Published in final edited form as:
Sci Stat Database Manag. 2010 ; 6187: 461–470. doi:10.1007/978-3-642-13818-8_32.

Provenance Context Entity (PaCE): Scalable Provenance Tracking for Scientific RDF Data

Satya S. Sahoo[1], **Olivier Bodenreider**[2], **Pascal Hitzler**[1], **Amit Sheth**[1], and **Krishnaprasad Thirunarayan**[1]

[1] Kno.e.sis Center, Computer Science and Engineering Department, Wright State University, Dayton, OH, USA

[2] Lister Hill National Center for Biomedical Communications, National Library of Medicine,NIH, Bethesda, MD, USA

Abstract

The Resource Description Framework (RDF) format is being used by a large number of scientific applications to store and disseminate their datasets. The provenance information, describing the source or lineage of the datasets, is playing an increasingly significant role in ensuring data quality, computing trust value of the datasets, and ranking query results. Current provenance tracking approaches using the RDF reification vocabulary suffer from a number of known issues, including lack of formal semantics, use of blank nodes, and application-dependent interpretation of reified RDF triples. In this paper, we introduce a new approach called Provenance Context Entity (PaCE) that uses the notion of *provenance context* to create provenance-aware RDF triples. We also define the formal semantics of PaCE through a simple extension of the existing RDF(S) semantics that ensures compatibility of PaCE with existing Semantic Web tools and implementations. We have implemented the PaCE approach in the Biomedical Knowledge Repository (BKR) project at the US National Library of Medicine. The evaluations demonstrate a minimum of 49% reduction in total number of provenance-specific RDF triples generated using the PaCE approach as compared to RDF reification. In addition, performance for complex queries improves by three orders of magnitude and remains comparable to the RDF reification approach for simpler provenance queries.

Keywords

Provenance context entity; Biomedical knowledge repository; Context theory; RDF reification; Provenir ontology

1 Introduction

An increasing number of scientific applications are storing and disseminating their datasets using the Resource Description Framework (RDF) format [1] [2] [3]. The Biomedical Knowledge Repository (BKR) project at the U.S. National Library of Medicine is creating a

sahoo.2@wright.edu, pascal.hitzler@wright.edu, amit.sheth@wright.edu, t.k.prasad@wright.edu, obodenreider@mail.nih.gov

comprehensive repository of integrated biomedical data from a variety of sources such as biomedical literature, structured data sources (for example the NCBI Entrez system [4]), and terminological knowledge sources (for example, the Unified Medical Language System (UMLS) [5]) [6]. BKR represents the integrated information in RDF, for example, the RDF statement "lipoprotein→affects→inflammatory_cells"[1] was extracted by a text mining tool from a journal article (with PubMed identifier PMID: 17209178) and states that lipoprotein (denoted as "subject" of the RDF triple) affects (denoted as "property" of the triple) inflammatory_cells (denoted as the "object" of the triple).

In addition to the biomedical data, BKR also records and uses provenance metadata describing the history or lineage of the RDF statements. The provenance information identifies the source of an extracted RDF triple, temporal information (for example, the date of publication of a source article), version information for a database, and the confidence value associated with a triple (indicated by a text mining tool). The provenance information is essential in the BKR project to ensure the quality of data and associate trust value with an RDF triple. It has specific applications in the four services offered by the BKR namely, enhanced information retrieval (search based on the named relationship linking two entities), multi document summarization, question answering, and knowledge discovery.

The RDF reification vocabulary [7] has been traditionally used by Semantic Web applications to track provenance in RDF documents. The RDF reification vocabulary consists of the four terms rdf:Statement,[2] rdf:subject, rdf:predicate, and rdf:object. A variety of problems have been identified in the use of RDF reification vocabulary for provenance tracking in Semantic Web applications.

