Vol. 3 · September 2009 · ISSN 20123981
doi: http://dx.doi.org/10.7719/jpair.v3i1.83


# Exposition of the Different Schema Clustering (Information Integration) Techniques

**GREGG VICTOR D. GABISON**
University of San Jose – Recoletos
Cebu City, Philippines

***Abstract*** - The Internet today has provided more opportunities for entities to communicate, exchange information in a very dynamic manner. Explicitly, Web 2.0 and SaaS have provided new kinds of platform/ services which resulted to more web applications taking advantage of the universality of XML. Techniques regarding XML clustering between different heterogenous sources especially in a more open environment are becoming more necessary, which leads to a more efficient handling of the increasing volumes of XML schemas employed. Various (XML clustering) techniques are out in the open, and can be integrated for use in the different application areas, such as E-commerce, data warehousing, unique user supplied query parameters and possibly in any general data integration activity. In this paper, in each of the XML clustering technique, the researcher provides a general idea and its specific approach/ technique with relevance to an intended implementation. Following the approach/ technique will be the presentation of its key advantages and disadvantages.

***Keywords*** **-** Exposition, schema clustering, information-integration

## INTRODUCTION

A newer technique in deploying software is through the Software as a Service (SaaS) Model. As a result, it has revolutionized the way Software is developed, deployed and maintained. Moreover, the introduction of Web 2.0, as a business revolution in the computer industry caused by the move to the Internet as a platform, and an attempt to understand the rules for success on that new platform [O'Reilly, 2005], has changed in the ways software developers and end-users utilize the Web. And with these innovations, they have sparked more dynamic advancements where software applications over the Internet, in the near future, will be the ones dealing directly with one another. And with this advancement, this could lead to a much more efficient and faster transaction between two electronic entities with less or no intervention from humans. Such instance can actually happen between two heterogenic E-commerce sites, or for Data Mining and warehousing activities or even more in a more advance state of Web 2.0, where the need for information integration critically utilized.

The fundamental reason of this phenomenon is brought about by the functionality and universality of XML [XML, 2008]. Currently, incidence of data manipulation and integration over internet has never been this in abundance, as a result of the introduction of XML. XML today has become a popular standard for effectively and appropriately interchanging and presenting data over the internet. As a result, an increasing amount of XML schema has been created [Biron and Malhotra, 2001], [XML Schema: Data Types, 2004]. With this increased number, XML schema clustering is one of the favorite topics for research and innovation.

## MATERIALS AND METHOD

The paper utilized the qualitative research method including descriptive analytical technique of available web and print sources.

## RESULTS AND DISCUSSION

Several research papers regarding XML Schema clustering has already been worked out, specifically in the context of schema translation, knowledge representation, information retrieval and machine learning. In this paper, we will start off by presenting the different XML schema clustering techniques and with emphasis of its specific approach followed by the challenges in undertaking XML Schema clustering.

## FRAMEWORK

Extensible Markup Language (XML) is a simple, very flexible text format derived from SGML. Originally designed to meet the challenges of large-scale electronic publishing, XML is also playing an increasingly important role in the exchange of a wide variety of data on the Web and elsewhere [Bray, 2001]. The fundamental principle behind XML is the ability for developers and the like to create custom tags that can be used to describe any kind of information. A custom tag is a user defined tag like if you would like to refer to a name attribute, you can represent it as <name>. As a result, more specific tags can be created in order to convey its direct function or use, meaning having more contexts to content. The clear of advantage of XML extends to not just putting more information to its strucutre, but rather providing a clear and simple way of storing and transferring information between different kinds of Computing Systems, both on and off the Internet, such as online stores.

The W3C XML Schema use XML as its expression language. It takes the namespace, not the document, as the fundamental unit of interest in validation. It is made of two normative specifications, one for Datatypes and the other for Structures and everything else. All typing uses type hierarchies, by which one can restrict and, in some cases, extend other types [Thompson, 2001]. The Datatypes specification approach defines a set of values, such as the maximum value of a number; types are derived from these primitive types by restricting values or ranges. The intent of W3C XML Schema is to reconstruct the facilities provided by DTD's parameter entities and marked sections

with type inheritance, type extension and type restriction. However, many of the uses of parameter entities and marked sections could not be reconciled with element or attribute "types" and so a basic module system of import and include declarations is also provided.

