Computers in Human Behavior xxx (2014) xxx–xxx

Contents lists available at ScienceDirect

Computers in Human Behavior

journal homepage: www.elsevier.com/locate/comphumbeh

Interoperability between platforms without a defined referential model: A semi-automatic learning system for structural pairing

Secundino José González Pérez *, Benjamín López Pérez, Daniel Machado Fernández, José Emilio Labra Gayo

University of Oviedo, Spain

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
Interoperability
Web Services
String Metrics
eGovernment
XML

ABSTRACT

This paper deals with the exchange of information between universities problematic, but could influence other sectors like banking, health, etc. where there are not standards defined for information exchange. We propose a semi-automatic learning and assessment system that is capable of unifying the way in which each of the universities work by employing a three-step system that will be based on the exploitation of the information offered by the web services in their WSDL files and the way in which each of the universities that had been previously evaluated work. This three-step system would be focused on: (a) taking advantage of the information that is obtained on a structural level from the WSDL, which, using the String-Metrics (Cohen, Ravikumar, & Fienberg, 2003) application, will be able to connect the structures between universities, (b) reinforcing any prior knowledge with a classification system for cases that could not be matched, and (c) applying a reasoning system or rules to support the previous two steps. The main result obtained is a system that is able to interoperate two or more platforms between themselves to automatically share information in spite of their structural difference.

1. Introduction

We must bear in mind that, throughout the past decade, the World Wide Web (WWW) has evolved from a static and monolithic Web, with hypertext and hyperlink applications—or, in other words, static HTML—towards the Web 2.0 (known for its social networks and degree of participation), a Semantic Web. Moreover, we have been hearing people talking about an Internet of Services and Things, where everything is or will be connected with and between everything. A similar tendency is the symbology of the electronic government or administration, which was barely known a decade ago as a concept, an identified activity and a research matter (Heeks & Bailura, 2007), is explicitly known as what "would have seemed as a utopia if dreamed barely a decade ago" (Garson, 2004). The lack of interoperability appears, in this context, as the most enduring and hard issue that our enterprises and government institutions have to face today; due to the development of propietary software or extensions of the current information systems, the lack or excess of standards, and the heterogeneity of the software and hardware platforms (The Yankee Group Report, 2003).

Recognizing the benefits of improving the efficiency of providing government services, through electronic means, the initiatives of Electronic Administration have quickly spread during the past few years (Weerakkody, Choudrie, & Currie, 2004).

In the field of e-government, the i2010[1] initiative, the strategic action plan of the European Committee (CEC, 2006a), presents interoperability as a pre-requirement so that "devices and platforms can talk to each other" and so that "services become portable between different platforms", and identifies it as one of the main building blocks for the common European space for eServices information (CEC, 2006b).

In fact, the establishment of a Pan European system with cross-border interoperability is a key element and a necessary condition for every electronic government initiative of the EU, such as the European Framework of Interoperability (EIF, 2010), the Service Directive 2006/123/CE[2] (European Commission, 2007) and the

* Corresponding author at: Campus de Barredo (3° planta) | Gonzalo Gutiérrez Quirós s/n, 33600 Mieres, Asturias, Spain. Tel.: +34 985458091.

E-mail addresses: cundi@uniovi.es (S.J. González Pérez), benja@uniovi.es (B. López Pérez), danielmf@innova.uniovi.es (D. Machado Fernández), labra@uniovi.es (J.E. Labra Gayo).

[1] http://europa.eu/legislation_summaries/information_society/strategies/c11328_en.htm retrieved August 15, 2014.

[2] http://europa.eu/legislation_summaries/employment_and_social_policy/job_creation_measures/l33237_en.htm retrieved August 15, 2014.

http://dx.doi.org/10.1016/j.chb.2014.11.011
0747-5632/

Directive 2013/37/UE[3] of the European Parliament and of the Council of June 26, 2013, which modified the Directive 2003/98/CE[4] related to the reuse of information in the public sector.

The Ministers of the European Union (EU) in charge of the policies of Electronic Administration in 2009, and other allies of the EU, adopted unanimously the Declaration of Malmö,[5] which aims at taking advantage of the potential of ICTs as catalysts of a greater degree of transparency and participation of citizens in their relations with the public administrations. Its main contribution is the strategic boost it provides to the so-called third-generation services, directed at satisfying the needs of the citizenship, who assumes a central role in its definition and design. Its objectives include the promotion of the reuse of public data; the multi-channel attention; the implication of the citizens in the process of establishment of public policies; the simplification of procedures and the reduction of administrative loads.

In turn, the Declaration of Granada of April 2010[6] constitutes the first reference draft for the establishment of the European Digital Agenda,[7] incorporating the principles related to the electronic administration to the Declaration of Malmö. Therefore, this agenda is also one of the most iconic initiatives of the Europe 2020 strategy, whose aim is to develop an ambitious plan that seeks to articulate the European economy also in the Web and in a common digital market.

On a national level, Law 11/2007 LAECSP "Law for Electronic Access of Citizens to Public Services"[8] was a law that made it compulsory for Public Administrations to offer their services to citizens via online means. For obvious reasons, such as the economic recession and crisis that we are going through, the implementation of this Law, which was compulsory since January 1, 2010, was relaxed and the administrations began to adapt to it progressively as they were able to.

