Dynamic Public Service Mediation

Wout Hofman, Mark Staalduinen

Dynamic Public Service Mediation

Wout Hofman, Mark van Staalduinen

TNO, P.O. Box 5050, 2600 GB, The Netherlands
{Wout.Hofman, Mark.vanStaalduinen}@tno.nl

Abstract. This paper presents an approach to dynamic public service mediation. It is based on a conceptual model and the use of search and ranking algorithms. The conceptual model is based on Abstract State Machine theory. Requirements for dynamic service mediation were derived from a real-world case. The conceptual model and the algorithms are developed and implemented by a proof of concept for this case. This model is build on existing means for sharing public services, which would enable migration of existing applications to our proposed approach. The conceptual model is compared to other models for service mediation to illustrate differences and identify similarities.

Keywords: public services, service discovery, abstract state machines, WSMO

1. Introduction

Over the past decades, services have become important [1]. The service economy refers to the services sector, but it is also applicable to governments. Many government organizations have defined and published their services on the Internet. Interoperability of these public services is of importance to the EU (European Union) market [2]. Standards are being set, both at European [3] and national level [4] and have been developed in EU funded projects [5]. Earlier research indicated that discovery of these public services does not satisfy citizens and organizations requirements [6], and therefore it requires new strategies.

One of the strategies is 'life events' for service discovery (see for instance [7]). A complicating factor is that life events are containers, e.g. 'birth', 'marriage', and 'decease', whereas the relevant services for a specific event may be provided by different government organizations. In addition, it is impossible to capture each real-world event by a life event. Furthermore, each organization still offers its particular services without taking into account the relations with services of other organizations. Thus, it is difficult for a citizen to discover the complete set of services that matches with his requirement.

This paper investigates the issue of public service discovery and mediation. The main objective of our solution is dynamic mediation of a user his goal with public services. First, the conceptual model is introduced. Thereafter, the service discovery algorithms are discussed. These are based on natural language and search technology. Subsequently, a dynamic interaction model is introduced to support interactive mediation of requirements and services. A realistic case is used to validate this model. Based on the model and the use-case a proof of concept has been constructed. Furthermore, the proposed conceptual model is compared with existing approaches. Finally, conclusions are drawn, and recommendations for further research are made.

2. The conceptual service model

This section introduces a model for service discovery to create the public service experience. This solution is part of a larger model for enterprise interoperability [8]. First, the overall model is introduced. Thereafter, the different concepts of the model are given.

2.1. The model

Figure 1 shows the model and its concepts for public service discovery. It allows a user to formulate in natural language his goal, which is a real world event. Processing of this event leads to a dynamic interaction model that flexibly composes a selection of services that are required to achieve a user's goal.

