An Ontological Framework for Integrated Public Service Delivery

Sietse Overbeek[1], Marijn Janssen[1], and Patrick van Bommel[2]

[1] Faculty of Technology, Policy and Management, Delft University of Technology, Jaffalaan 5, 2600 GA Delft, The Netherlands
{S.J.Overbeek, M.F.W.H.A.Janssen}@tudelft.nl
[2] Institute for Computing and Information Sciences, Radboud University Nijmegen, Toernooiveld 1, 6525 ED Nijmegen, The Netherlands
P.vanBommel@cs.ru.nl

Abstract. Present-day governments are experiencing a shift from supplying common, non-electronic services towards supplying demand-driven and personalized e-services. An ontological framework for integrated public service delivery is presented to anticipate on the need to match supply and demand of public services. This framework contributes to distinguish key concepts and relations that form the basis for coordinating the activities necessary for integrated service delivery. The ontology has been realized by studying organizational processes, by conducting interviews as part of an expat case, and by literature study. In the case of this research, expats are persons who live in another country and want to come over to The Netherlands for their work. For this purpose, they need to request services that are integrated in a whole. The proposed ontology provides a foundation for an architecture blueprint that can enable demand-driven integrated service delivery in practice.

1 Introduction

Governments are experiencing a shift from supplying common, non-electronic services towards more demand-driven and personalized electronic service (e-service) delivery. To accomplish these goals, governments are becoming more externally oriented instead of focusing on internal functions. They are focusing more on their client's needs and less on their own functionality, organizational structure, and boundaries. Initially, public organizations focused on recurring client needs instead of on incidental needs. As such, assessing needs and reacting to needs do not provide the flexibility to react to new needs or even changes in laws and regulations. Government functions are fragmented due to constitutional, legal, and jurisdictional limitations. As a consequence, governments are often acting in silo structures, but nowadays are forced to cooperate with other government agencies and partners in the private sector.

One of the initiatives that anticipates on the aforementioned governmental developments is the long-term 'B-dossier' research project [1]. Results of this project are specifically aimed at providing computer-based support for government agencies to realize a more Integrated Service Delivery (ISD). For ISD the tasks of several types

of actors need to be coordinated. In this case, an actor can be defined as an entity that is able to perform a task, such as a human or a computer.

The focus of the research reported in this paper is to determine relations and interdependencies between the main concepts for ISD. ISD requires that public organizations collaborate with each other, which in turn requires insight in the relationships among the functionalities and services provided by such organizations. This understanding contributes to distinguish concepts and relations that form the basis for coordinating the activities necessary for ISD. This is realized by the development of an ontology for ISD, which aligns and abstracts domain knowledge found in a case in the public domain and by studying relevant literature. Section 2 clarifies background knowledge necessary to develop an ontology for ISD. From an organization-centric view, knowledge is acquired from the case by applying a bottom-up approach which consists of studying organizational processes that include information on service supply during process fulfillment. Several process models have been created as a result of this study. Besides studying organizational processes, we have analyzed how actors would fulfill their part of a process. This has been realized by conducting interviews. Thus, we have applied an organizational view and an actor-centric view when studying processes in public organizations. These views have been extended by incorporating other views on public processes from current literature before proceeding to the realization of an ontological knowledge framework in section 3. Finally, section 4 concludes this paper.

2 Background

Ontologies are becoming essential for organizations, because ontologies are machine-processable semantic resources for many application areas [2]. An ontology is an agreed understanding of a certain domain, formally represented as logical theory in the form of a computer-based resource. By sharing an ontology, autonomous and distributed software applications can meaningfully communicate to exchange data and thus make such data transactions interoperate independently of their internal technologies. Relating the notion of ontology to the research described in this paper, it can be noticed that organizations sharing an ontology which includes semantics related to the public domain create a starting point for realizing ISD.

