developers to choose between two sources of the same content. Although the first option is provided by an authoritative and recognizable source (usually with deep domain knowledge), this data is often not uniformly accessible and not linked to other data. Meanwhile, the second option is uniformly accessible and linked to other datasets, but is not provided by an authoritative source. These third party sources are often experts in technology, but not the particular subject matter. When consumers require more than a vague citation for a transformed dataset, the benefits of Linked Data cannot outweigh the potential risks introduced by a non-authoritative and non-transparent third party.

One obvious approach that third party aggregators can take is to provide transparency for the transformations that they perform as well as the sources used. Application developers would then be able to choose Linked Data instead of the primary source because its lineage is available for inspection. To demonstrate this kind of transparency, RPI's Linking Open Government Data project used the csv2rdf4lod conversion toolset [17] to capture provenance at each stage of Linked Data production. But after 18 months of capture, only a fraction of it has been used in applications.

The pitfall of capturing more provenance than is used is not new. As Chapman warns, *Don't just maintain provenance, maintain good provenance* [2]. But as closed provenance systems become open, homogeneous systems become heterogeneous, and local coordinations become distributed, how does one know what good provenance is *a priori*? Similarly, are there different notions of good provenance in different contexts? With these opening trends, less control of overall systems, and less knowledge of expected usage contexts, the problem of defining and maintaining good provenance becomes more challenging. Indeed, Linked

[3] http://richard.cyganiak.de/2007/10/lod/

[4] http://logd.tw.rpi.edu/twc-logd

[5] http://purl.org/twc/links/iogds

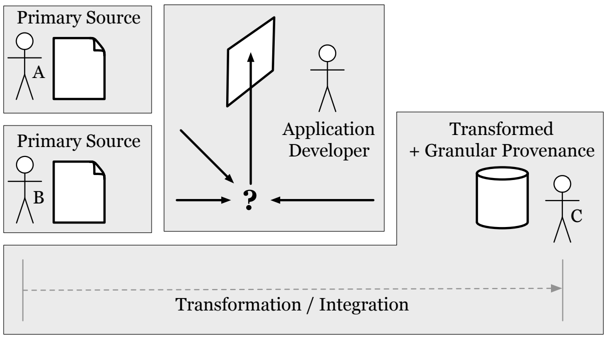

**Fig. 1.** Linked Data is often produced by third parties that transform data from various sources, which introduces a tradeoff between authoritative content and easier to use linked data. Although provenance enables transparency, excessive granularity may inhibit its use.

Open Data is an open system not only in the Open Data sense, but also in the sense of Moreau's [13] Open Provenance Vision. In the general conversion scenario that we describe above, *many* publishers offer primary data, *many* aggregators convert primary data to Linked Data, *many* developers choose among the data for their applications, and *many* audiences use them. All of these activities are performed across the world with loose, if any, coordination.

As systems become more open and information flows across multiple systems, we need reliable strategies for handling the disparity between what application developers need and what they get. We believe that framing these strategies around provenance granularity promises to address these growing challenges. In this paper, we describe a method to resolve *incongruent provenance granularities* using an open system design.

## 2 Related Work

Granularity is a widely studied problem in provenance. Gibson et al. [7] point out that the clutter of provenance capture obfuscates the conceptual view of processes. This observation parallels the **provenance granularity disparity** that our approach offers to address: too many details when fewer statements would adequately answer the question at hand. Gibson et al. show how user-defined views along with a high level summary of execution history can improve user understanding of provenance. ZOOM [1] focuses on user needs by offering provenance of customized granularity to achieve benefits like abstraction, privacy, and reuse between workflows. We expand on these ideas by showing how summarizations can be derived by third party consumers in an open environment.

Chapman and Jagadish [2] point out that provenance support needs more than a simple capture-store approach, which is a challenge we address. They also note that a choice in granularity is required and distinguish between a coarse-grained file level and a fine grain attribute level. However, the situation we present here motivates a distinction between coarse and fine granularities about the file level itself. Finally, while they note the importance of enabling *users* to actively view provenance at multiple levels, the approach we present resolves incongruent granularities between *systems* themselves. Ikeda and Widom [11] state that while existing work on provenance primarily focuses on modeling and capturing, there has been inadequate support for querying and using provenance. They also propose that provenance be captured at a variety of granularities. Alternatively, the approach we present here facilitates query and use by providing a mechanism to derive simpler, more abstract provenance that is more suitable for particular, possibly unanticipated uses.