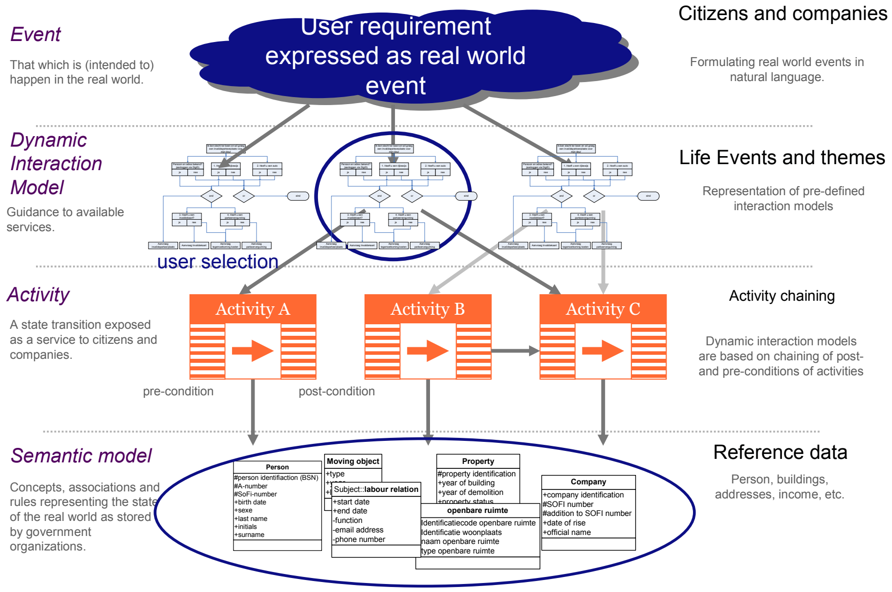

Figure 1. The overall model

'Event, 'activity' and 'reference data' are the basic concepts of our model. These concepts are presented hereafter. The 'dynamic interaction model' is part of the mediation process, and therefore it is discussed in the next section.

In general, a public service should not only specify what a government organization does, but also how this can be achieved, e.g. the interaction choreography, durations, and other process characteristics [18]. These aspects of a 'service' are not within the scope of this paper. It focuses on the concept 'activity' to represent the behavior of a service.

2.2. Event

There are many definitions of ‘event’, e.g. from different perspectives. We will use a philosophic approach: ‘an event is an operation on an object with given properties at a specific time or within a time interval’ [20]. Events can be intentional, e.g. taking a holiday, or not, e.g. a tornado. Not all real world events are of interest to government organizations. In the context of this paper, an event can be a trigger to a user to initiate an activity with one or more government authorities with a specific goal: the event has already happened in the real world and needs to be administrated by a government authority, e.g. ‘birth’ and ‘unemployment’, or a user intends to execute an event for which permission of a government authority is required, e.g. ‘building a house’ or ‘transportation of livestock to another country’. Intentional events that need registration or permission can also take place without permission in which case we call it fraud, theft or another form of crime. Each event could result in related services, e.g. ‘birth’ and ‘unemployment’ lead to financial support by a government (‘child benefit’ and ‘unemployment benefit’ respectively). Events requiring permission can be performed as soon as permission has been given. Note, that it is never sure that they also are actually executed (e.g. it is never sure that a house will also be built based on the granted permission). For this reason, government organizations have inspections that ensure that these events are executed according to the granted permission.

2.3. Activity

Like ‘event’, there are many definitions of ‘activity’, e.g. ‘the intentional behavior of and actor to achieve a certain goal’ [21]. An activity can be initiated by a user based on an event that has occurred or is to occur. In the context of this paper an activity can be performed by a government authority to handle the (intended) real world events triggered by a user. Activities are provided by government organizations and have to be matched with a user’s representation of ‘event’. Activities have the same properties as a state transition [9]:

- *Pre-condition*: (complex) predicates that must all be true before an activity can be executed.
- *Post-conditions*: the actual result of the execution of an activity. This result is defined as the state in case of an activity that is executed successfully.
- *Firing rules*: the (ordered) set of rules that are executed when the pre-condition is met and results in a post-condition. In an administration of an authority a firing rule actually changes the state of the representation of the operation.

From a service mediation viewpoint, a firing rule is not of interest. It is the objective to retrieve the complete set of services to achieve the user his goal. This set is determined by chaining activities using the pre- and post-conditions.

Pre- and post-conditions are expressed as the state space represented by the reference data. These abstract specifications are determined by means of logical expressions. These activities are specified by civil servants who do not understand the formalization of the pre- and post-conditions. An abstract specifications is also not sufficient for service mediation, which requires user interaction (e.g. ‘Do you have a parking permit?’ in case parking is only allowed with a permit in a particular area).

An important issue is activity chaining to discover related services during mediation. Chaining means the discovery of activities with a post-condition that met unsatisfied pre-conditions. This paper shows a practical case to obtain a parking place for a physically disabled person. This example requires a parking permit, which is another activity that results in granting this permission. That latter activity has ‘granting parking permission’ as a post-condition.

A method should be developed to establish links based on a formal specification of pre- and post-conditions that could be defined by civil servants. This method is left for further research, while this paper shows the result that is achieved if this method would be defined. The model is tested using the following specification of an activity (Figure 2 shows the representation of ‘activity’):

- An activity has a name, a description, one or more pre-conditions, and a post-condition.
- A pre-condition, which is repeatable and represents a predicate, has a description, a question, an input state, and potential a validation service (which is not required from a conceptual point of view). The input state is used for linking and the validation service calls a web service in case the input state is not specified.
- A post-condition has a description and a resulting state that could be linked to an input state.