The RDF specification [8] states that the RDF formal semantics does not extend to the reification vocabulary, and the intended interpretation of an RDF document using reification is application dependent (i.e., it may vary across applications) [7]. Further, the RDF specification states that entailment rules do not hold between an RDF triple and its reification [8]. The use of blank nodes, which have no "global meaning" outside a particular RDF graph [8], makes it difficult to use reasoning [9] and increases the complexity of query patterns since the queries have to explicitly take into account an extra entity. In addition to the limited formal semantics, use of RDF reification approach leads to a disproportionate increase in the total size of the RDF document without corresponding enhancement in information content of the RDF document. This adversely affects the scalability of large projects, such as BKR, that track provenance of hundreds of millions of RDF triples. A detailed discussion of the limitations of RDF reification and related approaches such as RDF named graph is given in [10].

In this paper, we introduce a new approach for RDF provenance tracking called Provenance Context Entity (PaCE). PaCE is part of a broader framework for provenance management in scientific applications called PrOM [10].

[1] We use the courier new font to represent RDF and OWL statements.

[2] The rdf namespace represents the http://www.w3.org/1999/02/22-rdf-syntax-ns Internationalized Resource Identifier (IRI).

1.2 Contributions and Overview

The contributions of this paper are three-fold:

1. Define the PaCE approach to track provenance in RDF-based Semantic Web applications without use of reification vocabulary and blank nodes (Section 2),

2. Define the formal semantics of PaCE, using model theory, by extending the existing RDF and RDFS formal semantics to ensure compatibility with existing RDF tools and implementations (Section 2),

3. Demonstrate the practical feasibility of PaCE through implementation in the BKR project (Section 3), and evaluate the advantages of PaCE in terms of storage and query performance as compared to the RDF reification approach.

2 Foundations of Provenance Context Entity

The intuition for the PaCE approach is that the provenance associated with RDF statements provides the necessary contextual information for applications to interpret two RDF statements to be equivalent or distinct. Contexts as formal objects have long been used in Artificial Intelligence (AI) applications, such as Cyc [11] and also to a limited extent in the Semantic Web, to facilitate processing of information that do not have a global frame of reference [12]. A detailed discussion of the existing work in context theory is given in [10].

2.1 Provenance Context and RDF Generation

The contextual information in the BKR project consists of the provenance information about the source of an RDF statement, that is, the journal identifier or the UMLS identifier or the Entrez Gene identifier. In other words, this *provenance context* is a formal object instantiated in the form of set of concepts and relationships that capture the necessary contextual provenance information to enable application to correctly interpret RDF statements. Similar to the provenance context defined in the BKR project (Figure 1(a)), other Semantic Web applications can also define a relevant provenance context for interpreting their RDF dataset. For example, an application in the sensor domain can define its provenance context to consist of sensor used to collect data readings, the geographical location of the sensor, and the timestamp value associated with a data reading (Figure 1(b)). To formalize the notion of provenance context, we define it in terms of the foundational model of provenance called provenir ontology (Figure 1 (c)) [10]. The provenir ontology is a upper-level provenance ontology representing a minimum set of provenance concepts common across domains and is modeled using the description logic profile of the W3C Web Ontology Language (OWL-DL) [13].

The provenir ontology consists of three primary concepts of “data”, “agent” and “process” linked by ten relationships adapted from the upper-level Relation Ontology [14] (Figure 3 (c)). An application can define its provenance context either in terms of the provenir ontology or in terms of a domain-specific provenance ontology, which extends provenir ontology. The use of the provenir ontology to define a provenance context has several advantages including the flexibility to model domain-specific provenance at a fine level of granularity, while ensuring consistent modeling and the support for RDF and OWL inferencing [8].