Different techniques have been used to model provenance granularities. Stephan et al. [16] presents a multi-tier provenance model in which each tier has a unique purpose, different characteristics, and distinct levels of granularity. They use the Open Provenance Model (OPM) to encode their provenance and disseminate higher level provenance that are abstracted from provenance captured in different tiers, e.g. instruments used, parameters used, and quality/confidence level, to produce Value Added Products (VAPs). Ding et al.[4] propose *RDF molecules* as a way to handle granularities between a single triple and an entire graph. RDF molecules are generated by decomposing a graph into separate sub-graphs. Although this technique can be used to track the movement of RDF subgraphs across systems, it fails to apply when the graph is abstracted to new forms.

Other granularity techniques are oriented towards the end user or domain expert. WDo [15] is a framework for provenance granularity where domain experts use a graphical interface to specify process composition. Methods are treated as black boxes at one general level and further described at more specific levels in terms of how they transform information types. The results are described using an OWL ontology that extends the Proof Markup Language (PML) ontology [12]. While this approach is helpful to elicit appropriate abstractions from experts, its information types do not allow one to specify the detailed structures that are required for application consumption. Garijo [6] does something similar, but has the same publisher bias for abstraction instead of enabling third parties to derive their own abstractions for their own purposes.

## 3 csv2rdf4lod's Assertions of Granular Provenance

RPI's Linking Open Government Data (LOGD) project began collecting provenance on June 25th, 2010 using a strategy to encode provenance that might be useful to our anticipated applications. One persisting purpose is to enable transparency for third party transformations when creating well structured and highly connected Linked Data from various disparate sources. To date, the conversion automation has recorded more than a half million instances of the major PML classes (Information, SourceUsage, NodeSet, and InferenceStep) and used more than 200 InferenceEngines. We continue to reflect on what is there, how we are using it, and how we can get more value from it. This section provides an overview of the kinds of provenance captured and highlights some patterns that have worked well throughout the project's development. The granular, context-free provenance that we describe here will contrast with the abstract, user-driven provenance that we describe in the following section.

The Linked Data creation process has four principal stages: retrieval, preparation, conversion, and publishing.[6] While below we list the types of provenance captured in each, different themes emerge between stages. In the **retrieval** stage, it is paramount to distinguish between the materials obtained from the *source* and those that the third party integrator has *derived* from them. In the **preparation** stage, it is important to maintain a distinction between results produced automatically and results produced manually. Finally, in the **conversion** and **publication** stages, it is useful to maintain a distinction between data results and their provenance, which may change even when the results are identical.

During retrieval, data files are obtained from their primary source. A script that retrieves a given URL also records the person and user account initiating retrieval, the URL requested, time requested, HTTP interactions, and the checksum of the file received. This is perhaps the most critical capture because it maintains the connection between the local file on disk and the original URL.

During preparation, manually modifying files retrieved from authoritative sources is avoided because it is error prone and cannot be reliably repeated. Custom software and manual adjustments are minimized by specifying declarative conversion parameters to a common converter. However, human intervention may be necessary in some situations. Transparency of any necessary manual activity is maintained by storing results separately from their originals, associating the adjusted files to their predecessors, indicating the type of process applied, and citing the person and user account reporting the modifications.

Conversion and publishing are automated and is started by software and human agents. Each activity's inputs, parameters, and outputs are published at URLs and are commonly referenced by each actor's provenance assertions. Tabular data files are converted to RDF by csv2rdf4lod, which records its invocation time, version and hash, input file, transformation parameters, the person and user account invoking the conversion, and the generated dump files. Because metadata typically mentions time, it changes more regularly than the generated data and is thus stored in separate files. This way, hashes of unchanging data files can persist through reconversions. Finally, when RDF URLs are loaded into a triple store's named graph, provenance of the activity is stored in the same.

## 4 SemantAqua's Need for Abstracted Provenance

SemantAqua is one application that uses the provenance captured by csv2rdf4lod during the stages of Linked Data integration. SemantAqua is a water quality web portal[7] that demonstrates a semantic approach to environmental monitoring [18]. It integrates water test results from different government sources and allows

[6] http://purl.org/twc/links/ipaw/2012/conversion-stages highlights the principal provenance captured for an example dataset.

