Towards an interface-based web service registry

Allard Brand
University of Twente
Overijsselaan 5, 7543 WC, Enschede
The Netherlands
a.brand@student.utwente.nl

ABSTRACT
Two of the problems faced in the process of data integration nowadays are the lack of uniformity in the interfaces of similar web services, and the discrepancies in the underlying semantics of data. This paper researches the question whether an interface-based web service registry could help improving the data integration process by providing a transparent platform to register and lookup web service interfaces, as well as web services implementing them. Interviews with several experts made clear that although the benefits are evident, a lot of barriers exist and need further research. We give functional requirements for the registry that are needed at least for it to be successful.

Keywords
Interface-based web service registry, SOA, functional requirements, similar web services, UDDI

1. INTRODUCTION
As computer and information technology develop, interconnection between various information systems is becoming an integral part in the information strategy of today’s enterprises. It is hard to imagine an enterprise in the 21st century without its supportive database and information systems. Whereas in the early days enterprise information technology mainly consisted of ‘stand alone’ programs, information systems nowadays start to rely increasingly on the exchange of data with other applications.
A lot of techniques such as SOAP, REST, RPC, etc. are developed to provide a technical basis to meet this need of data and system integration. However, the integration of information systems and their data still raises several issues. Consider for example the issue of discrepancies in the semantics of data. Information systems model the containing data in their own, specific ways. When attempting to integrate the data of these systems, a common problem is the mismatch between their specific datamodels.

As an example, consider the length of an holiday. One system could model this as the number of days, while the developers of another system could have made the decision to model the same attribute as the number of nights. This might not sound like a big problem, but in reality these small discrepancies - as well as more complex issues - are the basis of long and expensive integration processes [11].

One of the main problems following on this is that although a lot of web services providing the same kind of service exist, their interfaces and underlying datamodels often differ. It is therefore not possible to access these web services in a uniform way, increasing the time and money needed in the integration process when working with multiple equivalent service providers.

Therefore, concrete and obvious agreements of the semantics of data have to be made between the various parties in the integration process. At the moment no central platform for organizing these standard interfaces and the web services implementing them exists. As a solution to this problem, the idea of a central registry for the organization of these interfaces and the web services implementing them arose, which will be defined in section 1.2. The goal of this paper is to research the usefulness of, and the requirements for such a registry.

1.1 Definitions
Before working out the idea of an interface-based web service registry in more detail, we will first introduce some definitions which will be used throughout this paper:

Web service: A software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format. Other systems interact with the web service in a manner prescribed by its description [13].

Interface: The technical specification of a web service. A commonly used format for these descriptions is the ‘Web Services Description Language’ (WSDL) [12].

Web service providers: The provider of a web service, publishing one or more services for web service consumers to access and use.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

14th Twente Student Conference on IT Januari 21st, 2011, Enschede, The Netherlands.
Copyright 2011, University of Twente, Faculty of Electrical Engineering, Mathematics and Computer Science.
An association between a web service interface binding: web service provider to exchange data. Web services, making use of the web service published by a web service provider, can only be in a transparent organization of web services based on the interfaces, which explains the use of the term ‘interface-based’ in the title of this paper.

1.2 Interface-based web service registry

The main idea of the interface-based service registry as aimed at in this paper, is a central platform to facilitate both the registry and lookup of web service interfaces as well as web services implementing those interfaces. First of all, interface descriptions can be registered at any moment in the registry. Web service providers can lookup the these interfaces, implement the most popular interfaces, and subsequently register their web service under the interface being implemented. Web service consumers can lookup both interfaces as well as web services implementing them. Using this approach, such a registry provides a transparent organization of web services based on the interface being used.

The basic structure of the registry is shown in figure 1. As can be seen in this figure, the registry will contain interfaces, web services and bindings. The main functionality exists of the registry and lookup of both interfaces as well as web services implementing them.

An important aspect is that a web service will only be included in the registry if it conforms to one or more specific interfaces. In this case a binding between the interface and the web service will be added to the registry, available for others to be found. This means there will not be any web services in the registry ‘unbound’ to one or more of the interfaces, which explains the use of the term ‘interface-based’ in the title of this paper.

At this moment only a preliminary model for the registry is given. Throughout this paper we will work towards a clear view on how such a registry should function, resulting in a list of functional requirements.

2. RESEARCH

2.1 Research questions

The main research question is as follows:

- Can the process of data integration with respect to web services be improved by using an interface-based web service registry?

In order to answer this question, this research question will be split up into three subquestions:

- Which problems are being faced in the data integration process nowadays?
- Could an interface-based web service registry be a possible solution for these problems?
- What functional requirements are needed for such a registry to function?

