Linked provenance data: A semantic Web-based approach to interoperable workflow traces

Li Ding *, James Michaelis, Jim McCusker, Deborah L. McGuinness
Tetherless World Constellation, Rensselaer Polytechnic Institute, 110 8th street, Troy, NY 12180, United States

ARTICLE INFO

Article history:
Received 16 December 2009
Received in revised form 12 October 2010
Accepted 18 October 2010
Available online xxxx

Keywords:
Representation
Information networks
Data sharing
Provenance
Linked Data
Semantic Web
Interoperability

ABSTRACT

The Third Provenance Challenge (PC3) offered an opportunity for provenance researchers to evaluate the interoperability of leading provenance models with special emphasis on importing and querying workflow traces generated by others. We investigated interoperability issues related to reusing Open Provenance Model (OPM)-based workflow traces. We compiled data about interoperability issues that were observed during PC3 and use that data to help describe and motivate solution paths for two outstanding interoperability issues in OPM-based provenance data reuse: (i) a provenance trace often requires both generic provenance data and domain-specific data to support future reuse (such as querying); (ii) diverse provenance traces (possibly from different sources) often require preservation and interconnection to support future aggregation and comparison. In order to address these issues and to facilitate interoperable reuse, integration, and alignment of provenance data, we propose a Semantic Web-based approach known as Linked Provenance Data, where: (i) the Web Ontology Language (OWL) can be used to support complex domain concept modeling, such as subtype taxonomy and concept alignment, and seamlessly connect domain extensions to OPM core concepts; (ii) Linked Data can enable open and transparent infrastructure for provenance data reuse.

1. Introduction

Provenance can be viewed as contextual metadata describing the origin or source of something. It can help describe: intention for use, manner of creation or evolution, history of subsequent owners, and place/time of creation or evolution. The importance of provenance has been widely recognized with respect to many areas including databases (e.g., [1,2]), the World Wide Web and the Semantic Web (e.g., [3,4]), and scientific workflow systems (e.g., [5,6]).

Interoperability is an important issue in provenance research, as provenance data enable reproducibility of results in scientific research thereby making provenance critical in collaborative efforts. Recently, a few provenance interlinguas including the Open Provenance Model (OPM) [7] and the Proof Markup Language (PML) [3,8] have been proposed to address interoperability issues in provenance data reuse. The Third Provenance Challenge (PC3) was organized to focus on interoperability issues in provenance using a real world data-processing workflow. Given the documentation of this workflow, each PC3 team was asked to accomplish three consecutive tasks: (i) record provenance traces for executions of the workflow, (ii) use these traces to answer a selection of assigned provenance queries, and (iii) import traces exported by other teams to answer these queries. While each team was allowed to use its own representation to accomplish the first two tasks, the third task required teams to export and import traces using OPM.

The Tetherless World Constellation team (TetherlessPC3) participated in PC3 using a Semantic Web based approach leveraging OPM along with Inference Web’s PML and the Web Ontology Language (OWL) [9]. While most PC3 teams successfully exported OPM data, interoperability issues with OPM 1.01 [7] emerged. A number of these issues were discussed in detail shortly thereafter by participants. First, the exported workflow traces, encoded in OPM using XML syntax, contained limited information about the workflow elements. For example, it was often hard to identify the type of an artifact (e.g. “detection”) by directly looking at text-based identifiers,

...
especially when unclear identifiers (e.g., numerically-based) were used. Second, each PC3 team recorded provenance traces at varying levels of abstraction, in addition to having different interpretations of the assigned provenance queries. Therefore, different teams often obtained different results for individual provenance queries. These interoperability issues introduced representation and infrastructure challenges related to PC3 provenance data sharing. This paper presents a Semantic Web-based approach, known as Linked Provenance Data, to address the challenges using our experiences from both PC3 and the last 8 years of provenance research related to the Inference Web project [3]. The contributions of this work are two-fold: (i) we provide an analysis of outstanding interoperability issues with OPM in the context of PC3, and (ii) we demonstrate selected merits of Semantic Web technologies, especially OWL and Linked Data [10], in enhancing the interoperability of OPM compatible provenance data.

The rest of this paper is structured as follows: Section 2 quantitatively describes interoperability issues related to OPM 1.01 that were exposed in PC3. Sections 3–5 show how Semantic Web technologies, especially OWL and Linked Data, can help improve the interoperability of OPM-compatible provenance data in the context of data modeling, publishing, integration, reuse and querying. Section 6 reviews related work, and Section 7 discusses our conclusions and future plans.

