
Milena Guessi, Lucas Bueno Ruas de Oliveira, and Elisa Yumi Nakagawa

Dept. of Computer Systems
University of São Paulo - USP
PO Box 668, 13560-970, São Carlos, SP, Brazil
milena@grad.icmc.usp.br, {buenolro, elisa}@icmc.usp.br

Abstract. Aspect-Oriented Programming (AOP) has arisen, intending to contribute to the development of reusable, maintainable, and evolvable software systems. Aspects have been also explored in the early life cycle phases, aiming at contributing to more adequate development of aspect-oriented software systems. In this perspective, a diversity of extensions of the UML has been proposed to represent these systems, considering its relevance and dissemination as language to model software systems. However, there is a lack of a complete panorama that identify all these extensions. Thus, this paper intends to present an overview about which are possibly all extensions of the UML to represent aspect-oriented software systems. For this, we have used systematic mapping, a technique proposed by Evidence-Based Software Engineering (EBSE). Achieved results show a diversity of work; however, there is not a consensus about which are the more adequate or more used5 extensions. Based on this overview, interesting and important perspectives for future research can also be found.

1 Introduction

Aspect-Oriented Programming (AOP) has arisen as an approach that supports a better SoC (Separation of Concerns) and more adequately reflects the way developers think about the system [1]. Essentially, AOP introduces a unit of modular implementation – the aspect – which has been typically used to encapsulate crosscutting concerns in software systems (i.e., concerns that are spread across or tangled with other concerns). Modularity, maintainability, and facility to write software can be achieved with AOP [2]. Aspects have also been explored in the early life cycle phases. According to Baniassad et al. [3], aspects in early phases support easier identification and analysis of aspects during later activities and, as a consequence, a better SoC in the aspect-oriented software systems, i.e., systems that contain aspects. In order to adequately model these systems, it is observed a diversity of initiatives [4,5]. In particular, UML (Unified Modelling Language) [6] has been investigated as a language to represent aspect-oriented software systems, since it is already widely used as in the academy as in the industry.
In another perspective, Evidence-Based Software Engineering (EBSE) [7] has recently arisen, inspired in the medicine area, and has given considerable contribution to the software engineering area. EBSE provides means by which the best current research evidence can be coupled with practical experience and human values into the decision making process regarding software development and maintenance [8]. Thus, EBSE can contribute to software engineering researchers and professionals to make decisions concerning the adoption of new technologies, including processes, techniques, methods, and so on. In this context, systematic mapping is a technique that provides a systematic way in order to comprehend a topic of research [9,10]. Considering its relevance, systematic mapping for different domains can be found and, specifically for software engineering area, mapping for software testing [11] and for requirement specification [12] are two examples. In this perspective, the conduction of systematic mapping in order to identify all work that propose UML extensions to represent aspect-oriented systems seems also to be interesting.

The main objective of this paper is then to present a panorama containing possibly all research work that have investigated and/or extended UML to represent aspect-oriented software systems. For this, we have adopted and conducted a systematic mapping to investigate this topic of research. As main achieved results, we have observed that there are many research groups publishing different initiatives to the aspect-oriented modeling using UML. However, there is not a consensus about a unified solution for that modeling.

This paper is organized as follows. In Section 2 we present a brief overview on aspect orientation, UML, and systematic mapping technique. In Section 3 we present the conducted systematic mapping. In Section 4 we discuss about achieved results. Finally, in Section 5 we summarize our contributions and discuss perspectives for further work.

2 A Brief Overview

In the context of Aspect-Oriented Software Development (AOSD), identification of aspects during early phases of the software life cycle — i.e., requirement specification and architecture design — has been a recurrent issue in the literature. According to Baniassad [13], aspects in early phases support an easier identification and analysis of aspects during later activities, such as design and coding phases. In the last years, conferences, such as Aspect-Oriented Software Development (AOSD) Conference, Early Aspects workshops series and Workshop on Aspects in Architectural Description, have broadly discussed the use of aspects in these early phases. In order to deal with aspects in early phases related to modeling of aspect-oriented software systems, there is a diversity of initiatives [14,15,16]. In special, extensions to the UML can be also found [17,18].

Currently, UML has been widely used in order to mainly facilitate development of object-oriented software systems. This standardized general-purpose modeling language includes a set of graphical notation techniques (also UML techniques on UML diagrams) to create visual models of software-intensive sys-
tems [6]. Its newest version (version 2.2) presents 14 techniques (such as Class Diagram, Sequence Diagram, and Component Diagram) divided into two categories (structural and behavioral) [6]. The diversity of UML techniques has therefore been basis for modeling of software systems with diverse characteristics, including aspect-oriented software systems.