The PaCE approach allows an application to decide the level of granularity in modeling provenance of an RDF triple. For example, Figure 2 illustrates the three possible implementations of the PaCE approach in the BKR project that create distinct RDF triples extracted from two separate journal articles (though they share the same S, P, and O). The first implementation (Figure 2 (a)) is an exhaustive approach and explicitly links the S, P, and O to the source journal article and the second implementation (Figure 2 (b)) is a minimalist approach that links only the S of a RDF triple to the source article. The second implementation, on the other hand, requires the application to make additional assumption while processing the RDF triples, that the whole triple is extracted from the same source as the source of S.

The third implementation (Figure 2(c)) takes an intermediate approach that creates two additional provenance-specific triples but requires the application to assume that the source of the O is the same as the S, and the P. The choice to associate explicit “derives_from” property with one particular RDF component (S or P or O) in the minimalist (Figure 2 (b)) and the intermediate (Figure 2(c)) is arbitrary and has minimal impact on the provenance tracking functionality of the application.

It is important to note that, in contrast with the reification approach, none of the three variants of the PaCE approach requires the use of RDF reification vocabulary or the use of blank nodes. Further, the reification approach creates a total of six RDF triples (Figure 1) for each RDF triple, while the exhaustive implementation of the PaCE approach creates a total of four triples for one RDF triple. Overall, the PaCE approach is an incremental and simple mechanism that does not define additional vocabulary or require changes to existing RDF data stores. We now introduce the model theoretic semantics of PaCE inferencing.

2.3 Model Theoretic Semantics of PaCE Inferencing

The primary motivating factor for defining the formal semantics of PaCE is to provide a way to determine the validity of the inferencing process for Semantic Web applications that use the PaCE approach to track provenance. The definition of the model-theoretic semantics of PaCE is a straightforward modification of the existing RDFS semantics and allows us to infer additional provenance information for triples by virtue of having similar source. Let provenance context pc of an RDF triple α (= (S, P, O)) be the common object of the predicate provenir:derives_from associated with the triple. We define an *RDFS-PaCE-interpretation* I of a vocabulary V to be an RDFS-interpretation of the vocabulary $V \cup V_{PaCE}$ that satisfies the following additional condition (meta-rule):

- For RDF triples $\alpha = (S_1,P_1,O_1)$ and $\beta = (S_2,P_2,O_2)$, (provenance-determined) predicates p and entities v, if $pc(\alpha) = pc(\beta)$

then $(S_1, p, v) = (S_2, p, v)$ and, $(P_1, p, v) = (P_2, p, v)$ and, $(O_1, p, v) = (O_2, p, v)$

Provenance-determined predicates and entities are specific to an application domain.

Furthermore, a graph R_1 PaCE-entails a graph R_2 if every *RDFS-PaCE-interpretation* that is a model of R_1 is also a model of R_2. To illustrate the PaCE inference process, we consider two RDF statements in the BKR project (Figure 3). Given that the two RDF statements have equal provenance contexts (PubMed identifier: PMID17209178) additional provenance information, such as the confidence score (formalized via provenance-related predicate has_confidence_value and value confidence_value_2), associated with one of the triples can be inferred for the other RDF triple (flow_cytometry→measures→interleukin-1_beta) denoted by dotted arrows in Figure 3. We note that PaCE-entailment is strictly stronger than RDFS-entailment in the sense that all inferences which can be drawn using simple, RDF, or RDFS-entailment are also PaCE entailments. This is a deliberately conservative step on top of the existing Semantic Web recommendations that enables PaCE to be compatible with existing OWL and RDF tools and applications, and also allows implementing the PaCE-semantics by making reference to RDF reasoners as black boxes. In the next section, we describe the implementation of the PaCE approach in the context of the BKR project.