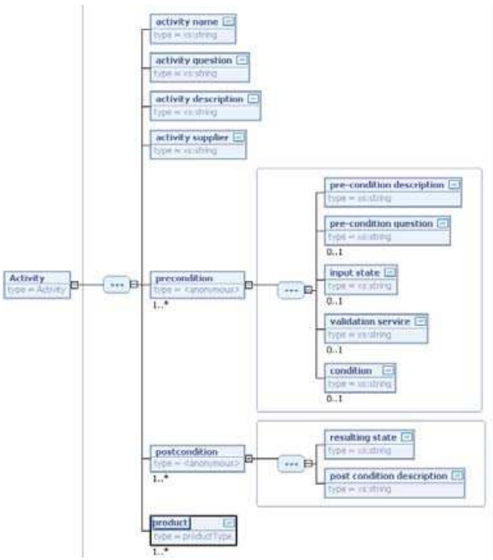

Figure 2. The structured representation of an ‘activity’

The representation of an ‘activity’ (Figure 2) is an extension of the XML structure of the shared public services [4]. The XML element ‘productType’ is defined in [4]. The figure also shows that an activity can identify more than one product, which reflects the situation that (Dutch) governments have defined products for individual events [8]. The figure shows the XML elements ‘input state’ and ‘validation service’, which are obsolete in case formal expressions are used for modelling pre- and post-conditions. Each question has to be answered by a user during interaction to select a proper (set of) activity. However, the answers to these questions have to be validated by the pre-condition of an activity based on the actual data provided by a user after selecting that activity. Data provision and validation is outside the scope of this paper.

2.4. Reference data

This section presents a brief outline of reference data as contained by government organizations. It actually contains the administrative description of real world state as perceived by government organizations. The complete set with reference data is complex and large. Examples of reference data are ‘natural persons’, ‘buildings’, ‘organizations’ and associations between these concepts like ‘person’ to ‘address’, and ‘address’ to ‘building’. Figure 3 shows a draft of the reference data modelled using a UML (Universal Modelling Language, [19]) class diagram. It shows the complexity of the state determined by government organizations.

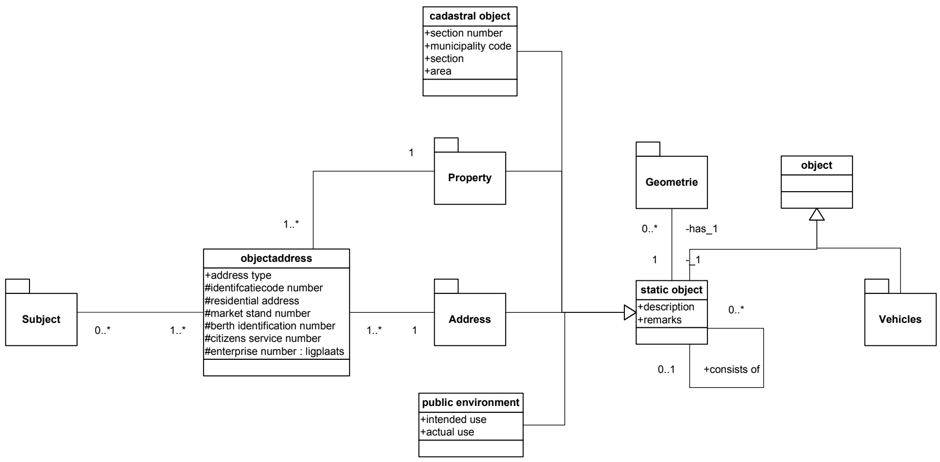

Figure 3. Overview of reference data

3. Service Mediation

This section presents functionalities to determine the service mediation process. It is determined by a self-learning search algorithm and the construction of dynamic interaction model based on activity chaining.