Electronic procedures and interoperability demanded the need to change the administrative procedures (Steward & Walsh, 2007) (Law 30/1992, of Legal Regime of Public Administrations and Common Administrative Procedure)[9] to which we were used to, of which most were conducted on paper and in person with a civil servant, which in turn often required long waiting queues for certain services, and sometimes even redundant information that the administration already had, such as a photocopy of the national ID; and current payments of our expenses. This led to changes in 2013, when a reformation of the education system took place (Law 7/2013),[10] whose modification involved the creation and simplification of procedures, which in turn made the citizens avoid having to present information that, in some way or another, was already in the hands of the administration or public bodies themselves. Here is where the new concept of Administrative Interoperability was born, since it demands that different administrations are in contact with each other to exchange information so that the citizens do not have to present redundant information, and that they are requested permission to use this information by the corresponding administration in a case per case basis. Previously, it also required the creation and approval of the National Security Scheme (NSE) and the National Interoperability Scheme (NIS)—Royal Decree 3[11] and Royal Decree 4[12] in 2010—which establish the technical regulations of security and interoperability of information. The National Schemes are very advanced and demanding regulations whose goal is to create objective conditions of interoperability and security that facilitate the exercise of rights and the compliance of duties through electronic means.

The NIS is defined in Section 1 of article 42 of Law 11/2007 as "...the set of criteria and recommendations in terms of security, conservation and normalization of the information, the format and the applications that must be taken into account by the Public Administrations to make technological decisions that guarantee interoperability."

Moreover, the Technical Regulations of Interoperability[13] developed concrete aspects of different matters that are required to ensure the most practical and operative aspects of the interoperability between the Public Administrations and the citizens. The set of regulations includes the standards of electronic documents and record; digitalization of documents; policies of electronic signature and certificates of the Administration, and even the way that information exchange files should be handled, which are not more than the standard definition of the XML file structure. All of this is not enough for the exchange of information between bodies—such as, for example, the Universities that want to exchange the academic record of a student who is changing from one University to another: currently, this procedure must be made with the academic record printed on paper. If we want the procedure to be conducted electronically, then we would require the creation of common information structures, such as the standard proposed for the e-EDS[14] (Electronic European Diploma Supplement) [15] and through a layer of services, information may be exchanged so that Universities retain the same system of information, but being able to interact with a service layer to exchange the academic record. This service layer must have a certain intelligence or semantics to interpret the data model of an academic record or, even, to interpret the information of the record. The aforementioned proposal is currently being developed in the Platform for Electronic Administration through the SUE[16] interoperability node project, which aims at interoperate the different universities between themselves and with other administrative services, like the Income, Revenue and Public Administrations Service (MINHAP).

2. Background and related work

Software has been evolving in many dimensions: the programming tools and languages that are used to create software; the ways in which we approach the process of creation; the models that describe the elements of software and the way they relate and interact with the real (or virtual) world; their complexity and frontiers, from small programs of a few hundred lines of code that are executed in a computer to gigantic distributed systems to

[3] https://www.boe.es/diario_boe/txt.php?id=DOUE-L-2013-81251 retrieved August 15, 2014.

[4] http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2003:345:0090:0096:ES:PDF retrieved August 15, 2014.

[5] http://ec.europa.eu/digital-agenda/sites/digital-agenda/files/ministerial-declaration-on-egovernment-malmo.pdf retrieved August 15, 2014.

[6] http://www.minetur.gob.es/es-es/gabineteprensa/notasprensa/documents/declaraci%C3%B3ncastellano.pdf retrieved August 15, 2014.

[7] http://ec.europa.eu/digital-agenda/ retrieved August 15, 2014.

[8] http://www.boe.es/diario_boe/txt.php?id=BOE-A-2007-12352 retrieved August 15, 2014.

[9] http://noticias.juridicas.com/base_datos/Admin/l30-1992.t6.html retrieved August 15, 2014.

[10] http://noticias.juridicas.com/base_datos/CCAA/517405-l-7-2013-de-21-nov-ca-castilla-la-mancha-adecuacion-de-procedimientos-administrativos.html retrieved August 15, 2014.

[11] http://www.boe.es/buscar/doc.php?id=BOE-A-2010-1330 retrieved August 15, 2014.

[12] http://www.boe.es/buscar/doc.php?id=BOE-A-2010-1331 retrieved August 15, 2014.

[13] http://administracionelectronica.gob.es/pae_Home/pae_Estrategias/pae_Interoperabilidad_Inicio/pae_Normas_tecnicas_de_interoperabilidad.html#.U-3AsmAcTIU retrieved August 15, 2014.

[14] http://ecctis.co.uk/europass/documents/Europe%20Unit%20Diploma%20Supplement%20Guide.pdf retrieved August 15, 2014.

[15] http://ec.europa.eu/education/tools/diploma-supplement_en.htm retrieved August 15, 2014.

[16] https://www.rediris.es/jt/jt2013/ponencias/jt2013-jt-sesion1b-a10b4c1.pdf retrieved August 15, 2014.

Please cite this article in press as: González Pérez, S. J., et al. Interoperability between platforms without a defined referential model: A semi-automatic learning system for structural pairing. *Computers in Human Behavior* (2014), http://dx.doi.org/10.1016/j.chb.2014.11.011

interoperate with other systems, organizations and people; the names of the roles that we assign to the creators of the analyst programs, architects, and developers, and, last but not least, software is becoming a key element of our lives, as well as a way to improve and transform business processes (García-Peñalvo, Forment, & Lytras, 2005). We have to highlight that the application of the technological model to the construction of an architecture for interoperability (Zelkowitz & Cuthill, 1997) is very important.