2. Interoperability issues in PC3

The Third Provenance Challenge (PC3) was structured with provenance interoperability issues in mind. During PC3, the participating teams were asked to share workflow traces using OPM 1.01. As shown in Example 1, OPM provides a generic data model for capturing important provenance concepts (artifacts, processes and agents) as well as provenance relations (e.g. an artifact was generated by a process).

Example 1. A fragment of OPM data (i.e. a workflow trace encoded using OPM/XML syntax). It shows that an artifact “DBEntryP2Detection_0_ForIter3” was “created” by a process “LoadCSVFileIntoTable_2_ForIter3” within account “ALL”.

```
<artifact id="DBEntryP2Detection_0_ForIter3"/>
<value xsi:type="xs:string">Fortier3,DBEntry_P2Detection,
261887437030025144</value>
<account id="ALL"/>
<artifact/>
<wasGeneratedBy/>
```

During PC3, many teams experienced challenges in working with OPM 1.01 based provenance data exported by other teams, resulting in numerous revision suggestions. These challenges were caused by the diversity in managing, modeling, and querying provenance data. In what follows, we analyze some underlying causes of these challenges using statistics and grounded examples collected from PC3 teams via email correspondence. We also used information from the PC3 website when we did not receive direct answers from a team.

2.1. Diversity in provenance data infrastructure

Interoperability issues often arise when provenance data needs to be shared across infrastructures that store and query provenance data differently. Table 1 supports an observation O1: A diverse collection of query languages/tools were used to store provenance data and answer provenance questions; and only a few teams reported success in importing traces from other teams. Of the twelve teams that attempted any form of querying, almost all used different query languages/tools (e.g. SQL, XQuery7 and SPARQL8). The varying approaches to provenance storage naturally led to alternative querying approaches. For example, it is natural to use SQL to query an RDBMS and SPARQL for RDF stores. Eight out of the twelve possible sets of OPM data were imported by other teams. Although the exported OPM data (in either XML or RDF) could be directly queried using, e.g. XQuery or SPARQL, only three teams (i.e. TetherlessPC3, Vistrails3 and PASS3) claimed success in answering provenance questions using their imported OPM data.

2.2. Diversity in OPM extensions

Even with a common infrastructure, interoperability issues may arise when parties extend a common data model differently. Table 1 shows another observation O2: Users extended their exported OPM data with additional structure and information. Of the twelve teams that exported OPM data, six teams enriched their XML-based OPM data with additional schema and data. Interestingly enough, three teams published not only OPM data (not extended) in XML but also extended OPM data in RDF. With further investigation of the OPM data (see Examples 2 and 3), many of the PC3 provenance questions were found to be hard to answer without including additional information in the OPM data. Moreover, the extended OPM data often used non-standard serializations which may be difficult to be processed by existing OPM tools. Observation O2 suggests a direction for extending OPM 1.01. OPM enhanced with additional terms and structures could be more understandable and users could answer provenance questions using structured queries rather than needing to resort to using heuristic approaches to find artifacts using regular expression matches to literal identifiers and labels.

Example 2. In order to answer Core Query 1 (CQ1) “For a given detection, which CSV files contributed to it?” SwiftPC3 extended XML-based OPM data using a complex data structure as the value of opm:value. The fragment below additionally annotates a process “x3” with its subtype “execute”, label, and unique identifier. Similar extensions were also done by other PC3 teams, such as UCDGC, KCL, and Vistrails3.

```
<process id="x3"/>
<account id="base'/">
```

5 A list of PC3 team web pages are located at: http://twiki.ipaw.info/pub/Challenge/ParticipatingTeams3.
6 Eight out of twelve PC3 teams have responded to our email questionnaire. This data in this paper is not used to rank teams but instead is used to inform an analysis of the challenges and motivate provenance language proposals, so even though some of the webpage obtained information may be slightly outdated, we feel it is sufficient for use in this paper.
7 XQuery 1.0, http://www.w3.org/TR/xquery/.
8 SPARQL Specification, see http://www.w3.org/TR/rdf-sparql-query/.

Please cite this article in press as: L. Ding et al., Linked provenance data: A semantic Web-based approach to interoperable workflow traces, Future Generation Computer Systems (2010), doi:10.1016/j.future.2010.10.011
Table 1
PC3 team self-reported query, export and import information.

