ISSN 2319-7595

International Journal of Information Systems and Computer Sciences, Vol.7. No.1, Pages : 08-15 (2017)
Special Issue of ICSIC 2017 - Held during 23-24 September 2017 in Amman Arab University, Amman-Jordan
http://www.warse.org/IJISCS/static/pdf/Issue/icsic2017sp15.pdf

A literature Survey on Conversion between Relational and XML Models

Husam Ahmed Al Hamad
Information Computer Systems Department
Amman Arab University, Amman, Jordan
hhamad@aau.edu.jo, hushamad@yahoo.com

ABSTRACT

Extensible Markup Language (XML) becomes widely used over the web to exchange and share the data, its operations and tags help to reduce memory, storage and processing of the data; these features and more were the reason behind rapid spread and adoption using of XML model by many companies. The main contribution of this work is to present a literature survey of different conversion techniques and methods between relational and XML databases models, as well as raising the awareness of these techniques and methods. We review the different researches approaches and techniques that developed for XML conversions. These techniques include but not limited to Document Type Definition (DTD), Document Object Model (DOM), clustering and matching, query languages Structured Query Language (SQL), XPath, XQuery, relational storage, relational catalog and other methods.

Key words: Extensible Markup Language, XML Conversion, XML Mapping, XML Database, Relational Model.

1. INTRODUCTION

Greater reliance on the use of the internet to exchange information between the integration applications and systems have caused the need of using a standard database model dealing with common data format [1] which are interoperability features represented by the web, one of these models that support a common data format is XML model.

XML is a database model focuses on what data, self-describing, it uses tags (elements) for describing the data itself [2], store its contents as plain text format by allows nested structures, this mechanism makes text and data much easier to understand, read, exchange, share, and interpret the data between incompatible programs and applications. It contains a set of strong capabilities to modeling of data and information such as flexibility, heterogeneity, and extensibility [3]. For these reasons, XML markup language becomes a common widely used standard data format in the databases and applications of organizations as well as a common semi-structured language for managing frequent storage and retrieval data over the internet [4]. This means XML document can interpreted in multiple ways, as well as filter, and restructure its contents in order to fit application needs [5].

XPath and XQuery are the common queries languages for XML documents to navigate through XML elements, attributes and contents. XPath uses path syntax such as file systems paths to select and navigate nodes in the XML document. XQuery supported by all major databases and built based on XPath expressions, it selects and extracts tags and attributes from XML documents.

The aim of this paper is to survey the different techniques and methods of conversion between relational and XML databases, in addition to presents a brief approach of each method that will be clarified.

2. CONVERSATION APPROACHES

This paper attempts to identify and categorizes all approaches and techniques that used for conversation between XML and relational databases, intended of this work is to give the researchers a general overview of what has been accomplished in literature regarding on this subject. This section presents six major transformation approaches, these approaches are DTD and DOM approach, clustering and matching approach, Query SQL, XPath, XQuery approach, relational storage approach, catalog-based approach, and other approaches. Other approaches contains the techniques that not categorized within the first five groups.

1.1 DTD and DOM Approach

XML DTD and XML DOM languages are using for transformation between XML and relational databases. DTD considers a markup language that declares the structure (schema) of XML document with a list of its legal elements and attributes, it uses also by applications and algorithms to verify wither XML data valid or not. XML DOM is an interface defining structure and values of XML documents and presents document as a tree-structure, this allows programs to create and build documents, access, manipulate elements and contents, structure, and style of XML documents. Figure 1 illustrates a general architecture of this approach.

Many papers focuses on DTD, DOM, and XML tree In order to extract information elements, attributes, and data using Functional Dependency (FD), where the conflicts of heterogeneous database systems has studied in many researches. Arenas and Libkin [6] proposed a tool for exchanging data between heterogeneous databases and systems, they restructured XML documents using DTD and target dependencies of data hierarchical structure, and then move to query answering between source and target.

Fong et al. [7] propose integrating relational schema into XML schema, the research normalizes relational schema into XML tree structures by mapping it into DOMs based on constraints of data dependencies, it integrates the results into XML trees and transform it into XML schema in the form of DTD. The method consists join dependency, multi-valued dependency, M: N cardinality, and functional dependency.