Globalization and internationalization have led the world to an era of constant changes, which translates into societies led by enterprises that focus their attention not only on technological aspects but also in trying to find a new way to manage the tools available for enterprises (González González, Serradell-López, & Castillo Merino, 2012).

After having implemented systems and offered them to their users, not without their fair share of internal transformation and sacrifices, the institutions are now realizing that these systems are limited, closed and far behind the cutting-edge in terms of innovation. This is a problem whose solution seems to reside in the domain of the Service Sciences, from the point of view of both SOA standards and interoperability, and the business model, ways in which the software development teams collaborate and the strategies to provide the corresponding licenses (García-Peñalvo et al., 2005).

We hear each day more and more people talking about software as a service (SaaS). There have been eight options identified as the most important features of the design of SaaS business models. These include: (1) the features of the SaaS service; (2) the source of the SaaS value; (3) destination group for the SaaS user; (4) architectural model for the configuration and provision of data; (5) the basic competences for managing government SaaS and demand/supply; (6) model of deployment on the cloud; (7) the strategy of integration and provision of SaaS, and, finally, (8) the SaaS price structure (Joha & Janssen, 2012).

The proposal offered by this article seeks to provide a solution to the interoperability between platforms (Ruggaber, 2006) on the level of Web Service (Youyuan & Tianyang, 2010). This article aims at solving the issue in the field of the University, but it may be exported to other cases, for example, banking, and even to health services to exchange patient histories between hospitals. More precisely, this article will deal with the exchange of academic records; academic statuses; grades; scholarships; mobilities, soften the pressure made on procedures that require documentation, etc. When we first started working on the field of universities with interoperability, we saw that it was a matter that interested the public administrations, in particular the Node of interoperability for SUE: Roadmap CRUE-TIC, 2013,[17] which lays out the work to be done by the public administration in the following years. We must not forget that in the Public Administration any exchange of information or interoperability between systems is even more complex than normal, since it requires facing factors in matters of legal, political and socio-cultural aspects (Cestari, Rocha Loures, Portela Santos, & Lezoche, 2014). More and more students demand from their universities electronic services and social channels (Access eGovernment, 2014).

First, we thought that it would be wise to employ more "futuristic" technologies and that they may be used to solve the problem that is presented in this article, as, for example, the Semantic Web (Berners-Lee & Hendler, 2001) and even Semantic Web Services (García Rodríguez et al., 2012) for the definition of an ontology from this referential model on which to apply reasoning, inference rules and other mechanisms offered by this technology, but we then saw through as series of experiments that its use and behavior were not correct. For example, we could not ask to the ontology questions of the type "field X is really field Y", since the Semantic Web may solve other types of relations (similar to those that may appear in a language such as Prolog), but never generate the pairing of the fields that are sought in this article.

In general terms, a great volumen of information is published by the governments and organizations that use formats of the semantic web and formats such as RDF, implicit structures using standard languages of the W3C: RDF Schema or OWL. But the new models of flexible coding to process and exploit this data is needed, which requires the creation of algorithms to process them semi-automatically, with the aim of finding relevant and related information in this new realm of data. Nevertheless, the efficient exploration of the great knowledge bases has not been resolved yet and this is why new paradigms are now emerging to promote the definitive deployment of the Data Web (González Lorenzo, Labra Gayo, & Álvarez Rodríguez, 2012), despite it being a new development system and requiring a great economic effort by the enterprises and organisms.

The proposal offered here is not more than an integration of university platforms that would, in turn, interoperate with a higher platform that will be the Revenue and Public Administrations Service, though it may be an even higher-rank organization, on a European level, mainly concerned with substituting manual procedures with electronic ones (SCP, Substitution of Certificates on Paper), promoted by Article 6 of Law 11/2007[18] "To not provide data and documents that belong to the Public Administrations...". One of the greatest current drawbacks when dealing with this problem is the lack of a referential model on which we may work to share this information in a standardized way. It is true that the Administration, in order to carry out the aforementioned Node of Interoperability for the SUE, has been promoting the e-EDS (Electronic European Diploma Supplement)[19] as the work base and the first building block to maintain said project, and which may constitute the foundations of the interoperability between different government platforms, through a model that improves the services to the citizens (Christiansson, 2011). This way, we will take as our reference the e-SDT proposed, since there is no official one. To work with theoretical models, it will always be better to work with a European official model (SDT). What we still do not truly know is how the interoperability between universities that we propose will be achieved, since the most common outcome will be redefining the web services published by the universities—a matter that is highly unlikely to be carried out due the current economic situation and the general lack of disposition of the administration towards modifying their services—or choosing a middleware service that performs a series of translations to convert the information published in each of the universities to adapt it to the e-SDT.

With this article, through the application of a number of techniques that will be developed in the following sections, we aim at achieving that several system share information and become capable of working with it without modifying their original service, while a service catalog is being built[20] (which does not exist currently on a user level) and a semi-automatic learning system is generated, as each of the iterations of the process will be new knowledge rules to be applied to the proposed system for future iterations.

[17] https://www.rediris.es/jt/jt2013/ponencias/jt2013-jt-sesion1b-a10b4c1.pdf retrieved August 15, 2014.

[18] http://noticias.juridicas.com/base_datos/Admin/l11-2007.t1.html retrieved August 15, 2014.