<table>
<thead>
<tr>
<th>Team</th>
<th>Query languages/tools</th>
<th>Exported OPM/XML</th>
<th>Exported in other formats?</th>
<th>Imported by</th>
</tr>
</thead>
<tbody>
<tr>
<td>NCSA²</td>
<td>Tupelo API,²,³</td>
<td>Yes</td>
<td>RDF/XML (Tupelo, extended)</td>
<td>SotonUSCISIPc3, UoM</td>
</tr>
<tr>
<td>SwiftPc³</td>
<td>Swift²</td>
<td>Yes, Extended</td>
<td>n/a (possible in database dump)</td>
<td>UCDGC, SotonUSCISIPc3</td>
</tr>
<tr>
<td>Trident</td>
<td>n/a</td>
<td>No</td>
<td>XML (XOML)</td>
<td>UCDGC</td>
</tr>
<tr>
<td>UCDGC⁴</td>
<td>SQL</td>
<td>Yes, Extended</td>
<td>XML (comad-kepler)</td>
<td>SotonUSCISIPc3, UoM, TetherlessPC3, PASS3</td>
</tr>
<tr>
<td>SotonUSCISIPc3</td>
<td>Java Code⁴</td>
<td>Yes</td>
<td>RDF/XML (Tupelo)</td>
<td>UCDGC, UoM</td>
</tr>
<tr>
<td>UoM⁴</td>
<td>Taverna API⁴</td>
<td>Yes</td>
<td>RDF/XML (Tupelo, extended)</td>
<td>TetherlessPC3</td>
</tr>
<tr>
<td>TetherlessPC³</td>
<td>SPARQL</td>
<td>Yes</td>
<td>RDF/XML (PC3OPM, extended)</td>
<td>UCDGC, PASS3</td>
</tr>
<tr>
<td>UvA/VI+</td>
<td>SQL</td>
<td>Yes</td>
<td>n/a</td>
<td>UCDGC</td>
</tr>
<tr>
<td>SDSCPc3</td>
<td>XQuery</td>
<td>Yes</td>
<td>n/a</td>
<td>VisTrails3, PASS3</td>
</tr>
<tr>
<td>VisTrails3</td>
<td>XQuery</td>
<td>Yes, Extended</td>
<td>n/a</td>
<td>VisTrails3</td>
</tr>
<tr>
<td>KCL⁵</td>
<td>Java Code</td>
<td>Yes, Extended</td>
<td>n/a</td>
<td>UC3GCL</td>
</tr>
<tr>
<td>PASS3⁵</td>
<td>Path Query Language</td>
<td>Yes</td>
<td>n/a</td>
<td>UCDGC, PASS3</td>
</tr>
<tr>
<td>Karma³</td>
<td>XPath⁵</td>
<td>Yes, Extended</td>
<td>n/a</td>
<td>UC3GCL</td>
</tr>
<tr>
<td>UCSB/raKe3</td>
<td>n/a</td>
<td>n/a</td>
<td>n/a</td>
<td></td>
</tr>
<tr>
<td>UTEP</td>
<td>n/a</td>
<td>n/a</td>
<td>n/a</td>
<td></td>
</tr>
<tr>
<td>UTEP</td>
<td>n/a</td>
<td>n/a</td>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

¹ Denotes PC3 teams that answered our questionnaire, the other teams' data was obtained from PC3 website.
² Uses a combination of SPARQL and procedural code to accomplish querying.
³ Part of the Java code is from PASOA (http://twiki.passoa.ecs.soton.ac.uk/bin/view/PASOA/WebHome) Project.
⁴ Java code (http://www.ci.uchicago.edu/swift/).
⁵ http://www.taverna.org.uk/.
⁶ http://www.eecs.harvard.edu/syrah/pql/.
⁷ http://www.w3.org/TR/xpath/.

Example 3. For Core Query (CQ) 1, NCSA published extended RDF-based OPM data: the fragment of RDF/XML data contains additional file path and subtype information for the artifact “http://pc3#FileEntryArtifact2.” It is also interesting to see that the corresponding fragment of XML-based OPM data has no extension and lacks information to answer CQ1. Note that UoM and TetherlessPC3 took similar approaches.

```
<op:artifact rdf:about="http://pc3#FileEntryArtifact2">
  <a:pathToFile>C:\dev\workspace\PC3\SampleData\j062941\P2_j062941_B001_P2fits0_20081115_P2Detection.csv</a:pathToFile>
  <a:rowCount>0</a:rowCount>
  <a:columnCount>0</a:columnCount>
  <a:imageWidth>0</a:imageWidth>
  <a:imageHeight>0</a:imageHeight>
  <a:filePath>C:\dev\workspace\PC3\SampleData\j062941\P2_j062941_B001_P2fits0_20081115_P2Detection.csv</a:filePath>
  <a:contentType>application/octet-stream</a:contentType>
  <a:isExistsCSVFileConditional1>true</a:isExistsCSVFileConditional1>
  <a:isCSVReadyFileExistsProcess>true</a:isCSVReadyFileExistsProcess>
  <a:createEmptyLoadDB_0_main>true</a:createEmptyLoadDB_0_main>
  <a:isExistsCSVFileConditional1>true</a:isExistsCSVFileConditional1>
  <a:isCSVReadyFileExistsProcess>true</a:isCSVReadyFileExistsProcess>
  <a:createEmptyLoadDB_0_main>true</a:createEmptyLoadDB_0_main>
  <a:csvColumnNames>ReadCSVFileColumnNames_1_ForIter2, CreateEmptyLoadDB_0_main, ReadCSVFileColumnNames_1_ForIter2, ReadCSVReadyFile_0_main</a:csvColumnNames>
</op:artifact>
```

