Provenance Querying for End-Users: A Drug Resistance Case Study

Bartosz Baliś[1,2], Marian Bubak[1,2], Michal Pelczar[2], and Jakub Wach[2]

[1] Institute of Computer Science, AGH, Poland
{balis,bubak}@agh.edu.pl, wach.kuba@gmail.com
[2] Academic Computer Centre – CYFRONET, Poland

Abstract. We present a provenance model based on ontology representation of execution records of in silico experiments. The ontologies describe the execution of experiments, and the semantics of data and computations used during the execution. This model enables query construction in an end-user oriented manner, i.e. by using terms of the scientific domain familiar to researchers, instead of complex query languages. The presented approach is evaluated on a case study Drug Resistance Application exploited in the ViroLab virtual laboratory. We analyze the query capabilities of the presented provenance model. We also describe the process of ontology-based query construction and evaluation.

Keywords: e-Science, Grid, ontology, provenance, ViroLab.

1 Introduction

The importance of provenance tracking in e-Science environments has been pointed out many times. However, providing an adequate end-user support for provenance querying is also an important challenge. It has been recognized that the need for provenance queries goes beyond the lineage of a single data item and searching or mining over many provenance records might be useful [8] [12]. However, there is a technical and conceptual barrier preventing or making it difficult for researchers or specialists who are the end users of e-Science environments, to construct complex queries using the query languages such as XQuery, SQL or SPARQL, or even dedicated APIs built on top of those languages or particular data models. Therefore, there is a need for provenance query methodology which allow end-users to construct powerful queries in an easy way.

The goal of this work is to present a provenance model which enables complex queries over many provenance records. In addition, the model should support end-user oriented querying in order to be usable by non-IT experts. Inspired by the vision of a future e-Science infrastructure, which brings together people, computing infrastructures, data and instruments, and in which semantics and knowledge services are of paramount importance [9] [5], we propose a provenance model based on ontologies which model the execution of scientific workflows. We argue that ontology models describing the execution of experiments (including provenance) are a convenient inter-lingua for: (1) *end-users* who use ontologies as a query language, (2) *query tools* using them to represent and evaluate queries, and *provenance repository* in which the ontologies serve as the data model.

M. Bubak et al. (Eds.): ICCS 2008, Part III, LNCS 5103, pp. 80–89, 2008.

Most existing approaches concentrate on a provenance model sufficient for correct computation of and querying for derivation paths of selected data items [4] [13] [10]. Some approaches introduce simple provenance models that abstract from any computation model, e.g. the Read – Write – State-reset model [4], and the p-assertions model [6]. It has been pointed out that those low-level models limit the provenance query capabilities [13]. A complex RDF and ontology-based provenance model is introduced in myGrid/Taverna [15]. This model has been shown to support complex provenance queries [14]. However, the aspect of query construction in an end-user-friendly manner is not explored in the mentioned works.

The approach to provenance tracking and querying presented in this article is evaluated in the context of the ViroLab Project[1] [11] which provides a virtual laboratory for conducting in silico experiments related to diagnosis of infectious diseases, especially the HIV virus[2] [7].

The remainder of this paper is structured as follows. Section 2 describes the provenance model. Section 3 presents the case study Drug Resistance Workflow. In Section 4, query capabilities of the presented provenance model are investigated. Section 5 introduces the ontology-based provenance query methodology oriented towards end-users. Finally, Section 6 summarizes the current achievements and discusses future work.

2 Provenance Model

Our provenance model is based on the concept of *Experiment Information*, an ontology-based record of experiment execution. The base for this record is an ontology describing in silico experiments in which the workflow model is used as the computation model. The execution stages of the experiment and the data items used in the workflow are linked to domain ontologies of applications and data in order to enhance the semantic description of experiment data and computations. Fig. 1 presents part of the ontology tree in which it is shown how the generic concepts describing the experiment execution are connected do domain ontologies concepts.

Provenance tracking is based on monitoring of the execution of a workflow. The creation of an ontology-based record of the experiment execution is done as a process of translation from low-level monitoring data composed of monitoring events into high-level Experiment Information. This process is depicted in Fig. 2. A workflow enactment engine executes a workflow according to some plan. Monitoring events generated by the instrumentation come from different distributed sources, among others, the workflow enactment engine and the workflow activities. A monitoring system collects and correlates the monitoring events, and passess them to a Semantic Aggregator component which aggregates and translates the monitoring data into an ontology representation, according to an ontology Experiment Information model. The ontology individuals are published into the provenance tracking system (PROToS) [2] and stored in a permantent Experiment Information Store. The process of monitoring, event correlation, aggregation and translation to ontologies is described in detail in [1].

