Supporting QoS Negotiation with Feature Modeling

Marcelo Fantinato[1], Itana Maria de S. Gimenes[2], and Maria Beatriz F. de Toledo[1]

[1] Institute of Computing, University of Campinas, Brazil
[2] Department of Computer Science, University of Maringá, Brazil
mfantina@ic.unicamp.br, itana@din.uem.br, beatriz@ic.unicamp.br

Abstract. Feature modeling is a technique that has been widely used for capturing and managing commonalities and variabilities of product families in the context of software product line. This paper presents a feature-based approach to be applied in QoS negotiation during the establishment of a Web services e-contract. Its motivation is that the e-negotiation process, aiming at defining attributes and levels for QoS – in a particular business domain, usually involves a set of well-defined common and variation points.

Keywords: e-contracts; Web services; QoS; information reuse; features.

1 Introduction

E-contracts are used to describe details of the supply and the consumption of e-services within a business process [1], [2]. An important part of e-contracts are the levels for QoS attributes agreed between the involved parties [3], [4]. The current complexity involved in e-contract establishment and QoS negotiation include aspects such as: great amount of necessary information; increasing number of parameters to be considered; potential long-duration of e-negotiations; and involvement of different profiles (business and development teams) of distinct organizations.

In order to overcome these drawbacks, it is necessary to tackle information structuring and reuse, which is normally tried using e-contract templates [2], [5]–[10]. Templates are commonly treated as simple documents that have empty fields to be fulfilled with some value, usually from a pre-defined list. In general, existing template approaches do not offer suitable mechanisms to manage common and variable elements in similar e-contracts. Therefore, they provide a limited potential for information reuse between similar e-contracts.

In this paper, a new approach to reduce the complexity of QoS negotiation, inside a process to establish e-contracts for Web services, is proposed. It is based on the feature modeling technique [11], which was developed inside the software product line (PL) [12]-[13] context. Its major contribution is to offer a systematic and efficient way for information structure and reuse, thus optimizing the QoS negotiation process. The approach provides a mean to represent QoS attributes and levels, besides other e-contract elements, in feature models that can be transformed into e-contract templates. This paper extends two previous works on e-contract establishment [14], [15] with: a new e-contract metamodel and emphasis on QoS negotiation.

B. Krämer, K.-J. Lin, and P. Narasimhan (Eds.): ICSOC 2007, LNCS 4749, pp. 429–434, 2007.

In brief, the proposed approach consists of a set of five stages. Feature modeling allows the representation of abstract e-services and possible levels for QoS attributes. The activities of the e-contract establishment process, including QoS negotiation, will be oriented by the feature model and feature model configurations. The generic contracted e-services will be mapped to the Web services implementing them, in a one-to-one relationship. These Web services will be referred to in the resulting e-contract, for which specific levels for some QoS attributes can be defined.

This paper covers the following information: e-contract and feature modeling background concepts; proposed approach; related work; conclusions and references.

2 Electronic Contracts

A contract is an agreement between two or more parties interested in creating mutual relationships on business or legal obligations. It defines an activity set to be carried out by each party, which must satisfy a set of terms and conditions – known as contract clauses. An e-contract is an electronic document used to represent an agreement between organization partners carrying out business using the Internet, in which the negotiated services are e-services, currently implemented as Web services.

An e-contract consists of [2]: *parties* – representing the organizations involved in a business process; *activities* – representing e-services to be executed throughout the e-contract enactment; and *contractual clauses* – describing constraints to be satisfied throughout the e-contract enactment. Contractual clauses can represent three different types of constraints [1]: *obligations* – what parties should do; *permissions* – what parties are allowed to do; and *Prohibitions* – what parties should not do.

Obligations include QoS clauses associated with e-services which define attributes related to non-functional properties. They affect the definition and execution of an e-service, regarding to, for example: availability, integrity, reliability, performance, security and reply time [3], [4], [10]. For each QoS attribute, a value must be defined to be used as a tolerable level (e.g. a minimum, a maximum or an exact value).

3 Feature Modeling

Feature modeling is a type of computing ontology that has been applied for capturing and managing commonalities and variabilities in software PL [12]. It was originally proposed in the domain engineering context, as part of the Feature-Oriented Domain Analysis (FODA) [11], and has been applied in a range of domains including telecom systems, template libraries, networks protocols and embedded systems.