2.3. Diversity in provenance data and query modeling

Even when a common data model and extension is used by all parties, interoperability issues may arise when users model data differently (e.g., modeling workflow traces at different levels of abstraction, referring the same entity with different IDs) and/or model queries differently (e.g., interpret provenance questions into different formal queries). Table 2 shows an observation O3: The number of answers varied significantly across PC3 teams for certain queries. There is high variation in the number of answers for Core Query (CQ) 3 and Optional Query (OQ) 8 because they ask for all (directly and indirectly) relevant processes. Note that CQ2, OQ3 and OQ6 have low variation because they expect just one answer. Example 4 shows that different teams captured provenance traces at different levels of abstraction. Moreover, Example 5 shows that the PC3 teams may have interpreted the English descriptions of provenance questions into different queries. This observation shows an inherent diversity across teams, even when they are using the same provenance model to record the same workflow execution under the same configuration. Meanwhile, it also suggests research paths aimed at linking relevant workflow traces: can we connect two different traces by using some notion of similarity (instead of causal relations) and then find their differences or use them to jointly answer provenance questions?

Example 4. TetherlessPC³ and NCSA found different answers to CQ3 “Which operation executions were strictly necessary for the Image table to contain a particular (non-computed) value?” In this case, NCSA recorded a more detailed workflow trace (e.g. additionally recording “http://pc3#Iterator1” as a process) than TetherlessPC³. NCSA also interpreted the provenance query differently by returning additional control flow processes (e.g. “IsExistCSVFileConditional1”).

```
TetherlessPC³ [4 answers]
LoadCSVFileIntoTable_1_ForIter2,
CreateEmptyLoadDB_0_main,
ReadCSVFileColumnNames_1_ForIter2,
ReadCSVReadyFile_0_main
```

```
NCSA [26 answers]
[http://pc3#IsCSVReadyFileExistsProcess]
[http://pc3#LoadCSVFileIntoTableConditional0]
[http://pc3#IsExistsCSVFileConditional1]
[http://pc3#LoadCSVFileIntoTableProcess0]
[http://pc3#IsExistsCSVFileProcess1]
[http://pc3#UpdateComputedColumnsConditional0]
[http://pc3#Iterator1]
[http://pc3#ReadCSVFileColumnNamesProcess0]
```

Source: http://twiki.ipaw.info/bin/view/Challenge/TetherlessPC³
Source: http://twiki.ipaw.info/bin/view/Challenge/NcsaPc3

Example 5. While SDSC returned 13 answers for CQ3, VisTrails3 imported SDSC’s OPM data and then returned two extra answers (15 answers in total¹⁰). We list the labels of the two extra answers below:

- “LoadForEachCompositeActor.IsMatchCSVFileColumnNames fire 1”
- “LoadForEachCompositeActor.StopOnFalse2 fire 1”

Please cite this article in press as: L. Ding et al., Linked provenance data: A semantic Web-based approach to interoperable workflow traces, Future Generation Computer Systems (2010), doi:10.1016/j.future.2010.10.011
3. Interoperability challenges and linked provenance data

Based on observations and experiences from PC3, we identified the following two complementary challenges concerning provenance data interoperability:

- **Representation Challenge**: Many provenance questions must be answered using both domain-independent provenance data and domain specific data. Therefore, a provenance Interlingua, such as OPM, should include common models for representing interoperable provenance, entity referencing, and querying. Additionally, guidelines for interoperable domain-specific extensions should be provided.

