Semantic Interoperability Middleware – Enabling Collaborative Networks in Public Sector

Alejandro Sanchez[1], Elsa Estevez[2], Tomasz Janowski and Adegboyega Ojo[3]

United Nations University-International Institute for Software Technology
Center for Electronic Governance
P.O. Box 3058, Macau
{ale, elsa, tj, ao}@iist.unu.edu

Abstract. Electronic Government initiatives, such as seamless public services delivered at one-stop government portals, require establishing collaborative networks among public- and private-sector organizations. However, semantic interoperability problems emerge as these organizations may differ in the terms and meanings they use to communicate, express their needs and describe resources they make available to each other. This paper describes typical semantic interoperability problems and presents a middleware solution to address them, called Semantic Interoperability Middleware (SIM). The paper illustrates the problems through three case studies in a collaborative network for the delivery of welfare benefits. Subsequently, the requirements for SIM are presented, and the architecture and design of the solution are specified using UML. SIM assumes organizations have agreed on ontologies that reflect the meaning of terms they use in communicating. It comprises three services: Mediation – resolves differences in terms and meaning; Validation – detects inconsistent terms and meaning; and Discovery – mediates and matches need with resource descriptions. Finally, the case studies are resolved applying SIM.

Keywords: Semantic Interoperability, Electronic Government, Collaborative Networks

1 Introduction

Electronic Government is one of the responses of governments around the world to social pressures demanding higher quality of public services and efficiency in government operations through. It refers to the use of Information and Communication Technologies (ICT), particularly the Internet, as a tool to achieve better government [1]. In particular, governments typically aim to facilitate access and improve the delivery of public services.

In order to realize these objectives, governments make efforts to deliver electronic public services through an interface that reflects the needs of citizens and businesses, rather than the structure of the government. An internet portal that provides access following this approach is called one-stop government portal [2]. To complete requested services, various public- and private-sector organizations must seamlessly collaborate following the principles of cross-organizational ownership to government objectives and goals [3]. When services are delivered in this way they are called seamless public services and the collaborations realized by the various involved participants establish ad-hoc collaborative networks (CN).

A CN is a group of entities, such as individuals or organizations, largely autonomous, geographically distributed, heterogeneous in various aspects including their goals, but collaborating

[1] Departamento de Informática - Facultad de Ciencias Físico Matemáticas y Naturales, Universidad Nacional de San Luis, San Luis, Argentina
[2] Departamento de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur, Bahía Blanca, Argentina
[3] Department of Computer Sciences, Faculty of Science, University of Lagos Lagos, Nigeria

together to better achieve common or compatible goals, and interacting through computer networks [4]. However, interoperability problems emerge as these organizations may differ and have its own IT platform and software applications to support their business processes.

Interoperability is the ability of Information and Communication Technology (ICT) systems and the business processes they support, to exchange data and to enable sharing of information and knowledge [5]. In particular, semantic interoperability requires that the precise meaning of exchanged information is understandable by the recipient application, even when the application was not initially developed for this purpose [6]. Achieving semantic interoperability becomes difficult when organizations differ in terms and meanings they use in communicating, and middleware to mediate and translate the semantic differences is required. For instance, two schools that exchange information about students may interpret information incorrectly, if they use a common schema but different grading scales and fail to detect this situation.

Semantic interoperability can be addressed applying one of two approaches: one based on schemas and one based on ontologies. Schemas can be used to show meaning, as organizations can agree on how to understand each section of a particular schema [7]. Ontologies can be used to explicitly specify how information must be interpreted [8]. They are logical theories that partially specify conceptualizations – the set of rules used to isolate and organize objects when tasks are performed. In the context of this paper, we refer to the first approach as schema-based and the second as ontology-based. Although the most popular is the first one, after the advent of the semantic web, the use of ontologies for sharing and reusing semantics has gained recognition [9].

In this paper we describe the semantic interoperability problems facing CNs in the public sector and present a middleware solution, called Semantic Interoperability Middleware (SIM), to address such problems. We illustrate the problems through three representative case studies in a CN for the delivery of welfare benefits. Subsequently, we specify the requirements for SIM to address the problems and model the architecture and design of the solution using UML. SIM follows an ontology-based approach and assumes CN partners have agreed on ontologies that reflect the meaning of terms they use in communicating. It also provides a bridge to the schema-based approach with the purpose of making existing information systems semantically interoperable. SIM comprises three services: Mediation – resolves differences in terms and meaning; Validation – detects inconsistent terms and meaning; and Discovery – mediates and matches need with resource descriptions. Finally, the approach is demonstrated by applying SIM to resolve the case studies. In addition, we reviewed other projects and software solutions addressing semantic interoperability.

The rest of the paper is organized as follows: Section 2 presents related work; Section 3 describes semantic interoperability problems facing CNs in the public sector through three case studies; Section 4 presents the requirements, architecture and design for SIM; Section 5 describes the application of SIM to address the case studies; finally, Section 6 draws conclusions.

