Semantic Web Enabled E-Government Services

Brahim Medjahed Athman Bouguettaya Mourad Ouzzani
Department of Computer Science, Virgnia Tech
7054 Haycock Road, Falls Church, VA 22044
{brahim,athman,mourad}@vt.edu
Project URL - http: //www.nvc.cs.vt.edu/dgov

Abstract

We propose a novel approach for organizing and describing e-government services on the envisioned *Semantic Web*. We combine the emerging concepts of *Web services* and *ontologies* to cater for Semantic Web enabled e-government services. This would lay the ground for the automatic selection, interoperation, and composition of e-government services.

1. Introduction

The *Semantic Web* is defined as an extension of the existing Web, in which information is given well-defined meaning [1]. It aims at providing machine-processable services and information on the Web. Efforts towards enabling the envisioned Semantic Web are gaining momentum. The emerging concept of *Web services* is slated to be the backbone of tomorrow's Web [2]. A *Web service* is a set of related functionalities that can be programmatically accessed through the Web [3]. One of the benefits of Web services is that users need no longer think in terms of *data* but rather *services* they wish to receive. This powerful concept is well suited for *e-government* applications where citizens are not expected to know about or understand the data infrastructure behind the provided services.

The semantics of Web services is crucial to the development of a Semantic Web enabled e-government. E-government agencies would automatically discover, "understand", and share each other services. *Ontologies* are expected to play a central role to empower Web services with semantics [1]. An *ontology* is a shared conceptualization based on the semantic proximity of terms in a specific domain of interest [2]. In this paper, we propose an ontological approach for organizing e-government services on the *Semantic Web*. To the best of our knowledge, this work is one of the first to combine the concepts of Web services and ontologies in e-government. We use welfare and social services within the *Family and Social Services Administration* (FSSA) as a case study.

2. Ontological Organization of E-Government Web Services

The number of Web services offered by government agencies can be large and continuously changing. This calls for techniques to organize government Web services in a way they can be efficiently discovered and "understood". For that purpose, we introduce the concept of *community* (Figure 1).

Communities provide means for an ontological organization of the available space. Each community is specialized in a specific area of interest (e.g., disability, adoption). Organizing e-government services into communities aims at reducing the overhead of discovering those services. Communities are themselves services; they are created, advertised, discovered, and invoked as "regular" Web services are. Community providers include government administrations, non-profit organizations, and businesses dealing with governments. We assume that the description of each community would be agreed upon by its providers ahead of time. E-government service providers (e.g., FSSA bureaus) identify the community of interest and register their services with it.

A community provides domain-specific information and terms of interaction with the community and its underlying Web services. Each community's definition includes three parts: *category*, *operations*, and *services*. The *category* contains three attributes: *domain*, *synonyms*, and *specialization*. *Domain* gives the area of interest of the community (e.g., "healthcare"). The *synonyms* attribute contains a set of alternative domains for the

category (e.g., “medical” is a synonym of “healthcare”). *Specialization* is a set of characteristics of the current category (e.g., “insurance” and “children” are specialization of “healthcare”). Communities are invoked through their *operations*. The invocation of each community operation is translated into the invocation of an e-government service registered with the *community services* element. Details about community operations and services are presented in the following sections.