- **Infrastructure Challenge**: In order to share provenance data (encoded using OPM in PC3), parties benefit from agreeing upon data storage formats and data access/query protocols. Common best practices for an OPM-based provenance data sharing infrastructure should be detailed. This allows users to have enough information to choose the same infrastructural options (such as using compatible provenance storage options and/or query languages) or to build bridges from their infrastructure to other party’s infrastructural components.

In order to address the interoperability challenges related to provenance data reuse, we propose a Semantic Web-based approach—**Linked Provenance Data**. Towards the representation challenge, Linked Provenance Data leverages OWL to enable flexible and reusable domain-specific extensions to OPM and enriches connections across related workflow traces (see Section 4). Towards the infrastructure challenge, Linked Data technologies can be used to build an open and transparent infrastructure for provenance data reuse (see Section 5). It should be noted that OPM 1.1 [11] offers a similar but parallel approach (mainly via the “profiles” extension), and this paper makes a point of highlighting the merits of a Semantic Web-based approach to the challenges.

4. Provenance ontology revisited

In PC3, we created an OWL ontology called PC3OPM [11] to serve as the foundation of provenance encodings for Linked Provenance Data. Unlike other OWL ontologies (e.g., [12]) from previous provenance challenges, PC3OPM used OPM 1.01 as its core provenance data model, and extended OPM 1.01 with PC3-specific concepts. A similar OWL ontology [12] for OPM 1.01 was contributed by NCSA’s Tupelo project in PC3, but it contains no domain-specific extensions. Work has begun on an OWL ontology [13] for OPM 1.1, which includes newly added annotation properties such as opm:pmname, and opm:type. In the rest of this section, we describe the modular design of the PC3OPM ontology and selected best practices for using OWL to address the representation challenges: (i) how OPM is encoded in OWL to provide a common data model that supports the expected concept modeling and inferences; (ii) how domain knowledge can be added using OWL constructs to support interoperable domain extensions; and (iii) how “equivalent” links can be used to align OPM data represented at different levels of abstraction.

**Modular ontologies.** Since OPM only focuses on the most domain-independent provenance concepts and relations, we should separate OPM core concepts from domain extensions into different modules. The first version of PC3OPM simply contains both OPM and PC3 concepts. After PC3, it was determined that the PC3OPM ontology could be split into two modular ontologies: one for OPM concepts and one for PC3 specific concepts — following the design of the modularized PML2 ontologies [8]. This design helps isolate the development of domain extensions from the development of the OPM concepts.

**Mapping OPM concepts.** OWL can be used to declaratively define OPM concepts and thus support a common data model for OPM. An OWL ontology for OPM should maintain a one-to-one mapping to its corresponding OPM specification. While it is straightforward to map OPM’s “nodes” (i.e. Artifact, Agent and Process) to OWL classes, all OPM related ontologies (from TetherlessPC3, Tupelo and OPM 1.1) unanimously mapped OPM’s “edges” to OWL classes to preserve additional information associated with edges, such as time and role. For example, in PC3OPM the causal relation Used is mapped to an OWL class “PC3OPM:Used” (see links from Fig. 1(a)–(e)). The class is defined as a subclass of “PC3OPM:Dependency” and inherits the latter’s properties, including “PC3OPM:hasRole” and “PC3OPM:hasAccount”. Note that PC3OPM ontology (see Fig. 1(e)) contains a full mapping for OPM 1.01.

**Adding inferred causal relations.** OWL also supports common modeling for OPM inference. In order to differentiate the directly asserted causal relations (defined as instances of e.g. “PC3OPM:Used”) from the inferred binary causal relations (e.g. a process indirectly needs an artifact as input), we created several OWL object properties for causal relations (e.g. “PC3OPM:opUsed”). Such properties can be used to declaratively represent the inferred binary relations.

---

and support provenance graph inference. While these properties were defined as transitive at the time of extending OPM 1.01, they now can be aligned with the definition of “multi-step relations” in OPM 1.1: only “PC3OPM:opWasDerivedFrom” is defined as an OWL transitive property (to leverage OWL native transitive inference), and all these properties can be derived by SPARQL-based rule inference (see Example 6 in Section 5.2).