[7] http://tw.rpi.edu/web/project/SemantAQUA

users to explore results on a map, see their severity, and hypothetically apply different regulations from different political jurisdictions. One could, for example, classify water tests taken in a particular state against local state regulations, federal regulations, or regulations in states that are known to have stricter rules. SemantAqua introduces a provenance-based search facet that allows the user to select the data organizations he/she trusts, so that the portal will use only data from the selected organizations. This is done by restricting queries to only named graphs that are known to come from the selected organizations.

To achieve this functionality, SemantAqua needs to know the organizations that are attributed to each named graph. The project considered three different strategies to address this need. First, SemantAqua could depend on the attribution made by the data integrator, which is done automatically by csv2rdf4lod using the `source` identifier chosen by the curator. This assertion, however, may not be completely accurate and more cautious consumers may demand more detailed justification. For example, data-gov is commonly cited as a source, when the data is actually provided by specific agencies such as epa-gov or usgs-gov. Second, the application developer could manually maintain the list of attributions. This approach is undesirable because it requires additional effort and cannot be reapplied in other applications. The third approach is to use an automated abstraction of the granular provenance captured at each stage of the data integration process. Although this offers the most accuracy and justification for the attributions, it is not straightforward from the application's perspective to determine the connection from the named graph to the organization. Further, adequate support for this third option was not available prior to this work. To determine the attribution, software would trace the provenance of the named graph load, the conversion invocation, any and all preparations performed, and the retrieval of the original data files provided by the primary source.

Because the first option did not meet application requirements and the third option was not supported at the time, SemantAqua's initial prototype constructed and maintained a separate graph to provide the abstract provenance required to support the data source search facet. This custom work took developer resources away from other portal features and the intermediate solution is difficult to reuse. A more desirable solution is to build on a reusable framework that supports abstracted provenance, which we describe next.

## 5 Deriving Abstractive Provenance

The prior sections describe two example systems that participate in an Open Provenance Environment. The disparity between application needs and linked data aggregator services provides one example of incongruent granularity issues that we anticipate to grow as more systems realize the Open Provenance Vision.

Our strategy is to resolve incongruent provenance granularities that occur between two systems in an open environment by adding a third, independent component into the same environment. Figure 2 depicts an independent party D creating a service that abstracts the original provenance in a way that the application can use with relative ease. The service is available for invocation by any system and can be called dynamically or accumulated for local use. We adopt the SADI Semantic Web Services Framework [19] for the design of the services and apply the DataFAQs linked data evaluation framework[8] to accumulate results for specific portions of provenance while capturing the provenance of accumulation. This approach can be applied to resolve incongruences between any systems that expose their provenance in any RDF vocabulary, including the PROV-O vocabulary in development by the W3C Provenance Working Group. The steps to our approach are as follows.

1. Define the type of entity whose provenance is required by the consumer
2. Define the type of provenance required by the consumer
3. Implement and deploy the independent service
4. Optionally find the service based on Steps 1 and 2
5. Accumulate results for the entities of interest, capturing provenance

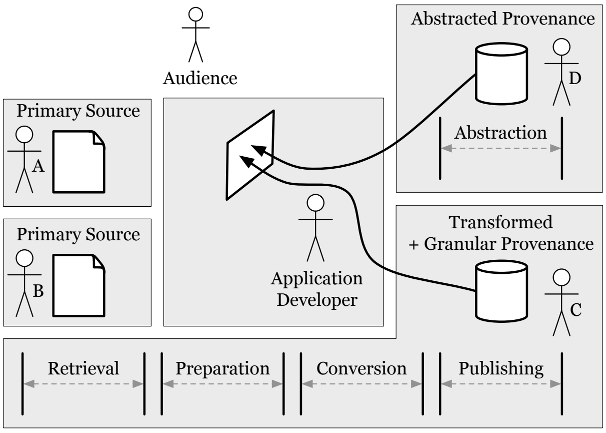

**Fig. 2.** In addition to Figure 1's situation, party D derives abstracted provenance from party C, which the application developer uses to determine which data to use from C.

