Schema matching methodologies and run-time solutions in SOA based enterprise application integration

PhD Thesis

Martinek Péter

Advisor: Dr. Szikora Béla

Budapest University of Technology and Economics
Department of Electronics Technology

2009.
Preliminaries – Enterprise Application Integration, Service Oriented Architecture

The widespread of enterprise information systems began in the ‘70s of the last century. The improvement of hardware tools and information sciences implicated the exchange of paper based data handling and human activity with information systems in everyday working tasks. Furthermore complex, manually non-manageable processes started to be supervised by the help of specific computer aided solutions and software applications. After the millennium there was not any organization, which did not use any information system for supporting its everyday working processes.

Various information systems of different vendors are applied in the last few years for covering many areas of the everyday working processes of organizations. The continuous spreading of enterprise information systems implicated also a wide heterogeneity, which is present among the information systems of different organizations or even among the systems within the same company. So the stored data and working processes of organizations are divided into more non-natural parts by the frames of enterprise applications. The need for the reunion of this parts and the need for implementing complex business processes affecting lots of parts of different systems demands the integration of enterprise applications. The integration of enterprise application systems, often called as enterprise application integration (EAI) became one of the main challenges of the modern information sciences in recent years.

The service oriented architecture (SOA) and corresponding methodology have arisen only 4-5 years ago. In spite of this it revolutionized also the area of enterprise computing. Together with the methodology and implemented tools of SOA the former only in theory existing process based view and approach became a practical solution also in productive enterprise environments.

After all, it is obvious that the application of the SOA concept and corresponding tools can result a great step forward in the field of enterprise application integration. It may facilitate the data exchange both within one system in one organization and among different applications of several companies by the creation of composite processes. The SOA provides open standards and free accessible tools for enabling the standard enterprise applications for SOA based integration. Furthermore it makes possible to implement standard business processes realizing the integration of different software applications.
Open issues of the research area

The heterogeneity of used application systems also implicates the diversity of their interfaces. To solve this problem SOA provides proper methodological and tool support to convert (or adapt) the so far used standard interfaces and protocols into SOA compatible solutions. Well defined standards and protocols like the WSDL (Web Service Description Language) for the specification of Web service interfaces or the SOAP (Simple Object Access Protocol) for the messaging grant the common language and medium for the proper collaboration of enterprise applications. Nevertheless one can face with serious problems during the integration because of the differences in the data structure of different applications. These dissimilarities come from the diversity of used systems developed by various vendors.

Experts developing enterprise application systems often implement the same real word entities with different data structures. For example consider, that an address entity contains separated fields for storing the street and number of the address in application System A. Contrary to this it is possible, that in another system, called here System B, this information is presented in one common field, where actual values of street and number values are separated with the special character of semicolon. Consider, that service 1 of System A provides an output of address type, and service 2 of System B consumes an input of address type, come after each other within a composite process. Although their interfaces seem to be compatible for the first time, so service 1 could be connected directly to service 2 in this order, system B will not be able to resolve the gaps between the two different data representation structures and will not be able to process the message coming from System A correctly. Problems similar to this are often arising during the integration of enterprise applications. Thus solving these is a live and important task.

There are numerous approaches for process based integration of enterprise system. Most of these works strongly focus on the process orchestration i.e. the finding of proper services for given places of given processes during the integration [2][4][5][7]. There are more of them taking also into consideration data incompatibility problems like my example above [5][7].

Bridging of data representation problems is actually solved by defining transformations and gluing them to the given service interfaces. For example, if we consider the example described above, we need to define a transformation which concatenates the street and number values coming from system A in separated fields, and places a semicolon between the values before forwarding it towards system B. Another transformation is also necessary into the reverse direction. It must split the values of the address coming from system B at the semicolon into two separated values representing the street and
number before replicating it towards system A. Applying these transformations in this example the data incompatibilities of the two systems can be bridged in their communication.