In general, a feature model is a description of the relevant characteristics of some entity of interest. A feature can be defined as a system property that is relevant to some stakeholder and is used to capture commonalities or discriminate systems in a family. They may denote any functional or non-functional characteristic at the requirement, architectural, component, platform, or any other level. According to the original FODA method, features can be mandatory, optional or alternative.

A feature model describes the configuration space of a system family. A member of the family can be specified by selecting the desired features from the feature model within the variability constraints defined by the model. This process is called feature

configuration. The rules to elaborating features models or diagrams can be specified by feature metamodels. The one being used here is proposed by Czarnecki at. al. [12].

Features can be organized in a feature diagram, which is a tree-like structure where each node represents a feature and each feature may be described by a set of sub-features represented as children nodes. Feature diagrams offer a simple and intuitive notation to represent variation points without delving into implementation details. The diagrams are especially useful to drive the feature configuration process.

4 QoS Negotiation and e-Contracts Establishment

This section presents the process to negotiate QoS attributes, inside a global process to establish WS-contracts (Web services e-contract) between two organizations. The global process consists of five stages, adapted from the FORM method [13]:

1. Feature model elaboration: two feature models are elaborated to represent the e-services and QoS attributes from each organization;
2. WS-contract template creation: having two feature model as the basis, a WS-contract template is created;
3. Web services development and publication: Web services that implement the e-services must be developed and published – which is out of this paper scope;
4. Feature model configuration: the two feature models are then configured to represent the exact e-services and QoS levels for a particular business process;
5. WS-contract establishment: a WS-contract is produced by refining the WS-contract template, based on the previously defined pair of feature model configurations.

Fig. 1 represents, as a class diagram, the artifacts produced throughout the stages above. The *feature model* is the basic artifact from which a unique *WS-contract template* is generated and one or more *feature model configurations* are derived. For each *feature model configuration*, a particular *WS-contract* is established. The *WS-contracts* are established based on the same *WS-contract template*. Each `Web` *service* implementing an abstract e-service of the feature model is referred to by the *WS-contract template*. Only the `Web` *services* implementing e-services of the feature model configuration are referred to by the corresponding *WS-contract*.

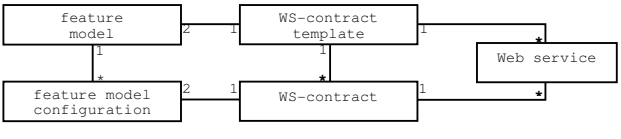

Fig. 1. Artifacts relationship

The QoS attributes, associated to the e-services, are treated as common points and variabilities in feature models. They can be specified by mandatory, alternative and optional features. A feature metamodel [12] was chosen to drive the modeling of this information as features. And a specific feature diagram structure for e-services and QoS attributes representation is being developed, since the inherent flexibility of the metamodel would allow the definition QoS attributes in too many ways.

A WS-contract metamodel was defined to represent rules to create both WS-contracts and templates. The metamodel was created by unifying the main concepts related to: (i) Web services – described by WSDL language; (ii) business processes involving Web services – described by WS-BPEL language; and, (iii) QoS of Web services – described by WS-Agreement language.

The creation of the WS-contract template is carried out in two steps: at first, the WSDL and WS-Agreement sections are created directly from the e-services feature models. For the first step, there is a mapping from elements of the feature metamodel to elements of the WS-contract metamodel. In the second step, the WS-BPEL section is created from WSDL definitions and further information is defined during this stage.

To enable contract instantiation, the WS-contract template is instrumented with a set of annotations linking the contract elements to the respective features used as basis for its creation. During contract instantiation, the feature model configurations are used by a parser in a removal process. This process is driven by the mandatory features and the optional/alternative features that have been selected or not.

A support tool is being developed to aid the proposed process. The tool, named FeMoWS-Contract (Feature Modeling WS-Contract establishment tool), includes a series of software components related to different stages of the approach. One of the component part of the tool is FeaturePlugin tool [16], used for specification of feature models and support theirs configurations.

An approach evaluation was undertaken on a pseudo-scenario to evaluate the approach proposed here. It is concerned with the integration between two business and operation support systems, in the telecom context: customer relationship management (CRM) and dunning systems. The success on its has made possible to demonstrate the feasibility of the approach.

