Computers in Human Behavior xxx (2014) xxx–xxx

Contents lists available at ScienceDirect

Computers in Human Behavior

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

Knowledge-based public service transactions: An intelligent model-driven approach in co-learning contexts

Guillermo Infante Hernández *, Aquilino A. Juan Fuente, Jose Emilio Labra-Gayo, Benjamín López Pérez, Edward Rolando Núñez-Valdez

Computer Science Department, University of Oviedo, Spain

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
Model driven engineering
e-government platforms
Collaborative learning
Layered software architecture
Domain experts
Knowledge base

ABSTRACT

Public administrations pursue the efficiency and quality of administrative services they offer as well as the reduction of time and operational costs in executing service transactions. However, some issues arise when trying to achieve these goals: (a) the lack of procedure formalization to describe public services, (b) a mechanism to guarantee services' obligatory compliance with frequently changing legal regulations, and (c) the efficient deployment of service transactions in e-government platforms. The aim of this work is to identify the phases of the development cycle of eGovernment transactions and provide the support to automatize them efficiently by using a model driven engineering (MDE) and knowledge-based approach. The main source of the knowledge extraction process comes from a collaborative learning environment where public servants share acquired domain knowledge. A web survey has been conducted to evaluate the approach acceptance degree by software developers and domain experts. The main conclusion is that 61% of the surveyed experts strongly agree that our approach improves actual eGovernment transactions practices and the phases needed to develop them.

1. Introduction

The objective behind Information Technology (IT) projects in e-government contexts as argued in (Buhl & Löffler, 2011) are: simplification and realization of information, communication, and transaction processes within and between public administrations and citizens by using digital information and communication technology. Technologies that use workflow management have shown its effectiveness for the integration or automation of business and industrial processes using Enterprise Resource Planning (ERP) systems (Aalst & Hee, 2002). However, the procedure transactions automation in the context of public sector with the single use of workflow management may not be a suitable approach as discussed in (Yang, Tong, Ye, & Wu, 2006). The main reason to support this is that the administrative procedures essentially differs from business process in that the former requires intensive domain knowledge (Lytras & Pouloudi, 2006; Papavassiliou, Ntioudis, Abecker, & Mentzas, 2003). Moreover, these procedures have high probability to change, disappear or evolve to new states over time since they are under the influence of political decisions that suffers frequently variations. These kind of procedures are characterized by frequent dependencies on domain knowledge like regulations that enclose certain rules to fulfill its tasks or actions; they can be classified as a set of high complexity procedures with a strong dependency on domain knowledge (Papavassiliou et al., 2003). At this point the knowledge extraction in e-government domain presents itself as a complex task. This complexity gets bigger when the rules that affect this domain are extracted from public regulations that control the execution of procedures. From the legal point of view it is difficult to comprehend and interpret the regulations by the public employees, even knowing the domain. Different interpretations are made coming from a single regulation that affects the same procedure, which is therefore executed in different ways at various public organizations. Taking this into account, there is a need to have a mechanism that formalizes the implicit knowledge involved in procedures transactions. Domain experts can be a very valuable knowledge source besides legal regulations. The domain expert's knowledge must be consistent with the tools it uses to do its work in procedure transactions. In conclusion, it can be said that it is necessary to formalize this knowledge in a comprehensible way for the expert that enable the extraction of the rules needed to automate procedure transactions.

* Corresponding author at: Calvo Sotelo s/n, 33007 Oviedo, Asturias, Spain. Tel.: +34 985103397.

E-mail addresses: infanteguillermo@uniovi.es (G.I. Hernández), aajuan@uniovi.es (A.A. Juan Fuente), labra@uniovi.es (J.E. Labra-Gayo), benja@uniovi.es (B.L. Pérez), nunezedward@uniovi.es (E.R. Núñez-Valdez).

http://dx.doi.org/10.1016/j.chb.2014.09.044
0747-5632/

The use of Model Driven Engineering (MDE) technologies in the definition of procedure transactions models in e-government domain requires a close coordination between domain experts, developers and resources in order to create representative and accurate transaction models. Making the public employee the eGovernment domain expert and involving it in the process of the proper formalization of the procedures can be a suitable approach to accomplish the desired efficiency in administrative transactions. Procedure transaction modeling in e-government domain it is defined in this work as the task to formalize an e-government procedure from its late execution point of view going through the supervision of a domain expert. The transaction term in this context gets the meaning from the execution of certain automated administrative actions to achieve a procedure completion. Defined models' validation is also a critical activity as stated by (Bertolino, De Angelis, Di Sandro, & Sabetta, 2011) since models are the start point of many later transformations where deficiencies in the source models can have detrimental implications. In the case of eGovernment domain models (e.g. procedure transaction models), the accuracy which they have been defined determines the compliance with specific regulations or mandates as the context of their usage is usually the public sector. Therefore, models' validation becomes of crucial importance in this domain since it can prevent from failing in unlawful procedures. The goal of this paper is to provide an approach which leverages domain experts' knowledge as well as legal regulations in order to automatize the modeling, validation and deployment cycle of public services in e-government platforms. The proposal assumes that the automation of this cycle providing domain experts with an intelligent model based toolset can improve the efficiency and quality of public services and reduce operational costs since the time and effort from designing to deployment of services it's greatly reduced. In order to model e-government procedure transactions in a balanced way between the supervision of human actors and automated validation, we provide a knowledge base composed by a set of supervision rules integrated in a modeling environment and executed by a rule engine. Therefore, the approach enables public sector employees or procedure modelers to transform, validate and deploy knowledge based requirements models in platform specific-applications. Furthermore, software developers can benefit from this approach since its model driven nature fosters the code reuse and reduce the development effort.