A typical XML Schema clustering can be exempified as a matched entity which indicates that certain elements of schema S1 (see Figure 1) are mapped to certain elements in Schema S2 [Bernstein and Rahm, 2001] based on certain matching algorithms. Furthermore, supplementary or auxillary information, if there is, like Data-Dictionary, comments, etc, can be utilized in order to intensify the matching expression which specifies how the S1 and S2 elements are related. As presented, the matching actions will undertake the process where the two schemas S1 and S2 as input and returns a clustering (mapped) report between these two schemas as output, called the data integration result. The fundamental expectation of the data integration will be that for each mapped element, the result will specify that certain elements of schema S1 logically corresponds to certain elements of S2.

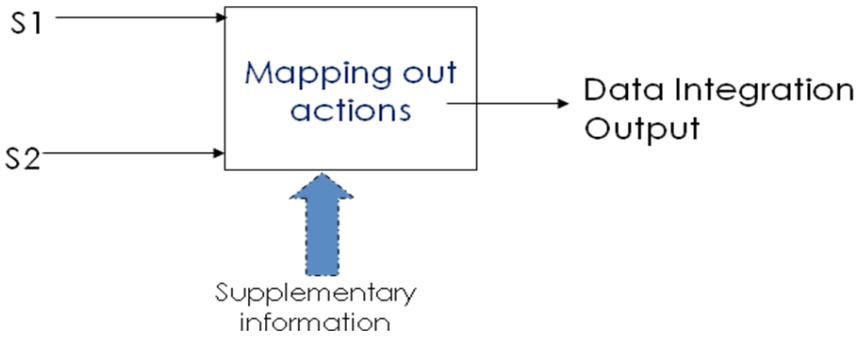


Figure 1. Typical schema clustering

In effect, the procedure in general contains four main steps: first, (1 )the schemas that correspond between their elements are imported (and may be transformed into the internal representation). (2)Then, the preprocessing step in which the schemas are traversed to determine schema elements for which match algorithms calculate the similarity values. (3)After identifying these elements, the mapping activity is undertaken. (4)Finally, the result obtained from the mapping operation is exported to desired/ designated applications.

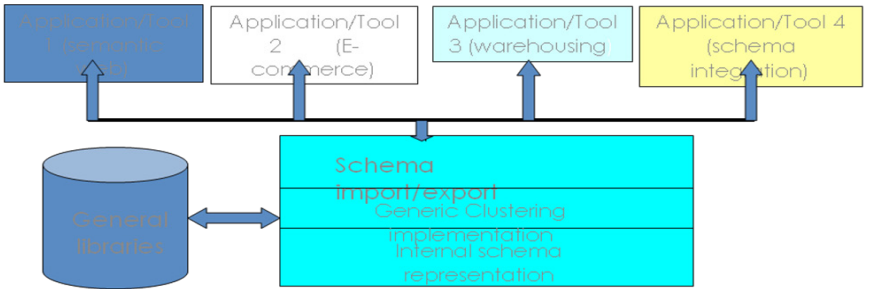


Figure 2. Typical schema clustering architecture

In Figure 2, it illustrates the structure for generic schema clustering, supporting different applications and multiple schema types such as XML and relational schemas. The clients are schema-related applications and tools from different domains. Each client uses the generic clustering implementation to automatically determine matches between schemas. Tools that are tightly integrated with the framework can work directly on the internal representation. Other tools need import/export programs to translate between their native schema representation and the internal representation. The implementation of match may also use the libraries and other auxiliary information, such as dictionaries to help find matches [Bernstein and Rahm, 2001].

**Application Domains**

**E-Commerce**. E-commerce has led to new undertakings for schema clustering: message translation. In an E-Commerce (ex. Business to Business) infrastructure where two or more entities interact/ transact resulting to an exchange of predominantly unique formats specific to each party. Trading partners exchange message that describe business transactions. Usually, each partner uses its own format. Message formats may differ in their syntax, structures. They may also use different message schemas. Translating between different message schemas is a schema matching problem.