There are various approaches for applying transformations in current literature as well. One of the most popular solutions is the semantic enrichment of service descriptions referring to the entities of special taxonomies e.g. ontologies. After that data incompatibilities can be easily detected and solved based on the extra information gathered from enriched service interface descriptions and using logical reasoning over the ontology. Most of these approaches have serious theoretical background [3][15][14] and mainly focus on the theoretical aspects instead of creating a solution working also in run-time.

On the other hand there are proposals optimizing the run-time aspects i.e. the improvement of non functional Quality of Service (QoS) parameters (for example availability, error-rate, response time of services) of composite processes in current literature [1][3][11].

Besides it is important to use existing standard tools (standard enterprise systems, process designers, process run-time engines, etc.) during the enterprise application integration. This prevents the usage of most semantic solutions because usually they do not have the proper implementation and solid software tools. Using of custom developed frameworks, which requires the continuous monitoring and dynamic re-connection of participating services for the run-time optimization is also not possible in EAI. Instead a solution built only on widely adopted standards and solid tools should be applied. This should also be able to deal with data transformations and must have only a slight overhead in the resource consumption compared to already existing standard solutions.

Estimating the values of non-functional QoS parameters – like response time and throughput – can also be found in many current researches. Some of these studies can also be applied for the run-time of composite processes in service oriented architecture based integration [6][16][17].

Besides creating a proper theoretical model for solving the problem of enterprise application integration it is also important to have an entire solution covering also the run-time of the integration. Hence after the definition of a correct methodology and complex architecture we also need a framework facilitating a solution working also in practice. Introducing a universal solution handling also data transformations, providing both the correct theoretical background and run-time solution would be a great breakthrough in the area of enterprise application integration in productive environment.

Furthermore taking also into consideration the optimization methods of non-functional QoS parameters found in the literature, the performance analyzez of
SOA based systems also containing proposals for the application of data transformations is a current issue. A universal model and methodology for examining and comparing of the performance of such proposal is also required.

Another complex problem of the area is the comparison of enterprise data schemas (i.e. the comparison of services interfaces of applications), which is a precondition of the transformation creation task. Because we can meet interfaces containing even 50-100 entities in the industrial area, finding of the corresponding entities – probably standing in the arguments of the same transformation rules – is also an important task. Finding and matching of the corresponding entities and possible creation of a schema compatible with all input/output schemas is called schema matching in the literature [8][10][12].

Integrating more schemas is usually realized by the definition of such a global schema for preventing the high number of necessary point-to-point alignment of interfaces. The service interfaces of participating enterprise applications should be aligned only to the global schema which means a much lower number of applied data transformations than in the point-to-point case. However mapping of interfaces to the global schema makes the comparing task of entities more difficult, because global schemas covering whole fields of application (domain) probably contain thousands of entities [18][19]. Hence alignment of enterprise service interfaces to a global schema is hardly performable for a human user without any (semi)-automatic computational support.

Schema matching approaches are mostly based on a method calculating the strength of the similarity of the entity pairs. The algorithm performing this typically requires an input containing of two schemas, and provides an output with similarity values between 0 and 1. Values close to 1 probably show entities which are semantically relevant. Thus by the creation of transformations we can limit our examinations only on that pairs of entities. In other words, these pairs of entities must be applied by creating the rules within the same transformations. The semantic not relevant pairs of entities (i.e. having much lower similarity values) are probably not required to be used as arguments of the same transformation rules. Moreover schema matching algorithms are also able to automatically determine a part of the global schema which must be used by the mapping of a given service of a specific enterprise application to the global schema.

There are many interesting researches and works published about schema matching approaches and similarity measurement methods. The wide heterogeneity and the high number of presented approaches definitely show that this is an important and live research area. Researchers present even novel
and novel algorithms and prove the efficiency of them compared to other published approaches based on widely accepted efficiency indicators [9][10]. The authors prove the effectiveness of their schema matching approaches mostly in practice. Several tests and experiments are run on more implemented test schemas and on schemas of productive enterprise applications as well [8][13].