[1] ViroLab Project: `www.virolab.org`

[2] ViroLab virtual laboratory: `virolab.cyfronet.pl`

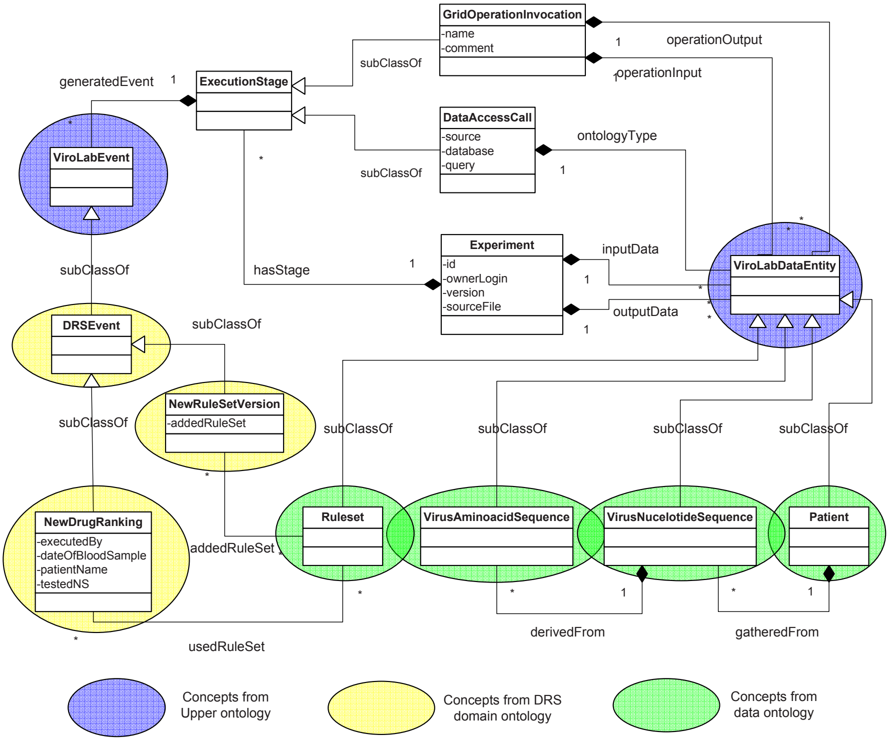


Fig. 1. Generic experiment ontology and related domain ontologies

3 Drug Resistance Workflow

The workflow which is used to illustrate the capabilities of the presented provenance model is the Drug Ranking System (DRS) application exploited in the ViroLab project. The goal of this application is to determine best drug combinations for a particular HIV virus taken from the blood sample of a patient. In the simplest DRS scenario, the inputs for an experiment are *a list of mutations for a given gene of an HIV virus*, and the *name of a ruleset* (chosen out of a few available) to be used to compute drug rankings. A drug ranking tool is executed to compute the drug rankings which are returned as the experiment's results.

We will consider an extended Drug Ranking scenario which contains the following stages:

Experiment Input: HIV virus isolate nucleotide sequence

Stage 1. Align input nucleotide sequence. Result: genes responsible for particular HIV proteins.

