Representing Interoperable Provenance Descriptions for ETL Workflows

André Freitas[1], Benedikt Kämpgen[2], João Gabriel Oliveira[1], Seán O'Riain[1], and Edward Curry[1]

[1]Digital Enterprise Research Institute (DERI)
National University of Ireland, Galway
[2]Institute AIFB
Karlsruhe Institute of Technology

Abstract. The increasing availability of data on the Web provided by the emergence of Web 2.0 applications and, more recently by Linked Data, brought additional complexity to data management tasks, where the number of available data sources and their associated heterogeneity drastically increases. In this scenario, where data is reused and repurposed on a new scale, the pattern expressed as Extract-Transform-Load (ETL) emerges as a fundamental and recurrent process for both producers and consumers of data on the Web. In addition to ETL, *provenance*, the representation of source artifacts, processes and agents behind data, becomes another cornerstone element for Web data management, playing a fundamental role in data quality assessment, data semantics and facilitating the reproducibility of data transformation processes. This paper proposes the convergence of this two Web data management concerns, introducing a principled provenance model for ETL processes in the form of a vocabulary based on the Open Provenance Model (OPM) standard and focusing on the provision of an interoperable provenance model for Web-based ETL environments. The proposed ETL provenance model is instantiated in a real-world sustainability reporting scenario.

Keywords: ETL, Data Transformation, Provenance, Linked Data, Web

1 Introduction

Extract-Transform-Load (ETL) is a fundamental process in data management environments. In Data Warehousing, data preprocessing is crucial for reliable analysis, e.g., reporting and OLAP; data coming from large databases or data derived using complex machine-learning algorithms may hide errors created in an earlier step of the analysis process. As a result, the design of ETL processes such as retrieving the data from distributed sources, cleaning it from outliers, and loading it in a consistent data warehouse demands up to 80 percent of data analysts' time [12].

The growing availability of data on the Web provided by Web 2.0 applications and, more recently through Linked Data, brought the computational pattern expressed as ETL to reemerge in a scenario with additional complexity, where the number of data sources and the data heterogeneity that needs to be supported by ETL drastically increases. In this scenario, issues with data quality and trustworthiness may strongly impact the data utility for end-users. The barriers involved in building an ETL infrastructure under the complexity and scale of the available Web-based data supply scenario, demands the definition of strategies which can provide data quality warranties and also minimize the effort associated with data management. In this context, *provenance*, the representation of *artifacts*, *processes* and *agents* behind a piece of information, becomes a fundamental element of the data infrastructure.

However, in an environment where data is produced and consumed by different systems, the representation of provenance should be made interoperable across systems. Interoperability represents the process of sharing the semantics of the provenance representation among different contexts. Although some systems in the areas of data transformation [1] and databases [17] provide history information about information pieces, those descriptions cannot be easily shared or integrated. *Provenance* and *interoperability* walk together: provenance becomes fundamental when the borders of a specific system or dataset are crossed, where a representation of a workflow abstraction of the computational processes can enable reproducibility, improve data semantics and restore data trustworthiness. Ultimately, provenance can make the computational processes behind applications interpretable at a certain level by external systems and users.

Standardization efforts towards the convergence into a common provenance model generated the Open Provenance Model [11] (OPM). OPM provides a basic description of provenance which allows interoperability on the level of workflow structure. The definition of this common provenance ground allows systems with different provenance representations to share at least a workflow-level semantics (the causal dependencies between artifacts, processes and the intervention of agents). OPM, however, is not intended to be a complete provenance model, but demands the complementary use of additional provenance models in order to enable uses of provenance which requires higher level of semantic interoperability. The explicit trade-off between the semantic completeness of a provenance model and its level of interoperability imposes challenges in specifying a provenance model.

In the context of ETL, provenance descriptions have a large spectrum of applications [13] including *documentation & reproducibility* and *data quality assessment & trustworthiness* and *consistency-checking & semantic reconciliation.* This paper focuses on the provision of a solution that allows the improvement of the semantic completeness and interoperability for provenance descriptors in complex data transformation/ETL scenarios. To achieve this goal, a vocabulary focused on modeling ETL workflows is proposed. The proposed model is built upon the workflow structure of OPM, being designed to extend the basic semantics and structure of OPM-based provenance workflows. In this work the ETL acronym is used in a broader context, focusing on generic data transformation patterns, transcending the original Data Warehouse associated sense. The contributions of this work are summarized in the following: **(i)** proposal of a set of requirements for a provenance model for ETL workflows, **(ii)** provision of a Linked Data ETL vocabulary, **(iii)** application of the proposed model in a real-world ETL scenario.