Adding domain information. In order to support interoperable domain-specific OPM extensions, one can add annotation properties similar to the extensions shown in Examples 2 and 3. While OPM 1.1 has adopted this approach through the concept “annotation”, we additionally perceive some potential enhancements: (i) Sub-types of artifact were modeled as subclasses of “PC3OPM:Artifact” instead of a string value in PC3OPM. As shown in Fig. 1, the links from Fig. 1(b)–(e) exemplify how the domain specific concepts in PC3OPM are determined by the descriptions of provenance questions, source code of workflow systems, and graphical workflow diagram respectively. This design makes it easy to find direct instances of PC3OPM:CSVFileEntry as well as all (direct and indirect) instances of PC3OPM:Artifact. Additionally, users can model a taxonomy of subtypes and even specify mapping relations among the subtypes. (ii) Some domain data should be modeled as complex objects rather than a flat list of property-value pairs. For example, a table cell in a CSV file should be described by its row–column indices and a reference to the object denoting the hosting CSV file. This design enhances flexibility for encoding domain data.

“Equivalent” artifacts. Many PC3 workflow traces were generated from the same workflow using the same configuration. Although each team generated workflow traces with their own systems, the workflow code and input data (i.e. the content of CSV files) are essentially equivalent. In order to resolve the diversity of reference to the same entity, with declarative “equivalence” relations among artifacts, users could integrate multiple traces for comparison, and potentially provide more thorough answers to the PC3 provenance queries. We can use owl:sameAs to declaratively connect semantically equivalent artifacts across different workflow traces. Such explicit mappings support users in aligning workflow traces modeled using different levels of granularity.

5. Infrastructure design for linked provenance data

This section focuses on how Semantic Web technologies are used in addressing the infrastructure challenge from Section 3. Our infrastructure design leverages prior work from the Inference Web project[3], where provenance data has been published as linked data since 2003.

5.1. Architecture

The infrastructure for Linked Provenance Data involves tools and mechanisms to publish, enhance and consume provenance data on the Web. Fig. 2 depicts the architecture of TetherlessPC3, a domain specific implementation of our infrastructure, where ovals denote data and block arrows denote operations. Below, we discuss how this infrastructure can be used to approach the three tasks of PC3 and address the infrastructure challenges from Section 3: Recording provenance traces. In order to generate workflow traces in PC3, we use terms from the PC3OPM ontology that cover both OPM
In addition to OPM’s provenance relation, provenance data The following examples show SPARQL queries that
Example 6
13
– By reusing URIs or identifier strings in recording workflow
Once assigned a unique URI, an entity can be described in dis-
RDF enables a graph model for representation and further sup-
By reusing URIs or identifier strings in recording workflow

texts are accessible to legitimate users, we
available for controlling Web-based data access. Ultimately, as
Publication and access; however, there are many security options
If we consider our best practices for access met.
Publish provenance data as linked data. While OPM/XML syntax can
help users import/export OPM-based workflow traces, publishing
provenance data in RDF following linked data principles carries
additional benefits related to data reuse:

• Once assigned a unique URI, an entity can be described in dis-
• RDF enables a graph model for representation and further sup-
• By reusing URIs or identifier strings in recording workflow
In Incremental and declarative data-centric computation. By publishing
OPM data as linked data, users can incrementally infer provenance
relations from existing data and answer provenance questions using
declarative SPARQL queries. As shown in Example 6, we can use both
a SPARQL “CONSTRUCT” query (enabling rule-based inference15) and
OWL inference to enhance linked OPM data. This approach allows teams to incrementally build up OPM data,
and it also enables transparency by declaratively tracking “why”
provenance [1] for inferred data.

Example 6. The following examples show SPARQL queries that
follow the OPM 1.01 specification to infer multi-step “WasDerived-
From” and “WasTriggeredBy” relations from directly asserted OPM graph data. Note that OWL reasoning will take care of the transitive
 closure computation on PC3OPM:opWasDerivedFrom.

PREFIX PC3OPM: <http://www.cs.rpi.edu/~michaj6/provenance/PC3OPM.owl#>
CONSTRUCT { ?d1 PC3OPM:opWasDerivedFrom ?d2 .
} } .
PREFIX PC3OPM: <http://www.cs.rpi.edu/~michaj6/provenance/PC3OPM.owl#>
?d1 PC3OPM:opWasDerivedFrom ?d2. } } }

Users can also consume the original (or enhanced) linked
provenance data using SPARQL queries. For example, PC Core
Query 3, “Which operation executions were strictly necessary for
the image table to contain a particular (non-computed) value”,
can be translated into the following SPARQL query, where PC3:
provVarDbEntry2ImageMeta_0 is the URI of the referred process.