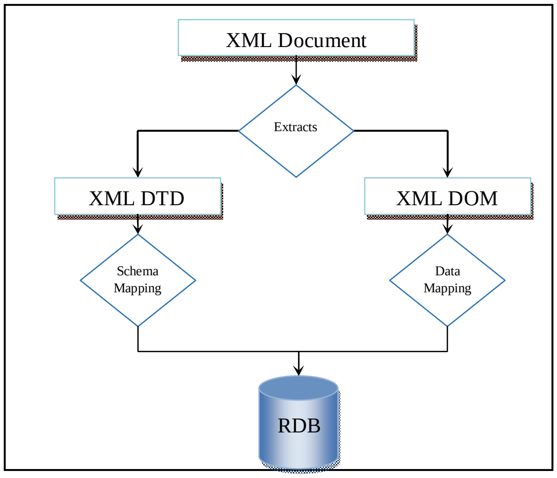

Figure 1: A General architecture of XML DTD and XML DOM approach.

A linear algorithm has proposed by Atay [8]; the algorithm uses DTD for mapping XML documents to relational tables, it converts each node into DTDs and combines each single child node with its parent node, then mapping them into the same table. The goal behind this idea is to reduce number of tables that corresponding relational schema and improve query processing. Mapping scheme for conversation between XML and relational database using data loading and processes query has discussed by Subramaniam *et al.* [9], their technique supports structure of skew dataset as compared to relational document type definition DTD.

Converts XML file into a DOM tree and extracts DTD is proposed by Zhou *et al.* [10], the technique transfers document tree into relational tables. Feng and Jingsheng [11] used and optimized an Absolute Data Group (ADG) technique for converting XML-DTD into relational model. Teng and Ping [12] proposed a technique for transforming relational schemas into XML-DTDs, the technique uses functional dependencies and the different keys of relational database for preserving the semantics implied. Moreover, an algorithm to access the components of XML file such as elements, attributes, contents and relationships of XML has proposed by Feng [13], the algorithm expresses the elements using DTD graph, optimize the components, and convert the result into relational database.

In general, the main limitation of this approach is the difficulty of converting integrity and referential constraints into DTD or DOM.

1.2 Clustering and Matching Approach

Clustering is a well-known technique developed and used in many areas of computing researches, XML conversation has benefited from the characteristics of these techniques, it usually divided data contents (elements, attributes and values) of the XML documents into set of non-hierarchical clusters or into nested set of hierarchal, then matching between the outputs in order to produce the final schema. This approach is often used in heterogeneous databases. Figure 2 shows a general architecture of clustering and matching approach.

Dividing the conflicts between values, attributes, and tables has proposed by Rajeswari and Varughese [14], their technique integrates microarray data sets using clustering heuristic and finding correspondent between popular majority rule and former consensuses. Create a mediated schema for integrating approach for XML structures has discussed by Saleem *et al.* [15], they used linguistic matchers that extract semantics of all node labels and tree-mining data structure and label clusters to find node context.

Structure method for enhancing XML clustering without summarize characteristics of XML structure is used by Shalabi and Elfatatry [16], the technique treats with different sizes of homogeneous and heterogeneous XML documents datasets. Al Hamad [17] developed a mediate schema for integrating heterogeneous XML, the technique decomposes the original schema into subschemas using three levels ancestor, root, and leaf. Matches the produced subschemas and return candidate subschemas. Thereafter, create the final mediate schema by obtain minimal of candidate subschemas.

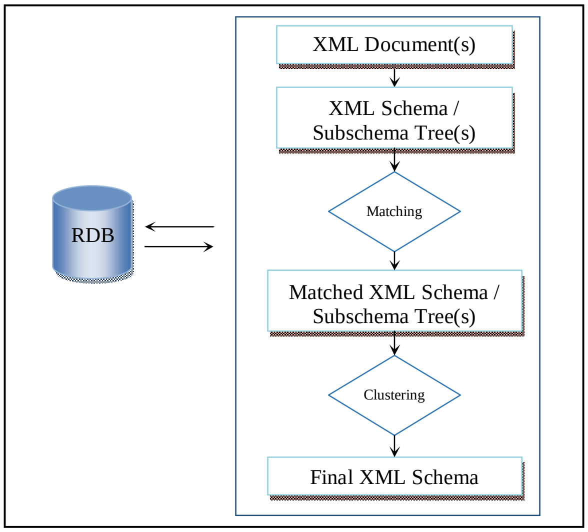
XML Document(s)
XML Schema / Subschema Tree(s)
Matching
RDB
Matched XML Schema / Subschema Tree(s)
Clustering
Final XML Schema

Figure 2: A general architecture of clustering and matching approach.