**Web 2.0**. Advancement of Web 2.0 had produced maturing Semantic query processing technology. A user specifies the output of a query and the system figures out how to produce that amount. The user's specification is stated in terms of concepts may not be the same

as the names of elements specified in the database schema. Therefore, the first step in query processing, the system must map the user-specified concepts in the query output to schema elements. This is a natural application of schema matching.

**Data Warehouse**. A data warehouse is a decision support database that is extracted from a set of data sources format into the warehouse format. The match operation is useful for designing transformation. Given a data source, one approach to create suitable transformations is to start by finding those elements of the source that are also present in the warehouse. This is a match operation.

Generally, the two major approaches are schema-based and instance based [Bernstein and Rahm, 2001]. However, in a realworld application, a combination of such approaches may be implemented in order to strengthen the match between two schemas or even more. Furthermore, on the more fundamental level, to compare such information whether as a result based on the schema or the instance approach, these approaches make use of a combination of Linguist and constraint-based algorithm in order to reinforce the creation of a clustering (data integration) result (Figure 3).

**Schema-level approach** only considers schema information, not instance data. The available information includes the usual properties of schema elements, such as name, description, data type, relationship types (part-of, is-a, etc.), constraints, and schema structure. Working at the element (atomic elements like attributes of objects) or structure level (matching combinations of elements that appear together in a structure), these properties is used to identify matching elements in two schemas [Kim and Seo 1991].

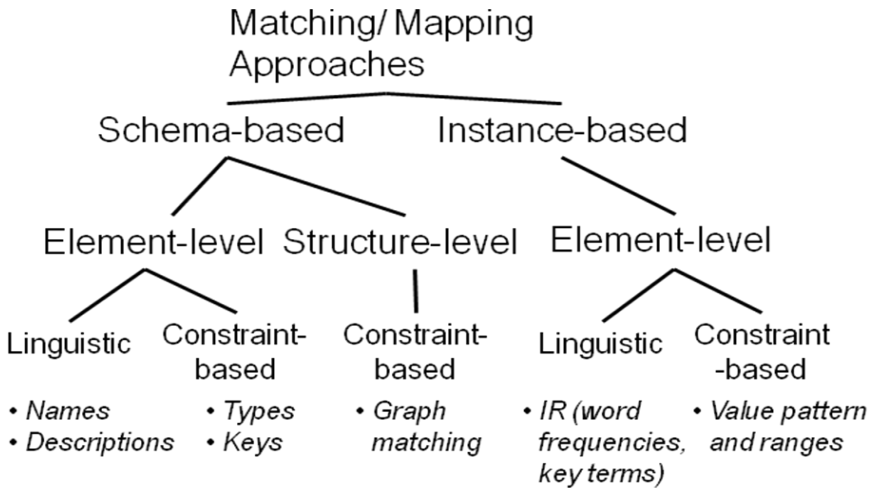


Figure 3. Taxonomy of Approaches to XML Schema Clustering by [Bernstein and Rahm, 2001]

**Specific approaches in Schema Level**

**Structure-level approach**. Structure-level approach or structure similarity as known by others, refers to clustering possible combinations of elements that appear together in a structure. A range of cases is possible, depending on how complete and precise a match of the structure is required. In an ideal scenario, all elements of the structures in the two schemas are fully matched. However, in reality, not all matches of components will be required to match. This instance is known as a partial structural match. Where based on an accepted fuzzy logic rule, partial structural match is acceptable to deem of a full match. The need for partial matches sometimes arises because subschemas of different domains are being compared.

**Element-level Approach**. Element-level approach refers to the matching of elements (attributes, objecst) in first schema and to the second schema. The general rule so far is to make use of elements at the most bottom level of the structure, which is usually referred as the atomic level. Examples are the attributes in an XML schema or columns in a relational schema.