One of the difficulties which hinder the spreading of schema matching solutions in practice is that most of the approaches require enormous high computational capacity to perform the task of similarity measurement. On one hand a lot of the publications focus on the accuracy of the algorithms. On the other hand computational requirements should also be evaluated because it essentially influences the practical applicability of a solution. A fairly accurate algorithm with not so high computational requirements could provide a real solution for everyday work in the preparation for SOA based integration.

The authors evaluate and compare the accuracy of their proposals mostly executed only one test scenario using only one given set of the parameters (like threshold values and weights). Considering the wide variety of the schema matching approaches and the heterogeneity of data schemas in enterprise applications it would be reasonable to create a universal method to determine the accuracy of different algorithms. The dependency on given input schemas and given configuration settings must also be taken into account.

**The aim and the method of my research**

Aiming to provide solutions for the problems and open issues of the research area the following goals were set for my work:

I intended to design and create a methodology and framework, which
- support the alignment of standard enterprise services to a global data schema,
- facilitates the application of transformations,
- is able to run transformations in run-time,
- has a predictable but not significant overhead, hence it does not show significant decreasing in the non-functional QoS parameters like throughput and response time compared with a standard SOA based collaborative process containing no transformations,
- is built on open standards and standardized solid tools exclusively, and
- does not require the modification of the standard enterprise application services or their interfaces during the integration.
Furthermore I intended to develop an efficient schema matching algorithm which can also be used in the everyday practice. It should be able to run on the input and output schemas of enterprise application services participating in SOA based integration scenarios. Hence I needed to create a solution working on data conforming XML i.e. XML schema standards and is able to find corresponding entities of given services and global schemas. Using an algorithm like this can be a great help for the experts preparing for the implementation of transformations bridging data incompatibilities.

Because there are a lot of proposals for schema matching in current literature it is obvious that the accuracy and computational requirements of my algorithm should be compared to other published solutions. I intended to perform this on various test schemas both in an analytical and in a practical – experimental – way as well.
Results

Part I.

Schema matching approaches, analyzing the efficiency and complexity of schema matching approaches

I.1. I have developed a universal methodology and algorithm for comparing entities of different data schemas

I have developed a method and algorithm for determining the strength of semantic correspondences of different entities. Hence data transformations are built upon these entities, performing of this task is a certain precondition of creating transformations. To support the alignment of local and global schema entities a function returns the desired similarity values in the form of a weighted sum of three sub-functions. The sub-functions are analyzing the names of entities, the similarity of the set of connected terms and compare the set of connected attributes. This approach is called shortly NTA (Name, connected Terms, Attributes). The approach NTA is applicable in various enterprise application integration scenarios to perform the comparison of service interfaces and provides valid result under custom circumstances.

The similarity of entities is calculated by the following function:

\[
F(C_i, G_j) = w_1 * N(C_i, G_j) + w_2 * T(C_i, G_j) + w_3 * A(C_i, G_j),
\]

where

\[
N(C_i, G_j)
\]

compares entities according their naming,

\[
T(C_i, G_j)
\]

compares the set of terms connected to entities \( C_i \) and \( G_j \).

\[
T(C_i, G_j)
\]

is calculated as follows:

\[
T(C_i, G_j) = \frac{2 * \left[ \| \text{Term}(C_i) \cap \text{Term}(G_j) \| + \| \text{Term}(C_i) \setminus \text{Term}(G_j) \| \right]}{\| \text{Term}(C_i) \| + \| \text{Term}(G_j) \|}
\]

\[
A(C_i, G_j)
\]

analyzes the set of connected attributes of entities \( C_i \) and \( G_j \).

\[
A(C_i, G_j)
\]

is calculated as follows:

\[
A(C_i, G_j) = \sum_{i,j} \frac{C(a_i, a_j)}{\| \text{Attr}(C_i) \| + \| \text{Attr}(G_j) \|}
\]

See details in Chapter I.1 of the PhD Thesis.