2. Background and related work

Related work concerns various aspects of the problem we address, namely use of process modeling, knowledge-based systems, code generation, etc., for e-government transactions automation. Recent research in the field of process modeling and workflow management in e-government domain has raised a significant interest in scientific research community (Osterweil, Schweik, Sondheimer, & Thomas, 2005). In (Klischewski & Wetzel, 2002), the authors have modeled the services flow using Extensible Mark-up Language (XML) by identifying the points where the specific tasks supplied by the public administration are captured. As shown in (Osterweil et al., 2005), the use of Process Definition Language (PDL) can be useful to analyze the inconsistencies and errors in administrative processes such as a license renovation. Another proposal have been developed in eGov project, which is built as an e-government integrated service platform where services are specified, registered and deployed using Governmental Markup Language (GovML) (Tambouris, 2001). As shown in (Ciaghi, Weldemariam, Villafiorita, & Kessler, 2011), process modeling in public administrations has mainly been performed with general-purpose languages. These modeling languages are flexible mechanisms to describe miscellaneous processes in many different domains. However, they do not consider in particular public sector specific issues like: (1) services integration in transactions performed (e.g. interoperation with external services) to retrieve or send information required, (2) real modeling time validations and completion of modeled transactions, and (3) conformance to legal regulations (Wimmer, 2003). This results in the conclusion that these generic approaches are not suitable to represent all relevant aspects of this domain. Therefore, there is a need for a new, e-government transactions modeling language. Regarding to knowledge management in e-government domain there are various approaches that addresses this topic. Among them can be found SmartGov (Tambouris, Boukis, Vassilakis, & Lepouras, 2002), which have been developed as a knowledge based platform to aid public employees in online generation of transactions for electronic forms. It uses ontology to build the domain map associated with knowledge units that represents the regulations. The approach depicted by (Papavassiliou et al., 2003) leverages the potentialities of workflow and knowledge management to automate or semi-automate administrative procedures that requires domain intensive knowledge. In (Savvas & Bassiliades, 2009) the authors adopt a process oriented approach, through a web-based knowledge management system that provides an interpretation of legal framework. Other approaches that were taken into account in this work from the knowledge management perspective in e-government transactions were the knowledge sharing (Zhang, Vogel, & Zhou, 2012) between domain experts in public administrations and coordination among the former and software developers (Yuan, Zhang, Chen, Vogel, & Chu, 2009). These approaches emphasize in the importance of sharing the domain knowledge effectively among the experts of the domain and the tasks coordination with the IT experts of the organizations. This subject is also addressed in (Zhang, De Pablos, & Zhou, 2013) where the authors examines how the visibility of knowledge sharing impacts on incentive-based relationship in IT-based knowledge management systems. Although we consider the domain knowledge extraction from domain experts, our work essentially differ from the above in that we propose a formal representation and generic mechanism that can be integrated in actual and future e-government platforms. We keep this formal representation close to the domain as possible providing a modeling environment to express the information structures and rules that affect the execution of the procedures. Furthermore, we implement such rules using an intelligent mechanism that aids the adaptation of procedure transactions to new changes in legal regulations, this way administrations guarantee the constant updating of their already defined procedures.

3. Approach

This approach starts from the fact that every transaction in eGovernment domain is described by repetitive phases that may

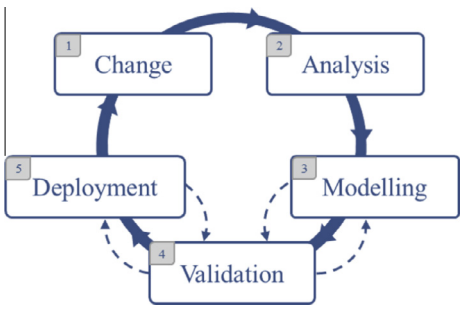

Fig. 1. Development cycle phases of eGovernment transactions.

Please cite this article in press as: Hernández, G. I., et al. Knowledge-based public service transactions: An intelligent model-driven approach in co-learning contexts. *Computers in Human Behavior* (2014), http://dx.doi.org/10.1016/j.chb.2014.09.044

vary from one administration to another depending on each current regulations. The common identified phases as shown in Fig. 1 are the following: (1) Change phase where emerges a need for a change in the current transactions performance derived from political decisions, normative updates or organizational issues; (2) Analysis phase where domain experts from public administrations establish the main requirements to meet within the new transaction proposed; (3) Modeling phase intended to represent in a semi-formal way the previously identified requirements in order to better understand and communicate the proposed solution; (4) Validation phase responsible of verifying the actual transaction compliance with legal regulations, and finally a (5) phase of Deployment to execute newly created or changed transactions as public services that can be accessed by citizens or other administrations through eGovernment platforms. This development cycle repeats each time a new change is needed and thus it is susceptible to be automatized. Although these changes takes place in a very complex domain where interdisciplinary actor intervenes, transaction systems in this domain share a common abstract information structure that can be formalized in order to be accessible to other systems that can use it.

In parallel to the identification of the development cycle, this approach proposes five main steps to undertake the tasks derived from it. The starting point is the definition of a metamodel to formalize the real-world entities involved in eGovernment transactions domain. The resulting metamodel is used for the construction of a graphical modeling language designed to visually represent domain experts' vocabulary regarding to administrative transactions (Section 3.1.1). A visual editor is developed in order to enable domain experts to use this language (Section 3.1.2). Section 3.2 describes the construction of a knowledge base formed by a set of validation rules shared between modeling process and deployment of procedure transactions. This work takes the Spanish regulation context as the source of the rules definition in order to validate the modeled procedures. Despite this specific regulation context, the defined metamodel has an open perspective to absorb other forms of regulations. This rule set represents the central business logic source used to monitor and validate procedure transactions' models as well as to control its execution in deployed platforms. Finally Section 3.3 depicts the construction of a code generator in order to transform the validated procedure transactions and inject them in a specific eGovernment platform prototype. Last step closes the entire development cycle of an eGovernment transaction system, from its modeling to deployment passing by the validation process. The final result is represented as a layered architecture that encloses three main modules as depicted in Fig. 2.

The proposed architecture follows the layered style described in (Clements, Garlan, Little, Nord, & Stafford, 2003), where each layer represents a grouping of modules that offers a cohesive set of services. In this case there is only one module per each layer to keep the design simple, even though, there are constraints on the allowed-to-use relationship among the layers: the relations must be unidirectional. The architecture informal notation shown in Fig. 2 is represented as a stack of boxes. The allowed-to-use relation is denoted by geometric adjacency and is read from top to down. The correlation between architecture modules and development cycle phases is established in Fig. 3.

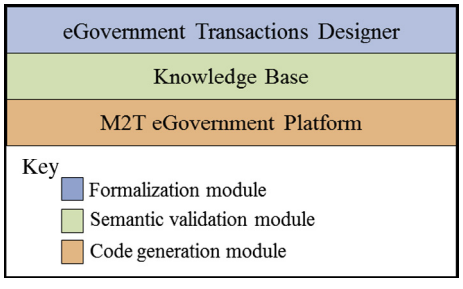

Fig. 2. eGovernment transactions layered architecture.

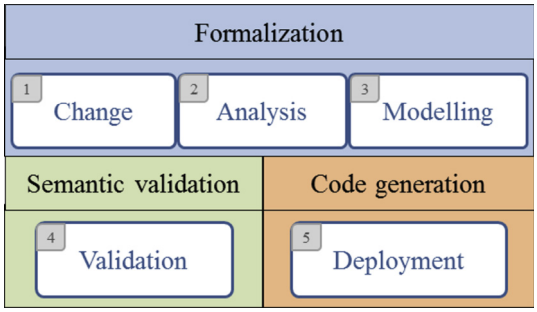

Fig. 3. Architecture modules and development phase correlation.

3.1. Formalization module

This section describes the top architecture module responsible to formalize procedure transactions. A modeling language called eGotML is presented along with a visual editor (eGotDesigner) to perform procedure transactions modeling tasks. When a system is effectively modeled against a well-defined modeling language, the model may also be used to predict the behavior of the system, and at a significantly lower cost than the implementation of the modeled system (Selic, 2003). These modeling benefits can be taken into eGovernment domain to formalize this very changing context and to gain an abstract representation of its repetitive tasks in order to automatize them. From an analysis performed to the literature in this field and previous studies (Muñoz-Cañavate & Hípola, 2011) of legal regulations as well as interviews with public administration experts, some common concepts were identified. In order to gain a meaningful representation of every concept as well as the relations between them, a metamodel (Fig. 4) is constructed using Ecore notation.