2 Related Work

This section reviews projects and software solutions addressing semantic interoperability problems in CN. The first two projects/solutions are domain-independent and the last two are focused in public sector. For each one, a brief description and a comparison against SIM are included.

ATHENA [10] (Advanced Technologies for Interoperability of Heterogeneous Enterprise Networks and their Applications) is a domain-independent project. It aimed at supporting research activities enabling enterprises to seamlessly interoperate. One of the project deliverables is the ATHENA Interoperability Framework (AIF). AIF includes a semantic reconciliation suite based on ontologies providing tools as follows: A* - A semantic annotation method and tool; ATHOS – An authoring and ontology management system; THEMIS – A repository for storing, managing and retrieving RDF schemas; ARES – A semantic rules engine for reconciling exchanged messages; and ARGOS – A tool to define, create, store and manage transformation rules used to reconcile documents. The semantic reconciliation suite addresses the problem of edition and storing ontologies which is not addressed by SIM. It includes ARES, which is comparable to the mediation service, but it does not include any equivalent service to discovery and validation.

WSMO [11] (Web Service Modelling Ontology) environment is a domain-independent software solution. It comprises the ontology (WSMO), the Web Service Modelling Language (WSML), and the Web Service Execution Environment (WSMX). In particular, WSMX is a comprehensive software framework for the discovery, selection, mediation, invocation and interoperation of web services based on semantic descriptions. It obtained the best results in the SWS Challenge (http://sws-challenge.org/). Its architecture can be seen as a P2P Network of nodes, each one containing pluggable components. WSMX covers several aspects which are not in the scope of SIM, but presents a strong dependence on web services. These aspects include resource management, and non-functional and functional selection of services, in addition to service discovery, and process mediation. SIM does not depend on any particular communication middleware and includes a validation extension which seems not to have an equivalent in WSMX .

SmartGov [12] (A Governmental Knowledge-Base Platform for Public Sector Online Services) is a project focused in the public sector. It aimed at specifying, developing, deploying and evaluating a knowledge-based platform to assist public sector employees to generate online transaction services by simplifying their development, maintenance and integration with existing IT systems. The project delivered a repository of XML documents for organizing knowledge, arranged according taxonomies derived from an Electronic Government Service Ontology. SmartGov follows a schema-based approach to address the problems. Although it considers the use of an ontology, it is only for generating taxonomies and not for executing the offered functionality. In contrast, SIM does not assume any particular ontology.

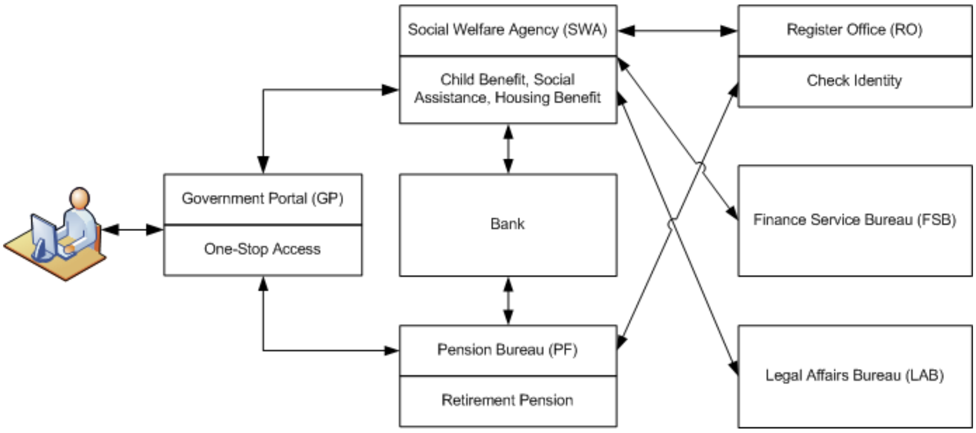
SemanticGov [13] (Providing Integrated Public Services to Citizens at the National and Pan-European level with the use of Emerging Semantic Web Technologies) is a project focused in the public sector. It aims at building the infrastructure (software, models and services) required for offering semantic web services in Public Administration. This infrastructure will support semantic interoperability among several Public Administration agencies within and among different countries. Semantic interoperability requirements to be supported include: automatic discovery of services by customers and execution of services spanning multiple agencies in inter-workflows. It relies on WSMO environment and a refinement of the ontology for the public sector. In contrast, developing a concrete ontology is out of the scope of SIM.

3 Semantic Interoperability Problems

We describe semantic interoperability problems to be addressed by SIM through three representative case studies. The case studies were selected upon analyzing the business processes supporting the major services delivered by the Government of Macao SAR [14], and literature about Electronic Government initiatives [15].