The paper is organized as follows: section 2 analyses related work on the representation and formalization of ETL provenance workflows; section 3 provides a list of requirements for an ETL provenance model; section 4 describes the construction of the ETL provenance model, describing *Cogs*, a provenance vocabulary for ETL. Section 5 describes the application of the ETL vocabulary in a case study for sustainable reporting. Section 6 finally provides conclusions and future work.

2 Related Work: Representing ETL workflows

Previous literature analyzed and formalized conceptual models for ETL activities. In the center of these models is the concept of *data transformations*. This section describes previous data transformation models, analyzing existing models regarding their suitability as interoperable provenance descriptions. Existing works can be classified in two major perspectives: *ETL Conceptual Models*, which focus on the investigation of ontologies and serialization formats for the ETL domain and *ETL Formal Models*, which concentrate on more abstract mathematical descriptions of the ETL domain.

2.1 ETL Conceptual Models

The Common Warehouse Metamodel (CWM) is an open OMG standard for data warehousing which defines a metadata model and an XML-based exchange standard. In CWM, a *transformation* is a basic unit in the ETL process which can be combined into a set of tasks. In addition, a transformation can be used in multiple transformation tasks. A transformation step is formed by combining multiple transformation tasks. Duong & Thi [15] propose a CWM compliant approach over an ontology-based foundation for modeling ETL processes for data from distributed and heterogeneous sources; the approach does not model the types of transformations explicitly but only provides a basic mapping infrastructure which can be used to reference external classes. Also, the approach lacks a concrete use case where its benefits are demonstrated.

Akkaoui & Zimani [5] propose a conceptual language for modeling ETL workflows based on Business Process Model Notation (BPMN). A set of BPMN constructs is used to define an ETL model. The artificiality of the solution lies on the fact that BPMN is not intended to be a universal data representation format, bringing questions on its suitability as an interoperable representation.

Becker & Ghedini [2] describe how to document Data Mining projects where descriptors are manually captured by the analysts. Their model includes tasks as an abstraction for various preprocessing activities, distinguishing between task definitions and task executions. Tasks can be annotated using free text and can be tagged using predefined concepts. Kietz et al. [9] introduce a cooperative planning approach for data mining workflows using the support of a data mining ontology (DMO). The DMO contains some similar concepts to the Cogs vocabulary. A difference is the fact that the DMO was not designed targeting a provenance representation, added to the fact that it focuses more on the data mining aspects, while this work targets data transformation operations.

2.2 ETL Formal Models

Cui & Widom [3] formalize the lineage problem on data warehouse environments proposing algorithms for lineage tracing. The authors define transformations as any procedure which takes datasets as inputs and produces datasets as outputs. Cui & Widom [3] define three transformation classes: dispatchers, aggregators and black-boxes. Vassiliadis et al. [16] investigate generic properties present in ETL activities across different ETL implementations. These properties build the base for the construction of a taxonomy. Vassiliadis et al. and Skoutas & Simitsis [16, 14] use a categorization of operations from different systems to build the type structure of the taxonomy. The classes present in their proposed taxonomy are designed to automatically capture the relationship between input and output definitions.

Davidson et al. [4] analyse the requirements for the construction of a formalism for modeling the semantics of database transformations and propose a declarative language for specifying and implementing database transformations and constraints. The motivation of their work is to generate a transformation formalism which can be used to verify the correctness of transformations.

Additionally, high-level declarative languages for data transformations have been defined (Galhardas et al. [8]). They propose logical specifications and physical implementations and describe the reasoning behind transformations, however, the solution is overly formal to be used as interoperable ETL descriptions.