3.1.1. eGovernment transactions modeling language (eGotML)

As discussed by Moody (2009), visual representations of models can be more effective than textual representations. Visual notations are also particularly important when communicating with end-users and customers. Visual representations can be particularly powerful when applied to the domain of model-driven development, as visual representations are themselves models. Modeling languages and domain-specific languages are two similar concepts; adapting the definition of a DSL provided by Fowler Fowler (2010), a domain specific modeling language (DSML) therefore represents a modeling language of limited expressiveness focused on a particular domain. The DSML proposed in this approach, the eGovernment transactions modeling language (eGotML), focuses on the eGovernment transactions domain concepts represented in Fig. 4 as a metamodel. This language presents a common visual notation that can be easily understood by domain experts. Table 1 shows the notation of eGotML elements along with a domain description.

3.1.2. eGovernment Transactions Designer editor (eGotDesigner)

In eGovernment domain as well as in any other domain, a modeling environment supported with software tools can maximize the benefits of using models to manage transactions. In this section it is proposed the eGovernment Transactions Designer (eGotDesigner)

Please cite this article in press as: Hernández, G. I., et al. Knowledge-based public service transactions: An intelligent model-driven approach in co-learning contexts. *Computers in Human Behavior* (2014), http://dx.doi.org/10.1016/j.chb.2014.09.044

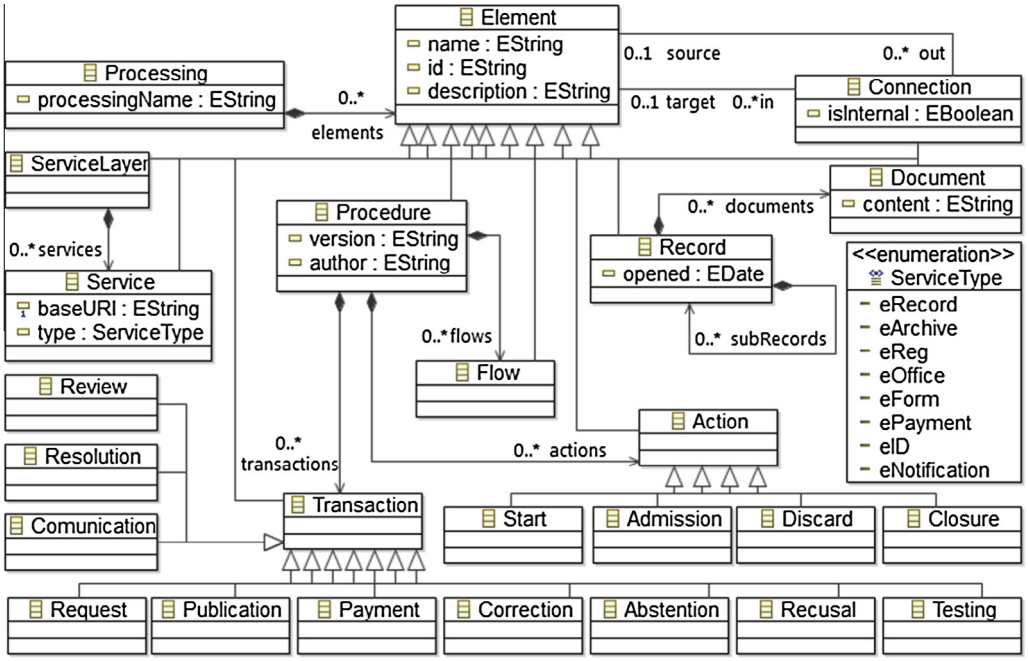

Fig. 4. eGovernment transactions metamodel.

Table 1
eGotML domain elements' description and notation.

Domain element	Description	Notation
Procedure	Formal channel to perform administrative actuations with the purpose to fulfil citizens or organizations requests	
Service layer	External services container	
Service	Interoperation mechanism that defines the protocols to share and retrieve information within procedure transactions	
Flow	Control structure that determines the transaction paths depending on the conditions expressed	
Actions	**Specific actuations within a procedure generally related to decision making**	
Start	Start action in a procedure	
Admission	Any sort of information input submitted to the administration that meet the requirements	
Discard	Rejection action that discards specific actuations or submitted information	
Closure	End action that terminates the procedure execution	
Transactions	**A processing unit composed by a subset of administrative actuations to perform some stage execution of the entire procedure completion**	
Abstention	Transaction where the procedure execution stops by the user decision	
Communication	Transaction used to trigger a notification procedure in order to communicate citizen records' changes	
Payment	Transaction usually performed by a payment gateway	
Correction	Used to ask citizens or organizations to fix or complete submitted information	
Publication	Transaction used to display general interest information in public official portals	
Recusal	Transaction brought by citizens if they are not agree with some administration decision	
Request	Performed by citizens or administrations to ask for some actuation of their interest	
Resolution	Procedure execution final decision	
Review	Administrative or technical review of procedure state or submitted information	
Data		
Document	Any sort of documentation	
Record	Persisted information derived from procedure transactions and linked to an specific citizen or organization	

which is intended to create model instances using the proposed eGotML language. Model-driven technologies such as the Eclipse Modeling Framework (Steinberg, Budinsky, Paternostro, & Merks, 2008) and the Graphical Modeling Framework (Gronback, 2009) are used for the implementation of the visual editor. Since these visual editors provide interactive representations of an underlying model instance, they map naturally to the model-driven development approach advocated in this paper. The visual editor was developed as rich client platform (RCP) that enables domain experts to model procedure transactions in a comprehensive way. A public procurement procedure (CENATIC, 2011) was modeled with eGotDesigner as Fig. 5 shows. This procedure is forward used in this paper as the source of information for the subsequent stages of the approach.

Please cite this article in press as: Hernández, G. I., et al. Knowledge-based public service transactions: An intelligent model-driven approach in co-learning contexts. *Computers in Human Behavior* (2014), http://dx.doi.org/10.1016/j.chb.2014.09.044