PREFIX PC3: <http://www.cs.rpi.edu/~michaj6/provenance/OurTrace.owl#>
PREFIX PC3OPM: <http://www.cs.rpi.edu/~michaj6/provenance/PC3OPM.owl#>
SELECT ?p FROM <http://www.cs.rpi.edu/~michaj6/provenance/OurTrace.owl#>
WHERE { PC3:provVarDbEntry2ImageMeta_0 PC3OPM:opWasTriggerBy ?p. }


15 SPARQL can be used to express acyclic Datalog rules, with negation [13].
Later, we can reuse the same SPARQL query on an imported workflow trace (from UoM) with small modifications — changing the dataset to be queried and changing the URI of the referred process. Note that the URI is manually obtained from the imported workflow trace.

```
PREFIX PC3Uom: <http://www.cs.rpi.edu/~michaj6/provenance/UoM.owl#>
PREFIX PC3OPM: <http://www.cs.rpi.edu/~michaj6/provenance/PC3OPM.owl#>
SELECT ?p
```

Supporting multi-trace queries. This could be especially useful when traces recorded complementary information due to their choices concerning the level of abstraction. Once we have asserted owl:sameAs relations across different traces, we could modify the SPARQL query for CQ3 to get a potentially larger set of processes from all imported traces. This query could be substituted by aggregating the query results from individual traces, but it declaratively reflects the semantics of the provenance questions.

```
PREFIX PC3: <http://www.cs.rpi.edu/~michaj6/provenance/OurTrace.owl#>
PREFIX PC3OPM: <http://www.cs.rpi.edu/~michaj6/provenance/PC3OPM.owl#>
PREFIX OWL: <http://www.w3.org/2002/07/owl#>
SELECT ?p
```

6. Related work

Workflow interoperability issues have also been reported in other work. Elmroth [14] provided a comprehensive review of how interoperability could be impacted by three factors: (i) the workflow’s execution environment, (ii) the model of computation used to structure a workflow, and (iii) the expressivity of the language used to encode a workflow.

A number of abstract provenance models have been proposed to enable workflow interoperability. OPM [7,11] defined a set of general-purpose primitive concepts for modeling workflow. PAOSA [15] introduced an abstract model for process documentation. The model was designed with the following features: recording only factual information, recursively attributing responsible actors and allowing automated creation. This line of work focuses on conceptual models and does not require that the implementation be tied to one specific language.

Semantic Web encodings for provenance have also been developed to provide deployable solutions, especially for sharing the provenance of workflow systems on the Web. PML [3,8] was introduced as an interlingua to support knowledge provenance and it has been used in a wide range of settings [16] to track and explain information manipulation workflows. Hartig [17] defined an abstract provenance model for the Web of Data. The model described is similar to OPM, but without the distinction between Processes and Artifacts. The author distinguishes between two dimensions of web data provenance: provenance about the creation of the data, or how it originates or is generated and provenance about access of the data, or the methods and sources to retrieve it. This work also discusses potential Semantic Web encodings for the provided provenance model. Watkins and Nicole [18] discussed the possibility of using named graphs to support large-grained provenance representations through the application of the Dublin Core Terms vocabulary [16] (e.g. dcterms:created and dcterms:replaces). Zhao et al. [19] implemented this technique to represent versioning of linked data named graphs and the mapping graphs that use them. This includes the discovery of the most recent version of a named graph without relying on an explicit versioning scheme, discovery of differences between versions of the graphs, and providing explanations for those differences. The authors argue that the named-graph level of granularity is far cheaper than finer grained provenance representations.

Zhao et al. [20] described the framework in MyGrid/Taverna for the generation and storage of workflow provenance and the associated provenance ontology. The authors identify open issues, including the current lack of user-appropriate provenance visualizations. Chebotko et al. [21] developed an RDBMS representation that uses an ontology and showed that it is optimized for query. They also described a SPARQL-to-SQL query translation method to support their use case.

Efforts by other provenance challenge teams have produced notable work. The first provenance challenge provided eight queries over a computational fMRI workflow. Golbeck and Hendler [12] provided an ontologically-driven approach to the challenge. Domain-specific primitives, such as File, Service Execution, and Workflow Execution were combined with a small number of inference rules to determine file ancestry. Notably, in addition to the successful query of their own data, the authors report success in importing provenance generated by the Harvard PASS [22] system, converting it into their OWL format, and successfully executing all eight of the provenance queries on that data. In PC3, Missier and colleagues were able to import OPM-based provenance by entailing plausible corresponding workflows [23].

The MyGrid/Taverna team also reported success in the first provenance challenge [24]. The authors report the successful generation and query of primary provenance using Ouzo, such as logs and data lineage, but also successful queries across secondary abstraction and interpretation of that provenance using ProQA. ProQA provides abstracted views based on user-contributed ontologies and subsets of properties and concepts from the Ouzo ontology.