The first step is to define the type of entity whose provenance is required by the consumer. In our example from the previous sections, the entity type is a named graph in a particular SPARQL endpoint. The type of entity whose provenance is needed will be the topic of a consuming application; whatever an application or system "discusses" is a potential type of entity about which we may want provenance. The second step is to define the type of provenance required by the consumer. In our example, the type of provenance we need for the named graph is the source organization(s) that should be attributed for providing its contents. The third step is to implement and deploy the independent service.

[8] http://purl.org/twc/id/software/datafaqs

Because the entity type from Step 1 and the provenance type of Step 2 can be described in RDF using any vocabulary, we use the SADI framework to implement the service. SADI services accept HTTP POSTs of RDF descriptions and return additional RDF descriptions of the same instances. In our example, the service `named-graph-derivation` accepts RDF descriptions of sd:NamedGraph[9] and returns additional descriptions using the prov:wasAttributedTo relation. The fourth step to find the service based on Steps 1 and 2 is necessary in cases where the consumer is not aware of the service. Use of the SADI framework facilitates this search because SADI services use the myGrid vocabulary to specify their input and outputs as OWL classes. The final step is to accumulate results for the entities of interest, capturing provenance of the accumulation. In our example, we create RDF descriptions of sd:NamedGraphs, HTTP POST them to the service `named-graph-derivation`, and store their results in a triple store for query by applications. Although this accumulation can be performed in a variety of ways, we use the DataFAQs linked data evaluation framework because it records the provenance of each service invocation and automatically publishes results.

Applying the five steps above creates an independent collection of abstracted provenance that is available to the application and other systems. Further, the provenance collection can be traced to the independent service regardless of where it has been accumulated. This provenance of provenance enables justifications for any of the abstract claims. Further, this also means that provenance is a first class object that can have its own provenance and has no limitations on the way it can be composed in complex applications.

To illustrate how the five steps can be applied, we show some materials used to solve our running named graph attribution example. To illustrate the longest derivation chain of the conversion process, we use an example dataset that begins as a compressed Excel file that is extracted and converted to CSV before becoming Linked Data. In the first step, we describe the named graph whose attribution we want, which includes the SPARQL endpoint's URL and the name of the graph. In the second step, we define the provenance needed by the application, which is a sd:NamedGraph with a prov:wasAttributedTo relation. The results of these two steps are shown in the RDF fragments below. The third step is to implement `named-graph-derivation`, a SADI service that accepts the description from Step 1 and returns the description in Step 2. The service answers the question, "For a given graph name in a specific SPARQL endpoint, what agent is responsible for the data it contains?" Figure 3 illustrates the output of `named-graph-derivation`. From this graph, one can directly find the attribution by following the prov:wasAttributedTo relations from the `#named-graph` node. It also includes the named graph in question, and a derivation chain that leads from the named graph to the original download URL. The result is an abstraction of the granular provenance captured throughout all four stages of conversion. The domain name of the original download is used to name the agent responsible for the file. Although this usually represents an organization, it could also represent a person or a specific software agent.

[9] See `http://prefix.cc/sd` `http://prefix.cc/prov` and `http://prefix.cc/moby`.

```
# Step 1: Describing the named graph for which we want attribution.
:service a sd:Service;
 sd:endpoint <http://logd.tw.rpi.edu/sparql>;
 sd:availableGraphs [
    a sd:GraphCollection, dcat:Dataset;
    sd:namedGraph :named-graph;
 ] .
:named-graph a sd:NamedGraph;
 sd:name <http://logd.tw.rpi.edu/source/lebot/dataset/golfers> .

# Step 2: Describing the provenance needed.
:named-graph a sd:NamedGraph;
  prov:wasAttributedTo <http://graves.cl>;
  sd:name <http://logd.tw.rpi.edu/source/lebot/dataset/golfers> .
<http://graves.cl> a prov:Agent .
```

## 6 Discussion

Despite the tendency to focus on modeling and collecting provenance, there are perhaps greater challenges to process and effectively use what has been collected. The approach we present encourages a separation of interests that permits systems to continue to collect with the level of granularity that they deem fit, while contextual applications of the granular provenance may be developed independently to provide direct, easily accessible abstractions derived from the original provenance. A further advantage of deriving abstract provenance from granular is that the provenance of provenance can be used to provide justifications for any high level claims, which can increase their trustworthiness. In contrast, directly asserted abstract records cannot be further justified.