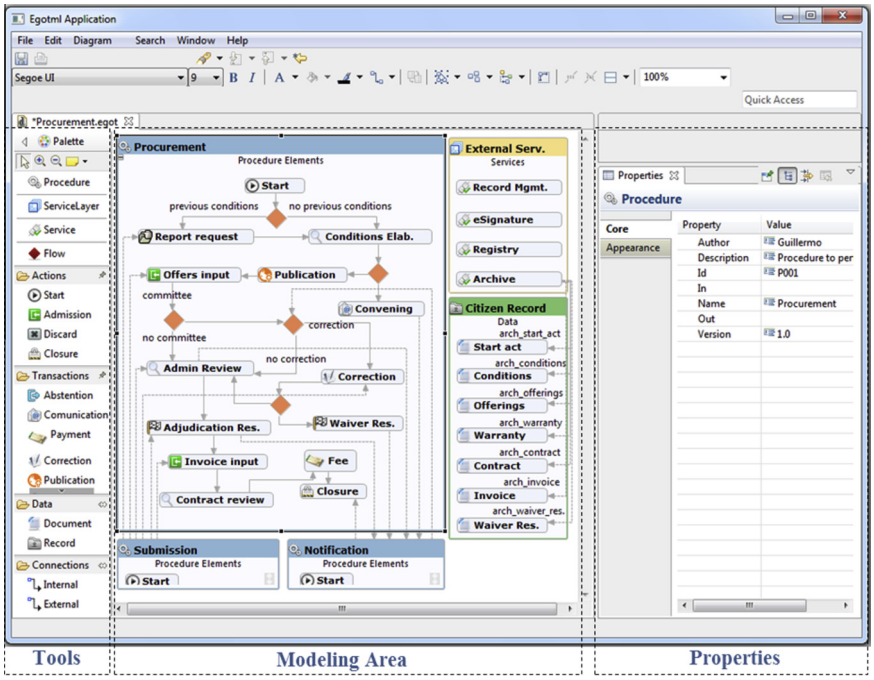

Fig. 5. Public procurement procedure fragment modeled with eGotDesigner tool.

3.2. Semantic validation module

This section describes the middle architecture module responsible to validate and complete eGovernment transactions models in real modeling time. The mechanism employed to achieve the semantic validation is the integration of a knowledge base composed by a set of automated rules which monitors the modeling process and aids modelers to create semantically correct models based on administrative and legal restrictions. The rule set is interpreted by a rule engine which reacts to the incoming information from the modeling process and fires the corresponding rules. In the context of this work, "knowledge base" (KB) means a collection of formal statement relating the inner content (the 'meaning') of administrative and legal regulations from Spanish normative (BOE, 1992, 2007). These regulations are originally presented under the form of unstructured natural language information and concern some general guidelines proper to the management of procedure transactions in the context of public administrations. There are a number of different technologies that may be used to express this sort of information and to set up the knowledge base. Almost any language that supports some form of rule-based inference can be used; this includes rule engines such as Drools ("Drools Expert," 2013), Jena (McBride, 2002) and Jess (Hill, 2003). Inference rules are evaluated using an enhanced implementation of the Rete algorithm (Sottara, Mello, & Proctor, 2010). The underlying model that Drools operates within is simple plain old java objects (POJOs), making it easy to integrate into an existing Java-based software system. The structure of inserted POJOs does not need to be defined as part of the rule base; this means that all metamodel properties and operations are always accessible to a Drools rule. These are the main reasons why to choose Drools in this approach in order to build and execute the eGovernment transactions knowledge base. A sample rule of the processing knowledge base is presented in Fig. 6 along with the previous sample of a public procurement procedure modeled with eGotDesigner. In this case the knowledge base validates and completes the model if the conditions are met. The conditions here states that: *if a process contains a procedure with a request and communication elements, two additional procedures must be added to handle such elements; the communication element needs to be handled by a notification procedure that manages the citizen notifications and the request element must receive a submission from an external procedure that handles the submitted documents.* This conditions are extracted from (BOE, 1992, 2007) and are related with citizen rights and duties in their interaction with administrations by electronic means. Citizens must be notified by the administration if his/her record state changes, and submit documentation when they are requested to.

As Fig. 6 depicts, there are three subsets placed within the rule in order to get a better understanding of the interaction process. The first (1) subset evaluates the existence of *request* and *communication* transactions in the same procedure following the left hand side (LHS) pattern of a production rule. If the pattern is matched, the second subset (2) in the right hand side (RHS) of the rule creates the *submission* and *notification* procedures to handle the modeled transactions that match with the rule's LHS statement. Finally, the third subset (3) connects the previously created procedures with their correspondent transactions establishing the correct cardinality between them. It is worth to mention that this whole process takes place dynamically once the rule is fired.

3.3. Code generation module

Model transformation involves well known concepts from the fields of MDE and Model Driven Architecture (MDA) (Miller, Mukerji, & Al, 2003). Concepts include meta-models and profiles, platform independent model (PIM), platform specific model (PSM), model transformation, model to model transformation (M2M), and model to text transformation (M2T). The particular description of each of the MDA concepts can be consulted in (Miller et al., 2003), in this section only M2T will be addressed in further detail. From our experience and general estimation, Acceleo (Obeo, 2012) was the M2T transformation technology choice for this approach. We based our decision in its simple template generation syntax, abundant documentation and support. Acceleo enables the creation of expressions using a subset of OCL in order to query the input models. In the case of this approach, input mod-

Please cite this article in press as: Hernández, G. I., et al. Knowledge-based public service transactions: An intelligent model-driven approach in co-learning contexts. *Computers in Human Behavior* (2014), http://dx.doi.org/10.1016/j.chb.2014.09.044

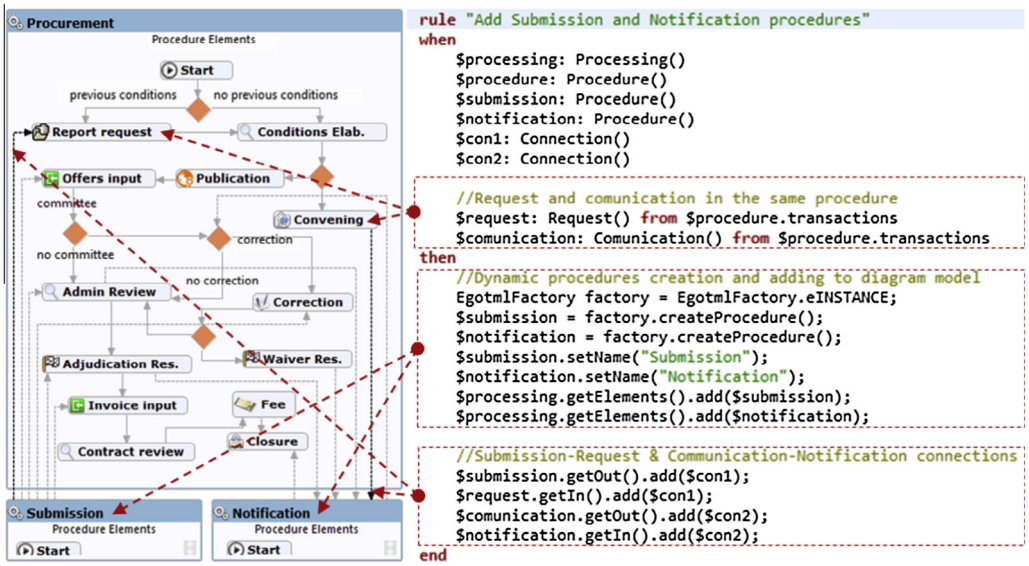

Fig. 6. Knowledge base rule and model instance interaction.

els are the ones generated by eGotDesigner and expressed in XMI format. One of the advantages of using the proposed approach is the use of platform independent technology. Since the model instances are persisted in XMI, the information can be transformed into the desired technology. In order to cover the deployment phase of the identified development cycle, a web platform prototype for performing e-government transactions was developed. The specific platform used to develop the prototype was Ruby on Rails (RoR) framework ("Ruby on Rails," 2013) due to its open-source nature and development productivity. Since RoR includes its own persistence engine (Active Record) and supports database introspection along with the Ruby language dynamism, some features like database seeding can be exploited to generate data from eGotML models and inject it into the web platform. Therefore, the code generation focuses on this particular feature in order to proof the concepts proposed. The final result from the generation process and the seeding execution can be seen in Fig. 7 where the citizen record management performed by the web platform is displayed. This web application prototype enables the management of the transactions and their triggering actions within a citizen record. This platform does not implements all the business logic required by a record management, this is actually beyond this investigation scope. Even though, the basic operations are supported as a proof of concept prototype for this approach.