We have identified a gap regarding the representation of provenance for ETL workflows. Previous literature has either presented models with limited interoperability (supported by a poor representation), or presented highly formalised models. Most previous work on describing ETL workflows lacks the potential of creating interoperable representations. The major part of ETL applications such as Kapow Software, Pentaho Data Integration, and Yahoo Pipes either do not create and use provenance information or do not allow to share and integrate such provenance data with other solutions.

3 Requirements

This section defines a list of requirements which summarizes the core usability and model characteristics that should be present on an ETL provenance model. The requirements are defined to satisfy two core demands which were found as gaps on the ETL literature (i) lack of a provenance representation from an ETL perspective and (ii) semantic interoperability across different ETL platforms and an additional (iii) usability demand (i.e. how easy is the instantiation of a correct and consistent model). The requirements are described below.

1. *Prospective and retrospective descriptions*: Provenance descriptors can represent workflows which were already executed (*retrospective provenance*) or workflow specifications (*prospective provenance*). Impacts: i, ii and iii.
2. *Separation of concerns*: ETL-specific elements should be separated from the provenance workflow structure, allowing at least a minimum level of interoperability between ETL and non-ETL provenance descriptors. This requirement is aligned with the OPM [11] compatibility. Impacts: ii.
3. *Terminological completeness*: Maximization of the terminological completeness of the provenance descriptor. Large terminological coverage of ETL elements. Impacts: i and ii.
4. *Common terminology*: Descriptors should allow the common denominator of representations of ETL elements. Ability to map elements present in different ETL platforms. Impacts: i and ii.
5. *Lightweight ontology structure*: Models with complex structures bring barriers for the instantiation and consumption of the model, including consistency problems, scalability issues, interpretability problems and additional effort in the instantiation of the model. The proposed provenance model should minimize these problems by providing a lightweight provenance model. Impacts: iii.
6. *Availability of different abstraction levels*: The vocabulary should allow users to express multiple abstraction levels for both processes and artifacts, varying from fine grained to coarse grained descriptions. Users should be able to express multiple levels of abstraction simultaneously. This requirements is also present in the OPM specification [11]. Impacts: ii and iii.
7. *Decentralization*: ETL provenance descriptors may be deployed on distributed database platforms without requiring cooperation among all databases. Impacts: ii and iii.
8. *Data representation independency*: Descriptors should be possible to refer to any data representation format including relational, XML, text files, etc. Impacts: iii.
9. *Accessibility*: The generated provenance descriptors should be easily accessible to data consumers. Data consumers should be able to query and automatically process provenance descriptors. Impacts: ii and iii.

4 Provenance Model for ETL Workflows

4.1 Multi-layered provenance model

The following approach was used to provide an ETL provenance descriptor addressing the requirements: (i) use of the Linked Data standards for representing provenance descriptors, (ii) construction of the provenance model from the OPM Vocabulary (OPMV) workflow structure, (iii) construction of an additional hierarchical workflow structure allowing the representation of nested workflows (complementing OPMV), (iv) design a complementary vocabulary for expressing the elements present in an ETL workflow and (v) make the solution extensible to describe domain specific objects. The following paragraphs describe the construction of the provenance model.

This paper uses a three-layered approach to represent provenance (depicted in Figure 1) as a principled way to provide interoperable provenance representations of ETL and generic data transformation workflows. OPM is a technology agnostic specification: it can be implemented using different representations or serializations. This work uses the OPM Vocabulary (OPMV) as the representation of OPM. In this representation, the bottom layer represents the basic workflow semantics and structure provided by OPMV, the second layer represents the common data extraction, transformation and loading entities and the third layer represents a domain specific layer.

The ETL provenance model layer is built upon the *basic workflow structure* of the OPMV layer. The ETL provenance model layer is designed to include a set of common entities present across different ETL workflows, providing a terminologically-rich provenance model instantiated as the *Cogs* vocabulary. The third layer consists of a domain specific layer which extends the second layer, consisting of domain-specific schema and instance-level information, e.g., of domain-specific source and target datasets or operations. For instance, e-Science operators from biological experiments would further specify classes of Cogs operators. This paper defines a conceptual model for the second layer and describes its interaction with the two complementary layers. The separation of the provenance model into the three-layered structure supports the requirement *(2) separation of concerns*.