3.1. Self-learning search technology

In general, a user is perfectly able to formulate his/hers particular goal using natural language. The objective is to match this goal to the correct service. This process is called mediation [10]. The match can be performed in a variety of ways. The field of text analytics provides solutions like language model-based text retrieval (e.g. [13]), syntactic analysis (e.g. [12]), and information ranking approaches that learn how to rank candidate answers to questions (e.g. [11]).

Our solution to query-answer matching consists of a mixture of language modelling and ranking techniques. First, a document index is constructed for a set of activities using commonly available index engines like Lucene ([14]) or Lemur (http://www.lemurproject.org/). Using such an engine, the formulated query is matched with candidate activities. If the description of a query is too short, character n-gram indexing can be of use to generate sufficient textual material for matching. Under this type of representation, words in both user events and indexed documents are split into overlapping sequences of characters, e.g. {cha, har,ara,rac,act,cte,ter}. The benefits of character n-grams in retrieval are well-known (e.g. [15]).

The candidate activities are subsequently presented to a user. If a user selects one of the answers, then it is assumed that this is the correct activity for his event. This selection is used to derive an improved ranking (see e.g. [16]). A Ranking Support Vector Machine (R-SVM) is trained using this feedback, which is used for re-ranking the results of the retrieval engine in subsequent passes. The R-SVM can be supplied with specific kernels geared towards bag-of-word representations, such as geodesic kernels (e.g. [17]). Using this approach, the performance of service mediation improves using the user feedback during its usage (the latter is tested with limited interaction tests).

3.2. Dynamic interaction model for service mediation

A dynamic interaction model supports user interaction based on selections made by a user, instead of having predefined interaction models that serve as a navigation structure, e.g. life event based navigation. Both a dynamic interaction model and its predefined version are decision trees. A dynamic interaction model meets all possible user requirements supported by government services, which is not feasible by a static, predefined model. Dynamic interaction models are constructed by chaining activities based on an activity selected by a user with the self-learning search techniques.

There are two important aspects for the construction of dynamic interaction models. First, the application of AND and OR operators as pre-conditions:

- AND indicates that two or more pre-conditions can be satisfied by two or more post conditions of other activities, e.g. a ‘drivers licence’ and a ‘parking permit’ is required.
- OR indicates that if one or more pre-conditions are not satisfied, then the activity can not be executed. The OR construction is also used to select one of the options of a set given using a pre-condition.

Second, identical pre-conditions of relevant activities should appear once in the interaction model. By backward chaining from pre-conditions of an activity selected by a user, a complete list of activities is constructed and identical pre-conditions should be mentioned once. We will illustrate this mechanism using in our practical case.

4. Model validation with a practical case

The case, which is a view of reality for the sake of our proof of concept, considers a physically disabled citizen who wants a reserved parking place near his or her door. In that particular area, a parking permit is required. Furthermore, a special card for parking on parking places for handicapped persons is required. In this particular case, parking permits are issued by the parking authority, handicap parking cards by a municipal social security authority, and an handicap parking place by a third department. Thus, there are at least three authorities involved. The legal aspects for arriving at a collaborated decision by these authorities are not discussed even as the orchestration of selected services. This paper focuses on service discovery and mediation. First, this section describes chaining of the relevant activities. Second, it presents the dynamic interaction model. Finally, some screenshots of the proof of concept are shown.

4.1. Activities

The case considers chaining of the following activities:

- a parking place reserved for handicapped drivers,
- a handicap parking card,
- a parking permit.

4.1.1. Backward chaining of relevant activities

Activity chaining includes the complete set of activities to achieve a user his goal. The complete chain of activities is constructed dynamically based on linked pre- and post-conditions. Since, pre-conditions are linked to post-conditions, it could be considered as a step backwards. For this reason, this chaining approach is called backward chaining. Figure 4 visualizes this process to obtain a parking place for a physically disabled citizen.