Corresponding results were published in [L2] and [K3].
I.2. I have deeply examined the most popular schema matching approaches. I have showed that their accuracy strongly depends on their configuration settings and applied input data. Therefore a fair method was created to analyze and compare their performance.

I have showed that schema matching algorithms are sensible for the given input and the working parameter settings in their accuracy. Thus a novel method is needed to compare them in a fair way.

To measure the efficiency the widely used and accepted measurement methods and metrics of current literature are applied. The measurement units are the accuracy, the recall and the F-measure\(^1\). The main idea of the method is that the partial optimum point of parameters must be determined first and this must be applied by calculating the final value of the efficiency. Finally, the evaluation of the accuracy can be performed by comparing these calculated values.

Using the novel method I have also showed that the provided results of my approach NTA – presented in I.1 –are probably more accurate than other current schema matching approaches.

The effectiveness and applicability of the approach NTA was also proved on the results of an experiment performed by the participation of 23 test subjects familiar with information sciences and enterprise computing in a cost model developed also by me.

The following table shows the values of accuracy (in F-measure) for more tested approaches and test scenarios by changing threshold values.

<table>
<thead>
<tr>
<th>Scenarios:</th>
<th>Company-customer</th>
<th>University-publication</th>
<th>Car-retailer</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NTA</td>
<td>SF</td>
<td>WN</td>
</tr>
<tr>
<td>Tre-NTA</td>
<td>0,2</td>
<td>0,04</td>
<td>0,15</td>
</tr>
<tr>
<td>Tre-Sim</td>
<td>0,25</td>
<td>0,05</td>
<td>0,175</td>
</tr>
<tr>
<td>Tre-WN</td>
<td>\textbf{0,3}</td>
<td>0,06</td>
<td>0,2</td>
</tr>
<tr>
<td>Tre-NTA</td>
<td>0,35</td>
<td>0,07</td>
<td>0,225</td>
</tr>
<tr>
<td>Tre-Sim</td>
<td>0,4</td>
<td>0,08</td>
<td>0,25</td>
</tr>
<tr>
<td>Tre-WN</td>
<td>\textbf{0,45}</td>
<td>0,09</td>
<td>0,275</td>
</tr>
<tr>
<td>Tre-NTA</td>
<td>\textbf{0,5}</td>
<td>0,1</td>
<td>0,3</td>
</tr>
<tr>
<td>Tre-Sim</td>
<td>0,55</td>
<td>0,11</td>
<td>\textbf{0,325}</td>
</tr>
<tr>
<td>Tre-WN</td>
<td>0,6</td>
<td>0,12</td>
<td>0,35</td>
</tr>
<tr>
<td>Tre-NTA</td>
<td>\textbf{0,65}</td>
<td>0,13</td>
<td>0,375</td>
</tr>
<tr>
<td>Tre-Sim</td>
<td>0,7</td>
<td>0,14</td>
<td>0,4</td>
</tr>
<tr>
<td>Tre-WN</td>
<td>0,75</td>
<td>0,15</td>
<td>0,425</td>
</tr>
<tr>
<td>Tre-NTA</td>
<td>0,8</td>
<td>0,17</td>
<td>0,475</td>
</tr>
<tr>
<td>Tre-Sim</td>
<td>0,85</td>
<td>0,17</td>
<td>0,475</td>
</tr>
</tbody>
</table>

See details in Chapter I.2 of the PhD. Thesis.

Corresponding results were published in [K1] and [K3].

---

\(^1\) Universal measurement unit of accuracy, calculated as the harmonic mean of accuracy and recall
I.3. I have created an analytic method to compare computational requirements of schema matching algorithms

I have created an analytic method to compare the complexity – i.e. the computational requirements – of schema matching algorithms. The complexity is estimated by the number of required computational steps. Calculating the number of computational steps it is possible to determine and compare the required computational costs of different approaches. The model was validated by sample implementations and run-time measurements run on various test data.

Using this method I have also proved, that approach NTA requires much lower computational costs to calculate results than other approaches found in current researches.