Moreover, Do and Rahm [18, 19, and 20] combined matching results of multiple methods in order to generate semantic correspondence between large XML schemas elements. The results indicate to superiority of combined match approaches and ability for matching large e-Business standard schemas. A hybrid XML schema matching algorithms has developed by Tansalarak and Claypool [21], the algorithm uses file paths with input schema that encoded as tree, the algorithm defines many classifiers that measure characteristics of various schema like QMatch path length and labels. The paper shows many experiments that describes the benefits of QMatch path of the schemas.

Some researchers have designed conversion algorithms based on intermediate schemes. Pottinger and Bernstein [22, 23] used a mediate schema merges a pair of heterogeneous schemas and integrates between databases including attributes, elements and relationships. An integration system of self-configuring schema using probabilistic of a set mediated schema has proposed by Sarma *et al.* [24], this system aims to enhance the integration processes; the research has used hundreds of experiments within five domains of data sources.

eXtensible Stylesheet Language Transformations (XSLT) is used by Jumaa *et al.* [25], the technique uses XSLT to create a mediate architecture of XML schemas in order to transform XML documents. An indirect method for converting relational schemas into XSD graph (XSD: XML Schema Definition) has applied by Fong and Cheung [26], the method maps the graph into XSD schema as a logical schema.

Since this approach merges two or more different databases work within different environments, many limitations may arise while reading, analyzing, matching, clustering schemas and contents which may leave the result either incomplete or inaccurate.

1.3 Query SQL, XPath, XQuery Approach

The Query languages consider an essential in many researches. In this filed, query languages such as SQL, XPath, and XQuery are currently using in many applications and algorithms for converting between relational and XML models. Figure 3 illustrates a general architecture of query approach.

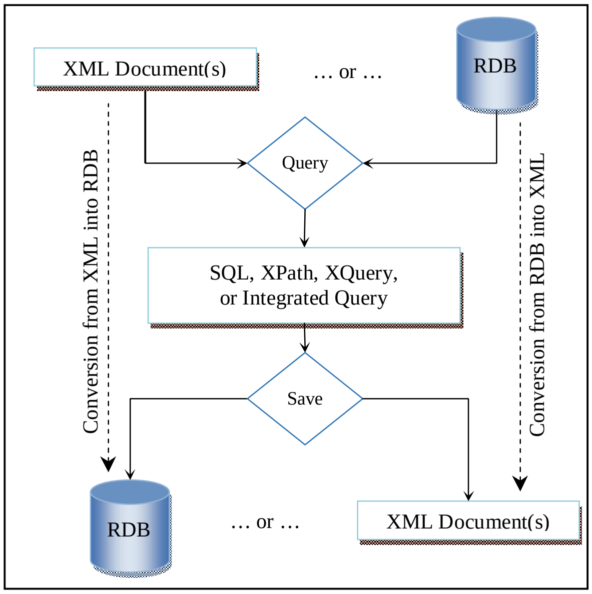
XML Document(s)
… or …
RDB
Conversion from XML into RDB
Query
Conversion from RDB into XML
SQL, XPath, XQuery, or Integrated Query
Save
RDB
… or …
XML Document(s)

Figure 3: A general architecture of query approach.

A new query language called RXQL (Relational XML Query Language) has purposed by Nappila *et al.* [27]. The query integrates data with other essential features in the heterogeneous XML documents; it manipulates XML data in a tuple-oriented way instead of path-oriented XML languages. The approach removes factors of heterogeneous from XML data and integrates the result with data features such as ordering, grouping, and aggregation.

An interpreter method for converting relational databases into XML model has developed by Fong and Shiu [28], the approach similar to XQuery, it integrates SQL language for constructing XML documents, it uses three keywords, these keywords are element name, attribute name, and attribute value. Another converting technique has described by Yoshikawa and Amagasa [29], the technique converts relational databases into XML documents, it decomposes an XML file into nodes-base tree structure and stores the results in database tables including path information of each node, they also propose an algorithm for converting XPath expressions into SQL queries.