2.2 Research method

To get insight in the current approaches in respect to web service registries, a literature study has been done. In order to answer the research questions stated above, several experts have been interviewed.

Due to the limited time available for this research, a case-based approach has been chosen. We will focus us on the touristic line of business and answer the research questions for this specific area. The touristic line of business has been chosen because of its complexity on one hand, and the need for improved data integration on the other hand.

In order to get insight in the current situation in the touristic line of business, we chose to interview two experts working in leading companies on the area of tourism and holiday specific software:


AICN Automatisering is a company that delivers software applications for use in touristic line of business and is one of the main players in this market. TSI-Solutions delivers a range of software solutions for the same line of business, but was mainly chosen because of its intermediary role. By means of example, TSI-Solutions delivers a platform to facilitate the integration of separate and various systems and applications in the touristic line of business.

To take a first step in getting insight in other lines of business as well, an expert working in the financial insurance line of business has been interviewed:

- Geert Vegter [10] Manager Information Management at Unive head-office, Assen

3. RELATED WORK

Currently a number of web service registries already exist. Three well-known web service registries are UDDI [7], ebXML [6] and Seekda! [8].

UDDI defines a standard method for publishing and discovering the network-based software components of a service-oriented architecture (SOA) [7]. The public UDDI registry, known as UBR, is no longer in use. The reason for this shutdown, as given by Microsoft, is that the primary goal of proving the interoperability and robustness of the UDDI specifications, was met and far exceeded. The UBR ran for 5 years, managing over 50,000 entries. Currently UDDI is generally used inside an organization, to catalog their services [5].

The ebXML Registry is similar to UDDI in that it allows businesses to find one another, to define trading-partner agreements, and to exchange XML messages in support of business operations [2].

Seekda! helps you to find Web Services based on a catalogue of more than 28,000 service descriptions. Services listed at seekda cover a wide range of functionality [8].

These approaches all focus on the registration and discovery of separate web services, often with their own specific interfaces. The interface-based service registry as aimed at in this paper differs from this approach in the important way that it focusses on interfaces. Instead of merely classifying web services and offering a catalog of these services, the goal is to classify web services based on the interface which they conform to.

We are not aware of any earlier work using this same interface-based approach for specifying a web service registry.

4. RESULTS

The following subsections will review and discuss the gained insights from the expert interviews.
4.1 Current problems
Both Eerko Vissering and Michel Leliveld acknowledged both the actuality of the data integration process and the presence of problems in this process. To paraphrase Michel Leliveld about the actuality of data integration: ‘The market of data integration is growing fierce, you can’t get away from it’ [4]. As stated in the introduction, discrepancies in the semantics of data turned out to take up the biggest part of these problems.

Consider the following statement from Eerko Vissering (paraphrased): ‘Even though clear agreements on web service interfaces have been established, it often occurs that parties still carry their own interpretation of the semantics of the data’. These differences in interpretation vary from the obscurity of the date format used to the difference of opinion in complex price-structures of holidays.

According to all three experts interviewed, the reason behind this problem is that organizations often use their own systems, sometimes consisting of old mainframes with various layers built on top of it. These systems work on basis of their own specific data models, with assumptions on how to interpret and model certain data. For example, some systems work with weeks starting on Sunday, while other systems work with weeks starting on Monday. This can have huge effects when for example working with week-pricing. The same is true when considering systems working with day-prices in comparison to systems using night-prices to calculate the cost of a holiday reservation.

Another example of a problem which often occurs is the question ‘where’ to save or apply certain attributes. Geert Vegter illustrated this by raising the question where to save bank account information when issuing an insurance policy: on customer or policy level? Again, the consequences of different choices in aspects like these can be significant in the process of data integration.

Lastly, it became clear that often months go by before agreements on questions like the above are settled and integration can commence. Obviously, this process of agreeing on interfaces and the semantics of data often consumes a lot of time and money. Both Eerko Vissering and Michel Leliveld agreed that efficiency in this process would be more than welcome.

4.2 Current solutions
When looking at the problems during the data integration process as described in the previous section, current solutions have to be mentioned as well to get a more complete view of the current situation.

One of the major solutions to facilitate efficiency in the process of data integration can be found in specialized companies, functioning as intermediary party between web service provider and consumer. Companies among which TSi-Solutions deliver web services integrating data from various organizations and offering this integrated data for other companies to use. Subsequently these companies can suffice with implementing this single web service, instead of having to implement the various interfaces themselves. HotelsCombined [3] delivers the same kind of service by integrating data from thirty of the most popular tour operators, and publishes this as their own web service. Again, implementing the single interface provided by HotelsCombined will suffice to access the data from all affiliated tour operators.