[19] http://administracionelectronica.gob.es/ctt/verPestanaGeneral.htm?idIniciativa=666#.U-yTyWAcTIU retrieved August 15, 2014.

[20] http://www.accenture.com/microsite/it-technology-trends-2014/Documents/TechVision/Downloads/Accenture_Technology_Vision_2014.pdf retrieved August 15, 2014.

Please cite this article in press as: González Pérez, S. J., et al. Interoperability between platforms without a defined referential model: A semi-automatic learning system for structural pairing. *Computers in Human Behavior* (2014), http://dx.doi.org/10.1016/j.chb.2014.11.011

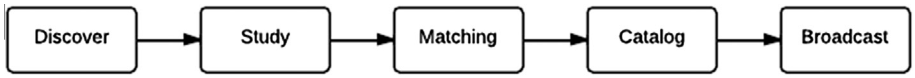

Fig. 1. Lifecycle.

In the end, this is a system that is capable of registering the Web Services of each University to conduct the pairing of their common referential data models, which will be useful for obtaining information from any source registered in a unified way, regardless of their point of origin.

In order to achieve this objective, we use techniques for introspection of languages (Java) to help the system to recognize the features of the Web Services (McIlraith, Son, & Zeng, 2001) to be registered. Moreover, we employ a series of functions called String Metrics (Simmetrics, 2014) that will be used as a tool to manage to "guess" the degree of similarity between two terms and therefore conduct the first phase of pairing of fields between the model to be registered and the referential model. Once the previous steps have been completed, we use a classifying system to solve the cases that could not be paired before, and, finally, as a last resort, we will follow the knowledge rules that exist in the system.

Consequently, with the proposed solution we seek to solve one of the main problems faced by the administration in general terms, which is the exchange of information between administrations: today, in the middle of the Digital and Information Era, the citizens are still being asked much redundant information that is already known by the administration. And not only that, but we also expect that we a very small technological and economic effort the administrations may exchange information. In other words, that information may be shared simply by making each organization offer their Web Services with the information that needs to be shared, and therefore avoiding having to make new developments and deep changes to the administrations and organizations.

3. Approach

This section describes the process that has been developed, which involves the definition of a series of modules that compose the architecture proposed, wherein each of the modules will perform a number of very concrete and specialized tasks and responsibilities. The relation between the modules will be sequential: it starts with an initial one and it goes through each of the following ones until reaching the final one. The modules are seen in Fig. 1 and they are: (1) Discover: Module to obtain the address of the Web Service (WSDL); (2) Study: Module to obtain information on the composition of the Web Service (Introspection); (3) Matching: Module to apply the corresponding techniques for three-step Matching to link the fields obtained in the previous steps with the model of reference; (4) Catalog: In this module should comprise not only the interoperation, but also allow the system to mark its presence in the services that it has registered, formed by a catalog of Web Services, and (5) Broadcast: Since it would not make any sense to design a homogenous system if its goal was not to share its information, this module establishes how this information should be exchanged in a unified way, so that the data hosted by the Web Services can be mined.

3.1. Discover

This module is the most important one in the system. It is the point where the services must be discovered, creating the need to link the Web Services of the Universities that are submitted to be cataloged. This section proposes the way to discover their Web Services to access the following steps of the architecture. As a starting point, in a simple way, there will be a form that will require the URL of the Web Services (WSDL file) that need to be registered, or in other words, the location of the services that offer the different operations related to university scholarships, the academic records of the students and the information that needs to be shared. Another, more-elaborate option would be a Web Service that is available to work as an operation where each of the Universities, or some other entity in charge of compiling this information, was in charge of maintaining these registries. In truth, any option is possible: these two are mentioned because the former would be the one most readily available for the public, while the latter provides an alternative. But what is truly important is that the information of the WSDL is provided to the system so that it can begin processing and analyzing the information.

The way in which the addresses of the WSDL are received is irrelevant for the process proposed, as what we need is to obtain that address. Since this is a system that aims at being automatic, the ideal scenario would involve any online system; like in our case we used for the tests a web form where we identified the University and its web service (see Fig. 2).

It is very important that this step is made in a correct, concise and exhaustive way, since all the information that is obtained will become the basis for the following modules. Moreover, on this will depend having a substantial catalog for the future, and so having available all the services not only in a unified way, but also independently.

3.2. Study

This module deals with the introspection of the information received by the previous module, and it can be seen as the first analysis of the information offered by the WSDL. Therefore, it is necessary that the programming language used is a language that has a large introspective potential; and although we employed Java as the language of our prototype, others, such as Python or Ruby, may be even better in terms of the specialization of this module, due to the high degree of alteration and dynamic typing of these programming languages, which offer a wide support for observation and modification of the structure of an object.

In order to extract that data that we need from the WSDL file, in the case of Java it is recommended that we used an Apache CXF library,[21] which is able to perform dynamic instances to a Web Service without needing to generate the whole stub of methods that would be needed with other technologies, and therefore facilitate the tasks of development and integration to the system.

More precisely, Apache CXF provides the DynamicClientFactory class, which is capable of interoperating directly with the web service by making calls to the methods of the service in an "abstract" way by using java.lang.Object classes and arrays of this type in the parameters and returns, so that it can obtain any result from these calls.

This is truly important, as the ultimate goal is to automatize the process as much as possible, and leave human intervention for extreme cases in which the system is not able to perform the matching (especially in early stages of the analysis), thus removing the complexity and nuisance of having to work with "Wizards" and clients that should be created manually to be able to work with the Web Services to be analyzed.