The context for the case studies, as shown in Figure 1, is a CN for the delivery of welfare benefits at a one-stop portal. Welfare benefits are provided to help in people’s living conditions and financial problems. However, not all citizens are eligible for all benefits. In our CN, the child benefit is for parents with at least two non-adult children; the housing benefit is for citizens that do not own any residence; the social assistance is for citizens whose income is less than a specified amount; and the retirement pension is for citizens above 65. These benefits can be accessed at the Government Portal (GP). The Social Welfare Agency (SWA) issues child benefits, social assistance and housing benefits, and the Pension Bureau (PB) issues retirement pensions. SWA and PB need to collaborate with other organizations: Register Office (RO) for checking the identity of citizens; Financial Bureau (FB) for checking the citizen's income; Legal Affairs Bureau (LAB) for checking the properties owned by citizens; and Bank (B) for paying the benefits. The next four sub-sections present the semantic interoperability problems and the first three are illustrated through case studies.

Figure 1: Collaborative Network for the Delivery Welfare Benefits


Social Welfare Agency (SWA)
Child Benefit, Social Assistance, Housing Benefit
Register Office (RO)
Check Identity
Government Portal (GP)
One-Stop Access
Bank
Finance Service Bureau (FSB)
Pension Bureau (PF)
Retirement Pension
Legal Affairs Bureau (LAB)

3.1 Semantic Heterogeneity

Case Study: SWA is responsible for granting social assistance to citizens. Before issuing the benefit, the agency must confirm the identity of all persons that live in the same residence as the applicant. To this end, SWA needs to exchange information with RO. However, the two agencies arrange their information around different concepts as depicted in Figure 2. SWA applies the concept of "household" – a group of people sharing a residence, and RO applies the concept of "family" – people related by family bonds. The two agencies share information about citizens but they use different terms for it: RO uses "person" and SWA uses "member".

Problem: Organizations may use different terms and meanings to communicate, express their needs and describe the resources to fulfill them. As a result, they may not be able to communicate or find an available resource to satisfy their needs.

3.2 Semantic Inconsistency

Case Study: An applicant completes a child benefit application and requests its submission. GP accepts the application as it is correct according to a generic check performed upon applications for any service. SWA receives the information and checks if the application is eligible. If not, the application is rejected and the applicant is notified through GP. It would be desirable to validate the eligibility of the application before accepting it, and provide immediate feedback to the applicant.

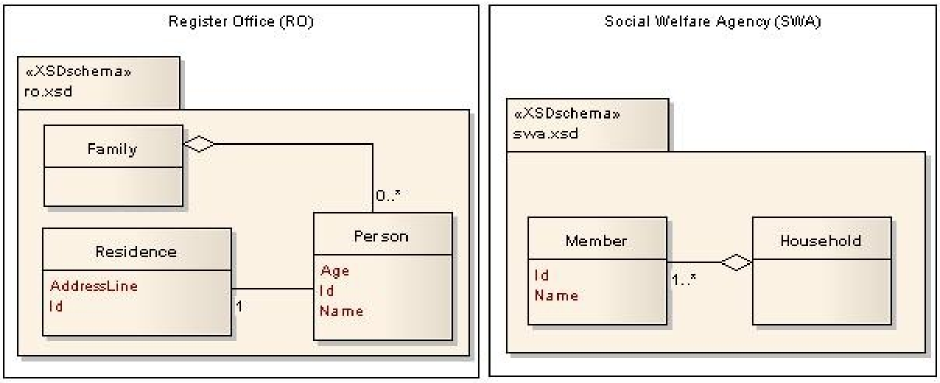
Problem: Organizations may exchange data but fail to recognize that they understand the exchanged terms differently. As a result, the exchanged data may cause semantic inconsistencies. In the case above, the application is considered valid by GP but it is not semantically valid according to the eligibility requirements in SWA.

3.4 Semantic Gap

Case Study: Citizens may not be aware that they are eligible for benefits that may help in their life situations. This problem becomes more relevant when citizens access public services through GP, as there is no officer to guide them. It would be desirable that GP proactively suggests benefits suited to citizens, based on their personal information and eligibility criteria.

Problem: Organizations and individuals express their needs and describe the resources to fulfill such needs using different levels of abstraction. Because of this, they may not be able to locate a resource that is available and able to fulfil their need.

Figure 2: Household and Family – Heterogeneous Data Models



3.4 Evolving Semantics

Any solution must consider the evolving nature of laws and regulations that affect how software applications interpret exchanged data. In Electronic Government, laws and regulations define how exchanged information must be interpreted. Changes are normal in this context and any solution must strive to minimize the effort to update software to reflect such legal/regulatory changes.

4 Semantic Interoperability Middleware

This section presents a middleware solution, called Semantic Interoperability Middleware (SIM) addressing the semantic interoperability problems of CNs in the public sector as described in Section 3. It was built following a development process comprising requirements specification, conceptual modelling, architecture design, detailed design, and implementation. Technical details as well as development artefacts are presented in the next sub-sections.

4.1 Requirements

Three functional requirements and two non-functional requirements were defined. The functional requirements are: F1) Mediation; F2) Validation; and F3) Discovery. The non-functional ones are: NF1) Explicit Semantics; NF2) Semantic Platform Transparency. The requirements relate to the problems in Section 3 as shown in Table 1.