In order to get a clear insight behind architecture modules integration details, Fig. 8 depicts them in a schematic manner. The integration between the Formalization module composed by the eGotDesigner tool and the Semantic Validation Module is performed by using the Eclipse Modeling Framework notification mechanism through content adapters. The same rules defined to monitor, validate and complete transactions models are reused by the eGovernment platform as part of its business logic. This

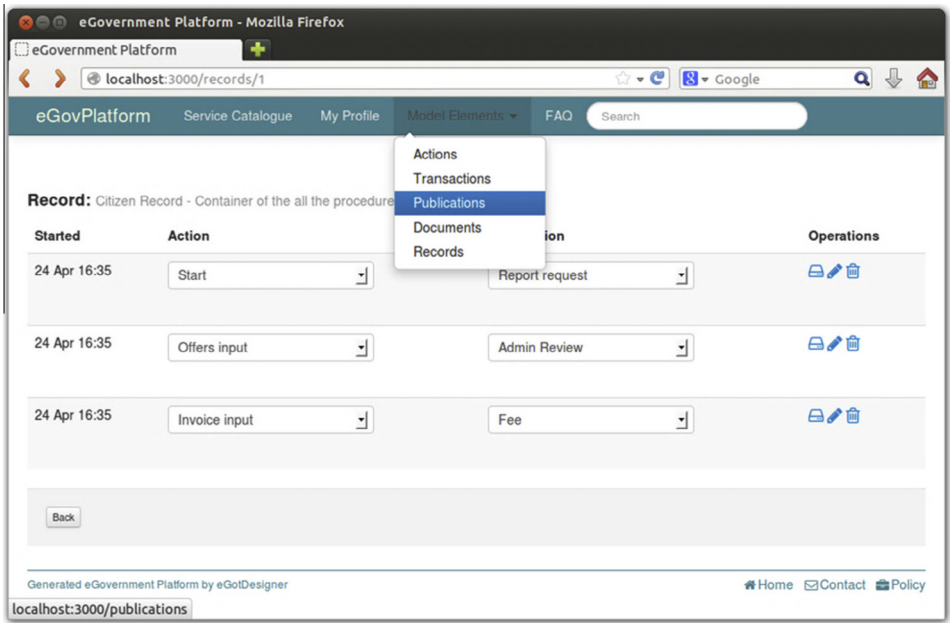

Fig. 7. eGovernment web platform prototype.

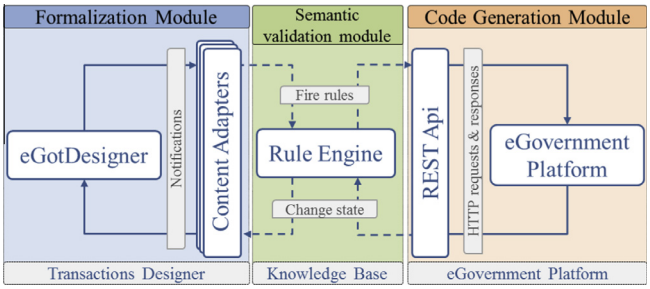

Fig. 8. Proposed Architecture modules' integration.

way the same logic from the modeling phase to deployment is shared in the whole cycle; this enables to avoid inconsistences between the transactions models defined by domain experts and the final services offered to citizens.

4. Evaluation

The context of this investigation hinders the performance of a comparative study where the same development cycle can be run twice since there are not similar approaches in this domain. The comparisons with current practice therefore needs to rely on domain experts' assessments. Surveys are a frequently used technique in software engineering research field to provide insight into complex issues and to support effective decision making (Kasunic, 2005). Therefore this technique can be a suitable approach for the context of this investigation. This section presents a survey designed with the purpose to gather experts' perceptions towards the presented approach. The survey is aimed to answer the following research question:

> Is the approach suitable to improve the eGovernment transactions development cycle?

Among the variety of measurement methods to employ, Likert Scale (Likert, 1932) is used in this study as it is the most commonly used measure in scale design. This study adopts the 5-point Likert Scale, with the responses rated as follows: 1 as strongly disagree, 2 as disagree, 3 as somewhat agree, 4 as agree, and 5 as strongly agree. The desired profiles of our survey responder are administrative employees or domain experts (DE) with experience in eGovernment practices, and software developers (SD). Since this is a very scattered target population, it is not possible to know the whole population from which the sample should be taken, therefore the snowball sampling method (Groves et al., 2004) has been used to create the sample. An invitation to participate in the survey was sent to potential participants and they were asked to invite other people with similar background. As the result of this process 41 people joined the group to participate in the survey. This proposal intention is to reduce dependencies among these profiles moving some SD work to DE. Hence, the feedback obtained from this group of experts enables to assess the viability of the approach in comparison to its current development processes in the eGovernment domain. Finally, the 90% of the people who were initially interested in participating in the survey completed it. To carry out the survey, we first did an interactive workshop where a demo was presented first to show the proposed features of the approach and later the experts explored the toolset in order to gain a deeper insight of the proposal. Right after the workshop the participants were asked to answer the questionnaire related with the workshop exploratory tasks (see Table 2).

4.1. Results

Studying each question individually, Table 3 and Fig. 9 show the descriptive statistics (min and max, quartiles 1–3 including the median, range, inter-quartile range, and mode) for the raw data by treating all the responses together as a whole.

5. Discussion

The main conclusions that can be drawn from the analysis of the results depicted in Table 3 are:

- From Q7 to Q12 (with Q10 exception) is concentrated the highest median, thus they are the questions that most of the participants agree with. One of the most important conclusions that can be drawn from the obtained results is that the 61% of the participants (see Fig. 10 for percentages) strongly agree that our approach seems to improve actual eGovernment transactions practices and the phases needed to develop such phases (Q7). This conclusion answers positively to our initial research question. The 56% of the responders strongly agree that using this approach the validation can be executed at design-time avoiding functional errors and a developer can do less repetitive work by using this approach with the correspondent increased productivity. Also, more than the 50% of the responders are strongly agree with this approach since it enforces the functional requirements of the proposed architecture with a 53% and makes explicit the implicit eGovernment domain knowledge by capturing it in a formal high level structure with a 56%.
- Q4 is the most polemic question since it presents the highest inter-quartile range with a value of 2, thus this represent the highest degree of variation. This is the question along with Q10 that most participants disagree with, i.e. 8% and 14% of the participants for both questions respectively, although 31%

Table 2
Questionnaire.