[21] http://cxf.apache.org/ retrieved August 15, 2014.

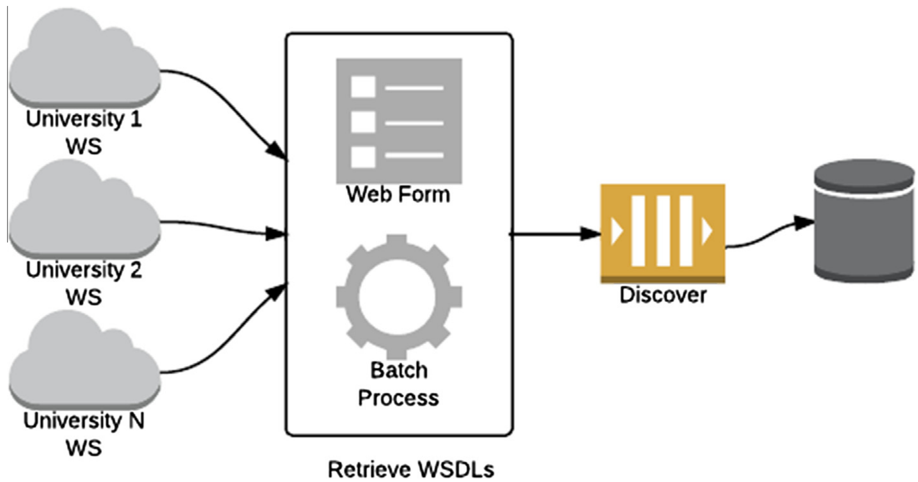

Fig. 2. Discover.

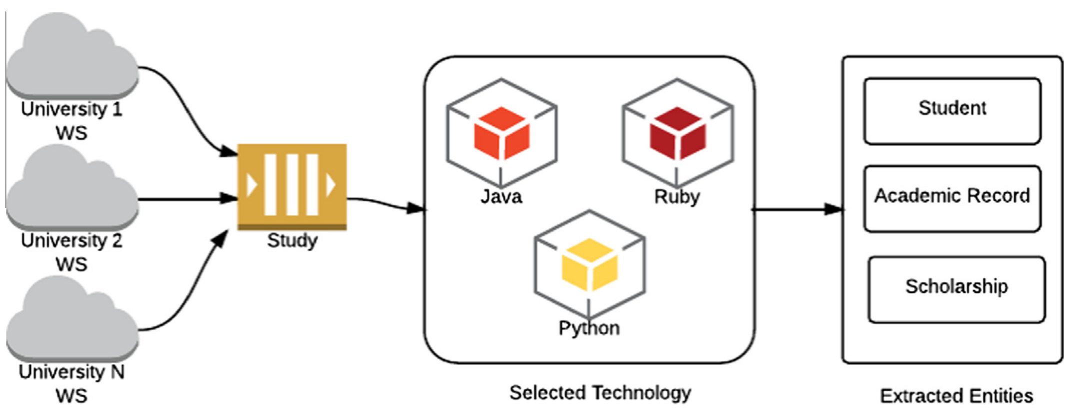

Fig. 3. Study.

In this stage, the Web Service has already been dynamically instanced, and the features of the models presented are analyzed. It is recommended that in the previous module (Discover), when the Web Service is registered, the relevant and important classes that we are working with are indicated (for example, the model Student, Academic Record, Scholarship, etc.), so that in this module it is easier to quickly and easily eliminate false positives with other classes of the system that may be introduced by the language itself (java.lang.String, java.lang.Integer, ...) and that the introspection of the language itself would interpret as classes.

Once the classes relevant to the model have been detected, the fields/attributes of each of them are obtained, so the following step would be to compare these fields with those of the reference model and previous matches, in order to ensure the matching process between models.

In conclusion, the result of this stage will be a list with the classes that compose the web services that are being checked, as well as their attributes and the method available from the web service (see Fig. 3).

3.3. Matching

This module is critical and very complex. Each of the attributes if the models obtained in the previous steps is analyzed and they are tried to be matched with reference model.

In order to achieve this goal, the following process is defined: We start with a No-SQL database of the key-value type, that allows us to perform a quicker search that a relation database (in this concrete case we used Apache Cassandra[22]), where each document is divided by each "class" of the reference model, or in other words, a scheme is created for Student, another one for Scholarship, another one for Academic Record, etc. For each document, the keys are each of the attributes for the classes that have the documents as their name. For example, the document Student may have as keys "name", "surname", "date_of_birth", etc. and the values will be each of the valid matches that the application makes for each attribute. Therefore, it can be understood as an array of values (["name", "student_name",etc.]). The container of information obtained is key for the application. The semi-automatic learning by the system is performed (we will have cases of matches that need to be done manually, either due to lack of information in our models or because it deviates too much from the case we are dealing with), and the knowledge base that is first used. The more registries are obtained or made, the more information will our system have and better will its "predictions" on the matching of the fields be.

With the information available, the system applies the String Metrics functions to each of the attributes of the document in relation to each of the attributes of the classes identified in the Web Service that is being registered. This way, we can obtain a document for the class Student similar to the following (Table 1).

Student will be the class of the reference model and name, surname, date_of_birth, etc. its attributes, and the arrays that

[22] http://cassandra.apache.org/ retrieved August 15, 2014.