3 Implementation and Evaluation

A practical challenge for implementing the PaCE approach in the BKR is to formulate an appropriate provenance context-based URI (URI_p) scheme that also conforms to best practices of creating URIs for the Semantic Web, including support for use of HTTP protocol [15]. The design principle of URI_p is to incorporate a “provenance context string” as the identifying reference of an entity and is a variation of the “reference by description” approach that uses a set of description to identify an entity [15]. The syntax for URI_p consists of the <base URI>, the <provenance context string>, and the <entity name>. This approach to create URIs for RDF entities also enables BKR (and other Semantic Web applications using the PaCE approach) to group together entities with the same provenance context. For example,

- http://mor.nlm.nih.gov/bkr/PUBMED_17209178/lipoprotein
- http://mor.nlm.nih.gov/bkr/PUBMED_17209178/affects
- http://mor.nlm.nih.gov/bkr/PUBMED_17209178/inflammatory_cells

are entities extracted from the same journal article. Using this URI scheme, RDF statements were generated for the original triples (extracted from the biomedical literature by a text-mining application or found in the UMLS Metathesaurus).

The base dataset (B) used in the evaluation comprises of 23,433,657 RDF triples extracted from two sources: the biomedical literature (PubMed) and the UMLS Metathesaurus. The open source Virtuoso RDF store version 06.00.3123 was used for the experiments running on a Dell 2950 server (Dual Xeon processor) with 8GB of memory. A total of 500,000 9kB buffers were allocated to Virtuoso RDF store.

3.1 Provenance-specific RDF Triples

To evaluate the number of provenance-specific RDF triples generated using the two approaches, we augment the base dataset B with provenance information representing the source information of each triple. For the PaCE approach, we create three datasets representing the exhaustive (E_PaCE), minimalist (M_PaCE), and intermediate (I_PaCE) approaches illustrated in Figure 2 (a), (b) and (c), respectively. For the RDF reification dataset (R), we use the standard method (presented in Section 1). Figure 3(b) shows that the reification approach requires twice as many RDF triples (~152 million) for the representation of provenance information compared to the E_PaCE approach (~89 million). This 49% difference between E_PaCE and R represents a significant reduction in storage requirements (~85 million fewer triples) for the BKR project. Analogously, the M_PaCE and I_PaCE approaches create 72% and 59% fewer provenance-specific triples compared to the reification approach.

3.2 Performance of Provenance Queries

We use four representative categories of provenance queries in the BKR project to evaluate the query performance on the four datasets (E_PaCE, M_PaCE, I_PaCE and Reification). We describe the *pattern* of the four queries and their significance in the BKR project:

Query Pattern 1: List all the RDF triples extracted from a given journal article (e.g., journal article identified by PMID17209178). This query is used to retrieve all the triples from a given source.

Query Pattern 2: List all the journal articles from which a given RDF triple was extracted (e.g., lipoprotein→affects→inflammatory_cells). This query identifies the source(s) of a given triple.

Query Pattern 3: Count the number of triples in each source (biomedical literature and UMLS Metathesaurus) for the therapeutic use (predicate = treats) of a given drug (e.g., Thalidomide). This complex query illustrates the use of the BKR as a knowledge base for a query answering application (e.g., which diseases are treated by a particular drug?).

Query Pattern 4: Count the number of journal articles published between two dates (e.f., 2000-01-01 and 2000-12-31) for a given triple (e.g., thalidomide → treats → multiple myeloma). This typical information retrieval query leverages the provenance information associated with each triple. A more complex version of this query is used in Section 3.3 for time series analysis.

We conducted the query performance evaluation in two phases. In the first phase the four queries are evaluated for fixed values, namely, the value underlined in the query description above. In the second phase, queries are evaluated using a larger set of values. The queries are expressed in SPARQL syntax, the RDF query language [16]. The queries are not listed in the paper due to space constraints and are available online along with the result set.[3] The numbers reported for the "fixed" value queries (first phase) are the average of last 5 of a total of 20 runs. The first phase of the evaluation starts with a "cold" cache for each query pattern. The results in Figure 4 demonstrate that query performance for PaCE is generally better than or similar to reification. As expected, M_PaCE generally performs better than I_PaCE, and I_PaCE better than E_PaCE. However, reification performs better than I_PaCE

[3] Query and result set available at: http://wiki.knoesis.org/index.php/ProvenanceContextEntity

for *Query 1* and better than both I_PaCE and E_PaCE for *Query 3*. *Query 4* is a complex query that uses the SPARQL FILTER to restrict publication dates to a particular range (January 1 to December 31, 2000). In this query, the query performance for E_PaCE is more than two orders of magnitude better than for R.