NF1) Explicit Semantics: SIM shall make possible a flexible solution to semantic interoperability. It shall allow partially describing the semantics of information exchanged between organizations through an ontology, and base its execution upon that ontology.

F1) Mediation: SIM shall provide a service to infer the contents of a sent message according to the receiver's terms and meanings. The service shall resolve semantic heterogeneity relying on an ontology that specifies how the terms used in the message expressed according to the sender's conceptualization relate to the terms in the receiver's conceptualization.

Table 1: Semantic Interoperability Problems and SIM Requirements

Semantic Interoperability Problem		SIM Requirement
Semantic Heterogeneity		Mediation (F1)
		Discovery (F3)
S2	Semantic Inconsistency	Validation (F2)
S3	Semantic Gap	Mediation (F1)
		Discovery (F3)
S4	Evolving Semantics	Explicit Semantics (NF1)

F2) Validation: SIM shall provide a service for ensuring that the data contained in a message is consistent with a concept, as defined in an ontology that specifies the semantics of the information.

F3) Discovery: SIM shall allow inferring which resources are related to a need. Given the descriptions of a need, the description of a set of resources, and particular criteria relating them, a discovery service shall infer a set of resources related to the need. The service shall rely on an ontology specifying the relations between terms at different abstraction levels and belonging to different conceptualizations that are used to describe resources and needs.

NF2) Semantic Platform Transparency: A client application shall be independent of the ontology language and inference tools used by SIM.

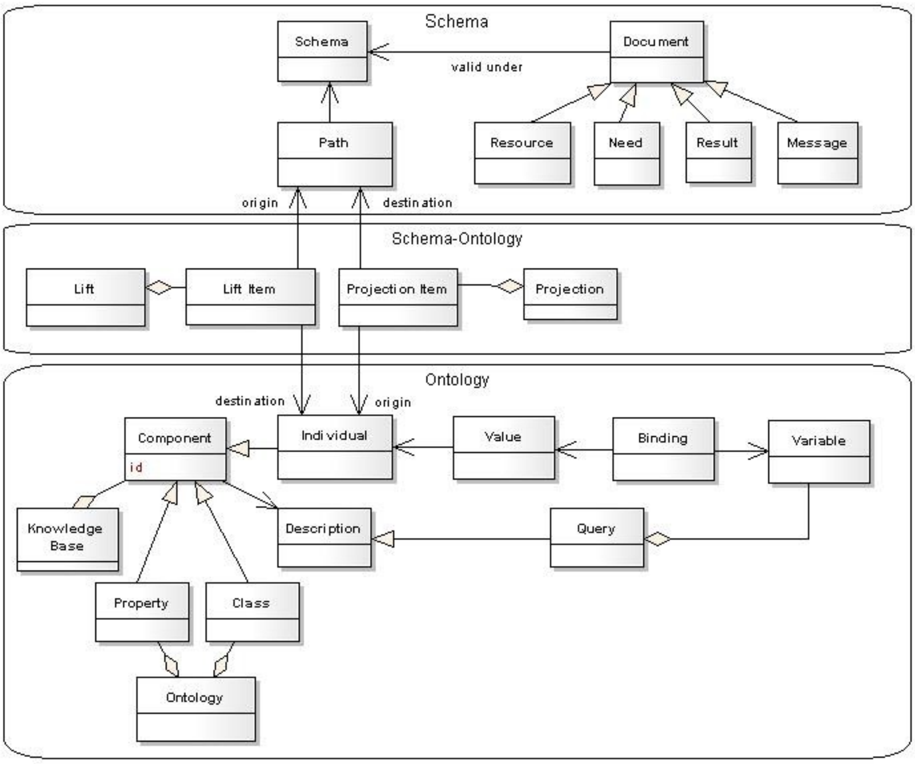
4.2 Conceptual Model

SIM key concepts are shown in Fig. 3. The concepts can be grouped into three sets: concepts related to Schema, concepts related to Ontology, and concepts for connecting both approaches (Schema-Ontology). The model aims at describing concepts relevant to the solution, but not at providing a full account of schema- and ontology-related concepts.

`Schema` is a concept upon which typical semantic interoperability approaches are based on. It is used to define valid types of documents by constraining and structuring their content. In particular, it establishes which elements are expected at a particular document section. The concept of `Document` denotes a container of structured data. A document may or may not have an associated schema. The concept of `Path` denotes a pattern expression identifying a section of a `Document`. SIM differentiates between the following types of documents: `Message` – for exchanging information, `Need` – for describing a need, `Resource` – for describing an available resource, and `Result` – for describing extracted resources. The document types `Need`, `Resource` and `Result` are used for implementing the discovery service.