Please cite this article in press as: González Pérez, S. J., et al. Interoperability between platforms without a defined referential model: A semi-automatic learning system for structural pairing. *Computers in Human Behavior* (2014), http://dx.doi.org/10.1016/j.chb.2014.11.011

Table 1
State of the NoSQL database in the middle of the execution.

Key	Value
Name	[“NAME”, “STUDENT_NAME”, “GIVEN_NAME”]
Surname	[“SURNAME”, “STUDENT_SURNAME”, “PATERNAL_SURNAME”,SURNAME”, “MATERNAL_SURNAME”]
date_of_birth	[“BIRTH”, “DATE_OF_BIRTH”, “BIRTHDATE”, “DATEOFBIRTH”]
Gender	[“GENDER”, “GENRE”, “SEX”]

accompany them will be registries that had been previously made (the first item by default is the attribute itself).

If, for example, the model being registered is similar to that of (Fig. 4).

What we will do is to pick each of the attributes of the Student class and register and apply String Metrics on each of the attributes of the Student class of the reference model. Thus, ‘‘givenname” will be compared with String Metrics to ‘‘NAME”, ‘‘STUDENT_ NAME”, ”GIVEN_NAME” and an average will be calculated. Then, String Metrics will be applied to ‘‘SURNAME”,”STUDENT_SURNAME”, ‘‘PATERNAL_SURNAME”,”MATERNAL_SURNAME” and the average will be calculated, and so on and so forth, until all the averages have been calculated for all the previous matching arrays of the reference model of this concrete case.

To calculate the averages, we performed several tests and, in our case, for the information on classes and attributes in Spanish, we made sure that all the averages were over 0.7, which is a good limit to make the selection of valid results.

The matching process is composed by three stages. The first one has been describe above, and some attributes of the registry model may still be left unmatched. In this case, we apply the second step, which involves a classification algorithm that obtains a ratio, which, depending on whether it fits the parameter configured, will be taken as valid or invalid for the matching. In this second stage, we can choose the suitable classification algorithm for the concrete situation we are dealing with: K-neighbors, neural networks, information fuzzy networks, etc. In order to implement this step, the ideal scenario would require an instantiation of the Web Service that is being used, in order to classify it in a data level, which is more representative. A possible instantiation may be performed during the first stage and would make the task much easier.

If, after performing the two aforementioned steps, there are still unmatched attributes, we perform the third step and last resource we have to conclude the automatic matching process. We apply a series of inference or knowledge rules that would be specific to each domain. One of the rules that we have implemented in an obvious way is the identification of patterns in regular expressions to identify the contents of the attributes (as in the previous stage, it is necessary to instantiate the Web Service). Attributes such as the National ID number; bank account numbers; academic record numbers; postcodes; and e-mail addresses, are very easily recognizable by regular expression patterns.

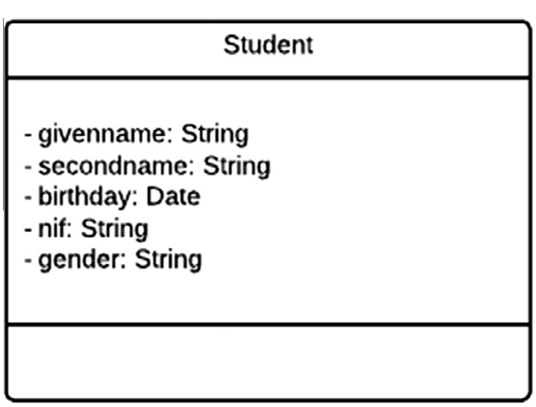

Fig. 4. Student class.

Once this three-step automatic process has been finished, the results of the matching are presented to the administrator of the system, who has the user rights to register the services, so they can be check them and correct them if they are not correct, as there may be complex fields or fields that did not have enough information on the system to be filled in during the matching process.

Once the results of the matching process have been confirmed by the aforementioned administrator (which justifies the semi-automatic nature of the process), these matching results will be checked against the No-SQL database mentioned before, which gives us an extra layer of knowledge for the next registration. The concrete matches will be backed up to another database in order to be able to perform the translations between the concrete service model that has been registered and the reference model (see Fig. 5).

3.4. Catalog

This module generates a catalog to be able to maintain information of each of the registered universities. This information may include the address of the Web Service; information about how many operations it offers; classification of those operations (for example, for which model each of them is useful), and in this way, we can know which are the best options offered by each University, and therefore the user may access a website in the form of a search engine where they may filter by categories and see which universities offer the services that interest them, or see concrete universities, how the matching process was performed, etc. We suggest that this is implemented with a search of the Apache Lucene[23] type, though other searches may also be used (see Fig. 6).

3.5. Broadcast

In order to perform the broadcasting of the Web Services that have been registered with other interoperability (Node of interoperability for SUE: Roadmap CRUE-TIC, 2013, for example), through a Web Service an endpoint is provided, offering a configurable operation (or several ones) so that the calls may be grouped according to whether certain universities support what is expected from them and which do not. For example, retrieving a student’s scholarships with their National ID number. In short, we built an interface that would be invoked to internally perform the study of the call, then decide what Web Services of the universities must be called, and, finally, to indicate that the results of the calls made to each of the different Web Services are used with the translations that had been previously stored in the Matching to present the information in a unified way to the entity that performs the service call (see Fig. 7).

An interesting option that can be made during this stage is the use of an Enterprise Service Bus. This way, we could easily secure the service and also control everything that happens around the Web Service offered, such as the inclusion of statistics, and performance.