To understand the relations and interdependencies between main concepts for ISD we have studied processes involved in the *expat* case in detail. Actor involvement in those processes has been studied by interviewing expats who search and request public services. Expats are, in this case, persons who live in another country and want to come over to The Netherlands for their work. For this purpose they will at least need a (temporary) residence permit, a registration in the citizens' registry, a bank account, a job, a health insurance, and housing. Process models have been developed for each of these scenarios. The expat case contains typical problems of ISD and involves organizations that need to collaborate. An example of a problem related to ISD is to bridge the digital divide [3]. Citizens lacking Internet access at home should be able to use e-services by other means, such as community self-service terminals.

A process model containing processes that are required to fulfill when requesting a residence permit for expats has been developed and is shown in figure 1.

Fig. 1. High-level process model for requesting and receiving a residence permit.

This model is based on the Business Process Modeling Notation (BPMN) [4], which is an industry standard graphic notation for representing organizational processes. The model is based on information concerning the processes related to the acquisition of a residence permit provided by the Dutch Immigration and Naturalization Service (INS). Figure 1 shows that there are three composite processes involved when obtaining a residence permit. First, a temporary residence permit has to be requested by an expat followed by the request of a permanent residence permit before registering at the municipality. To comprehend what the composite processes exist of, detailed process models for the composite processes have also been developed. It has been noticed that several parties are involved in the process to let an expat obtain a temporary residence permit. The process starts by the expat requesting to obtain a residence permit from the INS. The remaining process steps can then be fulfilled until the expat collects the residence permit from the INS. An ontology that is shared by public organizations is an agreed understanding of the public domain and as such enables to identify essential concepts and relations between concepts.

Studying public processes in which multiple parties interact is but one of the ways to achieve a better understanding of ISD. This can be regarded as an organization-centric view on ISD, because public processes are arranged by public organizations such as the INS, municipalities, and embassies. As part of the expat case, eleven interviews have been conducted with expats to understand how they participated in process fulfillment during their attempts to acquire a residence permit, a registration in the citizens' registry, and so on. Lessons learned from this actor-centric view can not only be used to improve current governmental processes and service delivery, but they can also be used to understand ISD from an actor's point of view.

Central issues related to the residence permit process that were experienced by the interviewed expats are concerned with: Information that is presented in Dutch only, governmental Web sites that are not functioning properly, contradictory information presented by multiple public organizations, and serious human mistakes during service delivery. The latter is related with loss of documents and failing / forgetting to inform other parties in the process. These issues obviously appeal to improve ISD, of which the ontological framework presented in section 3 can act as a step in the right direction. Our research contributes to at least partly resolve the aforementioned issues. The issue of presenting contradictory information can be resolved by letting public organizations share an ontology such as is presented in the next section, so that an agreed understanding is realized concerning the public domain. Based on the ontology, integrated public e-services can communicate to exchange data and thus make such data transactions interoperate independently of technology. This can at least partly resolve the issue of malfunctioning Web sites. Human mistakes during

service delivery can be diminished when more insight is provided how services can be offered and integrated for repeatable service requests.

Now that we have gathered both organization-centric and actor-centric insights on processes in the public domain with respect to a case in which typical problems for ISD are surfaced, it is possible to develop an ontology that contributes to distinguish the key concepts and relations for coordinating the activities necessary for ISD.

3 Ontological framework for integrated public service delivery

To accurately describe key concepts and relations between such concepts in the public domain, the domain knowledge that we have aggregated in the previous section is used as a basis for the creation of an ontological knowledge framework. Besides these insights the ARIS EPC (Event-Driven Process Chains) model [6] can be used for ontology creation, because it provides a description of a public process that extends our organization-centric and actor-centric views with service-centric, resource-centric, and event-centric views. The service-centric view describes concrete services offered by organizations that are required by actors for successful process fulfillment. The resource-centric view describes resources belonging to an organization and which processes consume which resources. To understand the event-centric view an explanation of this view is provided in the next section. First, an ORM representation of the ontological framework for ISD is presented in section 3.1. Second, an OWL specification of the ORM model is presented in section 3.2.