In the second phase of the evaluation, we aim to reflect the real-world requirements of the BKR project. Toward this end, each of the four query patterns is executed with different values, as if by different users. In practice, we use sets of 100 values for each query pattern. The resulting set of 100 queries is run 5 times (immediately following the first phase of evaluation for each dataset) and the average of the 100 queries for the last run is presented (Figure 5). The results confirm the trend seen in the first phase of evaluation, with the added observation that for *Query Pattern 3* the difference between E_PaCE and R has decreased (R no longer outperforms E_PaCE significantly). In contrast, for the complex *Query Pattern 4*, the query performance for E_PaCE has further improved and is more than three orders of magnitude better than for R. The second phase of evaluation also confirms that in a real-world scenario the query performance of PaCE is comparable to reification for simple provenance queries and significantly better for complex provenance queries. We now evaluate the query performance for an analytical query in the BKR project.

3.3 Application to Time Profiling of Scientific Results

An important objective of many applications and funding agencies is to understand the trend in research focused on a specific topic in biomedicine over a period of time. We extend the *Query Pattern 4* discussed in the previous section to define a query that collates the number of journal articles published over a period of 10 years for mentions of the therapeutic use of the drug Thalidomide over time. Figure 6 (a) shows a histogram created directly from the query results. The query performance is similar to what was observed for *Query Pattern 4*, that is, E_PaCE is three orders of magnitude faster than R (Figure 6(b)). This example query demonstrates the feasibility representing and exploiting provenance information in large triple stores serving real-world applications.

4 Conclusion

We show that that challenge of provenance tracking in RDF datasets can be effectively and efficiently addressed by using the PaCE approach in place of the RDF reification vocabulary. The PaCE approach uses the formal objects called provenance contexts that are defined in terms of the provenir upper-level provenance ontology to create provenance-aware RDF triple. The evaluations demonstrate that using the PaCE approach to create provenance-specific RDF triples not only reduces the number of triples by at least 49% but also improves the performance of complex provenance queries by three orders of magnitude.

Acknowledgments

This research was supported in part by the Intramural Research Program of the NIH, U.S. NLM and NIH RO1 Grant# 1R01HL087795-01A1. The authors would like to thank Tom Rindflesch, Marcelo Fiszman, Genaro Hernandez and Ramez Ghazzaoui for their extensive help. The open source version of the Virtuoso triple store is made available by OpenLink Software.