The concept of `Ontology` refers to a partial and explicit specification of a conceptualization. In the context of this work, a conceptualization refers to the semantics of exchanged information. Therefore, an ontology describes semantics of exchanged information. Ontologies comprise two types of elements: `Class` – an abstraction mechanism used for classification; and `Property` - assertion of facts about classes. The concept of `Knowledge Base` represents a software artefact that allows combining the elements of ontologies with `individuals. Individual` is the unit that can be classified. The concept of `Component` abstracts the common features of `class`, `property` and `individual`, i.e. `Description` and attribute `id`. A `Description` is the specification of either the semantics of a `Component` or the information contained by the `Component`, while `id` is used to identify a component. The concept of `Query` denotes another type of specification in which one or more parts are left underspecified in the form of variables. A `Variable` is a placeholder for values in knowledge base matching a specification in a query. When a `Query` is executed, a set of bindings is returned, each `Binding` relating a `Variable` with a `Value`.

Fig. 3. Conceptual Model



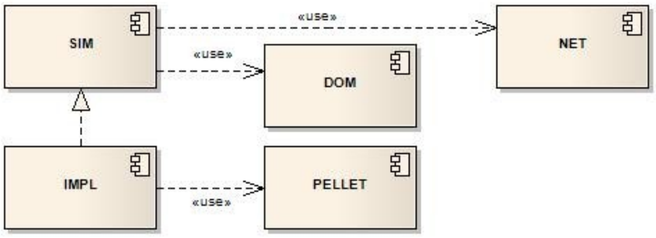
In order to exploit the benefits of an ontology-based approach to address semantic interoperability, and use the information in the documents specified through a schema-based approach, the concepts of `Lift` and `Projection` are introduced. A `Lift` specifies how the elements placed at a particular `Path` in a `Document` are transformed into individuals. A `Projection` establishes how an individual in a `Knowledge Base` should be located in a `Document` according to a `Path`.

4.3 Architecture

SIM architecture comprises five main components as shown in Fig. 4: `SIM` – providing the APIs for requesting the three services provided – validation, mediation and discovery; `DOM` – open source component providing APIs for handling XML documents; `NET` – open source component enabling to identify resources in Internet; `IMPL` – component implementing the interface defined in the SIM component; and `PELLET` – the inference tool [16] used in the current implementation. `SIM` component contains the API used by client applications for accessing the functionality. It relies on `DOM` and `NET` components for defining its methods. `SIM` includes a factory class allowing creating instances of the interface without depending on the concrete class implementing it. `IMPL` component represents an implementation of the interface defined in the `SIM` component for the Pellet inference tool. This component depends on `PELLET` component which contains Pellet API and other APIs used by Pellet to work with OWL [18] – the knowledge representation language.

The architecture addresses the non-functional requirements NF1 and NF2. NF1 – Providing Flexible Semantics - is achieved through the use of a knowledge representation language and an inference tool. NF2 – Providing Platform Transparency - is addressed through the factory class included in `SIM`, and the `SIM`, `DOM` and `NET` components, which decouple a client application from the concrete knowledge representation language and inference tool employed by a SIM implementation. As a result, if the knowledge representation language or/and inference tool are changed in a future version, the software code of client applications will not require to be updated. However, lift and projection specifications will do.

Fig. 4. Architecture



4.4 Design

SIM design is presented through a static view – design class diagram depicting the main classes and interfaces – and dynamic views – sequence diagrams depicting the collaborations for realizing the functionality. Only the sequence diagram for validation method is presented.

Static View: The static design diagram is shown in Fig. 5. The `SIM` interface defines the operations validate, mediate and discover. The operations rely on `DOM` and `NET` packages which represent the logical view of the components with the same name in the architecture. The `DOM` package defines the `Document` interface, which is used to model and operate on XML documents. `SIM` uses this interface to implement the Document concept presented in Section 4.2. The `NET` package contains a class for modelling and operating on the instances of the Uniform Resource Identifier (URI), which are used to identify ontologies, schemas, classes, properties, individuals and other types of resources. The `ProjectionSpec` and `Result` classes are defined to facilitate the access to XML Documents – these documents differ from the others since they have a fixed schema. These two classes provide implementation for the Projection and Result concepts, respectively. The `SIMFactory` class allows creating instances of the `SIM` interface without depending on the concrete class that implements the interface. The `SIM` class implements the `SIM` interface relying on `PELLET` and `OWL` packages. The `PELLET` package contains a set of interfaces and classes for using the Pellet inference tool (`Reasoner`) which in turn relies on the knowledge base. The `OWL` package includes classes for modelling and operating on elements of the OWL language. The `OWL` package classes and interfaces provide implementation for the Ontology, Class, Individual, Property, Query, Binding, Variable and Value concepts. The `Projection` and `Lift` classes define methods for operating on the Lift and Projection specifications, relying on `PELLET` and `OWL` packages. Note that `PELLET` and `OWL` represent the logical view of the component named PELLET in Section 4.3.