3.1 ORM representation of the ontological knowledge framework

Figure 2 shows an Object-Role Modeling (ORM) model of the proposed ontological framework for ISD. ORM is a conceptual data modeling technique, which can not only be used for the conceptual modeling of database models, but for a variety of modeling purposes such as the modeling of ontologies.

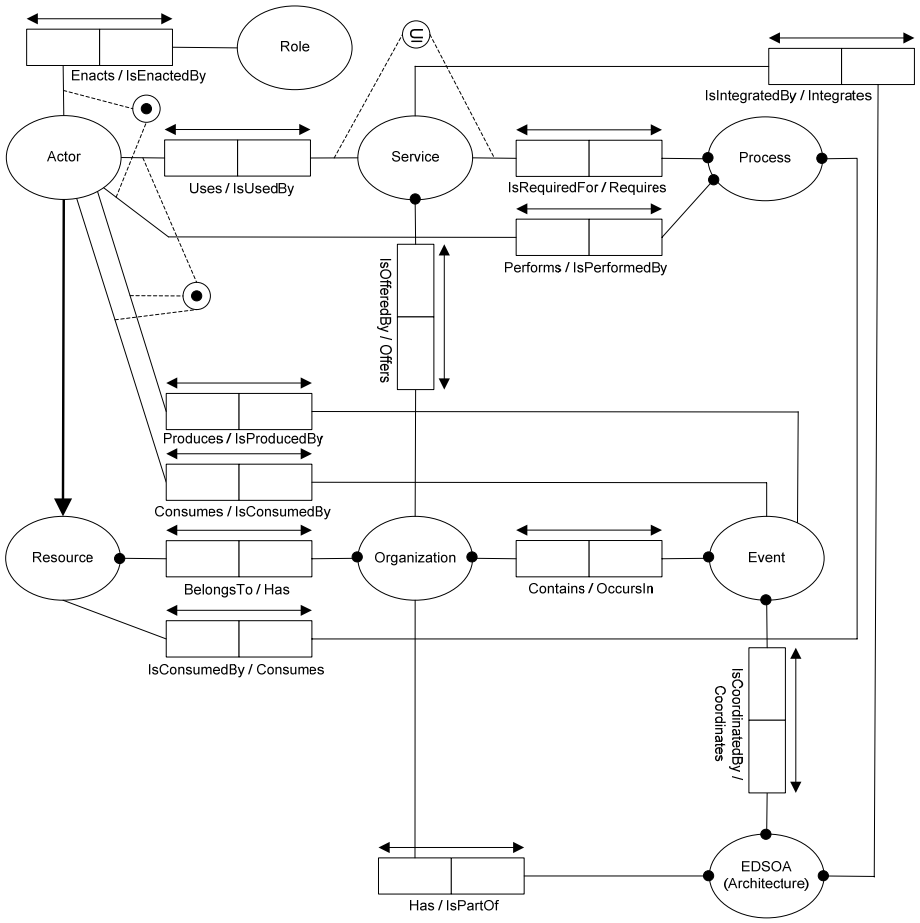

Fig. 2. An ontological framework for integrated public service delivery.

In an ORM model, ovals represent object types (which are counterparts of classes), whereas boxes represent relations between object types. These relations are dubbed as fact types. For more details on Object-Role Modeling, see e.g. [7].

There are eight central concepts that are part of the ontology. These are the concepts of role, actor, service, process, resource, organization, event, and an Event-Driven Service-Oriented Architecture (EDSOA) concept. A description of a public process from an actor-centric and organization-centric view, such as discussed in section 2, forms the basis of introducing the 'actor' and 'organization' concepts in the ontology. The concepts of service, resource, and event are part of the ARIS EPC model [6]. The role concept is introduced in the ontology to be able to denote a specification of an actor enactment. An actor is a resource of an organization that enacts a role during process performance or, on a more granular level, task performance. An employee enacting the role of registrar at a municipality is an example of such an actor at a public organization. These actors can use services that are offered by organizations. Services are on their turn required by actors during

process performance to assist actors in process fulfillment. For example, an expat that performs the process to acquire a residence permit is provided with an e-service to request a permit online and to provide those digital documents to government agencies that are necessary for the permit request.