It is important to note that while csv2rdf4lod uses the Proof Markup Language (PML) to record its provenance, the `named-graph-derivation` service provides its abstraction using W3C's PROV-O vocabulary. We are thus demonstrating interoperability at a fairly granular level between one relatively long lived provenance interlingua and the emerging W3C vocabulary. More importantly, we show how our approach can interoperate between two different provenance vocabularies as was envisioned by the W3C provenance incubator group.[10] This approach also helps advance the W3C Provenance Working Group's objective to enable provenance interchange.

The approach we present also highlights and motivates an outstanding need that, if addressed, would provide significant value in an Open Provenance Environment. In our example, the Linked Data aggregator C would benefit greatly if it were informed of any subsequent processing of its data or granular provenance (i.e., what the abstractor D and application developer did). This way, subsequent visitors to C could be led to derivations that may better suit their needs. Similarly, the primary sources would also benefit by being informed about subsequent uses of their publications (i.e., what aggregator C, abstractor D, *and* the application developer did). Consumers should be able to trace provenance in both directions, not just backward. This kind of information can also be essential for evaluating each party's contribution and return on investment. These and other benefits are lost unless the community establishes so-called "ping-back" capabilities in standards such as the W3C PROV recommendation.

[10] http://www.w3.org/2005/Incubator/prov/wiki/Provenance_Vocabulary_Mappings

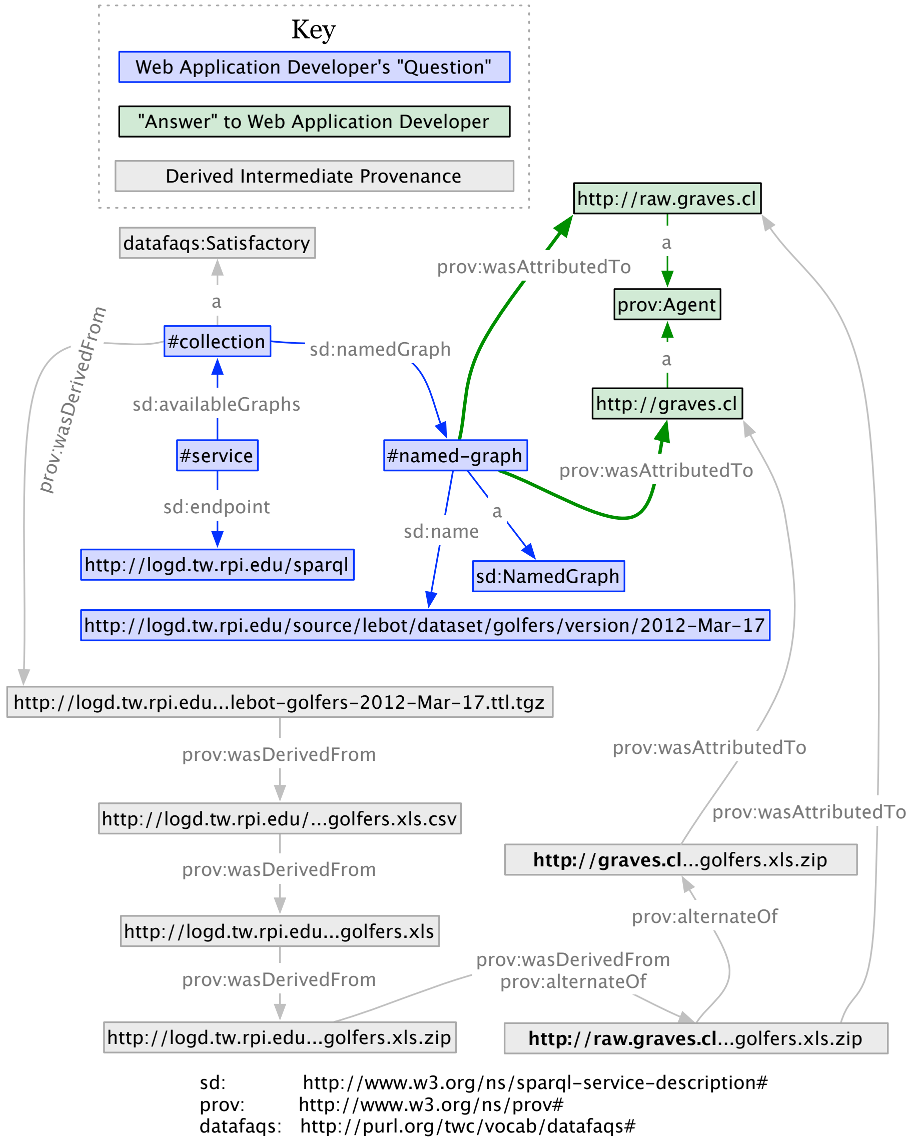
Key
Web Application Developer's "Question"
"Answer" to Web Application Developer
Derived Intermediate Provenance
http://raw.graves.cl
a
prov:Agent
a
http://graves.cl
prov:wasAttributedTo
datafaqs:Satisfactory
a
#collection
sd:namedGraph
prov:wasDerivedFrom
sd:availableGraphs
#service
#named-graph
prov:wasAttributedTo
sd:endpoint
a
sd:name
http://logd.tw.rpi.edu/sparql
sd:NamedGraph
http://logd.tw.rpi.edu/source/lebot/dataset/golfers/version/2012-Mar-17
http://logd.tw.rpi.edu...lebot-golfers-2012-Mar-17.ttl.tgz
prov:wasDerivedFrom
prov:wasAttributedTo
http://logd.tw.rpi.edu/...golfers.xls.csv
prov:wasAttributedTo
http://graves.cl...golfers.xls.zip
prov:wasDerivedFrom
prov:alternateOf
http://logd.tw.rpi.edu...golfers.xls
prov:wasDerivedFrom
prov:wasDerivedFrom
prov:alternateOf
http://logd.tw.rpi.edu...golfers.xls.zip
http://raw.graves.cl...golfers.xls.zip
sd: http://www.w3.org/ns/sparql-service-description#
prov: http://www.w3.org/ns/prov#
datafaqs: http://purl.org/twc/vocab/datafaqs#