References

1. [Jan 10 2010] Protein knowledgebase: Uniprot. http://www.uniprot.org/
2. Kanehisa M, Goto S, Kawashima S, Okuno Y, Hattori M. The KEGG resources for deciphering the genome. Nucleic Acids Res. 2004; 32:D277–D280. [PubMed: 14681412]
3. [Jan 10 2010] Reactome. www.reactome.org/
4. [Jan 10 2010] Entrez. http://www.ncbi.nlm.nih.gov/Database/
5. Bodenreider O. The Unified Medical Language System (UMLS): integrating biomedical terminology. Nucleic Acids Res. 2004; 32:267–270.
6. Bodenreider, O.; Rindflesch, TC. Lister Hill National Center for Biomedical Communications. National Library of Medicine; Bethesda, Maryland: 2006. Advanced library services: Developing a biomedical knowledge repository to support advanced information management applications..
7. Manola F, Miller E. RDF Primer. W3C Recommendation. 2004
8. Hayes P. RDF Semantics. W3C Recommendation. 2004
9. ter Horst HJ. Completeness, decidability and complexity of entailment for RDF Schema and a semantic extension involving the OWL vocabulary. Journal of Web Semantics. 2005; 3:79–115.
10. Sahoo, SS.; Barga, RS.; Sheth, AP.; Thirunarayan, K.; Hitzler, P. PrOM: A Semantic Web Framework for Provenance Management in Science. Wright State University; Kno.e.sis Center: 2009.
11. Guha, RV. Contexts: A Formalization and Some Applications PhD Thesis. Stanford University; 1991.
12. Guha RV, McCarthy J. Varieties of Contexts. CONTEXT. 2003; 2003:164–177.
13. Hitzler P, Krötzsch M, Parsia B, Patel-Schneider PF, Rudolph S. OWL 2 Web Ontology Language Primer. W3C. 2009
14. Smith B, Ceusters W, Klagges B, Kohler J, Kumar A, Lomax J, Mungall C, Neuhaus F, Rector AL, Rosse C. Relations in biomedical ontologies. Genome Biol. 2005; 6:R46. [PubMed: 15892874]
15. Ayers A, Völkel M. Sauermann L, Cyganiak R. Cool URIs for the Semantic Web. Woking Draft. W3C. 2008
16. Prud'hommeaux E, Seaborne A. SPARQL Query Language for RDF. W3C Recommendation. 2008

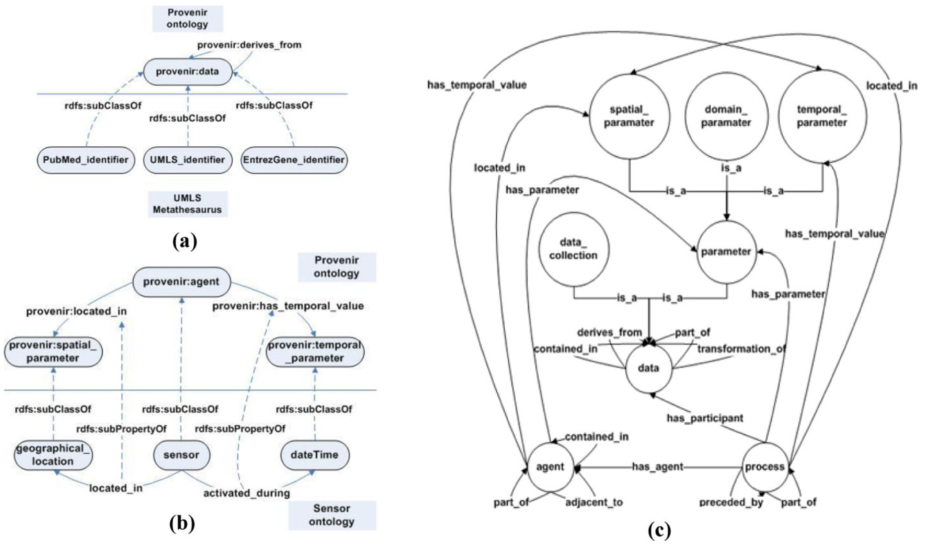

Figure 1.

(a) Representation of provenance context for the BKR project, (b) a sensor application, and (c) Provenir ontology schema

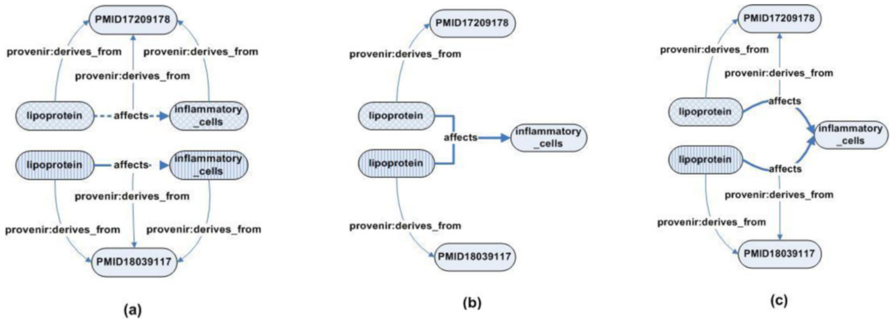

Figure 2.