4.2 Cogs: A Vocabulary for Representing ETL Provenance

In the construction of Cogs the core relationships are provided by *object properties* on the OPMV layer. The Cogs model specializes the core OPMV entities (Artifacts and Processes) with a rich taxonomic structure. The approach used in Cogs focuses on the design of a lightweight ontology (or vocabulary), which minimizes the use of logical features (such as transitive, inverse properties) and the consistency/scalability problems associated with the reasoning process (impacts requirement *(5) lightweight ontology structure*).

The methodology for building the Cogs vocabulary considered the following dimensions: (i) requirements analysis (ii) the core structural definition of modeling ETL workflows using the structure of OPMV workflows, (iii) an in depth analysis of concepts expressed in a set of analyzed ETL/data transformation tools (Pentaho Data Integration [1], Google Refine [2]) and (iv) concepts and structures identified from the ETL literature [3, 16, 10]. The core of the Cogs vocabulary captures typical operations, objects and concepts involved in ETL activities, at different phases of the workflow.

[1] http://kettle.pentaho.com.

[2] http://code.google.com/p/google-refine.

Cogs also extends the workflow structure of OPMV with additional object properties targeting the creation and navigation of hierarchical workflow structures. Hierarchical workflow structures allow the representation of both fine grained (important for machine interpretation and automated reproducibility) and coarse grained (important for human interpretation) provenance representation [6]. This features impacts both requirements *(6) availability of different abstraction levels and (1) prospective and retrospective descriptions.* Figure 1 depicts the core of the OPMV workflow model and the workflow extension of the Cogs vocabulary (with the cogs namespace).

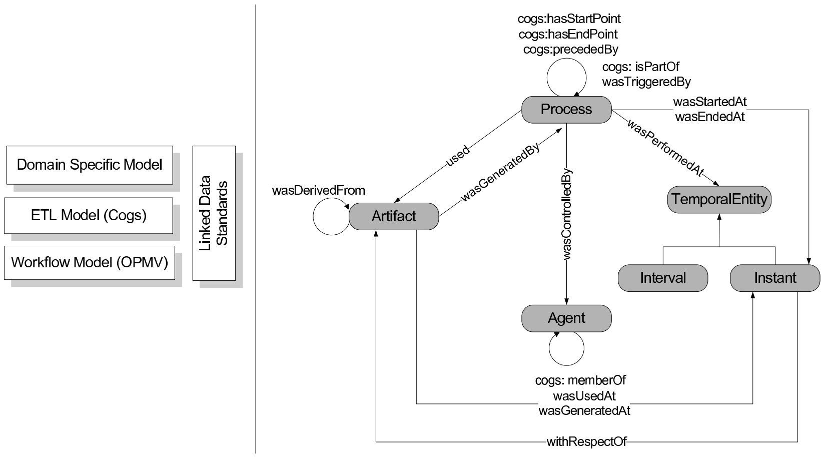

Fig. 1. OPMV workflow structure extended with additional Cogs properties.

The Cogs vocabulary defines a taxonomy of 151 classes. In addition, 15 object properties and 2 data properties are included in the vocabulary. The large number of classes allows a rich description of ETL elements supporting an expressive ETL representation (impacts requirements *(3) terminological completeness and (6) availability of different abstraction levels*). The classes, extracted from the ETL literature and from available tools covers the *terminological completeness* and *common terminology* requirements. The vocabulary taxonomy is structured with 8 high-level classes which are described below:

- *Execution*: Represents the execution job (instance) of an ETL workflow. Examples of subclasses include *AutomatedAdHocProcess* and *ScheduledJob*.
- *State*: Represents an observation of an indicator or status of one particular execution of an ETL process. These can range from execution states such as *Running* or *Success* to execution statistics, captured by the subclasses of the *PerformanceIndicator* class.
- *Extraction*: Represents operations of the first phase of the ETL process, which involves extracting data from different types of sources. *Parsing* is a subclass example. cogs:Extraction is an opmv:Process.