**Fig. 3.** PROV-O description returned by SADI service `named-graph-derivation` when given an RDF description of the named graph (blue) for which SemantAqua needed provenance (green). The service also returns an intermediate level of abstraction (gray) that can be used to justify the higher level of abstraction. Both the high and intermediate abstractions are derived from the detailed PML provenance about the named graph, which was provided by the data aggregator C.

Future work could lead in several directions. The framework presented could be used to quantify the interoperability among provenance systems, by building abstraction services to reflect the representation of one system using alternative ontological representations. We currently handle PML and PROV-O, but more could be added. Since PROV-O is an emerging standard, we also expect authors of many other provenance vocabularies to map to it, thus furthering the goal of interoperability. Those authors may use the approach we present to achieve this mapping without interfering with their existing systems. Another direction could use reasoning to chain SADI services based on their OWL inputs and outputs that could lead to some powerful and automated provenance derivations. This would be particularly useful within the Open Provenance Vision, where developers will not know all systems available, may require provenance available in a variety of original representations, but can gather appropriate service descriptions and determine a solution automatically. It would also be interesting to apply the approach we present to the variety of existing abstraction algorithms, including those intended for end users. One advantage is that the summary results would remain as alternative provenance accounts that could be queried, consolidated, and reused by other systems for other purposes at later times. This contrasts with the traditional approach where the abstraction remains in the system and is lost after use.

## 7 Conclusion

As Open Data grows in popularity, so will the need for and use of Linked Data principles to integrate disparate sources. However, current integration methods provide limited support for transparency, thereby minimizing trust of their results. This causes a trade-off between authoritativeness and ease of use that needs to be reconciled before Linked Data can be widely adopted. Linked Open Data is one environment that requires – and can realize – the Open Provenance Vision. Using an example from our SemantAqua water quality portal, we show how incongruent provenance granularities can inhibit the use of provenance between systems, and argue that this challenge will grow as more systems participate. We presented an approach and supporting technologies to resolve incongruent provenance granularities between two systems by adding a third independent component that derives abstract provenance from granular provenance sources. We showed how applying this approach fulfilled a real use case that attributes the source organization for the content in a SPARQL endpoint's named graph, which was determined by tracing granular provenance. This same approach can be applied to resolve other incongruent provenance granularities that we anticipate as more systems realize the Open Provenance Vision.