Another great step forward can be seen in the area of standardization, for example when looking at the G7-approach [11] in the touristic line of business. The G7 is an organization representing the seven considerable big tour operators in the Netherlands. In this organization the so-called G7-protocol is being developed to work towards standardized agreements on interfaces and semantics of data, used for publishing and obtaining data on availability and the process of searching and booking a holiday. Every time when a new version of the protocol is being agreed on, it is sent to all members of the affiliated touristic branch organization (the ANVR [1]). These affiliated members have to conform to this agreed protocol, and implement the described interface. The technical implementations are often provided by TSi-Solutions.

Interestingly, when looking at the financial assurance market, the same approach can be found, seeming to be even more exploited. SIVI, a Dutch abbreviation for what could be translated as ‘Standardisation Institute for Insurance Underwritings in the Intermediary branch’, develops and supervises standards and technology on behalf of chain integration in the insurance line of business [9]. So far, SIVI delivered four standards which are fully documented and publicly accessible on their website. SIVI also delivers basic online verification of implementations, and certification after integral verification to guard the correctness of implementations.

4.3 Opportunities
In order to get insight in the opportunities for a central interface-based web service registry, we asked the three experts about their views on such a registry. The main question was whether this registry would function, and if so, under which circumstances. The answer on this question will be reviewed in this and the following sections.

Both Eerko Vissering and Michel Leliveld realized and agreed on the advantages of an interface-based web service registry. First of all, such a registry could improve awareness of available standard interfaces and web services implementing these interfaces. This would make it easier to decide which interface(s) to implement, but also for companies to find other businesses and their web services to interact with.

Also, such a registry would give more insight and understanding into available web services, when comparing to WSDL. Available methods, and more importantly, the semantics of data used in their in- and outputs, could (and should) be added to web service entries in the registry.

The major advantage would be the improvement in efficiency with respect to the process of data integration. As explained in section 4.1, the process of agreeing on interfaces and semantics of data often consumes a lot of time and money. A well-known, trusted and central registry could reduce this time span, because insight is given in interfaces and agreements already available.

4.4 Barriers
Although the advantages of an interface-based web service registry became clear, all three experts stated various barriers and pitfalls that could threaten the effectiveness of a central interface-based web service registry. Of these barriers, the diverse interests of the stakeholders are the most important.

Before continuing, it is important to consider the three main stakeholders in the touristic line of business. First of all, there is the delivering party, offering holidays. The second party are the affiliates, selling these holidays on the delivering party’s behalf. Lastly, software integration parties like TSi-Solutions and HotelsCombined exist to integrate both the delivering and affiliate party, and act as
a intermediary.

All stakeholders have their own interests. When focussing on the delivering party, two visions can be noticed. In some cases, the delivering party wants their data to be available as far and much as possible. However, in other cases, the delivering party wants to be able to keep track and control of where and with whom data is being exchanged. In this case contracts are often being used, and data is exchanged with contracted parties only. When considering this last case, the delivering party would probably make no use of a registry to publish their web services, because it would make their services freely accessible to everyone.

When considering the affiliate party, another problem arises. Current players often have invested considerable amounts of time and money in the specification of interfaces being used, especially when looking at standardization projects. The question is whether they want other parties to take profit of this.

Lastly, another barrier indicated by Geert Vegter applies to all three stakeholders. In order to make centrally led initiatives work effectively, often a lot of time and money has to be invested. The question however is whether all these efforts are really worth the benefits. To paraphrase one of Geert Vegter’s statements concerning this question: “Do not expect too much from centrally led initiatives, but expect at least as much, if not more, from the natural growth of dominant market players. The interfaces delivered by these companies often become ‘de facto’ (unofficial) standards. The rest of the market will usually conform to these standards”. The fact that meanwhile a lot of interfaces and agreements already exist and are in use, increases the significance of this question.

4.5 Discussion

As reviewed in the previous sections, both advantages and barriers can be raised when considering the usefulness of an interface-based web service registry. In this section we will discuss the importance of these barriers, and possible solutions to soften these.

When considering the barrier of delivering parties that want to keep control and track of where and with whom data is being exchanged, and work on basis of contracts, some interesting things can be said. Usually the fact that parties want to keep control in their own hands, does not mean that these parties do not want to exchange their data at all. The main problem is that they do not want their services to be freely accessible. A solution to this problem can be found when introducing different possible publishing levels. For example, the endpoint of a web service could remain hidden until a contract has been agreed. In this way, insight is given that a company vice could remain hidden until a contract has been agreed. Another idea, proposed by Geert Vegter applies to all three stakeholders. In order to make centrally led initiatives work effectively, often a lot of time and money has to be invested. The question however is whether all these efforts are really worth the benefits. To paraphrase one of Geert Vegter’s statements concerning this question: “Do not expect too much from centrally led initiatives, but expect at least as much, if not more, from the natural growth of dominant market players. The interfaces delivered by these companies often become ‘de facto’ (unofficial) standards. The rest of the market will usually conform to these standards”. The fact that meanwhile a lot of interfaces and agreements already exist and are in use, increases the significance of this question.