- *Transformation*: Represents operations in the transformation phase. Typically this is the phase which encompasses most of the semantics of the workflow, which is reflected on its number of subclasses. Examples of classes are *RegexFilter*, *DeleteColumn*, *SplitColumn*, *MergeRow*, *Trim* and *Round.* cogs:Transformation is an opmv:Process.
- *Loading*: Represents the operations of the last phase of the ETL process, when the data is loaded into the end target. Example classes are *ConstructiveMerge* and *IncrementalLoad.* cogs:Loading is an opmv:Process.
- *Object*: Represents the sources and the results of the operations on the ETL workflow. These classes, such as *ObjectReference*, *Cube* or *File*, aim to give a more precise definition of opmv:Artifact (every cogs:Object is an opmv:Artifact) and, together with the types of the operations that are generating and consuming them, capture the semantics of the workflow steps.
- *Layer*: Represents the different layers where the data can reside during the ETL process. *PresentationArea* and *StagingArea* are some of the subclasses.

In practice it is not always possible to capture all data transformation operations into a fine-grained provenance representation. One important feature of the Cogs vocabulary is the fact that program descriptions (i.e. source code) or executable code can be associated with the transformations. The *cogs:programUsed* property can be used to associate transformations to their logical or executable code. This feature impacts the requirements *(3) terminological completeness, (6) availability of different abstraction levels and (1) prospective and retrospective descriptions.*

The use of Linked Data principles strongly supports requirement *(10) accessibility* by allowing a unified standards-based access layer to data. In the proposed model, the standards-based provenance representation is separated from the database representation (a relational database record or an element inside an XML file can have its provenance information represented using Linked Data principles). The use of (provenance) URIs to associate provenance information to data items is a generic solution which can be directly implemented to every data representation format, supporting the requirement *(8) data representation independency.* Additionally, by using RDF(S), HTTP and URIs provenance can be persisted in a decentralized way (requirement *(7) decentralization*). Users can access provenance through SPARQL queries or by the navigation over dereferenceable URIs.

Table 1 summarizes the requirements coverage by the proposed provenance model. The current version of the Cogs vocabulary is available at `http://vocab.deri.ie/cogs`.

5 Vocabulary Instantiation

In order to analyze the suitability of the proposed vocabulary as a representation of ETL processes, we have implemented an instantiation of the Cogs vocabulary using as a case study a platform for collecting sustainability information at

Requirement	OPMV	Cogs	LD Principles
Prospective and retrospective descriptions	+	+	-
Separation of concerns	+	+	-
Terminological completeness	+	+	+
Common terminology	+	+	-
Lightweight ontology structure	+	+	-
Availability of different abstraction levels	-	+	-
Decentralization	-	-	+
Data representation independency	+	+	+
Accessibility	+	-	+

Table 1. Requirements coverage of each element of the provenance model: '+' represents an effective impact on the requirements dimension while '-' represents the lack of impact.

the Digital Enterprise Research Institute (DERI). The organization-wide nature of sustainability indicators, reflecting the organizational environmental impact, means that potential information is scattered across the organization within numerous existing systems. Since existing systems were not designed from the start to support sustainability analysis, heterogeneous data present in distributed sources need to be transformed into sustainability indicators following an ETL process. The correctness and consistency of each sustainability KPI needs to be auditable through the publication of the associated provenance information, which should be interpretable by different stakeholders.

The ETL process for the construction of sustainability indicators consists of four workflows (printing emissions, paper usage, travel emissions and commute emissions). Data sources include RDF graphs for people, research units and different file formats containing raw data. The basic ETL workflow consists in a sequence of operations: file selection, filtering, transformation, CO2 emissions calculation and transformation into RDF conforming to the RDF Data Cube vocabulary. On the last step information in the data cubes is aggregated to generate a final report available on the Web. The ETL workflow is implemented in Java code.

To make the ETL workflow provenance-aware, the Prov4J-Light framework was used, a lightweight version of [7], which is a Java framework for provenance management, that uses Semantic Web tools and standards to address the core challenges for capturing and consuming provenance information in generic Java-based applications. Java objects are mapped to artifacts and processes in the OPMV + Cogs provenance model. The set of generated instances is persisted in a separate provenance dataset. The connection between the final data and its provenance descriptor is given by a provenance URI (provURI) which is a reflection of the annotated artifact in the provenance store, pointing to its associated retrospective provenance workflow. Each element in the provenance store is represented by a dereferenceable provenance URI. Applications and users can navigate through the workflow structure by following the graph links or by executing SPARQL queries.