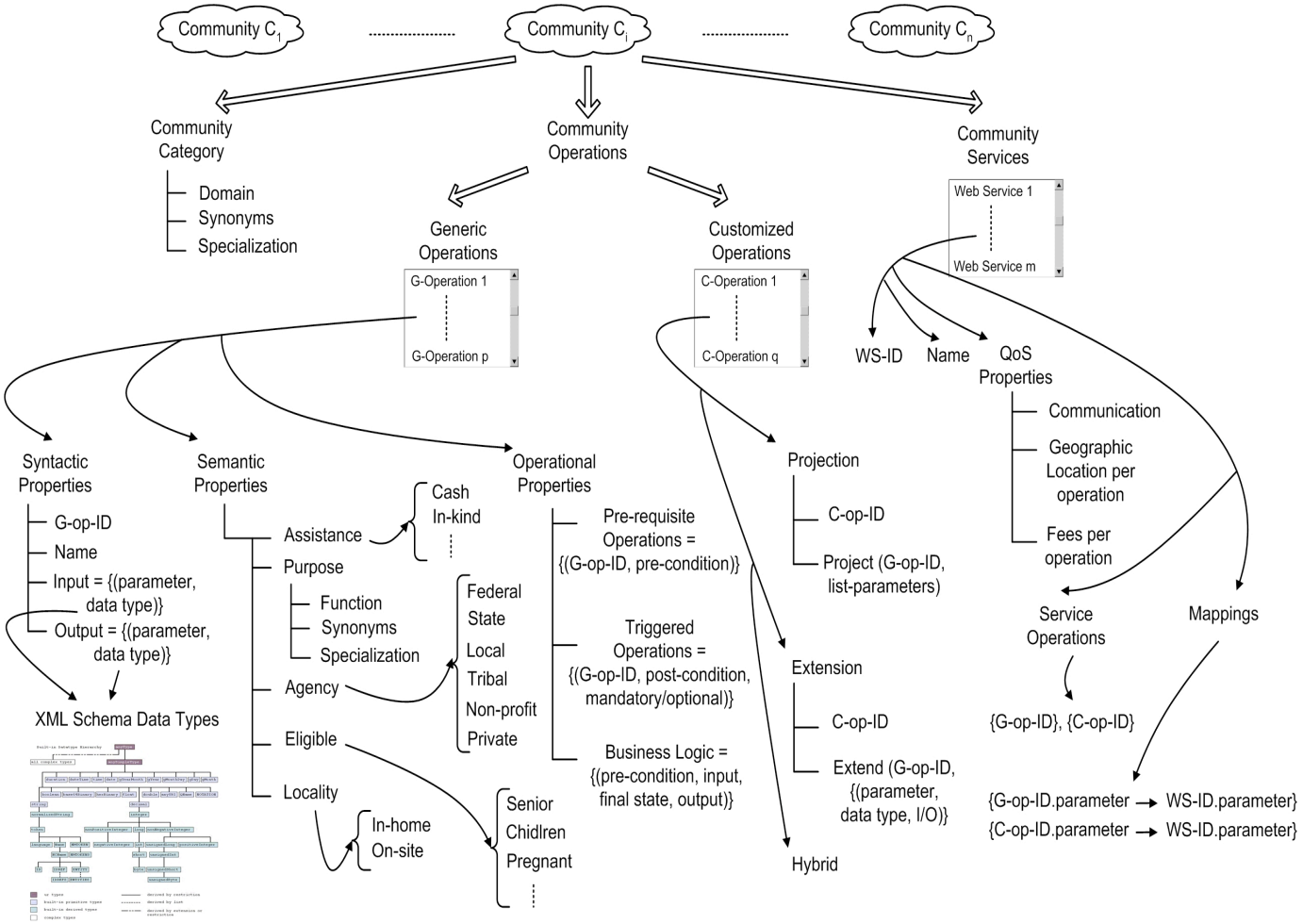

Figure 1. Organizing E-Government Web Services on the Semantic Web

3. Describing Community Operations

Communities are accessible through a set of pre-defined operations called *generic* operations. Generic operations may be used “as is” by e-government services or *customized* to best fit the capabilities of those services. They are described at three levels: *syntactic*, *semantic*, and *operational*.

Syntactic Properties – Generic operations are syntactically described by the following attributes: *G-op-ID*, *name*, *mode*, *input*, and *output* (Figure 1). *G-op-ID* is unique operation identifier that takes the form of a *Universally Unique ID* (UUID). The second attribute gives the name assigned to the current operation. The operation’s *mode* has one of the values *In*, *Out*, *In/Out*, and *Out/In*. *In* and *Out* modes represent one-way operations: *In* (*Out*) operation receives (sends) an input (output) message. *In/Out* and *Out/In* modes represent two-way operations: *In/Out* (*Out/In*) operation first receives (sends) an input (output) message and then replies (receives) by an output (input) message. Depending on the mode, each operation has input parameters, output parameters, or both. A parameter has a name and data type associated with it. We adopt XML Schema data types as a canonical type system for input and output parameters.

Semantic Properties – The semantics of operations is crucial to enabling e-government services on the semantic Web. Semantic properties defined for generic operations include: *assistance*, *purpose*, *agency*, *eligible*, and *locality* (Figure 1). *Assistance* gives the type of support provided by the operation. This may be *cash* or *in-kind*. For example, TANF (Temporary Assistance for Needy Families) is a welfare program within FSSA that provides financial support to needy families. Other types of support may be educational (e.g., enhancing communication skills of visually impaired people), informational (e.g., returning the list of Area Agencies on Aging), etc. The *purpose* of an operation includes three attributes: *function*, *synonyms*, and *specialization*. *Function* describes the business functionality that the current operation offers, such as "eligibility", "registration", and "mentoring". *Synonyms* and *specialization* attributes work as they do for community categories.

The *agency* property describes the party that offers the current operation. Government operations are available through federal, state, local, or tribal agencies. They may also be provided by non-profit organizations (e.g., Teen Outreach Pregnancy) or private businesses (e.g., meals providers participating in a subsidized government program). The *eligible* property specifies the groups of citizens (e.g., children, pregnant women) that are eligible to the operation's welfare benefit. For example, WIC (Women, Infant, and Children) is a program for pregnant women, lactating mothers, and children. The *locality* property specifies whether pr1oviding the current operation requires citizens moving to the agency location (*on-site*) or agency representatives moving to citizens' location (*in-home*). An example of *in-home* operation is `mealsOnWheels` which which provides home-delivered meals to homebound elderly.