The registered web services should be grouped accordingly on the delivering party, two visions can be noticed. In one of Geert Vegter’s statements concerning this question: “Do not expect too much from centrally led initiatives, but expect at least as much, if not more, from the natural growth of dominant market players. The interfaces delivered by these companies often become ‘de facto’ (unofficial) standards. The rest of the market will usually conform to these standards”. The fact that meanwhile a lot of interfaces and agreements already exist and are in use, increases the significance of this question.

5. FUNCTIONAL REQUIREMENTS

As became clear in the previous section, quite a lot of barriers can be raised in respect to the usefulness of an interface-based web service registry. Therefore, the question whether the proposed registry would be successful needs further research. However, due to limited time, we will continue and try to use the gained insights in formulating a list of functional requirements. These requirements define the criteria to which the registry should at least conform to be successful, and will be reviewed in this section.

When looking at these requirements, a good starting position is to look at the functionality of current web service registries. All three web service registries mentioned in section 3 provide the basic functionality to register and lookup web services and give insight into information about the provider. Interestingly, UDDI and ebXML provide web services to provide this functionality. Seekda! is more web oriented and offers a web portal only. A web portal increases the simplicity and accessibility, giving end users simple and user-friendly insight into available web service interfaces and the web services implementing them. On the other hand, using a web service to lookup web services has its own advantages. Applications can automatically include new web services by requesting all web services implementing the correct interface from the registry. Both approaches are desirable and should be included in the registry.

Another interesting functionality which can be seen at Seekda! is the online tester tool. Without the need of writing a single line of code web services can be called by filling in the necessary parameters, and the response can be examined in a user-friendly format. This functionality seems very useful, and could be very potential when combined with the option to publish a test service.

As stated in section 3, the main difference between current web service registries and the registry as aimed at in this paper is the focus and grouping on interfaces. The registry should therefore support separate registration of web service interfaces and the web services implementing them. The registered web services should be grouped accordingly to the interface implemented. To ensure only web services implementing the related interface can be registered, the registry should support basic validation of methods and their in- and output.

Because the importance of the correctness in semantics of data, extensive documentation for describing these semantics have to be supported, and validation also has to occur on semantic level. Because of the complexity on
one hand, and the importance of correct semantics on the other hand, this validation process should be done manually. The registry should however support the assignment of a quality inspection mark, guaranteeing the correctness of the web service.

Summarizing the above, the following functional requirements have been drawn up:

- The registry should support the registry of web service interfaces, including method descriptions and documentation about the semantics of data being used.
- The registry of web service interfaces should only be available to authorized representatives from each line of business.
- The registry should support the registry of web services, listing the web service under one of the web service interfaces available in the registry.
- The registry should only include web services after basic validation of the interface being implemented.
- The registry should support the lookup of web service interfaces and the web services implementing them through both a web-based portal and a web service itself.
- The registry should support the assignment of a quality inspection mark to specific web services to guarantee the correctness of these web services with respect to the semantics of data being used.
- The registry should support various publishing levels. When a web service is registered, a choice has to be given whether to publish the end-point (specific location for accessing the web service) of a web service or not.
- The registry should support the addition of a test service when registering a web service. This test service should work exactly the same (and thus also conform to the interface) as the real service, but using test data instead.

6. CONCLUSIONS

As an answer to the research questions stated in section 3, we can conclude that the problem described in the introduction is present and actual in both the touristic and financial insurance line of business.

Current registries are far from sufficient, because of their weak organizational basis and lack support for description of semantics.

An interface-based web service registry has a lot of opportunities. However, a lot of barriers have been perceived. The question whether a registry as aimed at in this paper could be successful turned out to be more complex than expected, and still needs further research. In this paper we gave the first steps in answering this question. Insight in the current situation and expected barriers has been given and discussed.

Lastly, functional requirements have been identified, to which a registry as aimed at in this paper should conform at least. These requirements can be found in section 5.

The main conclusion established during this research is that the presence of a strong organizational basis is the most important factor in determining the opportunities for an interface-based web service as aimed at in this paper. Again, further research is needed to get insight in how to reach this. Other lines of business have to be examined as well, where the outcomes might be different.

The list of requirements given in the previous section is not complete. The requirements listed are still rough and meant as a starting point. Further research could result in additional requirements.

7. ACKNOWLEDGEMENTS

I would like to thank Maarten Fokkinga and all the participants of the track B for their invaluable feedback and suggestions during this research.

8. REFERENCES