The calculated number of expected computational steps was verified by implementing more test schemas and the order of difference in the number of steps proved to be the same as the order of difference in the actual run-times measured in run-time experiments.

The computational complexity of approach NTA can be calculated by the following expression:

\[ \text{Steps}_{\text{NTA}}(C, G) = \text{Steps}_{\text{NTA-Perform}}(C, G) + \text{Steps}_{\text{NTA-Prepar}}(C, G) = \]
\[ = \sum_{i,j} 1 + \|\text{Term}(C_i)\| + 1 + \|\text{Term}(G_j)\| + 1 + \|\text{Attr}(C_i)\| \times \|\text{Attr}(G_j)\| + \]
\[ + \sum_{i} \|\text{Attr}(C_i)\| + \sum_{j} \|\text{Attr}(G_j)\| + \|C\| + \|G\| + \|C\| \times \|\text{Term}(C_i)\| + \|G\| \times \|\text{Term}(G_j)\| \]

The following table shows the measured run-time values in seconds. These values proved the correspondence between the number of computational steps and the run-time costs.

<table>
<thead>
<tr>
<th>Test scenario\Approach</th>
<th>NTA</th>
<th>Sim. Flood</th>
<th>WordNet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample vs Sample</td>
<td>0,03</td>
<td>0,06</td>
<td>0,08</td>
</tr>
<tr>
<td>Example1 vs Sample</td>
<td>0,03</td>
<td>0,06</td>
<td>0,08</td>
</tr>
<tr>
<td>Example1 vs Example1</td>
<td>0,03</td>
<td>0,08</td>
<td>0,11</td>
</tr>
<tr>
<td>Example2 vs Sample</td>
<td>0,08</td>
<td>0,63</td>
<td>1,00</td>
</tr>
<tr>
<td>Example1 vs Example2</td>
<td>0,08</td>
<td>1,11</td>
<td>1,95</td>
</tr>
<tr>
<td>Example2 vs Example2</td>
<td>0,92</td>
<td>25,94</td>
<td>98,23</td>
</tr>
<tr>
<td>Example3 vs Sample</td>
<td>3,56</td>
<td>41,91</td>
<td>101,88</td>
</tr>
<tr>
<td>Example1 vs Example3</td>
<td>2,52</td>
<td>406,48</td>
<td>159,06</td>
</tr>
<tr>
<td>Example2 vs Example3</td>
<td>40,52</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Example3 vs Example3</td>
<td>2264,84</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

See details in Chapter I.3 of the PhD Thesis. Corresponding results were published in [K2].
Part II.
Service oriented architectures, analyzing the performance of given approaches based on service oriented architecture

II.1. I have developed a SOA based solution and framework for handling data incompatibilities during enterprise application integration. My proposal is built on open standards and corresponding tools exclusively.

I have developed a methodology and framework for handling mismatches in data representation by applying transformations during composite process creation. It makes possible to adopt existing services of standard enterprise applications into SOA based integration scenarios. The proposed solution is modular, developed transformations and service interfaces are recyclable while interfaces of standards enterprise application services can remain intact. It uses open standards and corresponding free tools exclusively.

The practical applicability of the solution was proved by constructing a working framework and solving various test problems with it.

The main idea behind my approach is the encapsulation of standard services. The invocation of an encapsulated service is shown below:

See details in Chapter II.1 of the PhD Thesis. Corresponding results were published in [L3], [K4], [K5], [K6], [K7], [K8] and [K9].
II.2. I have developed a novel methodology for analyzing and compare the run-time performance of SOA based application integration solutions

I have developed an analytical and practical methodology for analyzing the run-time performance of SOA based application integration solutions also containing transformations to bridge incompatibilities in data representations. The novelty of my approach is the separated handling of resource requirements needed by the execution of transformations, by instantiating the processes and by performing the standard enterprise services. Furthermore I proposed also to separate the resource requirements between the enterprise application server and the process run-time server. The reference throughput and response time of given configurations must be determined first. Hence the expected response time and expected number of served request during a given time period is already calculable.