The Wings/Pegasus System was also used to track provenance in the first provenance challenge. Kim et al. [25] described the creation of valid specifications of computational workflows that can be efficiently executed in distributed shared environments. The Wings/Pegasus system produces both application-level provenance and execution provenance, which were used to answer the provenance challenge queries. Of note is the ability for the system to provide tracing on expected versus actual computation through analysis of workflow optimization steps.

7. Conclusion

In PC3, OPM 1.01 was chosen to serve as a common provenance representation to facilitate provenance interoperability, thereby providing opportunities for identification of remaining interoperability issues related to OPM. We have further documented some of these interoperability challenges and have discussed how our linked provenance data approach can help. For example, a shared workflow trace should carry enough information (both generic provenance relations and domain specific annotations) to support future reuse (e.g. answering provenance questions). In this paper, we evaluate these interoperability issues using statistics and examples collected from PC3. Following this, we identified two interoperability challenges related to representation and infrastructure. We also described a Semantic Web-based approach to enable Linked Provenance Data, which: (i) extends an OPM ontology to enable rich and interoperable domain-specific provenance descriptions by providing additional types of connections; and
(ii) shows selected merits of Semantic Web technologies in enabling open and transparent provenance data management on the Web. One concrete example of this approach is provided by looking at the TetherlessPC3 results that included a finer-grained provenance model that supported answering many of the more challenging PC3 provenance questions; see Table 2.

Both our work and the recently released OPM 1.1 showed approaches to interoperability issues with OPM 1.01, including ontological enhancements and guidance on infrastructure design. Sharing significant interests with OPM 1.1, our approach carries some unique merits. First, OWL can be used to support rich (domain-specific) descriptions of OPM nodes, such as taxonomies of subtypes, complex annotation structures, and interconnections across related traces. Meanwhile, modular ontologies enable better control of the development of OPM core concepts and domain-specific extensions. Second, Linked Provenance Data, in addition to current XML-based data reuse, offers an open and transparent infrastructure for provenance data reuse.

We will continue our research on interlinking provenance traces. This process depends heavily on use and interpretation of the owl:sameAs relation (e.g. when should two instances of opm:Process be considered not to be identical and under what conditions should two electronic documents be considered the same) [26,27]. We are evaluating efficient distributed provenance data storage and query technologies, e.g. federated query over distributed provenance data [28,29]. We are also working on describing additional best practices for generating linked OPM data related to defining similar but not completely-equivalent relationships (e.g. JDK 1.5.2 could be considered to be a version of JDK5 and a version of JDK). Further, we will continue our work aimed at providing solutions to provenance infrastructure (e.g. Inference Web) and Interlingua (e.g. the Proof Markup Language) challenges on the Web.

Acknowledgements

We thank Ankesh Khandelwal for valuable input during the preparation of this manuscript. We also thank Zhenming Shangguan and Rui Huang for their participation in TetherlessPC3 design and implementation.

References


L. Ding is a research scientist in the Tetherless World Constellation at Rensselaer Polytechnic Institute (RPI). His research interest includes semantic search, user interaction, web-scale computing, ontology, provenance, privacy and trust, information integration, and social computing. He is the inventor of Swoogle, one of the first semantic web search engines on the Web. He is a former Kodak postdoctoral fellow in the Knowledge Systems, Artificial Intelligence Laboratory (KSL) at Stanford University. He received Ph.D. in computer science from University of Maryland Baltimore County (2006), and received B.S. (2001) and M.S. (1998) in computer science from Peking University.

James Michaelis is a PhD. student in the Tetherless World Constellation at Rensselaer Polytechnic Institute. His research focus is on Semantic Web based provenance applications—specifically, in the context of scientific workflow driven systems.
Jim McCusker is a Ph.D. student in the Tetherless World Constellation at Rensselaer Polytechnic Institute, and is a Programmer-Analyst at the Yale School of Medicine. His research focus is on semantically-enriched provenance and knowledge representation in federated biomedical research grids.

Deborah L. McGuinness is the Tetherless World Senior Constellation Chair and Professor of Computer Science and Cognitive Science at Rensselaer Polytechnic Institute. Deborah’s research focuses on the semantic web, ontologies and ontology evolution environments, explanation, trust, and semantic eScience. Until recently, Deborah led the Knowledge Systems, Artificial Intelligence Lab at Stanford University. Deborah is also the CEO of McGuinness Associates consulting and acting Chief Scientist for Sandpiper Software. Deborah received her B.S. degree from Duke University, M.S. from the University of California at Berkeley, and her Ph.D. from Rutgers University.