Dynamic View – Validate: The `Validate` method is realized as shown in the sequence diagram in Fig. 6. The interactions start when a client application obtains a `SIM` instance through the `SIMFactory` class (`createInstance`) and requests to validate a message according to the ontology (`validate`). The parameters of the `validate` method include: the message to be validated - `message`, the lift specification detailing how to transform message elements to `OWLIndividual` instances - `lift_spec`, the name of the ontology class specifying the validation criteria - `class_id`, and a reference to the ontology to be used for validation - `ontology_id`. SIM loads the referenced ontology into an `OWLOntology` instance (`loadOntology`), generates a set of

OWLIndividual instances - message_individuals through a Lift instance based on lift_spec and the message, and loads both - the ontology (loadOntology) and message_individuals (loadOntology) into the Reasoner instance. The reasoner uses its knowledge base to store both ontology and instances. Finally, it requests the Reasoner whether the knowledge base is consistent or not (isConsistent). If consistent, SIM obtains an OWLClass instance (getOWLClass) for the class_id referenced as a parameter, and asks Reasoner to infer individuals of this ontology class (getIndividuals). If the returned set instance is empty (isEmpty), the message is invalid, otherwise it is valid.

Fig. 5. SIM (Design - Static View)

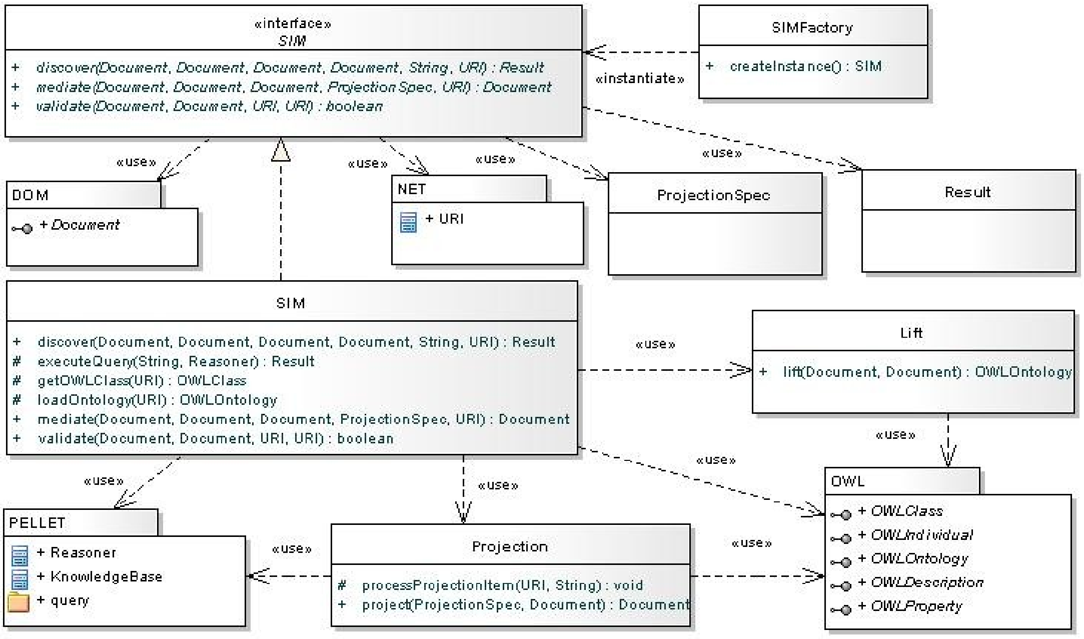
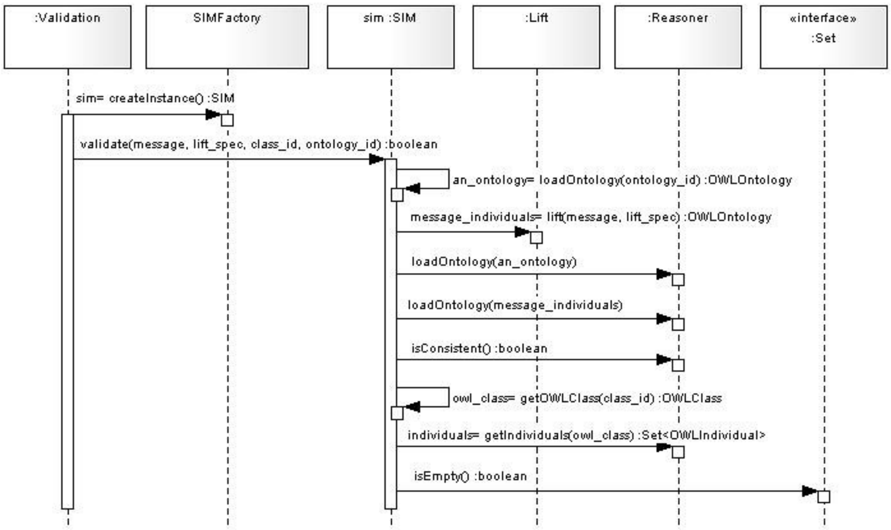