The purpose of the workflow usage should be determined in advance, where coarse grained data transformation representations are more suitable for human consumption (in particular, in the determination of quality assessment) while fine grained representations provide a higher level of semantic interoperability which is more suitable for enabling automatic reproducibility. The proposed provenance model for ETL handle both granularity scenarios. For this case study, since the main goal is to provide a human auditable provenance trail, a coarse grained implementation was chosen. Figure 2 depicts a short excerpt of the workflow in the provenance visualization interface with both OPMV and Cogs descriptors. The user reach the provenance visualization interface by clicking in a value on an online financial report. Readers can navigate through a workflow descriptor for the printing CO2 emissions on the web[3]. The final average linear workflow size of the high-level workflow consisted 4 processes and 5 artifacts.

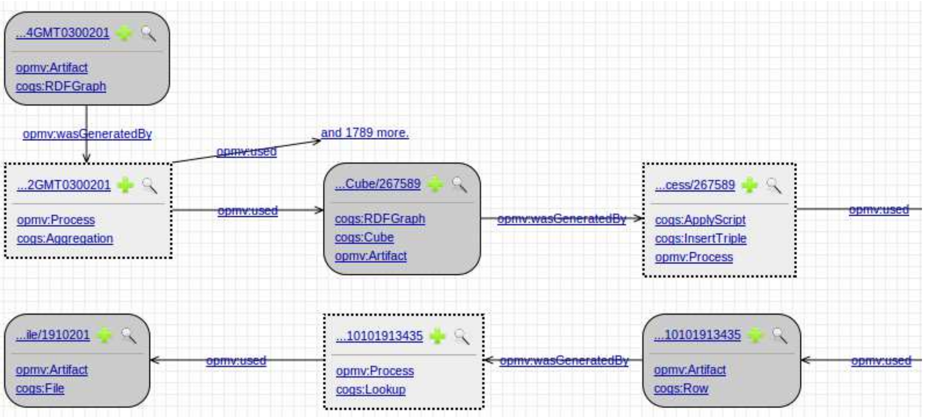

Fig. 2. Visualization interface for the retrospective provenance of the implemented ETL workflow.

One important aspect for a provenance model is the expressivity of the queries supported by it. The OPMV layer allows queries over the basic workflow structure behind the data, such as *What are the data artifacts, processes and agents behind this data value?*, *When were the processes executed?*. By adding a Cogs layer to the OPMV layer it is possible to define queries referring to specific classes within the ETL environment, such as *What are the RDF data sources used to generate this data value?*, *Which extractors are used in this workflow?*, *What are the schema transformation operations?*, *Which formulas were used to calculate this indicator?*, *Which is the source code artifact behind this data trans-*

[3] http://treo.deri.ie/cogs/example/swpm2012.htm

formation?. The third layer contains information which is domain-specific (not likely to be directly interoperable with other systems). It consists of specific operations (e.g., reference to specific data mining algorithms), schema-level information (such as table names and column names) and program code references (as in the example instantiation). The use of the Cogs vocabulary allows an increase of the query expressivity in relation to OPMV, allowing queries over the ETL elements. In addition to the direct interoperability increase provided by Cogs-compatible systems, the additional semantics of Cogs can facilitate knowledge discovery in provenance workflows, facilitating the inductive learning and semantic reconciliation of entities in the domain-specific layer.

Compared to previous works, the proposed provenance model focuses on providing a standards-based solution to the interoperability problem, relying on the structure of a community-driven provenance model (OPM) to build a provenance model for ETL. Linked Data standards are used for leveraging the accessibility of provenance descriptors. The proposed provenance model is able to provide a terminology-based semantic description of ETL workflows both in the prospective and retrospective provenance scenarios. The model is targeted towards a pay-as-you-go semantic interoperability scenario: the semantics of each workflow activity can be fully described using a fine grained mapping of each sub-operation present in each process of the ETL workflow.