Question	Description
Q1	By using eGotML language it is possible to model eGovernment procedures and transform them in software solutions
Q2	This proposal enables the validation of transactions defined within a procedure, which prevent from fall in unlawful procedures
Q3	By using this proposal an eGovernment expert can be directly involved in the creation of solutions that citizens can use
Q4	This approach establishes a common channel that improves the communication between actors involved in eGovernment transactions, i.e. experts, developers and citizens
Q5	Public administrations could benefit from this proposal to perform more efficient transactions and save the corresponding time and money involved
Q6	An eGovernment expert can adapt and validate the generated software solutions for a procedure by using this approach
Q7	Broadly, this approach seems to improve actual eGovernment transactions practices and the phases needed to develop them
Q8	When using this approach validation can be executed at design-time avoiding functional errors
Q9	A developer can do less repetitive work by using this approach with the correspondent increased productivity
Q10	This approach results in software being less sensitive to changes in eGovernment transactions requirements and technology used to implement them
Q11	The approach enforces the functional requirements of the proposed architecture
Q12	This approach makes explicit the implicit eGovernment domain knowledge by capturing it in a formal high level structure

Please cite this article in press as: Hernández, G. I., et al. Knowledge-based public service transactions: An intelligent model-driven approach in co-learning contexts. *Computers in Human Behavior* (2014), http://dx.doi.org/10.1016/j.chb.2014.09.044

Table 3
Overall descriptive statistics.

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12
Min	3	2	2	2	2	3	2	2	2	2	3	2
Quartile 1	4	4	4	3	4	4	4	4	4	3.75	4	4
Median	4	4	4	4	4.5	4	5	5	5	4	5	5
Quartile 3	5	5	5	5	5	5	5	5	5	5	5	5
Max	5	5	5	5	5	5	5	5	5	5	5	5
Range	2	3	3	3	3	2	3	3	3	3	2	3
Inter quartile-range	1	1	1	2	1	1	1	1	1	1.25	1	1
Mode	4	4	4	4	5	4	5	5	5	4	5	5

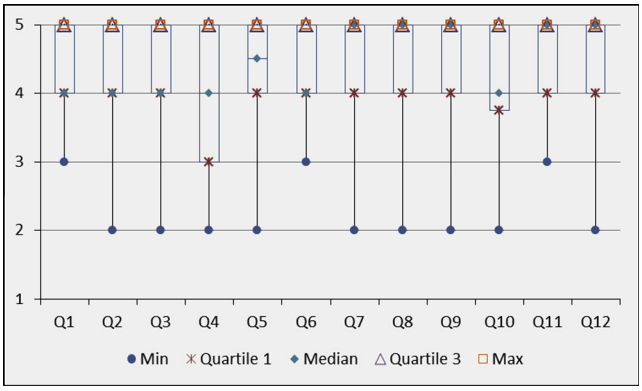

Fig. 9. Box and whiskers plot for each question.

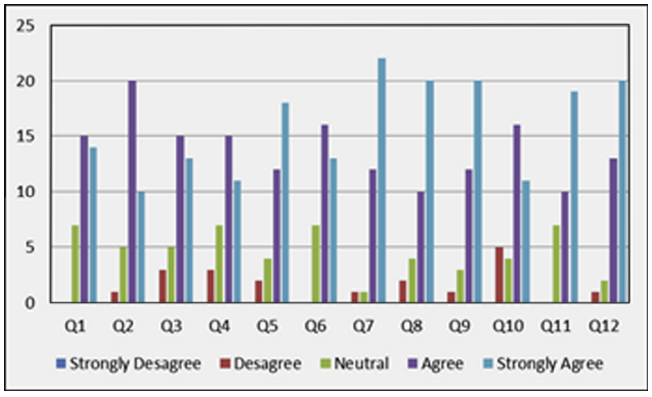

Fig. 10. Overall response distribution.

of the responders strongly agree that this approach establishes a common channel that improves the communication between actors involved in eGovernment transactions, and that it results in software being less sensitive to changes in eGovernment transactions requirements and technology used to implement them.

- The questions where responders did not express disagreement are Q1, Q6 and Q11, since the minimum value is 3 for each one. In the case of Q1, 42% of the participants agree that using eGotML language it is possible to model eGovernment procedures and transform them in software solutions. Regarding to Q6, the 44% of the participants agree in that an eGovernment expert can adapt and validate the generated software solutions for a procedure by using this approach. In the case of Q11 the 53% of the responders strongly agree with this approach in enforcing the functional requirements of the proposed architecture. The minimum value in the rest of the questions is 2, which represents the 36% of the total responders.
- To conclude the analysis of the answers, the statistical results have divided into DE and SD profiles. Figs. 11 and 12 show the descriptive statistics of both profiles answers' analysis.

The most significant differences among these profiles can be summarized as follows:

- In the case of question Q1, both groups agree with the modeling and transformation of eGovernment procedures in software solutions by using eGotML language since 4 is the median in each group, thus 50% of the participants in each group agree with this question. Even though, there is no disagreement with this question in both groups and the mode is 5 in the DE group, the degree of dispersion is higher among the responders of this profile (the inter-quartile range is 1.75). 28% of DE participants are not sure about using eGotML language to model eGovernment procedures and transform them in software solutions. This result is reasonable since domain experts are less related with terms such as software solutions or code generation, even though they were instructed in the workshop and doubts were clarified. There is a higher degree of consent among the SD respondents (56%), who are more used to such terms. In both groups, 39% of the participants strongly agree with Q1.
- With regard to question Q2, both groups agree with defining the validation of transactions within the procedure to prevent from falling into unlawful practices. Although one SD disagrees with it, there is no disagreement among the DE participants, 39% of them strongly agree with it.
- With regard to question Q3 the median for each group is 4, thus the 50% of the responders from both profiles agreed with the role of an expert directly involved in the creation of solutions that citizens can use. Although, there are not discrepancies between the profiles (the inter-quartile range is 1 in both) there are two DE and a SD that do not agree with this question. DE participants expressed that they do not see themselves as an active role in the development of new software solutions and the SD explained that this role should be played by a developer. Despite these discrepancies, 42% of both groups agree with this question whereas 36% strongly agree.
- With regard to Q7, the median in both groups match the maximum value of 5 with a 61% of strongly agree answers from both profile participants. As mentioned before this result answers positively to our research question in this case with a low degree of variation (interquartile range is 1) which means that there is unanimity between both profiles.
- In the case of question Q8, both groups agree with the validation executed at design-time to avoid functional errors since the medians are above 4 in each group. Although, there is an 11% of disagreement among SD group, the mode is 5 in both groups. It is reasonable that the degree of dispersion among responders of SD profile (inter-quartile range is 1.75) be higher than DE participants since the terms of "design-time" and "functional errors" are more familiar to SD experts which can