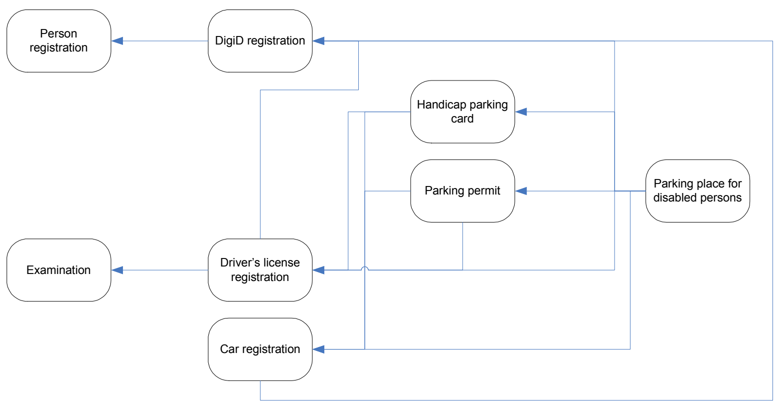

Figure 4. Backward activity chaining for user requirement 'handicapped parking place'

This figure shows that authentication by DigiD is used for a number of activities. Second, it shows that having a car and a driver's licence are required for a parking permit. These are obtained using different activities. Furthermore, it shows additional activities like 'driver's license registration', 'examination', 'person registration' and 'car registration'. Thus, a network of activities is obtained to describe this 'simple' case. As 'car registration' and 'examination' require an activity performed outside the scope of the government, the pre-condition of these two activities cannot be satisfied during user interaction. Thus, these activities are not considered for this case. As stated before, the questions are used for user interaction and the answers will be validated by actual data offered by a user to execute an activity.

4.1.2. Parking place for disabled persons

A parking place for disabled persons can only be used by persons with a handicap parking card. To be able to request such a parking place, several pre-conditions need to be met. These pre-conditions are given by:

1. The person needs to have valid identification means of the set {drivers license, passport, digital authentication mechanism}. In our proof of concept, we use the Dutch digital authentication mechanism: DigiD. The address of the person should be equal to the one retrieved by DigiD.
2. There should be sufficient parking space in the neighborhood to reserve a handicapped parking place. The related question is: 'Is there sufficient parking space near the place you live?'
3. The person should have a valid driver's license. The question is: 'Do you have a valid driver's license?'
4. The person should have a car. The question is: 'Do you have a registered car?'
5. The person should have a handicap parking card. The question is: 'Do you have a handicap parking card?'
6. The person should have a parking permit. The question is: 'Do you have a parking permit?'

An interesting extension on this model is the following. If web services are available that connect to public registers, then a number of questions in the interaction model could be answered automatically. For example, a check could be performed, whether the authenticated person has a driver's license.

4.1.3. Handicap parking card

A handicap parking card is required to park a vehicle on a parking place reserved for disabled persons in the Netherlands. It is recognized as a valid document by all parking authorities in the Netherlands. To request this card, the following pre-conditions should be satisfied:

1. The person should have a DigiD authentication.
2. The person should have a valid driver's license. The question is: 'Do you have a valid driver's license?'
3. The person should have a car. The question is: 'Do you have a registered car?'

4.1.4. Parking permit

A parking permit issued by the relevant authority for a particular area. It is only used to park a vehicle in that particular area and cannot be used in other areas. To request a parking permit, the following pre-conditions should be satisfied:

1. The person should have a DigiD authentication.
2. The persons should live in the municipality for which a parking permit is requested. A validation for this address could be identified using DigiD.
3. The person should have a valid driver's license. The question is: 'Do you have a valid driver's license?'
4. The person should have a car. The question is: 'Do you have a registered car?'

4.2. Constructing the dynamic interaction model

A dynamic interaction model is based on backward chaining of activities. First of all, a user should select the activity that fulfills his goal. A dynamic interaction model is constructed using this choice as figure 5 presents. The interaction model shows that although the three activities described before have identical questions, these questions appear once. Furthermore, the questions that can not be satisfied by another activity to achieve a user his goal appear first.