Krishnamurthy *et al.* [30] proposed a technique translates XML path expression into complex SQL queries; their technique uses two algorithm. First algorithm translates SQL queries and then optimize SQL results; Second algorithm uses an intelligent translation algorithm and generate efficient SQL clauses for queries of path expression over the tree schemas. Using XQuery query language for processing complex XML queries has evaluated by Shanmugasundaram *et al.* [31], the technique develops XML queries by pushing it into the relational engine. Moreover, they proposed using the same query processor for all relational databases and XML documents, the technique deals with relational and XML data as non-separated databases, it automatically creates relational tables and reconstruct XML view over them, in addition to query and store the contents of XML documents as rows inside these tables [32].

The main limitation of this approach is the difficulty of extracting all database constraints and connecting them to their correct contents.

In this approach, limited information can be extracted from the XML document and relational databases using query language. Many constraints such as referential, data type, keys, etc. possible to ignore during the conversion process and this leads to inaccurate mapping.

1.4 Relational Storage Approach

In order to take advantages of potential of using relational database models and SQL query language which consider an effective models. Therefore, some researchers uses relational model to store and index the contents of XML databases. This approach decomposes XML files into rows and stores them as relational tables created for this purpose. SQL, XPath, XQuery, or integrated Query languages can also be used over the created tables. Figure 4 displays a general architecture of relational storage approach.

One of the large institutions specializing in databases and applications solutions that used this approach is Oracle Company. Oracle developed and lunched in release Oracle 9i the XML DB standard, XML DB built based on World Wide Web Consortium (W3C) standards, it includes and supports XPath recommendation. XML DB is an independent storage, content, and a programming language supports all Oracle databases versions for storing and managing XML data. One of the advantages of this technique is increasing the performance of database by using XML storage and retrieval. XML DB uses SQL to transform, update query, update, and perform XML operations [33, 34].

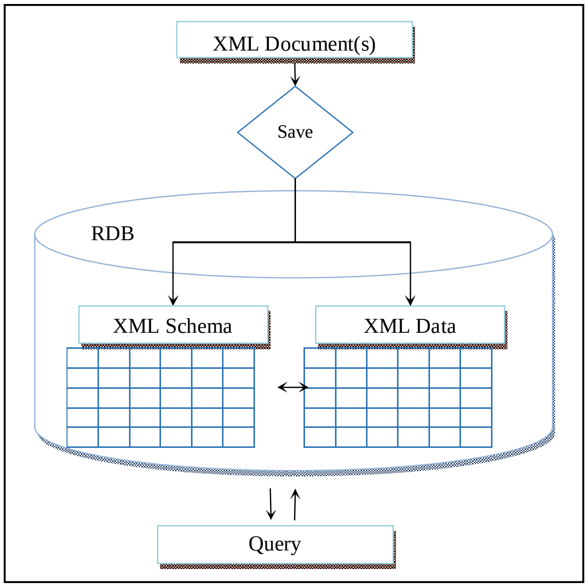

Figure 4: A general architecture of relational storage approach.

A conversion approach between XML heterogeneous databases based on relationship schemas has proposed by Ning *et al.* [35], the approach solves many problems such as values losing, misjudgment of relationships and changing of field attribute during conversion of the data. Qtaish and Ahmad [36] proposed an approach called XAncestor, the approach contains two mapping algorithms, first algorithm transfers XML file into relational database, second algorithm transfers XPath queries into SQL base on construction of the relations, the algorithms considered more effectiveness and scalability than other approaches. Merging XML documents into a relational data model in purpose of take advantages of relational query has suggested by Hong and Song [37], the technique proposes two data transformation from XML into relational model and from relational model into XML documents.

In some cases, query does not work effectively due to complexity of XML constructs that may contain many relationships and attribute values.

1.5 Catalog-based Approach

Another approach used for mapping between relational and XML models is catalog-based conversion technique; the idea behind this approach is to re-engineer the legacy relational databases and schema by extracting catalog of schema design and contents from the database using SQL. Figure 5 illustrates a general architecture of catalog-based approach.

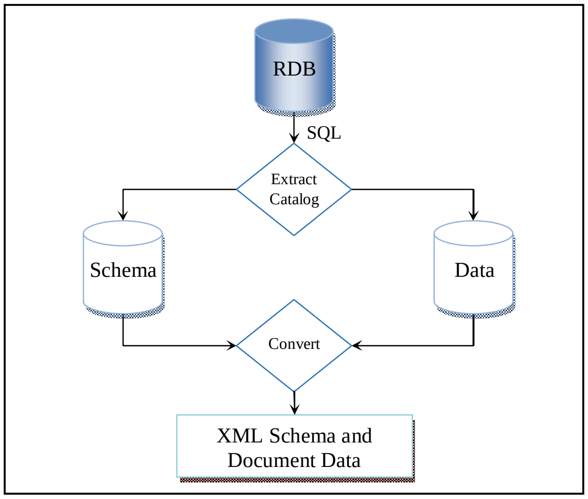

Figure 5: A general architecture of catalog-based approach.