Implementation of the PaCE mechanism to track provenance of RDF triples extracted from two journal articles

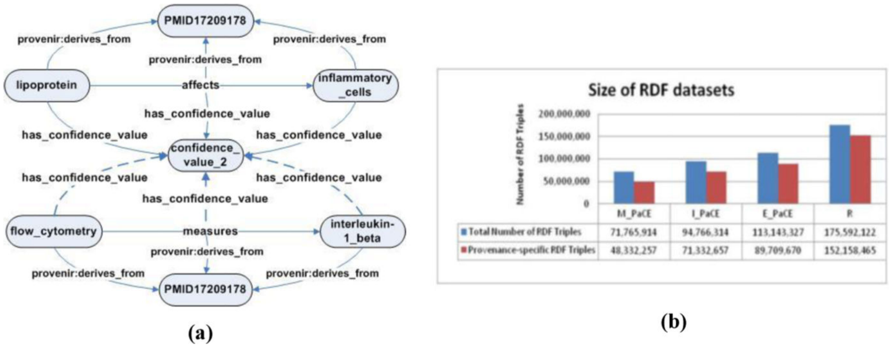

Figure 3.

(a) PaCE inferencing, (b) The relative number of provenance-specific triples created using PaCE and RDF reification

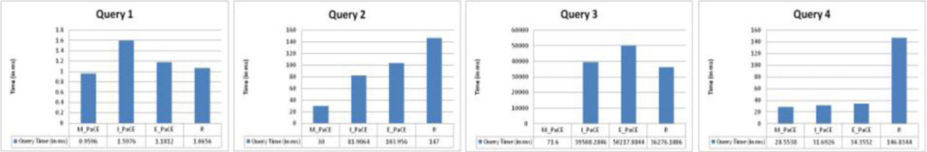

Figure 4.

Query performance for fixed values

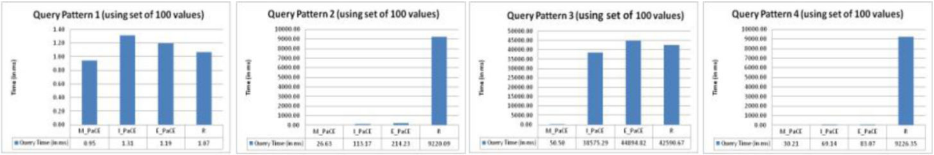

Figure 5.

Query performance for query patterns using a set of 100 values

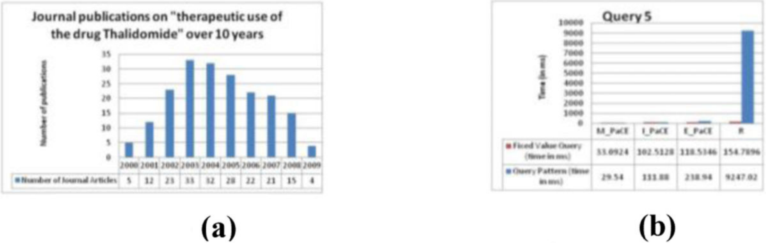
Journal publications on "therapeutic use of the drug Thalidomide" over 10 years
Number of publications
35
30
25
20
15
10
5
0
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
Number of Journal Articles
5
12
23
33
32
28
22
21
15
4
(a)
Query 5
Time (in ms)
10000
9000
8000
7000
6000
5000
4000
3000
2000
1000
0
M_PaCE
I_PaCE
E_PaCE
R
Fixed Value Query (time in ms)
33.0924
102.5128
118.5346
154.7896
Query Pattern (time in ms)
29.54
111.88
238.94
9247.02
(b)

Figure 6.