Finally, the concepts of event and event-driven service-oriented architecture need to be introduced. An EDSOA in the context of this research defines a methodology for designing and implementing computer-based applications and systems in which events are transmitted between a set of integrated and interacting services [8]. Such events are consumed or produced by actors in organizations. An actor that consumes an event can subscribe to an architecture that manages such events, and an actor that produces an event publishes to this architecture. When an event is broadcasted by an actor, the architecture facilitates that this event is forwarded to a demanding actor. If a demanding actor is unavailable, the architecture can facilitate the storage of the event and try to forward it later. This architecture-based coordination of events can be dubbed as *event orchestration* [1]. An example of an event in the residence permit process shown in figure 1 can be an event 'residence permit form received'. A subsequent event that can be produced by a receiving expat is a 'residence permit signed' event. Building applications and systems based on an EDSOA allows these applications and systems to be more responsive, since such systems are more oriented to unpredictable and asynchronous environments. Eventually, implementation of an EDSOA based on the ontological framework shown in figure 2 can enable ISD and orchestration of events between services in practice.

To increase usability for public organizations that wish to adopt the ontological framework shown in figure 2, the ontology might be specified in multiple specification languages, such as XML, RDF, RDF-S, OWL, etc. (see e.g. [2]). These languages are specifically designed for use by computer-based applications that need to process the content of information instead of just presenting information to human actors. However, the Web Ontology Language OWL facilitates greater machine interpretability of Web content than that supported by e.g. XML, RDF, and RDF Schema (RDF-S) by providing additional vocabulary along with a formal semantics [2]. To increase successful adaptation and machine interpretability of our ontological framework, an OWL specification of the ontological framework that has been visualized in ORM so far is presented in the next section.

3.2 OWL representation of the ontological knowledge framework

A partial OWL representation of the ontological framework for integrated public service delivery is shown in figure 3. This representation verbalizes the concepts, relations, and constraints of the ORM model shown in figure 2.

```
<owl:Class rdf:ID = "Resource" />
<owl:Class rdf:ID = "Actor">
  <rdfs:subClassOf rdf:resource = "#Resource" />
</owl:Class>
<owl:Class rdf:ID = "Role" />
<owl:Class rdf:ID = "Service" />
<owl:Class rdf:ID = "Process" />
<owl:Class rdf:ID = "Organization" />
<owl:Class rdf:ID = "Event" />
```

```
<owl:Class rdf:ID = "Architecture" />
<owl:ObjectProperty rdf:ID = "Enacts">
  <owl:inverseOf rdf:resource= "IsEnactedBy" />
  <rdfs:domain rdf:resource = "#Actor" />
  <rdfs:range rdf:resource = "#Role" />
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID = "Uses">
  <owl:inverseOf rdf:resource= "IsUsedBy" />
  <rdfs:domain rdf:resource = "#Actor" />
  <rdfs:range rdf:resource = "#Service" />
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID = "Produces">
  <owl:inverseOf rdf:resource= "IsProducedBy" />
  <rdfs:domain rdf:resource = "#Actor" />
  <rdfs:range rdf:resource = "#Event" />
</owl:ObjectProperty>
....
<owl:Restriction>
  <owl:onProperty rdf:resource="#Coordinates.Event" />
  <owl:minCardinality rdf:datatype=
   "&xsd;nonNegativeInteger">1</owl:minCardinality>
</owl:Restriction>
```

Fig. 3. A partial OWL representation of the ontological framework for integrated public service delivery.

Representing the ontological concepts and relations between those concepts by means of OWL yields differences compared to modeling it in ORM. However, both languages intend to express the same meaning. For example, the ORM uniqueness constraint that spans over 'Enacts / IsEnactedBy' cannot be expressed in OWL, as it is implied by definition [2]. I.e., the formalization of ObjectProperties in OWL does not allow the same tuple to appear twice in the same set, such as Enacts = {<actor1,role1>,<actor1,role1>}. The other uniqueness and mandatory constraints are all expressed as a cardinality restriction in OWL. For instance, the mandatory constraint on 'Coordinates' is expressed in OWL by the constraint 'owl:minCardinality'. An 'owl:minCardinality' constraint of one or more means that all instances of the class must have a value for the property.