Reverse engineering approach using catalog-based that extracts relational model into Extended Entity Relationship (EER) model has applied by Chunyan Wang *et al.* [38], the technique converts EER model into XML schema. Metadata schema management and data exchange for mapping between relational schemas has used by Kolaitis [39], the technique provides a justification for engineering data exchange between XML and relational schemas.

A conversation technique from relational database into XML schema (XSD) based on relational database catalog has proposed by Al Hamad [5], the technique uses SQL to extract contents and data of the relation model, thereafter transform the results into XML document tree, then convert the result to XSD schema to represent schema design and constraints.

1.6 Other Approaches

Many other researchers has touched other ways of transforming XML models, some of them converts semantic constraints, others uses mining algorithms and UML diagrams or converts between object-oriented and XML using SQL and XQuery languages.

Dongwon et al [40] presents three semantics conversion techniques between XML schema and relational schema, these methods convert semantics constraints of XML schemas into a relational schemas, nested XML schema structure from relational schema, and constraints of relational schema to XML schema. Zaki [41] formulates a mining algorithm to reconcile all frequent subtrees of XML structures using scope-list data structure, the algorithm illustrates how nodes of the tree represented as a list.

A mechanism to converts XML schema to UML diagram by clustering and unification of concepts has proposed by Zhang and Liu [42], the technique restructures the relationships and extends a global conceptual model. Terwilliger *et al.* [43] proposes a prototype for transforming queries of Object-oriented into queries of XML using declarative mappings between classes and XML schema types. The technique takes benefits of mapping capabilities of ADO.NET object-relational model; it uses features of Language Integrated Query (LINQ) that translated into a combination of SQL and XQuery.

Other surveys treat with different topics related to the XML. Wong *et al.* [44] surveys different approaches of XML indexing, the survey analyses indexing tree structures of XML path information. Hacherouf *et al.* [45] surveys indexing and conversion of XML documents into Ontology Web Language (OWL), the survey focuses on enrichment and population of ontology using XML data, the ontology uses Resource Description Framework (RDF) and formal languages of the semantic web. Haw and Lee *et al.* [46] surveys storage and query processing in XML databases, the survey reviews two approaches for storing and enhance query processing of XML data, first one employs traditional storage, the second creates an XML specific storage, as well as review different query optimization such as join algorithms, labeling and indexing classification methods.

The main limitation of this approach is the difficulty of extracting all database constraints and connecting them to their correct contents.

3. CONCLUSION

The main objective that presented throughout this paper is a brief literature survey introduces the conversation approaches and techniques between relational and XML models. In addition, the paper explains the general architecture for each approach.

This survey will help researchers to distinguish between the different types of mapping methods. Although there are many conversion techniques, still there is no preferred common technique because each one has its own application purpose.

ACKNOWLEDGMENT

The author would like to acknowledge the financial support provided by the Deanship of Scientific Research at Amman Arab University, under research project reference number (R.D./1/2/185), entitled, "Web-Based Model for Conversion between Relational Databases and XML Language".