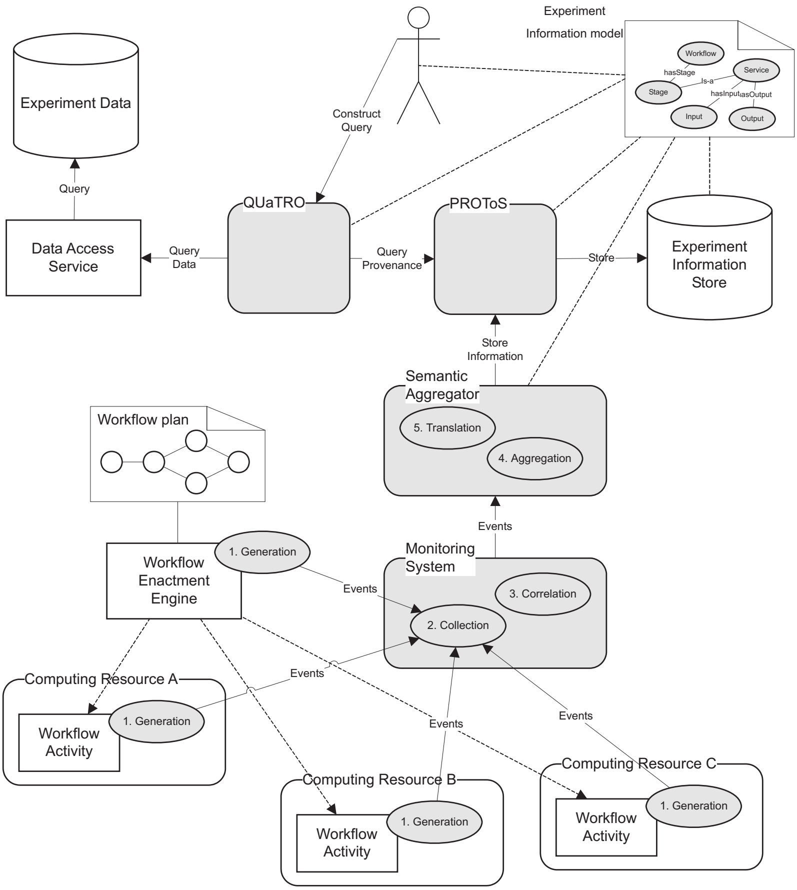


Fig. 2. Provenance tracking and querying in ViroLab

Stage 2. Compare obtained sequences to reference strains in order to determine mutations.
Stage 3. Apply selected rulestes to mutations to determin drug rankings.

In the course of provenance tracking of this simple scenario, multiple events are generated, collected and aggregated into the ontology representation of the experiment's execution. The events include those related to generic experiment enactment (experiment started/ended, operation started/ended, etc.). In addition, *domain-specific* events are generated to enhance the generic information about experiment execution with semantics related to the particular domain. For example, the three steps of the described scenario will generate respective domain events – 'Sequence Alignment', 'Computation of Mutations' and 'New Drug Ranking'. In consequence, appropriate information will be created to denote that the invoked operations were in fact sequence alignments, mutation computations, or drug ranking computations. Also, the input data items will be identified as a nucleotide sequence, a protein sequence, or a mutation list. Those semantic enhancements are achieved by linking generic ontology individuals (e.g. 'Operation Invocation') to domain ontology individuals (e.g. 'New Drug Ranking').

4 Query Capabilities of the Provenance Model

In [8], a number of challenging provenance questions have been formulated. Following were a few articles that attempted to answer those questions on the grounds of several provenance models, e.g. [10] [14]. In this section, we shall define a similar set of questions for our example DRS workflow and demonstrate how they can be answered in our provenance model. The questions are as follows:

Q1 Find the process that led to a particular drug ranking.
Q2 Find all operations performed after the alignment stage that led to a particular drug ranking.
Q3 Find operations that led to a particular drug ranking and were performed as 1st and 2nd ones.
Q4 Find all alignment operations performed on 10.10.2007 which operated on a given nucleotide sequence.

Question Q1 is a basic provenance question which returns a derivation path for a given data result. XQuery implementations of those queries over our provenance model are shown below.
Q1:

```
declare function local:variable-proces($varId as xs:string)
as list {
  for $goi in //GridOperationInvocation
      $dal in //DataAccessLoad
  where $goi//outData contains $varId
  return
    {
      for $input in $goi//inData
      return
        local:variable-proces($input)
    }
    $goi
  where $dal//variableId eq $varId
  return
    local:dataEntity-process($dal//dasId)
    $dal
};

declare function local:dataEntity-proces($dasId as xs:string)
as element {
  for $das in //DataAccessSave
  where $das//dasId eq $dasId
  return
    <process>
```

```
      local:variable-proces($das//variableId)
      $das
    </process>
};

<provenance>
  local:dataEntity-proces({dasId as passed})
</provenance>
```

Q2:

```
<provenance>
for $goi in //GridOperationInvocation,
  $exp in //Experiment
  where
    $exp/outData@[name()='rdf:resource'
    and . eq {drug ranking id}]
    and $exp/stageContext/@rdf:resource = $goi/@rdf:Resource
    and $goi/stageNumber > 1
  return $goi
</provenance>
```

Q3:

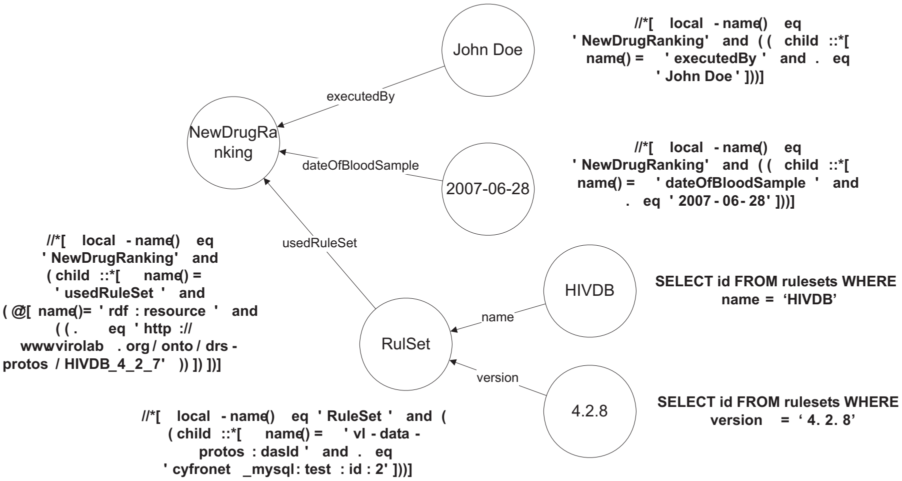
```
<provenance>
for $goi in //GridOperationInvocation,
  $exp in //Experiment
  where
    $exp/outData@[name() = 'rdf:resource'
    and . eq {drug ranking id}]
    and $exp/stageContext/@rdf:resource = $goi/@rdf:Resource
    and $goi/stageNumber in {1, 2}
  return $goi
</provenance>
```

Q4:

```
<provenance>
for $goi in //GridOperationInvocation,
  $exp in //Experiment
  where
    $exp/time eq '10.10.2007'
    and $goi/inData@[name() = 'vl-data-protos:dasId'
    and . eq {nucleotide sequence id}]
    and $exp/stageContext/@rdf:resource = $goi/@rdf:Resource
    and $goi/stageNumber = 1
  return $goi
</provenance>
```

5 Ontology-Based Query Construction

On top of the provenance tracking system we have built QUery TRanslation tOols (QUaTRO, Fig. 2), which enable end-user oriented approach to querying both repositories of provenance (through PROToS), and experiment data (through external Data Access Service). Both query construction in QUaTRO and provenance representation in PROToS are based on the ontology model of Experiment Information.



```
//*[ local - name() eq
' NewDrugRanking' and ( ( child ::*[
name() = ' executedBy ' and . eq
' John Doe ' ]))]
```

```
//*[ local - name() eq
' NewDrugRanking' and ( ( child ::*[
name() = ' dateOfBloodSample ' and
. eq ' 2007 - 06- 28' ]))]
```

```
//*[ local - name() eq
' NewDrugRanking' and
( child ::*[ name() =
' usedRuleSet ' and
( @[ name()= ' rdf : resource ' and
( ( . eq ' http ://
wwwvirolab . org / onto / drs -
protos / HIVDB_4_2_7' )) ]) ])]
```

```
SELECT id FROM rulesets WHERE
name = 'HIVDB'
```

```
SELECT id FROM rulesets WHERE
version = ' 4. 2. 8'
```

```
//*[ local - name() eq ' RuleSet ' and (
( child ::*[ name() = ' vl - data -
protos : dasId ' and . eq
' cyfronet _mysql : test : id : 2' ]))]
```