The differences in modeling the ontology as described above illustrate different ways of characterizing the ontology. The contrast in formalizations and constructs of both languages causes such differences. The choice of which language is more suitable for specifying an ontology depends on the application scenario and perspectives of the ontology [2]. For example, ORM and EER are suitable for database and XML-based application scenarios since they are extensive in their treatments of data set integrity. Description logic based languages such as OWL seem to be more applicable for deductive and reasoning-based application scenarios, as they focus on the expressiveness and the decidability of axioms.

As a next step in this research, an event-driven service-oriented architecture will be developed that adopts the proposed ontological framework for ISD as the knowledge base. The ontological framework provides a foundation for describing actor context, public processes, resources, etc. Therefore, the ontology helps the architecture to dynamically compose a personalized process flow and automate the execution of the process flow.

4 Conclusions

Governments are experiencing a shift from supplying common, non-electronic services towards more demand-driven and personalized electronic service (e-service) delivery. To anticipate on these developments, an ontological framework for integrated public service delivery is described in this paper. The key concepts and relations that form the basis for coordinating the activities necessary for integrated public service delivery can be distinguished by means of this ontology. It is also a foundation for an EDSOA that can integrate services and orchestrate events between services in practice. Knowledge to develop the ontology is acquired by studying a case in which expats search and request public services, and by studying relevant literature. Expats are persons who live in another country and want to come over to The Netherlands for their work. Public processes have been studied from an organizational viewpoint resulting in detailed process models. This has been illustrated by a process for requesting a residence permit. Furthermore, several expats have been interviewed to understand how actors participating in such processes would fulfill their part of the process resulting in an actor-centric view on public processes. Next, the studied processes in the expat case have been hierarchically structured by identifying the core processes, the actors that participate in each core process, and the tasks those actors fulfill as part of a core process. Finally, an ontological framework for ISD is realized by extending the results from the expat case with findings from related literature. The resulting ontology is represented as an ORM model and described in OWL.

References

1. Klievink, A.J., Janssen, M.F.W.H.A., Lankhorst, M.M., Leeuwen, D. van: An Event-driven Service-Oriented Architecture for Coordinating Flexible Public Service Networks. In: Wimmer, M., Jochen Scholl, H., Ferro, E. (eds.) Electronic Government, 7th International Conference, EGOV 2008, Turin, Italy, August 31 - September 5, 2008, Proceedings. LNCS, vol. 5184, pp. 133-140. Springer, Berlin (2008)
2. Jarrar, M., Meersman, R.: Ontology Engineering – The Dogma Approach. In: Dillon, T., Chang, E., Meersman, R., Sycara, K. (eds.) Advances in Web Semantics I. LNCS, vol. 4891, pp. 7-34. Springer, Berlin (2008)
3. Ke, W., Wei, K.K.: Successful E-Government in Singapore. Communications of the ACM. 47, 95-99 (2004)
4. OMG, Business Process Modeling Notation (BPMN) Version 1.0. OMG Final Adopted Specification, Object Management Group (2006)
5. Dang, J., Hedayati, A., Hampel, K., Toklu, C.: An Ontological Framework for Adaptive Medical Workflow. Journal of Biomedical Informatics. 41, 829–836 (2008)
6. Wikipedia. Event-driven Process Chain. Available from: http://en.wikipedia.org/wiki/Event-driven_process_chain
7. Halpin, T.: Information Modeling and Relational Databases, from Conceptual Analysis to Logical Design. Morgan Kaufmann, San Mateo, CA (2001)
8. Yuan, S.-T., Lu, M.-R.: An Value-Centric Event Driven Model and Architecture: A Case Study of Adaptive Complement of SOA for Distributed Care Service Delivery. Expert Systems with Applications. 36, 3671-3694 (2009)