REFERENCES

1. D. Juan, and Q. Zheng, **The research on the XML-based information exchange under heterogeneous environment in HR outsourcing enterprises**, *7th International Conference of Computer Science and Education (ICCSE)*, 14-17 July 2012, pp. 462-465.
2. J. Shanmugasundaram, K. Tufte, G. He, C. Zhang, D. DeWitt, and J. Naughton, **Relational databases for querying XML documents: limitations and opportunities**, *Proceedings of the 25th VLDB Conference, Edinburgh*, Scotland, 1999, pp. 302–314.
3. A.Silberschatz, H. F. Korth, and S. Sudarshan, *Database System Concepts*, McGraw-Hill, Sixth Edition, 2010, pp. 981-1025.
4. D. Joseph, **Current usage and future of XML Database Management Systems**, 2009, http://www.getallarticles.com/2009/12/28/4/.
5. H. A. Al Hamad, *Catalog-based conversion from relational database into XML schema (XSD)*, International Journal of Date Engineering (IJDE), vol. 6(2), 2015, pp. 9-22.
6. M. Arenas and L. Libkin, **Xml data exchange: consistency and query answering**, *In Proceedings of the 24th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS'05)*, Baltimore, USA, 2005, pp. 13-24.
7. J. Fong, H.K. Wong, and Z. Cheng, **Converting relational database into XML documents with DOM**, *Information and Software Technology*, vol. 45, 2003, pp. 335– 355.
8. M. Atay, Y. Sun, D. Liu, S. Lu and F. Fotouhi, **Mapping XML data to relational data: A DOM-based approach**, *eighth IASTED International Conference on Internet and Multimedia Systems and Applications (IMSA'04)*, Hawaii, USA, 2010, pp. 59-64.
9. S. Subramaniam, S. Haw, and P. Hoong, **s-XML: An efficient mapping scheme to bridge XML and relational database**, *Knowledge-Based Systems Journal*, vol. 27, 2012, pp. 369–380.
10. Zhou, H. Lu, S. Zheng, Y. Liang, L. Zhang, W. Ji and Z. Tian, **VXMLR: A visual XML-relational database system**, *Proceedings of the 27th International Conference on Very Large Data Bases*, September 11-14, 2001, pp. 719-720.
11. Y. Feng and X. Jingsheng, **Mapping XML DTD to relational schema**, Database Technology and Applications, *First International Workshop*, 25-26 April 2009, pp. 557-560.
12. L. Teng, and Y. Ping, **Schema conversion from relation to XML with semantic constraints**, *Fourth International Conference in Fuzzy Systems and Knowledge Discovery, FSKD.*, 24-27 Aug. 2007, pp. 619-623.
13. Y. Feng, **Converting XML DTD to database, Intelligent Systems and Applications**, *ISA 2009 International Workshop, 23-24 May 2009*, pp. 1-4.

14. V. Rajeswari and D. K. Varughese, **Heterogeneous database integration for web applications**, *International Journal on Computer Science and Engineering*, vol. 1(3), 2009, pp. 227-234.
15. K. Saleem, Z. Bellahsene and E. Hunt, **PORSCHE: Performance ORiented SCHEma mediation** *Information System Journal*, vol. 33(7–8), 2008, pp, 637–657.
16. R. Shalabi, A. Elfatatry, **Towards improving XML search by using structure clustering technique**, *Journal of Information Science*, 41(2), 2015, pp. 146–166.
17. H. A. Al Hamad, **XML-Based data exchange in the heterogeneous databases (XDEHD)**, *International Journal of Web & Semantic Technology (IJWesT)*, vol. 6(3), 2015, pp. 11-24.
18. H. Do and E. Rahm **COMA: A system for flexible combination of schema matching approaches**, *International Conference on Very Large Data Bases (VLDB)*, Hong Kong, China, August 20–23, 2002, pp. 610–621.
19. H. Do and E. Rahm, **Matching large schemas: Approaches and evaluation**. *Information System Journal*, vol. 32(6), 2007, pp. 857–885.
20. E. Rahm, H. Do and S. Massmann, **Matching large XML schemas**, *ACM SIGMOD Record*, vol. 33(4), 2004, pp. 26–31.
21. N. Tansalarak and K.T. Claypool, **QMatch: using paths to match XML schemas**, *Knowledge Data Engineering*, vol. 60(2), 2007, pp. 260–282.
22. R. A. Pottinger and P. A. Bernstein, **Creating a mediated schema based on initial correspondences**, *IEEE Data Engineering Bulletin*. Vol. 25(3), 2002, pp. 26–31.
23. R.A. Pottinger, **Processing queries and merging schemas in support of data integration**, Ph.D. thesis, University of Washington, Washington, USA, 2004.
24. A.D. Sarma, X. Dong and A. Halevy, **Bootstrapping pay-as-you-go data integration systems**, *ACM SIGMOD International Conference on Management of Data*, Vancouver, BC, Canada, 10–12 June 2008, pp. 861–874.
25. H. Jumaa, P. Rubel and J. Fayn, **An XML-based framework for automating data exchange in healthcare**, *12th IEEE International Conference in e-Health networking Applications and Services (Healthcom)*, 1-3 July 2010, pp. 26124-269.
26. J. Fong and S. Cheung, **Translating relational schema into XML schema definition with data semantic preservation and XSD graph**, *Information and Software Technology*, vol. 47, No. 7, 2001, pp. 437-462.
27. T. Nappila, K. Moilanen,and T. Niemi, **A query language for selecting, harmonizing, and aggregating heterogeneous XML data**, *International Journal of Web Information Systems*, vol. 7(1), 2011, pp. 62-99.
28. J. Fong and H. Shiu, **An interpreter approach for exporting relational data into XML documents with structured export markup language**, *Journal of Database Management*, 23(1), 2012, pp. 49-77.
29. Y. Yoshikawa and T. Amagasa, **XRel: A path-based approach to storage and retrieval of XML documents using relational databases**, *ACM Transactions on Internet Technology*, vol. 1(1), 2001, pp. 1-29.
30. R. Krishnamurthy, R. Kaushik and J. Naughton, **Efficient XML-to-SQL query translation: where to add the intelligence?**, *Proceedings of the Thirtieth International Conference on Very Large Data Bases*, vol. 30, Toronto, Canada, 2004, pp. 144-155.
31. J. Shanmugasundaram, J. Kiernan, E. Shekita, C. Fan and J. Funderburk, **Querying XML views of relational data**, *Proceedings of the 27th Very Large Data Bases (VLDB) Conference*, 2001, pp. 261-270.
32. J. Shanmugasundaram, R. Krishnamurthy and J.Tatarinov, **A general technique for querying XML documents using a relational database system**, *ACM SIGMOD Rec journal*, vol. 30(3), 2001, pp. 20-26.
33. Oracle, **Oracle XML DB: best practices to get optimal performance out of XML queries**, *Oracle White Paper*, 2013, http://www.oracle.com/technetwork /database-features/xmldb/xmlqueryoptimize11gr2-168036.pdf
34. Oracle, **Oracle XML DB in Oracle database 12c release 2**, *Oracle White Paper January*, 2017, http://www.oracle.com/technetwork/database-features/xmldb/overview/xmldb-twp-12cr1-1964803.pdf
35. L. Ning, L. Bin, Z. Xin and D. Zhongliang, **Database conversion based on relationship schema mapping**, *Internet Technology and Applications (iTAP)*, 16-18 Aug. 2011, pp. 1-5.
36. Qtaish, K. Ahmad, **XAncestor: An efficient mapping approach for storing and querying**, *Knowledge based Systems Journal*, vol. 114, 2016, pp. 167–192.