Fig. 3. Sample query tree and its evaluation

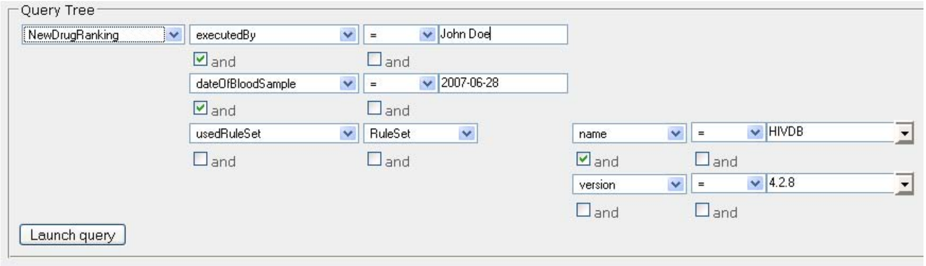
QUaTRO provides an easy-to-use graphical user interface. Query construction in QUaTRO amounts to building a query tree which begins with selecting a starting concept which determines the area of interest (for example 'New Drug Ranking', 'Experiment', 'Nucleotide Sequence', 'Ruleset'). The query form automatically expands and allows to follow properties of the starting concept which can lead to (1) other concepts (in which case the tree can be further expanded), (2) literal values, or (3) database mappings, both (2) and (3) becoming query tree leaves. When the query tree construction is completed, the tree is analyzed in a bottom-up fashion, starting from its leaves and following the edges up to the tree root. Subqueries represented in particular subtrees are evaluated and merged in upper nodes, if feasible. In principle, some subtrees might generate queries to relational databases of experiment data, while others – XQuery queries to the provenance repository. This approach enables queries such as *Select all experiments of type 'Drug ranking' whose results contained recommendation for 'Drug X'*. The provenance repository allows to select all experiments of type 'Drug ranking' which resulted in *some* drug recommendations. However, the actual results containing detailed descriptions of what drugs were recommended, are not part of provenance, but of actual experiment data which is stored in a database. Only a combined request to provenance repository and data repository will allow to construct queries of this kind.

Let us consider the following example provenance query: *Select all computations of drug rankings performed by 'John Doe' for blood samples taken on Jun-28-2007, and in which rule set 'HIVDB' in version 4.2.8 was used*. The constructed query tree is shown in Fig. 3, and the corresponding QUaTRO tool web form is presented in

Fig. 4. Query tree nodes denote either concepts ($NewDrugRanking$, $RuleSet$), or literals ($JohnDoe$, $2007 - 06 - 28$, $HIVDB$, 4.2.8). The tree edges are one of the following:

- Object properties which link to another concept ($usedRuleSet$);
- Datatype properties which link to a literal of a given datatype ($executedBy$, $dateOfBloodSample$);
- Database properties which denote that a literal they link to is actually stored in a database (e.g. in table T, column C) ($name$, $version$).

The query is evaluated as shown in Fig. 3. For database values, appropriate SQL queries are constructed. In this case, they return the IDs of rule sets which are compared to IDs stored in the provenance repository (as a property of the $RuleSet$ individuals). This allows us to pick exactly those $RuleSet$ individuals which satisfy the given criteria. Other subtrees are evaluated as XQuery requests to provenance repository and merged in upper-level nodes to minimize the total number of requests.



NewDrugRanking | executedBy | = | John Doe
☑ and ☐ and
dateOfBloodSample | = | 2007-06-28
☑ and ☐ and
usedRuleSet | RuleSet | name | = | HIVDB
☐ and ☐ and ☑ and ☐ and
version | = | 4.2.8
☐ and ☐ and

Fig. 4. Query construction in QUaTRO GUI

6 Conclusion and Future Work

We have presented an approach to tracking and querying provenance in which end-users are enabled to construct complex queries in a simple way. Ontologies as a model for provenance proved to be an adequate inter-lingua between: (1) *end users* who use ontologies as a query language while interacting with a graphical user interface; (2) *query tools* using the ontologies to represent and evaluate queries; (3) *provenance repository* in which the ontologies serve as the data model.

A unique feature of our approach is the support for subrequests to databases within a query tree. This enables even more detailed queries in which the structure of data items (not only provenance) is explored [3]. As a matter of fact, QUaTRO tools actually can be used to construct pure SQL-based queries thanks to mappings in the data ontology. This feature is also very useful, since the domain researchers often need to build complex queries to extract interesting input data for in silico experiments.

Currently, prototypes of PROToS and QUaTRO tools are implemented. They allow to record provenance and to construct queries in the described ‘wizard-based’ manner, starting from an ontology concept and following its properties to build the query tree.

Future work includes the implementation of distributed architecture of PROToS in order to ensure efficient querying. Most importantly, however, we plan several enhancements related to querying capabilities:

- We plan to extend our ontology with reverse relationships. This would allow to issue queries not only in the one-to-many but also in the many-to-one direction. For example, currently one can only query about Experiment which has ExecutionStage, but cannot query directly about ExecutionStages which are part of Experiment.
- Additional operators need to be added to QUaTRO, for example the logical or, and the in operator denoting that an attribute may have one value from a set thereof.
- Currently, an attribute can be compared only to literal values, but not to the evaluated values of other attributes. We plan to add this enhancement by allowing to explicitly create subqueries within a query, so that attributes could be compared against the result of the evaluated subquery (provided that data types would match).