As a result from the approach evaluation, some developed artifacts are presented. Fig. 2 presents an example of a feature model configuration for a system providing information for another one. The right side of the figure models some e-services whereas the left side of it models some levels for a QoS attribute. In both cases, a set of optional features is already selected. Fig. 3 presents a part of the WS-contract template related to this features model. Since only the level “15” was chosen during configuration, all the other options will be removed from the contract model to instantiate the resulting WS-contract – through a annotation removal process.

5 Related Work

In relation to e-contract establishment – in a general way, there are several projects involved in this research field. However, most of them use only metamodels as a basic and limited way to achieve information reuse. In some few cases, they also use e-contract templates as a more efficient way to achieve information reuse. Examples of such projects are [2], [5]–[9]. There are also some projects that work directly with QoS attributes, including [3], [4], [10].

Some works focus on the negotiation phase before specifying the business process, but they are commonly concerned with the process to be followed and the tools to be used during the negotiation between the parties. Some projects related to e-negotiation are presented in [17]–[20]. In these and other similar approaches, there is little emphasis in information reuse compared to the approach proposed by this work.

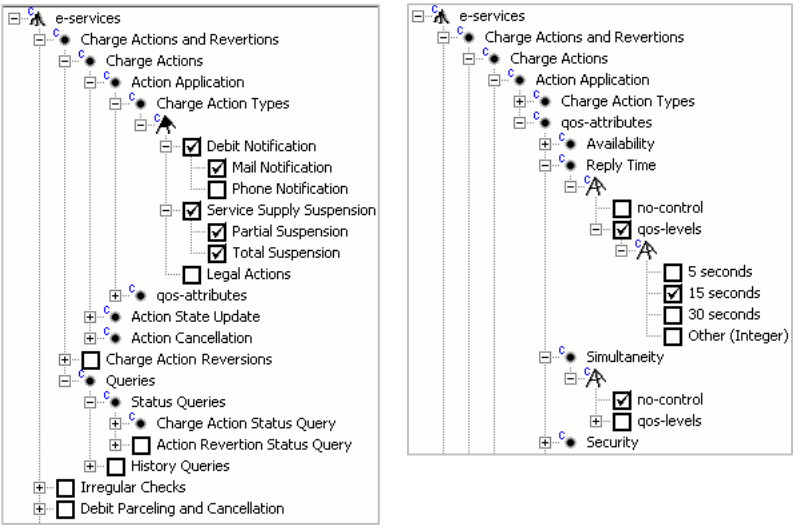

Fig. 2. Example of feature model configuration

```
<wsag:GuaranteeTerm Obligated="ServiceProvider">
  <wsag:ServiceScope ServiceName="applyChargeAction">
  </wsag:ServiceScope>
  <wsag:QualifyingCondition>…</wsag:QualifyingCondition>
  <wsag:ServiceLevelObjective>
    replayTimeSecond IS_LESS_INCLUSIVE
                None  <!-- f:Reply_Time_No_Control_ID -->
                5     <!-- f:Reply_Time_level_5_ID -->
                15    <!-- f:Reply_Time_level_15_ID -->
                30    <!-- f:Reply_Time_level_30_ID -->
                Other <!-- f:Reply_Time_level_Other_ID -->
  </wsag:ServiceLevelObjective>
  <wsag:BusinessValueList>…</wsag:BusinessValueList>
</wsag:GuaranteeTerm>
```

Fig. 3. Example of WS-contract template

6 Conclusions and Future Work

In this paper, a new approach to support QoS negotiation, as a step for establishing e-contracts for Web services, is proposed. Its main contribution is allowing a better management of common and variable points found in similar WS-contracts, including the QoS attributes and levels for different e-services; and information structure and reuse in a systematic way. Such improvement is achieved by the use of e-contract templates associated with feature models representing e-services and QoS attributes.

Future work includes: (i) finishing the development of a prototype tool to automate the establishment of WS-contract templates and resulting WS-contracts; (ii) searching for new ways to analyze the proposed approach effectiveness and compare it to other approaches to establish WS-contracts and QoS negotiation; and (iii) evaluating the approach extension for QoS negotiation between more than two parties.