«interface»
SIM
+ discover(Document, Document, Document, Document, String, URI) : Result
+ mediate(Document, Document, Document, ProjectionSpec, URI) : Document
+ validate(Document, Document, URI, URI) : boolean
SIMFactory
+ createInstance() : SIM
«instantiate»
«use»
«use»
«use»
«use»
DOM
+ Document
NET
+ URI
ProjectionSpec
Result
SIM
+ discover(Document, Document, Document, Document, String, URI) : Result
# executeQuery(String, Reasoner) : Result
# getOWLClass(URI) : OWLClass
# loadOntology(URI) : OWLOntology
+ mediate(Document, Document, Document, ProjectionSpec, URI) : Document
+ validate(Document, Document, URI, URI) : boolean
«use»
Lift
+ lift(Document, Document) : OWLOntology
«use»
«use»
«use»
«use»
PELLET
+ Reasoner
+ KnowledgeBase
+ query
«use»
Projection
# processProjectionItem(URI, String) : void
+ project(ProjectionSpec, Document) : Document
«use»
OWL
+ OWLClass
+ OWLIndividual
+ OWLOntology
+ OWLDescription
+ OWLProperty

Fig. 6. Sequence Diagram - Validate


:Validation
SIMFactory
sim :SIM
:Lift
:Reasoner
«interface»
:Set
sim= createInstance() :SIM
validate(message, lift_spec, class_id, ontology_id) :boolean
an_ontology= loadOntology(ontology_id) :OWLOntology
message_individuals= lift(message, lift_spec) :OWLOntology
loadOntology(an_ontology)
loadOntology(message_individuals)
isConsistent() :boolean
owl_class= getOWLClass(class_id) :OWLClass
individuals= getIndividuals(owl_class) :Set<OWLIndividual>
isEmpty() :boolean

5 Applications

The next sub-sections describe how SIM is applied to resolve each case study in Section 3.

5.1 Semantic Heterogeneity

SIM Mediation can be applied to resolve the case study in Section 3.1 and mediate between the heterogenous data models. The following artefacts must be developed by SWA and RO domain experts: `swa.owl`, `ro.owl`, `swa-ro.owl`, `lift-ro.xsl`, `projection-swa.xml`. The `swa.owl` and `ro.owl` files contain ontologies describing SWA and RO terms and meanings - structured through `swa.xsd` and `ro.xsd` schemas, respectively (see Figure 2). The `swa-ro.owl` shown in Table 2is the mediation ontology used by SIM Mediation service during data exchanges. In addition, the `lift-ro.xsl` transforms the information in the original schema format (`ro.xsd`) into class individuals as defined in the local ontology (`ro.owl`). The projection (`projection-swa.xml`) specifies which ontology components can be projected into a XML document following `swa.xsd`.

Table 2. Ontology Fragments for SWA-RO Mediation

swa-ro.owl	Class: Household EquivalentTo: Residence that hasResident min 1 Person Class: Member EquivalentTo: Person that hasResidence min 1 Residence ObjectProperty: hasResidence Inverses: hasMember
swa.owl	ObjectProperty: hasMember Domain: Household Range: Member Class: Household SubClassOf: hasMember min 1 Member
ro.owl	ObjectProperty: hasFamilyMember Domain: Family Range: Person ObjectProperty: hasResidence Domain: Person Range: Residence ObjectProperty: hasResident Domain: Residence Range: Person Inverses: hasResidence

5.2 Semantic Inconsistency

SIM Validation can be applied to resolve the case study in Section 3.2. The service can be invoked by GP to validate the application informing as parameters: the application form data, the lift for transforming the data into individuals, the reference to the ontology used by the CN, and the name of the class specifying the eligible applications (`ValidApplicationForms`). Table 3 shows a fragment of the ontology used for validating, written following the Manchester OWL Syntax [18].

Table 3. Ontology Fragment for Validating Eligibility

Axiom	Description
Class: ValidApplicationForm EquivalentTo: ApplicationForm that hasApplicant only EligibleApplicant and hasDependant only ValidDependant	A valid application form is any application form having an applicant who is eligible and has only dependants who are valid
Class: EligibleApplicant EquivalentTo: Applicant that hasChild min 2 NotAdult	Specifies that an eligible applicant is any applicant who has at least 2 children, and each of these children is not an adult
Class: ValidDependant EquivalentTo: Dependant that hasValidRelationWith min 1 Applicant	Specifies that a valid dependent is any dependant who has a valid relation with at least one applicant
Class: Applicant EquivalentTo: Person that appliesAsBeneficiary min 1 ApplicationForm	An applicant is any person who applies for a benefit submitting at least one application form as beneficiary
Class: NotAdult EquivalentTo: Person that hasAge < 21	Specifies that a non-adult is any person who has less than 21 years
Class: Dependant EquivalentTo: Person that appliesAsDependant min 1 ApplicationForm	A dependant is any person who at least applies as dependant in at least one application form
ObjectProperty: hasParent SubPropertyOf: hasValidRelationWith	The property hasParent is a valid relation to apply as dependant
ObjectProperty: hasSpouse SubPropertyOf: hasValidRelationWith	The property hasSpouse is a valid relation to apply as dependant

5.3 Semantic Gap

SIM Discovery can be applied to resolve the case study in Seciton 3.3. Domain experts of CN members must describe the offered benefits and citizen's data through an ontology, as shown in Table 4. When the citizen logs in, GP can suggest benefits suited to his/her personal situation.