Acknowledgments. This work was supported by EU project Virolab IST-027446 with the related Polish grant SPUB-M and by the Foundation for Polish Science.

References

1. Balis, B., Bubak, M., Pelczar, M.: From Monitoring Data to Experiment Information – Monitoring of Grid Scientific Workflows. In: Fox, G., Chiu, K., Buyya, R. (eds.) Third IEEE International Conference on e-Science and Grid Computing, e-Science 2007, Bangalore, India, December 10-13, 2007, pp. 187–194. IEEE Computer Society, Los Alamitos (2007)
2. Balis, B., Bubak, M., Wach, J.: Provenance Tracking in the Virolab Virtual Laboratory. In: PPAM 2007. LNCS, Springer, Gdansk, Poland (in Print, 2008)
3. Balis, B., Bubak, M., Wach, J.: User-Oriented Querying over Repositories of Data and Provenance. In: Fox, G., Chiu, K., Buyya, R. (eds.) Third IEEE International Conference on e-Science and Grid Computing, e-Science 2007, Bangalore, India, December 10-13, 2007, pp. 77–84. IEEE Computer Society, Los Alamitos (2007)
4. Bowers, S., McPhillips, T.M., Ludäscher, B., Cohen, S., Davidson, S.B.: A Model for User-Oriented Data Provenance in Pipelined Scientific Workflows. In: Moreau, L., Foster, I. (eds.) IPAW 2006. LNCS, vol. 4145, pp. 133–147. Springer, Heidelberg (2006)
5. Goble, C., Roure, D.D., Shadbolt, N.R., Fernandes, A.A.A.: Enhancing Services and Applications with Knowledge and Semantics. In: Foster, I., Kesselman, C. (eds.) The Grid 2: Blueprint for a New Computing Infrastructure, ch. 23, pp. 432–458. Morgan Kaufmann Publishers, San Francisco (2004)
6. Groth, P., Jiang, S., Miles, S., Munroe, S., Tan, V., Tsasakou, S., Moreau, L.: An Architecture for Provenance Systems, University of Southampton, Tech. Rep. (2006)
7. Gubala, T., Balis, B., Malawski, M., Kasztelnik, M., Nowakowski, P., Assel, M., Harezlak, D., Bartynski, T., Kocot, J., Ciepiela, E., Krol, D., Wach, J., Pelczar, M., Funika, W., Bubak, M.: ViroLab Virtual Laboratory. In: Proc. Cracow Grid Workshop 2007. ACC CYFRONET AGH (2008)
8. Moreau, L., et al.: The first provenance challenge. Concurrency and Computation: Practice and Experience 20(5), 409–418 (2007)
9. De Roure, D., Jennings, N., Shadbolt, N.: The Semantic Grid: A future e-Science infrastructure. In: Berman, F., Fox, G., Hey, A.J.G. (eds.) Grid Computing – Making the Global Infrastructure a Reality, pp. 437–470. John Wiley and Sons, Chichester (2003)

10. Simmhan, Y.L., Plale, B., Gannon, D.: Query capabilities of the Karma provenance framework. Concurrency and Computation: Practice and Experience 20(5), 441–451 (2007)
11. Sloot, P.M., Tirado-Ramos, A., Altintas, I., Bubak, M., Boucher, C.: From Molecule to Man: Decision Support in Individualized E-Health. Computer 39(11), 40–46 (2006)
12. Stankovski, V., Swain, M., Kravtsov, V., Niessen, T., Wegener, D., Kindermann, J., Dubitzky, W.: Grid-enabling data mining applications with DataMiningGrid: An architectural perspective. Future Generation Computer Systems 24(4), 259–279 (2008)
13. Zhao, Y., Wilde, M., Foster, I.T.: Applying the virtual data provenance model. In: Moreau, L., Foster, I. (eds.) IPAW 2006. LNCS, vol. 4145, pp. 148–161. Springer, Heidelberg (2006)
14. Zhao, J., Goble, C., Stevens, R., Turi, D.: Mining Taverna's Semantic Web of Provenance. Concurrency and Computation: Practice and Experience 20(5), 463–472 (2007)
15. Zhao, J., Wroe, C., Goble, C.A., Stevens, R., Quan, D., Greenwood, R.M.: Using Semantic Web Technologies for Representing E-science Provenance. In: McIlraith, S.A., Plexousakis, D., van Harmelen, F. (eds.) ISWC 2004. LNCS, vol. 3298, pp. 92–106. Springer, Heidelberg (2004)