4. Conclusion and future work

The solution proposed in this article aims at being as economical as possible for the organizations or administrations, and to not demand a great technological or economic effort from them, as they

[23] http://lucene.apache.org/ retrieved August 15, 2014.

Please cite this article in press as: González Pérez, S. J., et al. Interoperability between platforms without a defined referential model: A semi-automatic learning system for structural pairing. *Computers in Human Behavior* (2014), http://dx.doi.org/10.1016/j.chb.2014.11.011

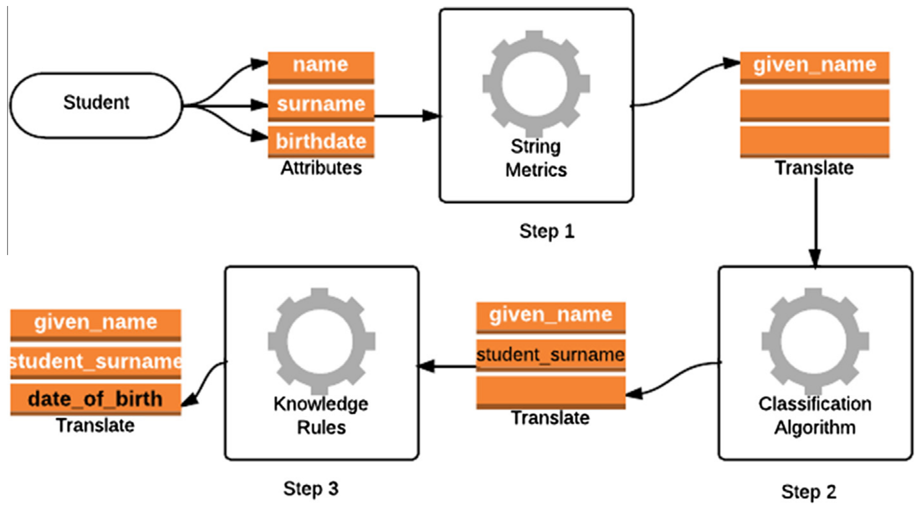

Fig. 5. Matching.

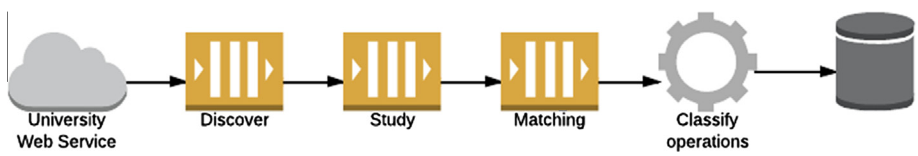

Fig. 6. Catalog.

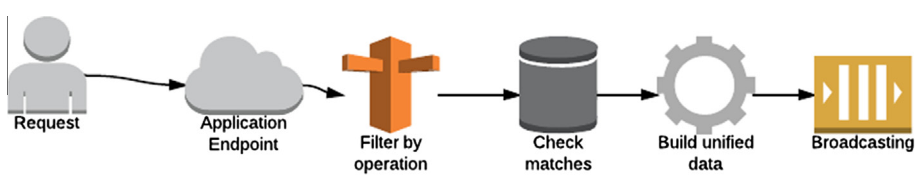

Fig. 7. Broadcasting.

will not require them to alter the current information systems, since this solution would be on a higher layer, leaving them unaltered. The system proposed in this article provides a solution for many of the problems currently faced by the Public Administration related to the exchange of information between their systems. The modules described in the article—according to the life cycle described in Fig. 1—goes from the initial task of how to gather the addresses of the different Web Services to how to disseminate the information of these Web Services, including how to study these Web Services and achieve a common model for these highly-different sources to be used in a unified and transparent way for the client system that needs them. This transparency and uniformity is achieved through the execution of a three-step system that will deal with the attributes of each of the models gathered from the different Web Services in relation to the previous registries, by applying functions of the String Metrics type, classification algorithms an and a predefined system of rules.

In this article, we reflected on how to interact or interoperate between two or more organizations of the same field—in this case, the university system, although they could belong to other sectors, such as banking, and the health system. Therefore, this solution may be applied to any situation that demands the need to interoperate between different information systems that already offer Web Services and a common thematic. Thus, it could be implemented in hospitals where a clinical record must be shared on a national level (a patient in Madrid may visit a hospital in Barcelona during their holidays there, for example), or in another environment such as in banking entities to share information about clients when transferring them from one bank to another, or in bank transferences. For example, in the recent case of the SEPA[24] (which involves a unique way to pay in Euros in a simple, secure and efficient manner for consumers and enterprises in Europe) it has taken several years to define a unique format, the European and national regulations, the migration deadlines, schedules, extensions, etc. With an interoperable system like the one we propose, this process could be made at different paces and different transaction systems could remain compatible, but above all, it would offer the conditions for the European payment system to continue evolving in the future, with the constant changes that will surely be demanded by the new challenges that the electronic payment system will have to face.

As a future project, we will proceed to choose a classification algorithm that will be used to perform the second stage in the three-step system presented in this article, as the alternatives that we have at the moment are very varied. It is possible that the algorithms that work for an environment such as the academic world may not be truly useful for another one, such as the healthcare system. Moreover, another of the tasks in which we will

[24] http://www.paymentscouncil.org.uk/what_do_we_do/european_payments/-/page/1141/ retrieved August 15, 2014.

continue to work is the description of the rules that should be defined for the third step of the system proposed, since they also depend on the context that is being studied.