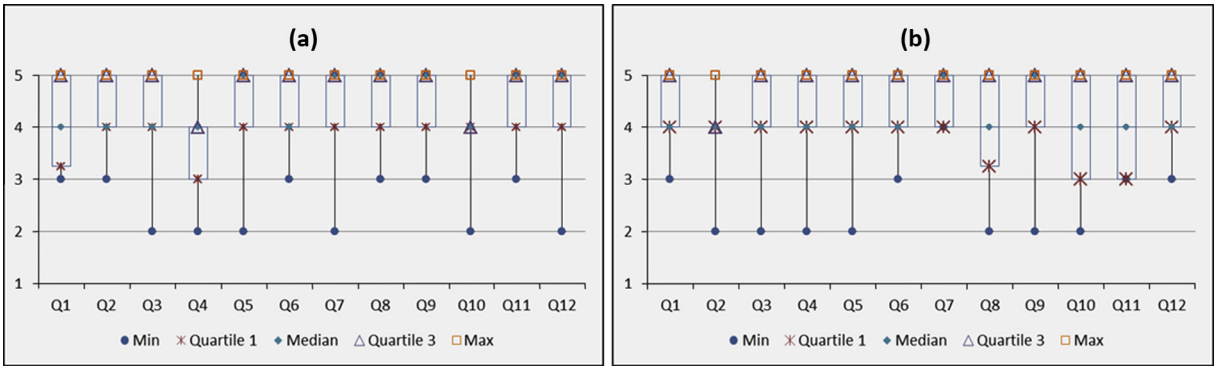

Fig. 11. Boxplot by profiles: (a) DE and (b) SD.

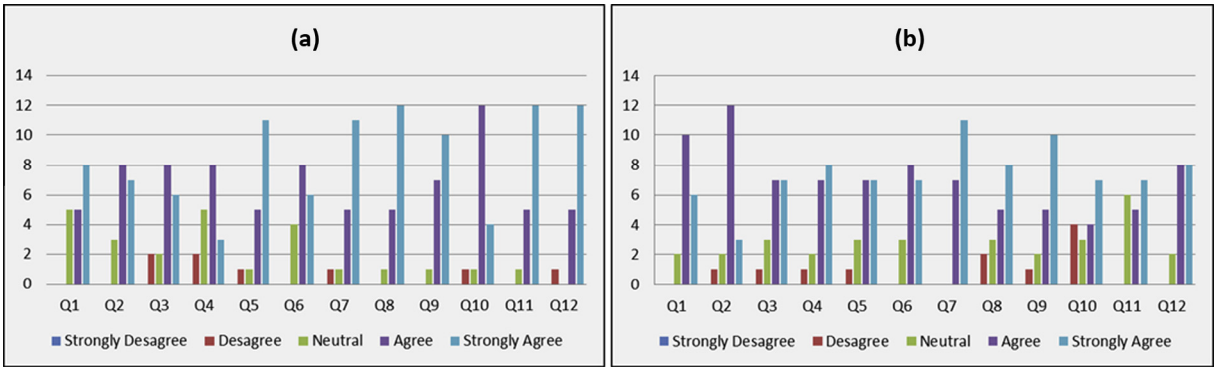

Fig. 12. Response distribution by profiles: (a) DE and (b) SD.

debate more accurately with regard to this matter. In both groups, 56% of the participants strongly agree with Q8.

6. Conclusion and future work

In this article we have described a new model-driven and knowledge based approach that enables domain experts to cover the modeling, validation and deployment of public services in e-government platforms. We have identified the repetitive phases involved in eGovernment transactions that suffer variations in their implementation and established a cycle with them. Since these phases repeat each time a new change emerges, they are susceptible to be automatized; therefore we have designed a layered architecture to establish the modules needed to undertake the phases' development tasks. Each module has been described along with their inner components. Our approach fosters the definition of transactions models and the construction of various generators in order to transform models' information into different specific platforms as needed. Finally we have integrated all the architecture modules as an environment that shares the same rules defined to monitor, validate and complete transactions models with the eGovernment platform as part of its business logic. This way the same logic from the modeling phase to deployment is shared in the whole cycle; this enables to avoid inconsistences between the transactions models defined by domain experts and the final services offered to citizens. We have further conducted a web survey to evaluate, based on experts' perceptions, if our approach is suitable to improve the eGovernment transactions development cycle. We can conclude that both consulted experts profiles of software developers and domain experts mainly agree with the presented proposal since the mode of the scores given is 4 of a maximum scale value of 5. In future work, we would like to test the approach with a wider sample of procedure transactions to compare the different performances, times and effort employed in each one. The approach itself can be extended to support the generation of different platform-specific applications, this could allow the interoperation among the applications against the same knowledge base. The improvement of the eGotDesigner editor usability is another task that we would like to do. Finally in the immediate future we pretend to solve the issues pointed by the experts in the questions where they express disagreement and conduct an experiment in which they can use the tools and test the approach.

References

Aalst, W. van der, & Hee, K. van. (2002). *Workflow management: models, methods, and systems*. <http://mitpress.mit.edu/catalog/item/default.asp?tid=8633&ttype=2>.

Bertolino, A., De Angelis, G., Di Sandro, A., & Sabetta, A. (2011). Is my model right? Let me ask the expert. *Journal of Systems and Software, 84*(7), 1089–1099. http://dx.doi.org/10.1016/j.jss.2011.01.054.

BOE. (1992). Ley 30/1992, de 26 de noviembre, de Régimen Jurídico de las Administraciones y del Procedimiento Administrativo Común. <http://www.boe.es/buscar/doc.php?id=BOE-A-1992-26318>.

BOE. (2007). Ley 11/2007, de 22 de junio, de acceso electrónico de los ciudadanos a los Servicios Públicos. <http://www.boe.es/buscar/doc.php?id=BOE-A-2007-12352>.

Buhl, H. U., & Löffler, M. (2011). The role of business and information systems engineering in E-Government. *Business & Information Systems Engineering*. http://dx.doi.org/10.1007/s12599-011-0182-4.

CENATIC. (2011). *Public procurement procedure*. <http://www.cenatic.es/laecsp/page30/page30.html> (Retrieved 21.03.13).

Ciaghi, A., Weldemariam, K., Villafiorita, A., & Kessler, F. B. (2011). Law modeling with ontological support and BPMN: A case study. *Business*, (c), 29–34. <http://www.thinkmind.org/index.php?view=article&articleid=cyberlaws_2011_2_20_70022>.

Clements, P., Garlan, D., Little, R., Nord, R., & Stafford, J. (2003). Documenting software architectures: Views and beyond. In *6 25th International conference on software engineering 2003 proceedings* (pp. 740–741). Addison-Wesley Professional. doi: http://dx.doi.org/10.1109/ICSE.2003.1201264.

Please cite this article in press as: Hernández, G. I., et al. Knowledge-based public service transactions: An intelligent model-driven approach in co-learning contexts. *Computers in Human Behavior* (2014), http://dx.doi.org/10.1016/j.chb.2014.09.044

Drools Expert. (2013). <http://www.jboss.org/drools/drools-expert.html> (Retrieved 21.03.13).