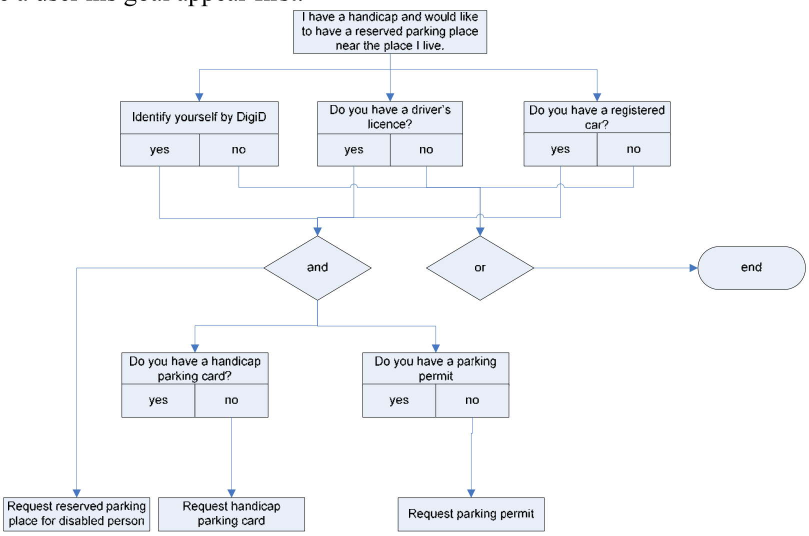

Figure 5. The dynamic interaction model that is constructed after selection of 'parking place for disabled persons'

4.3. Proof-of-concept

The Proof-of-Concept visualizes service discovery and mediation using the proposed techniques, e.g. the XML structure of activities and the shared product catalogue. It starts with the formulation of an event using natural language. Thereafter, a user receives a list with activities that could possibly satisfy the goal of a user with respect to the event. By improving the ranking, it is the aim to have the most suiting activity for a particular goal at rank one. It is possible to select multiple activities. The state of the user is determined by exploiting the scope of the selected activities based on a dynamic interaction model of selected activities. Finally, a set of one or more services is presented that should be executed to achieve the user his goal.

5. Discussion on alternative methodologies to model public services

Our conceptual model is based on the theory of Abstract Service Machines (ASM, [9]). A similar approach is taken by Web Service Modeling Ontology (WSMO, [10]), that is also based on ASM. WSMO defines 'capability', which is identical to our concept of 'activity'. WSMO defines service discovery and mediation as matching a customer requirement expressed as a goal with one or more capabilities offered by a service provider. However, both 'capability' and 'goal' should be expressed formally with identical pre- and post-conditions. It cannot serve as the basis for dynamic service mediation. Furthermore, capability chaining is not specified in WSMO [10].

Public services are formally specified in for instance WSMO-PA (WSMO for Public Administration, [5]). WSMO-PA specifies data requirements for public services, e.g. the requirement of identification with a valid document, like passport or driver's license, or another digital means. We have shown that these requirements depend on pre-conditions of activities, such that they could be derived from these pre-conditions. This topic is left for further research.

6. Conclusions and further research

This paper presented a conceptual model for public service discovery and mediation. The case, presented in section 4, showed that our conceptual model can be practically deployed. This is further illustrated using our proof of concept. Our conceptual model and software that supports this model has several advantages:

1. It is not required to construct decision trees manually to achieve a user his goal using public services. These are generated dynamically.
2. Our ranking algorithm learns from users with similar goals to show the best matching activities first. Learning from user feedback improves the system.
3. The conceptual model is build upon existing standards used by Dutch government organizations, which makes migration easier.

We are convinced that our approach leads to systems that are adaptive to changes in laws and regulations with lower costs for specification and maintenance, while it is provides a simple and accessible interface to end-users. However, the approach still needs further research in the following areas:

- A business case needs to be constructed to show the advantages of the proposed approach compared with existing approaches. Such a business case should be based on improved user experience. This improvement needs to be tested for more cases, also to validate and possibly refine the model.
- Research questions regarding the conceptual model are:
  - Dynamic backward chaining of activities on a formal description of pre- and post-conditions to support scalability.

