Onspect: Ontology Based Aspects

Parisa Rashidi
Washington State University
Pullman, WA
US, 99164
001-509-335-1786
prashidi@wsu.edu

Roger T. Alexander
Washington State University
Pullman, WA
US, 99164
001- 509-335-0922
rta@eecs.wsu.edu

ABSTRACT
In software engineering community, semantic interoperability usually has been ignored despite its significant importance. To achieve semantic level interoperability, ontology as a powerful means of expressing and sharing knowledge can be used to add meaningful standard semantics to syntactic annotations. In this paper we describe semantic pointcuts based on ontology modeling. Current AOP models, like many other programming models, primarily rely on a syntactic representation and mostly ignore pointcut expression at semantic level. We present a pointcut modeling approach based on semantics instead of underlying program's syntax, by using ontology modeling to conceptually modularize crosscutting concerns.

Categories and Subject Descriptors
D.3.3 [**Programming Languages**]: Language Constructs and Features – *Classes and object;* F.3.2 [**Logics and Meanings of Programs**]: Semantics of Programming Languages – *Denotational semantics.*

General Terms
Algorithms, Design, Languages.

Keywords: Semantic Pointcuts, Ontology, Onspect.

1. Onspect
In AOP community, like most other software engineering communities, less attention has been paid to semantic interoperability and semantic pointcuts, and most current mainstream AOP techniques separate crosscutting concerns based on mere syntax. Lack of semantics in AOP has led to problems such as fragile pointcuts, due to tight dependence of aspects on the syntax [1][4]. Various solutions have been proposed by different researchers to define semantic pointcuts, such as [5], [4][4], [2]. Though most of the above works address the fragile pointcut problem through use of semantic elements, neither focuses on a standard semantic modeling tool such as ontology to provide semantic interoperability and unambiguous sharing of pointcut semantics. In our work, we present a new approach toward semantic aspect modeling based on ontology, called Onspect. Ontology as an explicit specification of a conceptualization can be used as a powerful tool for adding semantics to syntactic forms, and for sharing knowledge unambiguously among different implementations and organizations [1]. Using ontology for modeling aspects allows for conceptual modularization of crosscutting concerns among heterogeneous nodes, thus reducing problem of fragile aspects.

To represent programming domain ontology, we introduce a formal model based on concepts, attributes, relationship and constraints to model basic ontology elements of a program. The program itself is modeled as an agent; its set of methods and functionalities are modeled as behaviors; and objects and their roles in the program are modeled as subjects and roles. We then map our formal template into a simple and easy-to-use set of Java annotations to annotate those semantic units. Java annotations are automatically converted into OWL constructs [6], which is a standard ontology modeling language providing a set of necessary reasoning and querying operators. The semantic pointcuts are then defined using a set of semantic quantifiers that refer to the Java annotation elements as semantic concerns. To achieve semantic interoperability between heterogeneous remote nodes, we use JAsCo's hook and connector style for declaring Onspects [3].

In summary, current approach provides a model for defining semantic pointcuts in a heterogeneous environment based on ontology to reduce dependability of crosscutting concerns on mere syntax. However as a first approach for modeling semantic pointcuts based on ontology, current model has its own limitation and shortcoming. In our future works, we plan to extend the ontology model to provide further semantic information and also to develop standard template contents for the ontology model.

FOAL 2008, April 1, 2008, Brussels, Belgium. Copyright 2008 ACM
ISBN 978-1-60558-110-1/08/0004 ... $5.00

2. REFERENCES
[1] T.R. Gruber. Towards Principles for the Design of Ontologies used for Knowledge Sharing. Journal of Human Computer Studies, 1993, pages 907--928.

[2] K. Gybels and J. Brichau. Arranging language features for more robust pattern-based crosscuts. In Proceedings of AOSD 2003, 2003.

[3] JAsCo: An aspect-oriented approach tailored for component based software development. In AOSD Proceedings, ACM Press, 2003, pages 21–-29.

[4] Andy Kellens, Kim Mens, Johan Brichau, Kris Gybels. Managing the Evolution of Aspect-Oriented Software with Model-Based Pointcuts. ECOOP 2006: 501-525.

[5] Klaus Ostermann, Mira Mezini, Christoph Bockisch. Expressive Pointcuts for Increased Modularity. ECOOP 2005:214-240.

[6] OWL Standard, http://www.w3.org/TR/owl-ref/.