Fowler, M. (2010). *Domain specific languages*. *International journal of computer applications IJCA* (vol. 1, pp. 39–60). Addison-Wesley Professional. <http://martinfowler.com/dslwip/>.

Gronback, R. C. (2009). ECLIPSE MODELING PROJECT a domain-specific language toolkit. In G. Doench, K. Hatt, & J. S. Nicolas-Shirley (Eds.), *Architecture* (p. 689). Addison-Wesley Professional. <http://portal.acm.org/citation.cfm?id=1530861>.

Groves, R. M., Fowler, F. J., Couper, M. P., Lepkowski, J. M., Singer, E., & Tourangeau, R. (2004). Survey methodology. In J. A. Harkness, M. Braun, B. Edwards, T. P. Johnson, L. E. Lyberg, P. P. Mohler, ... T. W. Smith, (Eds.), *Statistics* (Vol. 2, p. 448). Wiley-Interscience. doi: http://dx.doi.org/10.2307/1504821.

Hill, E. F. (2003). *Jess in action: Java rule-based systems. Environment*. Manning Publications Co.

Kasunic, M. (2005). Designing an effective survey. *Carnegie Mellon Software Engineering Institute, CMU SEI-20*(September). <www.sei.cmu.edu/pub/documents/05.reports/pdf/05hb004.pdf>.

Klischewski, R., & Wetzel, I. (2002). Service flow management: Caring for the citizen's concern in designing e-government transaction processes. In *Proceedings of the 35th annual Hawaii international conference on system sciences* (pp. 1637–1646). IEEE Comput. Soc. doi: http://dx.doi.org/10.1109/HICSS.2002.994074.

Likert, R. (1932). A technique for the measurement of attitudes. *Archives of Psychology, 140*(140), 1–55. http://dx.doi.org/10.1111/j.1540-5834.2010.00585.x.

McBride, B. (2002). Jena: A semantic Web toolkit, 6 IEEE Internet Computing 55–59. IEEE Computer Society. doi: http://dx.doi.org/10.1109/MIC.2002.1067737.

Miller, J., Mukerji, J., & Al., E. (2003). MDA Guide Version 1.0.1. *Object Management Group, 234*(June), 51. <http://www.omg.org/mda/faq_mda.htm>.

Moody, D. (2009). The "physics" of notations: Toward a scientific basis for constructing visual notations in software engineering. *IEEE Transactions on Software Engineering, 35*(6), 756–779. http://dx.doi.org/10.1109/TSE.2009.67.

Muñoz-Cañavate, A., & Hípola, P. (2011). Electronic administration in Spain: From its beginnings to the present. *Government Information Quarterly, 28*(1), 74–90. http://dx.doi.org/10.1016/j.giq.2010.05.008.

Obeo. (2012). Acceleo. <http://www.eclipse.org/acceleo/> (Retrieved 21.03.13).

Lytras, M. D., & Pouloudi, A. (2006). Towards the development of a novel taxonomy of knowledge management systems from a learning perspective: An integrated approach to learning and knowledge infrastructures. *Journal of Knowledge Management*. http://dx.doi.org/10.1108/13673270610709224.

Osterweil, L. J., Schweik, C. M., Sondheimer, N. K., & Thomas, C. W. (2005). Analyzing processes for e-government application development. *Journal of E-Government, 1*(4), 63–87. http://dx.doi.org/10.1300/J399v01n04_05.

Papavassiliou, G., Ntioudis, S., Abecker, A., & Mentzas, G. (2003). Supporting knowledge-intensive work in public administration processes. *Knowledge and Process Management, 10*(3), 164–174. http://dx.doi.org/10.1002/kpm.176.

Ruby on Rails. (2013). <http://rubyonrails.org/> (Retrieved 21.03.13).

Savvas, I., & Bassiliades, N. (2009). A process-oriented ontology-based knowledge management system for facilitating operational procedures in public administration. *Expert Systems with Applications, 36*(3), 4467–4478. http://dx.doi.org/10.1016/j.eswa.2008.05.022.

Selic, B. (2003). The pragmatics of model-driven development, 20 IEEE Software 19–25. IEEE Computer Society Press. doi: http://dx.doi.org/10.1109/MS.2003.1231146.

Sottara, D., Mello, P., & Proctor, M. (2010). A configurable rete-OO engine for reasoning with different types of imperfect information. *IEEE Transactions on Knowledge and Data Engineering, 22*, 1535–1548. http://dx.doi.org/10.1109/TKDE.2010.125.

Steinberg, D., Budinsky, F., Paternostro, M., & Merks, E. (2008). *EMF: Eclipse Modeling Framework*. In E. Gamma, L. Nackman, & John Wiegand, (Eds.) *Engineering* (p. 744). Addison-Wesley Professional. <http://portal.acm.org/citation.cfm?id=1197540>.

Tambouris, E. (2001). An integrated platform for realising online one-stop government: The eGOV project. In *DEXA '01 Proceedings of the 12th international workshop on database and expert systems applications* (p. 359). <http://dl.acm.org/citation.cfm?id=874050.874685>.

Tambouris, E., Boukis, B., Vassilakis, C., & Lepouras, G. (2002). SMARTGOV: A governmental knowledge-based platform for public sector online services. In *Proceedings of the KMGov2002 workshop* (pp. 173–185). Denmark. <http://www.sdbs.uop.gr/?q=node/50>.

Wimmer, M. A. (2003). Knowledge management in electronic government. In *4th IFIP International working conference*, KMGov 2003 Rhodes, Greece, May 26–28, 2003.

Yang, D., Tong, L., Ye, Y., & Wu, H. (2006). Supporting effective operation of e-governmental services through workflow and knowledge management. In K. Aberer, Z. Peng, E. Rundensteiner, Y. Zhang, & X. Li (Eds.), *Web information systems WISE 2006* (Vol. 4255, pp. 102–113). Springer, Berlin Heidelberg. doi: http://dx.doi.org/10.1007/11912873_13.

Yuan, M., Zhang, X., Chen, Z., Vogel, D. R., & Chu, X. (2009). Antecedents of coordination effectiveness of software developer dyads from interacting teams: An empirical investigation. *IEEE Transactions on Engineering Management, 56*(3), 494–507.

Zhang, X., De Pablos, P. O., & Zhou, Z. (2013). Effect of knowledge sharing visibility on incentive-based relationship in Electronic Knowledge Management Systems: An empirical investigation. *Computers in Human Behavior, 29*(2), 307–313. http://dx.doi.org/10.1016/j.chb.2012.01.029.

Zhang, X., Vogel, D. R., & Zhou, Z. (2012). Effects of information technologies, department characteristics and individual roles on improving knowledge sharing visibility: A qualitative case study. *Behaviour & Information Technology*.

Please cite this article in press as: Hernández, G. I., et al. Knowledge-based public service transactions: An intelligent model-driven approach in co-learning contexts. *Computers in Human Behavior* (2014), http://dx.doi.org/10.1016/j.chb.2014.09.044