## References

1. O. Biton, S. Cohen-Boulakia, S.B. Davidson, and C.S. Hara. Querying and managing provenance through user views in scientific workflows. In *Proceedings of the 2008 IEEE 24th International Conference on Data Engineering*, ICDE '08, pages 1072–1081, Washington, DC, USA, 2008. IEEE Computer Society.
2. A. Chapman and HV Jagadish. Issues in building practical provenance systems. *IEEE Data Eng. Bull*, 30(4):38–43, 2007.
3. M. Craglia, P. G. Almirall, M. M.. Bergadà, and P. Queraltó Ros. The socio-economic impact of the spatial data infrastructure of catalonia. *Institute for Environment and Sustainability, Joint Research Centre, European Commission*, 2008.

4. L. Ding, Y. Peng, P. Pinheiro da Silva, and D.L. McGuinness. Tracking RDF Graph Provenance using RDF Molecules. Technical report, UMBC, April 2005.
5. J.S. Erickson, E. Rozell, Y. Shi, J. Zheng, L. Ding, and J.A. Hendler. Twc international open government dataset catalog. In *Proceedings of the 7th International Conference on Semantic Systems*, pages 227–229. ACM, 2011.
6. D. Garijo and Y. Gil. A new approach for publishing workflows: abstractions, standards, and linked data. In *Proceedings of the 6th workshop on Workflows in support of large-scale science*, pages 47–56. ACM, 2011.
7. T. Gibson, K. Schuchardt, and E. Stephan. Application of named graphs towards custom provenance views. In *First workshop on Theory and Practice of Provenance*, TAPP'09, pages 5:1–5:5, Berkeley, CA, USA, 2009. USENIX Association.
8. A. Graves. A case study for integrating public safety data using semantic technologies. *Information Polity*, 16(3):261–275, 2011.
9. C. Hartung, Y. Anokwa, W. Brunette, A. Lerer, C. Tseng, and G. Borriello. Open data kit: Tools to build information services for developing regions. In *Proceedings of the International Conference on Information and Communication Technologies and Development*, pages 1–11, 2010.
10. T. Heath and C. Bizer. Linked data: Evolving the web into a global data space. *Synthesis Lectures on the Semantic Web: Theory and Technology*, 1(1):1–136, 2011.
11. R. Ikeda and J. Widom. Panda: A system for provenance and data. *IEEE Data Eng. Bull.*, 33(3):42–49, 2010.
12. D. McGuinness, L. Ding, P. Pinheiro Da Silva, and C. Chang. PML 2: A Modular Explanation Interlingua. In *Proceedings of the AAAI07 Workshop on ExplanationAware Computing*, volume 7, pages 49–55. Knowledge Systems Laboratory, Stanford University, 2007.
13. L. Moreau. The Foundations for Provenance on the Web. *Foundations and Trends in Web Science*, 2(2-3):99–241, 2010.
14. D. Robinson, H. Yu, W. Zeller, and E. Felten. Government data and the invisible hand. *Yale Journal of Law & Technology, Vol. 11, p. 160, 2009*, 2009.
15. L. Salayandia, P. Pinheiro, and A.Q. Gates. A framework to create ontologies for scientific data management. Technical Report UTEP-CS-12-03, University of Texas at El Paso, El Paso, TX, 2012.
16. E.G. Stephan, T.D. Halter, and B.D. Ermold. Leveraging The Open Provenance Model as a Multi-Tier Model for Global Climate Research. In *Proc of 3rd International Provenance and Annotation Workshop IPAW10*, pages 34–41, 2010.
17. T. Lebo, J.S. Erickson, L. Ding, A. Graves, G.T. Williams, D. DiFranzo, X. Li, J. Michaelis, J.G. Zheng, J. Flores, Z. Shangguan, D.L. McGuinness, and J. Hendler. Producing and Using Linked Open Government Data in the TWC LOGD Portal. In David Wood, editor, *Linking Government Data*. Springer, 2011.
18. P. Wang, J.G. Zheng, L. Fu, E.W. Patton, T. Lebo, L. Ding, Q. Liu, J.S. Luciano, and D.L. McGuinness. A Semantic Portal for Next Generation Monitoring Systems. *Proceedings of the 11th Interational Semantic Web Conference ISWC2011*, pages 253–268, 2011.
19. M.D. Wilkinson, B. Vandervalk, and L. McCarthy. The Semantic Automated Discovery and Integration (SADI) Web service Design-Pattern, API and Reference Implementation. *Journal of biomedical semantics*, 2(1):8, 2011.