Table 4. Ontology Fragments for Discovering Benefits

Child Benefit	`Class: EligibleForChildBenefit`
	`  EquivalentTo: Person that hasChild min 2 Person`

Housing Benefit	`Class EligibleForHousingBenefit`
	`  EquivalentTo: Person that owns exactly 0 Residence`

Retirement Pension	`Class: EligibleForRetirementPension`
	`  EquivalentTo: Person that hasAge >=65`

6 Conclusions

This paper introduced SIM, a middleware solution addressing semantic interoperability problems facing collaborative networks in the public sector. The problems were illustrated through case studies in a collaborative network for the delivery of welfare benefits. Technical details about SIM development as well as development artefacts were presented. The application of SIM to the case studies for delivering services demonstrates the approach. The applications are available on-line at http://egov.iist.unu.edu/projects/interoperability. In addition, related projects/solutions were identified and comparisons between SIM and their results were introduced.

SIM offers three main semantic services – validation, mediation and discovery, following an ontology-based approach. The usage of SIM presents various advantages for CN partners, such as: changes introduced in laws/regulations can be incorporated by introducing a change to the ontology used by SIM, reducing the likelihood of requiring software maintenance; SIM offers an API which is independent of the underlying technologies, therefore the replacement of the ontology representation language and inference tool will not affect client applications.

Future research tasks are focused in raising the abstraction level of formal specifications of semantics, and developing formal models to improve the development process of software solutions addressing semantic interoperability problems facing CNs in the public sector.

Acknowledgements

The authors thank Rilwan Basanya for assisting with research and Khoi Nguyen for support with development tasks. This work was jointly funded by Microsoft Corporation and the Center for Electronic Governance at UN University IIST.

References

1. OECD e-Government Studies. “The e-Government Imperative”. OECD, 2003.
2. M. Wimmer, E. Tambouris, “Online One-Stop Government: A working framework and requirements”. In Proceedings of the IFIP World Computer Congress, 2002, Montreal.
3. UNDESA, “UN E-Government Survey 2008, From E-Government to Connected Governance”, United Nations, Department of Economic and Social Affairs (UNDESA), ISBN 978-92-1-123174-8, 2008.
4. L. Camarinha-Matos et al. “A reference Model for Collaborative Networks”, ECOLEAD Project. 2007. http://www.veforum.org/projects/284/Deliverables/ D52.2_Final.pdf.
5. eGovernment Unit, European Commission, “Study on Interoperability at Local and Regional Level. Interoperability Study Version 3”, December 2006.

6. European Commission. “European Interoperability Framework for Pan-European eGovernment Services Version 1.0”. European Communities, 2004.
7. D. Bjørner. “Software Engineering 2: Specification of Systems and Languages”, Springer, 2006.
8. N. Guarino, P. Giaretta. “Ontologies and Knowledge Bases: Towards a Terminological Clarification”, Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing, pp 25-32. IOS Press, Amsterdam, 1995.
9. M. Uschold, M. Grüninger. “Ontologies and semantics for seamless connectivity”, SIGMOD Record 33(4), pp 58-64. ACM Press, December 2004.
10. Information Society Technologies 507849, 6th Framework Programme. “ATHENA - Advanced Technologies for Interoperability of Heterogeneus Enterprise Networks and their Applications”. http://interop-vlab.eu/ei_public_deliverables/athena-deliverables/.
11. The Web Service Modelling Ontology. http://www.wsmo.org
12. G. Lepouras et al. “A Governmental Knowledge-based Platform for Public Sector Online Services – Final Project Report”. February 2004.
13. Information Society Technologies, 6th Framework Programme. “SemanticGov - Providing Integrated Public Services to Citizens at the National and Pan-European level with the use of Emerging Semantic Web Technologies”. http://www.semantic-gov.org.
14. T. Janowski et al. (Eds), “The State of Electronic Government in Macao”, Volume 2: Agencies, e-Macao Report 2, April 2005.
15. R. Basanya, R. et al. “Semantic Interoperability for Electronic Governemnt - A Survey”. Center for Electronic Governance at UNU-IIST, 2007.
16. E. Sirin et al. “Pellet: A practical OWL-DL reasoned”, Journal of Web Semantics, 5(2), 2007.
17. S. Bechhofer et al. (eds). “OWL Web Ontology Language Reference”, World Wide Web Consortium Recommendation. http://www.w3.org/TR/owl-ref/. February 2004.
18. M. Horridge, P. F. Patel-Schneider, “Manchester OWL Syntax for OWL 1.1”. OWLED 2008 DC OWL: Experiences and Directions, Washington, April 2008.