References

[1] Marjanovic, O., Milosevic, Z.: Towards Formal Modeling of e-Contracts. In: Proc. EDOC, Seattle, pp. 59–68. IEEE Computer Society, Los Alamitos (2001)
[2] Hoffner, Y., Field, S., Grefen, P., Ludwig, H.: Contract-Driven Creation and Operation of Virtual Enterprises. Computer Networks 37, 111–136 (2001)
[3] Sahai, A., Machiraju, V., Sayal, M., van Moorsel, A., Casati, F.: Automated SLA Monitoring for Web Services. In: Proc. DSON, Montreal, pp. 28–41 (2002)
[4] Menasce, D.A.: QoS Issues in Web Services. IEEE Internet Computing 6(6), 72–75 (2002)
[5] Chiu, D.K.W., Cheung, S.-C., Till, S.: A Three Layer Architecture for E-Contract Enforcement in an E-Service Environment. In: Proc. HICSS, Big Island, p. 74 (2003)
[6] Rouached, M., Perring, O., Godart, C.: A Contract Layered Architecture for Regulating Cross-Organisational Business Processes. In: van der Aalst, W.M.P., Benatallah, B., Casati, F., Curbera, F. (eds.) BPM 2005. LNCS, vol. 3649, pp. 410–415. Springer, Heidelberg (2005)
[7] Krishna, P.R., Karlapalem, K., Dani, A.R.: From Contract to E-Contracts: Modeling and Enactment. Information Technology and Management 6(4), 363–387 (2005)
[8] Berry, A., Milosevic, Z.: Extending Choreography with Business Contract Constraints. IJCIS journal 14(2/3), 131–179 (2005)
[9] Hoffner, Y., Field, S.: Transforming Agreements into Contracts. IJCIS journal 14(2/3), 217–244 (2005)
[10] Keller, A., Ludwig, H.: The WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services. JNSM journal 11(1), 57–81 (2003)
[11] Kang, K., Cohen, S., Hess, J., Novak, W., Peterson, A.: Feature-Oriented Domain Analysis (FODA) Feasibility Study, Tech. Report CMU/SEI-90-TR-021, SEI/CMU (1990)
[12] Czarnecki, K., et al.: Staged Configuration through Specialization and Multi-Level Configuration of Feature Models. Software Proc.: Improv. and Prac. 10(2), 143–169 (2005)
[13] Kang, K.C., et al.: FORM: A Feature-Oriented Reuse Method with Domain-Specific Reference Architectures. Annals of Software Engineering 5, 143–168 (1998)
[14] Fantinato, M., de Toledo, M.B.F., Gimenes, I.M.S.: A Feature-based Approach to Electronic Contracts. In: Proc. IEEE CEC EEE, San Francisco, pp. 34–41 (2006)
[15] Fantinato, M., Gimenes, I.M.S., de Toledo, M.B.F.: Web Services E-contract Establishment Using Features. In: Dustdar, S., Fiadeiro, J.L., Sheth, A. (eds.) BPM 2006. LNCS, vol. 4102, pp. 290–305. Springer, Heidelberg (2006)
[16] Antkiewicz, M., Czarnecki, K.: FeaturePlugin: Feature Modeling Plug-in for Eclipse. In: Proc. eTX workshop, Vancouver, pp. 67–72. ACM Press, New York (2004)
[17] Streitberger, W.: Framework for the Negotiation of Electronic Contracts in E-Business on Demand. In: Proc. IEEE CEC, Munich, pp. 370–373. IEEE Computer Society, Los Alamitos (2005)
[18] Rinderle, S., Benyoucef, M.: Towards the Automation of E-Negotiation Processes Based on Web Services - A Modeling Approach. In: Ngu, A.H.H., Kitsuregawa, M., Neuhold, E.J., Chung, J.-Y., Sheng, Q.Z. (eds.) WISE 2005. LNCS, vol. 3806, pp. 443–453. Springer, Heidelberg (2005)
[19] Jertila, A., Schoop, M.: Electronic Contracts in Negotiation Support Systems: Challenges, Design and Implementation. In: Proc. IEEE CEC, Munich, pp. 396–399 (2005)
[20] Kaminski, H., Perry, M.: SLA Automated Negotiation Manager for Computing Services. In: Proc. IEEE CEC EEE, San Francisco, pp. 47–54 (2006)
[21] Andrieux, A., et al.: Web Services Agreement Specification (WS-Agreement), http://www. ogf.org/Public_Comment_Docs/Documents/Oct-2006/WS-AgreementSpecification