**Match Cardinality**. In addition to structural and element level approach, match cardinality which is consistent to exposing a queried recordset can be initiated as a simple element level matching or to a complex structural level matching. Simple matching comprises of 1:1, 1:n and n:1 match cardinality, whereas n:m match cardinality is considered to be complex matching. A schema element can participate in zero, one or many mapping elements of the match result between the two input schemas S1 and S2. Moreover, within an individual mapping element, one or more S1 elements can match one or more S2 elements. Thus, we have the usual relationship cardinalities, namely 1:1 and the set-oriented cases 1:n, n:1, and n:m, between matching elements both with respect to different mapping elements (global cardinality) and with respect to an individual mapping element (local cardinality). Element-level matching is typically restricted to local cardinalities of 1:1, n:1, and 1:n. Obtaining n:m mapping elements usually requires considering the structural embedding of the schema elements and thus requires structure-level matching.

**Instance Level Approach.** Instance level approach provides a richer and almost precise result to the contents and meaning of schema elements. This is especially true when schema information is limited, and can also be in the case for semistructured data. In the extreme case, no schema is given, but a schema can be constructed from instance data either manually or automatically [Bernstein and Rahm, 2001]. In its simplest sense, clustering happens at the actual data/ content.

Even when substantial schema information is available, the use of instance-level matching can be valuable to uncover incorrect interpretations of schema information. For example, it can help disambiguate between equally plausible schema-level matches by choosing to match the elements whose instances are more similar.

The finer approaches like linguist and constraint base can also be applied to instance-level approach. The main benefit of evaluating instances is a precise classification of the actual contents of schema elements. This can be employed in two ways. First, is to use it at Schema Level approach. For instance, a constraint-based matcher can then more accurately determine corresponding data types based, for example, on the discovered value ranges and character pattern, thereby improving the effectiveness of Match. This requires classifying the content of both input schemas and then matching the schemas with each other.

A second approach is to perform an instance level clustering. First, the instances of the first schema are evaluated to characterize the content of its elements. Then, the 2nd schema instances are matched one-by-one against the characterizations of first schema elements. The per-instance match results need to be merged and abstracted to the schema level, in order to generate a ranked list of match candidates from the first schema to the 2nd schema.

**Natural Language/ Semantics.** This approach makes use of names/ text (ex. Words, sentences) to find semantically similar schema elements. In most cases, Natural language or linguistic can be utilized at different levels, either at the Schema or Instance level. Two common approaches that are commonly available are the following:

**Name Based Approach**. Name based approached matches names of elements/ data with similar or equal value. In a schema based approached, the element will be evaluated. Typical name based approach can be undertaken in several ways:

*Equality of names*

- *Comparing basic name semantics with or without sensitivity to cases of letters.*

*Equality of category name representations*

- *This is important to deal with special prefix/suffix symbols (ex. CName to customer name, and EmpNO to employee number)*

*Equality of synonyms.*

- *ex. car to automobile, brand to make*

*Equality of hypernyms.*

- *Ex. book is-a publication and article is-a publication thus book is a match to publication, article is a match to publication, thus book is a match to article*

*Similarity of names*

- *based on common substrings, edit distance, pronunciation, soundex (an encoding of names basedon how they sound rather than how they are spelled),*
- *Ex. Representedby is a match to representative, shipto is a match to ship2*

*User-provided/ defined name matches*

- *Ex. reportsTo is a match to manager, issues is a match to bugs*

Utilizing synonyms and hypernyms requires the use of a thesaurus or dictionaries. Also Name matching can also make use of NLP (Natural Language Processing) algorithm/ engines making use of domain or enterprise specific dictionaries containing user defined names, synonyms and descriptions of schema elements, abbreviations, etc. However, one drawback is that these specific dictionaries require a substantial effort in compiling a rich resource.

Description Matching

In most cases, just like scripts and codes, schemas usually contain comments in plain human like language to provide description as to the nature of the element/ object. These comments can be evaluated linguistically to determine similarity between elements. An analysis can be a simple extraction of keywords with in the comment line to be used for comparison in the clustering process

**Constraint Based.** Constraint based makes use the of schema constraint definitions such as data types with its corresponding value ranges, uniqueness, options, relationship types and even cardinalities (recordsets). In addition, other schema information can also be utilized such as intra-schema references like foreign keys, and adjacency related information (ex. Part of relationships). In table 1, referring to the data type definition and key of the element custno in Customer schema suggest a possible match to cno in the Client schema. Furthermore, Birthdate of the customer schema can be matched to client.born considering that data type is date. The rest like custname, custaddress can be a match to clients cname and address with reference to the string definition.