37. S. Hong, and Y. Song, **Efficient XML query using relational data model**, *Eighth ACIS International Conference in Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, SNPD 2007., Aug. 2007, pp. 1095-1100.
38. W. Chunyan, A. Lo, R. Alhajj and K. Barker, **Novel approach for reengineering relational databases into XML**, *Data Engineering Workshops*, 21st International Conference, 05-08 April 2005, pp. 1284.
39. P. G. Kolaitis, **Schema mappings, data exchange, and metadata management**, *Proceedings of the Twenty-fourth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, 2005, pp. 61–75.
40. L. P. Dongwon, L. Dongwon, M. Murali and W. Wesley, **Schema conversion methods between XML and relational models**, *Information and Software Technology Journal*, vol. 48, 2006, pp. 245–252
41. M. J. Zaki, **Efficiently mining frequent trees in a forest: algorithms and applications**, *IEEE Trans, Knowledge Data Engineering (TKDE)*, vol. 17(8), 2005, pp. 1021–1035.
42. Y. Zhang and W. Liu, **Semantic integration of XML Schema**, *Proceedings of Machine Learning and Cybernetics International Conference*, 2002, pp. 1058-1061.
43. J. Terwilliger, S. Melnik and Ph. Bernstein, **Language integrated querying of XML data in SQL Server**, *Proceedings of the Very Large database (PVLDB)*, vol. 1(2) 2008, pp. 1396—1399.
44. K. Wong, J. Xu Yu, and N. Tang, **Answering XML queries using path-based indexes: A survey**, *World Wide Web Journal*, vol. 9, 2006, pp. 277–299.
45. M. Hacherouf, S. Bahloul and C. Cruz, **Transforming XML documents to OWL ontologies: A survey,** *Journal of Information Science*, vol. 41(2), 2015, pp. 242–259.
46. S. Haw, CH. Lee, **Data storage practices and query processing in XML databases: A survey**, *Knowledge-Based Systems*, vol. 24, 2011, pp. 1317–1340.