The method was validated by various tests results and test scenarios in practical experiments.

The response time of my proposed framework by a typical load can be estimated by the following expression:

\[
\sum_{\text{path}} T_{\text{service}} + \sum_{\text{path}} T_{\text{add}} + T_{\text{constructs}} < T_{\text{response}} < \left( \sum_{\text{path}} T_{\text{service}} + \sum_{\text{path}} T_{\text{add}} \right) \frac{\sum R_{\text{orchest}}}{R_{\text{avail,enterprise}}} > \sum R_{\text{constructs}}
\]

The throughput of the process server can be described by the following expressions:

\[
T_{p_{\text{Process}}} = T_{p_{\text{Process, max}}} \frac{\sum R_{\text{process}}}{R_{\text{avail}}} \quad \text{ha} \sum_{m_{conc}} R_{\text{process}} \leq R_{\text{avail}}',
\]

\[
T_{p_{\text{Process}}} = T_{p_{\text{Process, max}}} \quad \text{ha} \sum_{m_{conc}} R_{\text{process}} > R_{\text{avail}}',
\]

\[
T_{p_{\text{Process}}} < T_{p_{\text{Process, max}}} \quad \text{ha} \sum_{m_{conc}} R_{\text{process}} >> R_{\text{avail}}'.
\]

See details in Chapter II.2 of the PhD Thesis. Corresponding results were published in [L1], [K4] and [K5].
Applicability of the results

The methodology summarized in Thesis I.1 can be used to compare entities of data schemas and to define the strength of connections among them also in productive systems. The created prototype implementation can be extended and exchanged by an implementation in specific enterprise application environment based on the detailed description of the methodology and algorithms in the PhD Thesis. Hence it can also be applicable in the comparison of entities by the preparation for enterprise application integration in productive environment.

The solution presented in Thesis I.2 can be used to determine and compare the accuracy of specific schema matching approaches. The expected accuracy level can be estimated and algorithms can also be customized (optimized) for given productive environment and given input schemas of enterprise applications.

Based on the Thesis I.3 we can estimate the computational requirements of different schema matching approaches for given input complexity. This facilitates the sizing of hardware components for schema matching solutions, which is an important task by putting software solutions in practice in a productive environment.

The presented SOA based integration methodology and created framework is able to solve enterprise application integration problems in small sized organizations. Furthermore after implementing more effective tools for the framework it also provides an applicable solution for performing collaborative processes of larger applications. The methodology and the framework presented in Chapter II.1 of the PhD Thesis can also be realized by solid, industrial tools which provides a solution for data incompatibilities and SOA based integration problems of any size.

The solution presented in Thesis II.2 makes possible to analyze the run-time performance of specific SOA based integration implementations in enterprise application integration. The productive applicability and hardware sizing is supported by the analytical estimation of resource requirements and by the calculation of non-functional QoS (Quality of Service) parameters based on reference measurements. The necessary methods and tools for a possible productive applicability are all presented in the PhD Thesis.
Future work

There are many areas of connecting researches and possible future work:

- after determining the strength of semantic correspondences between schema entities the implementation of the actual data transformations can also be supported under specific circumstances. This can help or even totally save the task of transformation creation performed mostly by a human expert today.

- The run-time performance of the presented schema matching approach can also be improved by taking into account some local structural aspects of the specific input data – e.g. the comparing of entities found in nodes of very different level of abstractions can be passed by during the tree traversal of schemas.

- The presented solution for analyzing and optimizing run-time performance can be extended by cluster-system applications where the continuous monitoring of loads facilitates the reallocation of available resources among the cluster components implicating an improvement in the overall system performance.
The list of corresponding publications can be found below:

Papers published in Hungarian journals in English:


Papers published in international journals in English:

Papers published in proceedings of international conferences in English:


Further publications

Papers published in Hungarian journals in English:

Papers published in proceedings of international conferences in English:

Electronically distributed laboratory guide in Hungarian:
References