Table 1. Constraint based approac

| Customer Schema | Client Schema |
|---|---|
| CustNo - int, primary key | Cno - int, unique |
| CustName – varchar(50) | Cname – string |
| Custaddress – varchar(2) | address – string |
| Age - int | Born – date |
| Birthdate – date | |

**Heuristic Approach.** Generally speaking, schema clustering is naturally considered a heuristic method, considering that in the different approaches mentioned (structure/schema level, element to instance), it simply starts with a discovery of matches, yielding strong possibility of a number of false results, which fortunately can be used as reference for another sequence of clustering (matching and mapping). In its basic sense, heuristic is defined as technique in problem solving where initial result are reused for next runs leading to learning and discovery. In most cases, it employs experimentation and trial and error techniques. Heuristics are "rules of thumb", educated guesses, intuitive judgments or simply common sense [Gigerenzer, Todd, 1999].

## CONCLUSIONS

Despite its pervasiveness and importance, schema matching remains a difficult problem: The semantics of the involved elements can be sourced out from only a few information sources and importantly it's created/ defined by people, which even if the two schemas are identical even at the dictionary level, it can still different on the syntactic level.

Schema elements are matched based on clues in the schema and data. Schema and data clues are often incomplete and worst have different formats. To make matters worse, mapping out correspondence is often subjective, depending on the application was designed.

Considering the general semantic heterogeneity of different schemas, schema clustering is largely performed semi-manually (semi automatic) [Bernstein and Rahm, 2001], despite the presence of new tools and algorithms, which as a result, sometimes becomes tedious, time consuming, error prone, and an exensive process.

## LITERATURE CITED

Batini, C, Lenzerini, and Navathe, S.B.
1986 A comparative Analysis of Methodologies for Database Schema Integration. ACM Computing Surveys, Vol 18, No.4.

Bernstein, Philip and Rahm Erhard.
2001 A survey of approaches to automatic schema matching. In VLDB Journal, pages 10: 334-350.

Biron, Paul V.and Malhotra Ashok
2001 XML Schema Part 2: Datatypes. W3C (World Wide Web Consortium), 2001. Available at http://www.w3.org/TR/xmlschema2/. [Accessed October 28, 2007]

Bray,Tim and Paoli, Jean et al.
2001 Extensible Markup Language (XML) 1.0 (Second Edition), W3C (World Wide Web Consortium. Available at http://www.w3.org/TR/REC-xml.

Gigerenzer, Todd, and the ABC Research Group
1999 Simple Heuristics That Make Us Smart. Oxford, UK, Oxford University Press. ISBN 0-19-514381-7

Jelliffe, Rick.
2001 The Current State of the Art of Schema Languages for XML. XML Asia Pacific Conference Sydney, Australia.

Kim, W. and Seo, J.
1991 Classifying Schematic and Data Heterogeneity in Multidatabase Systems.. Computer 24, 12.

O'Reilly, Tim.
2005 "What Is Web 2.0". O'Reilly Network. [Accessed October 31, 2007]

Shohoud, Yasser.
2002 Place XML message design ahead of schema planning to improve web service interoperability. MSDN Magazine. vol.17, no.12:81. December 2002.

Thompson, Henry S.and Beech, David et al.
2001 XML Schema Part 1: Structures. W3C (World Wide Web consortium). See http://www.w3.org/TR/xmlschema-1/

XML (Extensible Mark Up Language)
2008 Available at: http://www.w3.org/XML/ [Accessed October 12, 2008]

XML Schema Part 0: Primer, W3C Recommendation
2004 Available at: http://www.w3.org/TR/xmlschema-0/. [Accessed October 28, 2007] h

Pursuant to the international character of this publication, the journal is indexed by the following agencies: (1)Public Knowledge Project, a consortium of Simon Fraser University Library, the School of Education of Stanford University, and the British Columbia University, Canada: (2) E-International Scientific Research Journal Consortium; (3) Philippine E-Journals (4) Google Scholar.