Another topic that worries us in an international level is the interoperability of information systems in terms of the legal operative capacity of information exchange, and the establishment of the obligations and responsibilities of each of the parties involved in the interoperated system.

Another future line of work would be to establish a federated system for the proposed solution, either through a centralized system audited by an official body, or through a distributed solution that would need to reflect the obligations of this solution. Therefore, it would be advisable to conduct a prior research on the advantages and disadvantages of the solution chosen.

References

Access eGovernment (2014). *An educational program*. <http://access-egov.info/> Retrieved 15.08.14.

Berners-Lee, T., & Hendler, J. (2001). Publishing on the semantic web. *Nature Publishing Group*, 1023–1024. http://dx.doi.org/10.1038/35074206.

CEC, C. (2006b). *Interoperability for Pan-European eGovernment Services, COM (2006) 45 final*. <http://www.epsos.eu/uploads/tx_epsosfileshare/Communication-on-Interoperability_01.pdf> Retrieved 15.08.14.

Cestari, J. M., Rocha Loures, E., Portela Santos, E., & Lezoche, M. (2014). A research strategy for public administration interoperability assessment. In *Proceedings of the 2014 industrial and systems engineering research conference* (Vol. 10).

Christiansson, M. -T. (2011). A common process model to improve eService solutions – The municipality case. In *The proceedings of the 11th European conference on eGoverment 2011*. ISBN: 978-1-908272-00-3.

Cohen, W., Ravikumar, P., & Fienberg, S. (2003). *A comparison of string distance metrics for name-matching tasks*. <https://www.cs.cmu.edu/~pradeepr/papers/ijcai03.pdf> Retrieved 15.08.04.

Commission of the European Communities (CEC) (2006a). *i2010 e-Government Action Plan: Accelerating e-Government in Europe for the benefit of all, COM (2006) 173 final*. <http://eurlex.europa.eu/LexUriServ/LexUriServ.do?uri=C> Retrieved 15.08.14.

EIF (2010). *European interoperability framework for European Public Services, COM (2010) 744 final*. <http://ec.europa.eu/isa/documents/isa_annex_ii_eif_en.pdf> Retrieved 15.08.14.

European Commission (2007). *Handbook on implementation of the services directive*. <http://ec.europa.eu/internal_market/services/services-dir/index_en.htm> Retrieved 15.08.14.

García Rodríguez, M., Álvarez Rodríguez, J. M., Berrueta Muñoz, D., Polo Paredes, L., Labra Gayo, J. E., & Ordóñez de Pablos, P. (2012). Towards a practical solution for data grounding in a semantic web services environment. *J. UCS, 18*(11), 1576–1597.

García-Peñalvo, F. J., Forment, M. A., & Lytras, M. D. (2005). Some reflections about service oriented architectures. *Journal of Universal Computer Science*.

Garson, G. D. (2004). The promise of digital government. In A. Pavlichev & G. D. Garson (Eds.), *Digital government principles and best practices*. Hershey, PA: Idea Group Publishing.

González González, I., Serradell-López, E., & Castillo Merino, D. (2012). The implementation of process management: A system to increase business efficiency—Empirical study of Spanish companies. *International Journal of Knowledge Society Research*.

González Lorenzo, J., Labra Gayo, J., & Álvarez Rodríguez, J. (2012). A MapReduce implementation of the spreading activation algorithm for processing large knowledge bases based on semantic networks. *International Journal of Knowledge Society Research*.

Heeks, R., & Bailura, S. (2007). Analyzing egovernment research: Perspectives, philosophies, theories, methods, and practice. *Government Information Quarterly, 24*(2), 243–265. http://dx.doi.org/10.1016/j.giq.2006.06.005.

Joha, A., & Janssen, M. (2012). Design choices underlying the software as a service. *Journal of Universal Computer Science, 22*.

McIlraith, S., Son, T., & Zeng, H. (2001). Semantic web services. *IEEE Intelligent Systems, 16*(2). http://dx.doi.org/10.1109/5254.920599.

Ruggaber, R. (2006). *ATHENA – Advanced technologies for interoperability of heterogeneous enterprise networks and their applications*. London: Springer. http://dx.doi.org/10.1007/1-84628-152-0_45.

Simmetrics (2014). *A String Metrics framework for Java*. <http://sourceforge.net/projects/simmetrics/>.

Steward, J., & Walsh, K. (2007). Change in the management of public services. *Public Administration, 70*(4), 1467–9299. http://dx.doi.org/10.1111/j.1992.tb00952.x.

The Yankee Group Report (2003). *Interoperability emerges as new core competency for enterprise architects*. <http://www.intersystems.com/ensemble/analysts/yankee.pdf>.

Weerakkody, V., Choudrie, J., & Currie, W. (2004). Reasiling e-government in the UK: Local and national challenges. In *Proceedings of the 2004 American conference on information system*, New York, NK.

Youyuan, F., & Tianyang, G. (2010). Research on web service interoperability. *World Academy of Science, Engineering and Technology, 3*.

Zelkowitz, M., & Cuthill, B. (1997). *Application of an information technology model to software engineering environments*.

Please cite this article in press as: González Pérez, S. J., et al. Interoperability between platforms without a defined referential model: A semi-automatic learning system for structural pairing. *Computers in Human Behavior* (2014), http://dx.doi.org/10.1016/j.chb.2014.11.011