6 Conclusion & Future Work

This work presented a provenance model for ETL workflows, introducing *Cogs*[4], a vocabulary for modeling ETL workflows based on the Open Provenance Model (OPM). The proposed vocabulary was built aiming towards the provision of a semantically interoperable provenance model for ETL environments. The vocabulary fills a representation gap of providing an ETL provenance model, a fundamental element for increasingly complex ETL environments. The construction of the vocabulary is based on the determination of a set of requirements for modeling provenance of ETL workflows. The proposed provenance model presents a high coverage of the set of requirements and was implemented in a realistic ETL workflow scenario. The model relies on the use of Linked Data standards. A more thorough evaluation of interoperability gained using Cogs is planned. Future work includes refining the vocabulary based on feedback from users and the implementation of the proposed provenance solution in an open source ETL platform.

7 Acknowledgments

The work presented in this paper has been funded by Science Foundation Ireland under Grant No. SFI/08/CE/I1380 (Lion-2) and by the German Ministry of Education and Research (BMBF) within the SMART project (Ref. 02WM0800)

[4] http://vocab.deri.ie/cogs

and the European Community's Seventh Framework Programme FP7/2007-2013 (PlanetData, Grant 257641).

References

1. M. Altinel, P. Brown, S. Cline, R. Kartha, E. Louie, V. Markl, L. Mau, Y.-H. Ng, D. Simmen, and A. Singh. Damia: a data mashup fabric for intranet applications. In *Proceedings of the 33rd international conference on Very large data bases*, 2007.
2. K. Becker and C. Ghedini. A documentation infrastructure for the management of data mining projects. *Information & Software Technology*, 47, 2005.
3. Y. Cui and J. Widom. Lineage tracing for general data warehouse transformations. *The VLDB Journal*, 12, 2003.
4. S. Davidson and P. Buneman. Semantics of database transformations. *Semantics in Databases*, 1998.
5. Z. El Akkaoui and E. Zimanyi. Defining ETL worfklows using BPMN and BPEL. In *Proceedings of the ACM twelfth international workshop on Data warehousing and OLAP*, DOLAP '09, pages 41–48, New York, NY, USA, 2009.
6. A. Freitas, T. Knap, S. O'Riain, and E. Curry. W3P: Building an OPM based provenance model for the web. *Future Generation Computer Systems*, 2010.
7. A. Freitas, S. A. Legendre, S. O'Riain, and E. Curry. Prov4J: A Semantic Web Framework for Generic Provenance Management. In *Second International Workshop on Role of Semantic Web in Provenance Management (SWPM 2010)*, 2010.
8. H. Galhardas, D. Florescu, D. Shasha, E. Simon, and C.-A. Saita. Declarative data cleaning: Language, model, and algorithms. In *Proceedings of the 27th International Conference on Very Large Data Bases*, 2001.
9. J.-U. Kietz, F. Serban, A. Bernstein, and S. Fischer. Towards cooperative planning of data mining workflows. In *Proc of the ECML/PKDD09 Workshop on Third Generation Data Mining(SoKD-09)*, 2009.
10. R. Kimball and J. Caserta. *The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleanin.* John Wiley & Sons, 2004.
11. L. Moreau. The open provenance model core specification (v1.1). *Future Gener. Comput. Syst.*, 27, 2011.
12. K. Morik and M. Scholz. The miningmart approach to knowledge discovery in databases. In *In Ning Zhong and Jiming Liu, editors, Intelligent Technologies for Information Analysis*, 2003.
13. Y. L. Simmhan, B. Plale, and D. Gannon. A survey of data provenance in e-science. *SIGMOD Rec.*, 34, 2005.
14. D. Skoutas and A. Simitsis. Designing ETL processes using semantic web technologies. In *Proceedings of the 9th ACM international workshop on Data warehousing and OLAP*, 2006.
15. A. Thi and B. T. Nguyen. A Semantic approach towards CWM-based ETL processes. In *Proceedings of I-SEMANTICS 08*, 2008.
16. P. Vassiliadis, A. Simitsis, and S. Skiadopoulos. Conceptual modeling for ETL processes. In *Proceedings of the 5th ACM international workshop on Data Warehousing and OLAP*, 2002.
17. J. Widom. Trio : A System for Integrated Management of Data , Accuracy , and Lineage. *Innovative Data Systems Research (CIDR 2005)*, 2005.