- The ranking algorithm may be extended by creating so-called personae: users that have similar traits e.g. shared customer.

- Operational questions are:
  - Tooling that can be used by civil servants with limited knowledge of formal specification technologies like ASM.
  - Efficiency improvement by derivation of semantic models out of existing data models.
  - A methodological aspect is integration of distributed development of these semantic models by various government organizations responsible for particular laws and regulations.
  - We have only given a partly model of reference data. It needs to be refined and extended to include all relevant data maintained by government organizations, e.g. educational data and patient data.

Acknowledgments

We would like to thank our colleagues Stephan Raaijmakers, Joost de Wit, and Corne Versloot for their work on the realization of the proof of concept. Furthermore, we would like to thank Harrie van Houtum (IND – Dutch Immigration Service) for introducing the concept 'dynamic interaction model', ICTU for providing the product feeds, and GovUnited for offering their platform to perform tests.

References

1. J. Heineke, M. Davis, The emergence of service operations management as an academic discipline, Journal of Operations Management 25 (2007) 364–374.
2. European Interoperability Framework for Public Services (version 2.0), European Commission – IADBC, 2009.
3. Discovery of and Access to eGovernment Resources – part 1 to 5, CWA 15971-1, CEN, 2009.
4. L. van der Wijst, Sharing Government Services, version 2.11, May 2009 (in Dutch).
5. X. Wang, T. Vitvar, V. Peristeras, A. Mocan, S. Goudos and K.Tarabanis, WSMO-PA: Formal specification of Public Administration Service model on Semantic Web Service Ontology, Proceedings of the 40th Annual Hawaii International Conference on System Sciences, 2007.
6. Th. van der Geest, R. Klaassen, J. Karreman, Public information discovery – search strategies of citizens, University of Twente, 2005 (in Dutch).
7. OneStopGov home page, http://www.onestopgov-project.org.
8. W.J. Hofman, Concepts of enterprise interoperability, ICSOFT 2010 (for review).
9. E. Börger: "High Level System Design and Analysis Using Abstract State Machines", Proceedings of the International Workshop on Current Trends in Applied Formal Method: Applied Formal Methods, p.1-43, October 07-09, 1998
10. D. Fensel, M. Kerrigan, M. Zaremba (eds.), Implementing Semantic Web Services – the SESA framework, Springer-Verlag, 2008.
11. S. Verberne, H. Van Halteren, D. Theijssen, S. Raaijmakers, L. Boves, Evaluating Machine Learning Techniques for Re-ranking QA Data. Proceedings Workshop Learning to Rank for Information Retrieval. SIGIR 2009, Boston, USA.
12. M. Mittendorfer, W. Winiwarter, Exploiting syntactic analysis of queries for information retrieval. In Data & Knowledge Engineering, Volume 42 , Issue 3 (September 2002), pp. 315 – 325, 2002.

13. Djoerd Hiemstra. Using language models for information retrieval. PhD Thesis, University of Twente, 2001.
14. O. Gospodnetic, E. Hatcher, M. McCandless, Lucene in Action (2nd ed.). Manning Publications. pp. 475, June 28, 2009.
15. P. McNamee, Character N-Gram Tokenization for European Language Text Retrieval. In Information Retrieval, Volume 7 , Issue 1-2 (January-April 2004), pp. 73 – 97.
16. T. Joachims, Optimizing Search Engines Using Clickthrough Data, Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD), ACM, 2002.
17. S. Raaijmakers and W. Kraaij, Polarity Classification of Blog TREC 2008 Data with a Geodesic Kernel. Proceedings TREC 2008, Gaithersburg, USA.
18. OWL-S, Semantic Markup for Web Services, W3C member submission, 2004.
19. M. Fowler, UML Distilled, Third Edition, A brief guide to the standard object modeling language, Addison-Wesley, 2004.
20. Kim, Jaegwan, Supervenience and Mind: Selected Philosophical Essays, New York: Cambridge University Press, 1993.
21. M. E. Bratman, Faces of Intention. Cambridge University Press, 1999.