Operational Properties – Operational properties describe the *behavior* of a generic operation. The behavior of an operation refers to (i) the *relationship* between this operation and other operations and (ii) the operation's *business logic* (Figure 1). Government regulations may require going through a pre-defined process before applying for a given welfare program. This process generally involves the execution of several operations called *pre-requisite operations*. For example, senior citizens must first be registered with an Area Agency on Aging (`registerAAA`) before ordering meal from a participating restaurant (`orderMeal`) and asking for delivery (`mealsOnWheel`). Additionally, `mealsOnWheel` cannot be invoked before the invocation of `orderMeal` operation. Pre-requisite operations may belong to different communities. A pre-condition may be associated to each pre-requisite operation. For example, *mealsOnWheel* is executed only if the `orderMeal` operation has been approved. A generic operation is executable only if all its pre-requisites are executed and their pre-conditions are true.

Government regulations may also state that invoking a generic operation would initiate the invocation of other operations called *triggered operations*. For example, a pregnant women that registers for a food check program women (`registerFoodCheck`) should also register for nutritional counseling course (`registerNutritionCourse`). Triggered operations may be mandatory or optional. The invocation of a triggered operation may also depend on the results returned by the generic operation.

Another important property of generic operations is their *business logic*. This operation refers to the outcome expected after executing an operation given specific inputs. Each generic operation has one or several business logic rules. A rule takes the following form:

$$Rule_i \text{ (G-op-ID)} = \text{(ruleInput, precondition) / (ruleOutput, finalState)}$$

$Rule_i$ specifies that if the operation G-op-ID is executed and the *preCondition* over parameters in the *ruleInput* is true, then the parameters in *ruleOutput* have the values contained in *finalState.* The following is an example of rule associated with the operation `registerFoodCheck` is specifies that if a citizen has an annually income less than 22,090 dollars and if the household size is ≥ 2 then he/she is eligible for `registerFoodCheck` operation:

$$Rule_i \text{ (registerFoodCheck.G-op-ID)} =$$
$$\text{({householdSize, income},(householdSize} \geq 2 \text{ and income} < 22{,}090)) /$$
$$\text{({eligibility},{(eligibility,true))}$$

4. Joining Communities

E-government service providers can, at any time, select and register with a community of interest. Selecting communities is generally based on their category. The registration process involves defining a UUID unique identifier (*WS-ID*) and *name* for the service. Other required information includes *service operations*, *QoS properties*, and *mappings* (Figure 1).

Generic operations are used as templates by providers to identify their service operations. An e-government service may offer all or some of the generic operations defined within a community. In this case, the provider specifies the G-op-IDs of the operations available through its service. Providers may also customize generic operations to best fit their services. This is done through *customized operations*. Each customized operation (identified by a unique C-op-ID) is associated to a generic operation. We define two methods for defining customized operations:

- *Projection* – The customized operation uses a subset of the parameters defined in the corresponding generic operation. This is done by execution the function *Project(G-op-ID, list_parameters)* where G-op-ID is the identifier of the generic operation and *list_parameters* is the subset of G-op-ID's parameters that are used by the customized operation.
- *Extension* – The customized operation adds new parameters to the corresponding generic operation. This is done by execution the function *Extend(G-op-ID,{(parameter, data_type, I/O)}*. The attributes *parameter* and *data_type* are the name and data types (in XML Schema) of the new parameter respectively. *I/O* specifies whether the parameter is an input or output.

QoS (Quality of Service) properties provide qualifications for the service and its operations. We define three QoS properties: *communication*, *geographic location*, and *fees*. Since Web services may support different binding protocols (e.g., SOAP/HTTP), it is important to insure that they "understand" each other at the message format and protocol level. The *communication* property specifies the set of protocols supported by the e-government service. *Geographic location* and *fees* properties are assigned to each service operation. Some welfare programs require living within a specific geographic area. For example, senior citizens cannot benefit from programs offered by AAAs located outside their county. The geographic location of an operation contains a set of areas (zip codes) that can be served by that operation. The *fees* property gives the dollar amount required to execute the corresponding operation.

For a service to register with a community, the provider must define the mappings between operations defined in the community and those defined in the service. The following statement allows the specification of such mappings. For each service operation (defined by a name), the provider gives the corresponding G-op-ID or C-op-ID. The provider also gives a one-to-one mapping between G-op-ID's or C-op-ID's parameters and service operation's parameters.

5. Conclusion

In this paper, we presented a new approach for organizing e-government services on the envisioned Semantic Web. We combined the emerging concepts of Web services and ontologies to make e-government services machine interpretable and processable. We are currently investigating the design of techniques to enable the automatic selection and composition of e-government services using the concept of community.

Acknowledgment. This research is supported by grant 9983249-EIA from the National Science Foundation's Digital Government program.

References

[1] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, May 2001.

[2] S. A. Mclraith, T. C. Son, and H. Zeng. Semantic Web Services. *IEEE Intelligent Systems*, March 2001.

[3] B. Medjahed, A. Rezgui, A. Bouguettaya, and M. Ouzzani. Infrastructure for E-Government Web Services. *IEEE Internet Computing*, 7(1), January 2003.