In another perspective, it is noticed that when a research area is mature, there is almost always an increase in the number of reports and results made available. During the study of a new knowledge area, researchers usually conduct a bibliographical review (almost always an informal review) to identify publications related to a specific subject. However, this kind of review do not use a systematic approach and do not offer any support to avoid bias during the selection of the publications that will be analyzed. Thus, it is important to have mechanisms to summarize and provide an overview about an area or topic of interest [10]. For this, EBSE has investigated and proposed the use of systematic mapping technique [9,10]. In this context, an individual evidence (for instance, a case study or an experimental study divulged in a publication/paper) which contributes to a systematic mapping is called primary study, while the result of a systematic mapping is a secondary study. Systematic mapping aims at providing an overview of a research area to assess the quantity and type of primary studies existing on a topic of interest [10]. In short, systematic mapping is conducted by planning, conduction of search and screening of primary studies using inclusion and exclusion criteria [9]. Besides that, systematic mapping also conducts data extraction and analysis through the identification of categories and classification of the primary studies in these categories. As a result, this technique provides maps (for instance, in table or graphical format) containing condensed information about a research area.

3 Systematic Mapping Application

Our systematic mapping aims at identifying all primary studies that propose extension to the UML in order to represent aspect-oriented software systems. This systematic mapping was conducted from January 1st, 2010 to April 25, 2010 and involved three people (a researcher in software engineering, a specialist in systematic mapping, and an undergraduate student). In order to conduct our systematic mapping, we have used the process presented in Figure 1. In short, it is composed by four steps: (i) systematic mapping planning; (ii) conduction of the search; (iii) selection of the primary studies; and (iv) analysis, classification, and map building. These steps are explained in more details during presentation of our systematic mapping\(^1\). Following, we detail each step:

\(^1\) For the sake of space, only the main information related to our systematic mapping are presented in this paper. More details can be found in \url{http://www.icmc.usp.br/~buenolro/clei2010}
3.1 Step 1: Systematic Mapping Planning

In this step, it is established the systematic mapping plan that will be used as basis to conduct the systematic mapping. This plan consists in the formulation of the research questions, selection of the sources of primary studies, and establishment of the inclusion and exclusion criteria.

(i) **Research Questions (RQ):** These questions are structured corresponding to the objective that is intended with the systematic mapping that, in our case, it is the identification of a panorama involving UML extensions to model aspect-oriented software systems. The research questions of our systematic mapping are:

- **RQ1:** Which UML techniques have been applied to model aspect-oriented software systems?
- **RQ2:** Which are the terms (word or set of words) that are often associated with work that investigate UML extension to represent aspect-oriented software systems?
- **RQ3:** Which vehicles have more published work on UML extensions to model aspect-oriented software systems?
- **RQ4:** Which are the research groups that have investigated UML extensions to represent aspect-oriented software systems?

(ii) **Selection of Sources:**

In addition to the research questions, we established which search sources (i.e., publication databases) would be used to find the primary studies. The criteria used to select the sources were: **content update** (publications are regularly updated); **availability** (full text of the papers are available); **quality of results** (accuracy of the results returned by the search); and **versatility export** (since much information are returned through the search, a mechanism to export the results is required). These criteria are also discussed in [19]. Thus, the selected databases to our systematic mapping are shown in Table 1. According to Brereton et al. [20,21] and Dybå et al. [21], these databases are efficient to conduct systematic mapping in the context of software engineering. Furthermore, Scopus was added, since it is considered the largest database of abstracts and citations [22].
Table 1. Selected Databases

<table>
<thead>
<tr>
<th>Database</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACM Digital Library</td>
<td><a href="http://www.portal.acm.org">www.portal.acm.org</a></td>
</tr>
<tr>
<td>IEEE Xplore</td>
<td><a href="http://www.ieeexplore.ieee.org">www.ieeexplore.ieee.org</a></td>
</tr>
<tr>
<td>ScienceDirect</td>
<td><a href="http://www.sciencedirect.com">www.sciencedirect.com</a></td>
</tr>
<tr>
<td>Scopus</td>
<td><a href="http://www.scopus.com">www.scopus.com</a></td>
</tr>
<tr>
<td>ISI Web of Science</td>
<td><a href="http://www.isiknowledge.com">www.isiknowledge.com</a></td>
</tr>
</tbody>
</table>

(iii) **Establishment of Selection Criteria**: Another important element of the systematic mapping planning is to define the Inclusion Criteria (IC) and Exclusion Criteria (EC). These criteria make possible to include primary studies that are relevant to answer the research questions and exclude studies that do not answer them. Thus, the inclusion criteria of our systematic mapping are:

**IC1**: The primary study proposes UML extensions to represent aspect-oriented software systems;

**IC2**: The primary study proposes the use of UML techniques to represent aspect-oriented software systems; and

**IC3**: The primary study proposes a tool to support UML techniques to represent aspect-oriented software systems.

The exclusion criteria established are:

**EC1**: The primary study does not address modeling of aspect-oriented software systems;

**EC2**: The primary study is not related to UML;

**EC3**: The primary study proposes a new language to model aspect-oriented software systems;

**EC4**: The primary study does not present any abstract or it is not available for further reading;

**EC5**: The primary study is written in a different language than English;

**EC6**: The primary study presents an abstract and an introductory section that seem related to UML extensions to model aspect-oriented software systems; however, the rest of the text is not in fact related to;

**EC7**: The primary study is directly related to another primary study of the same author; and

**EC8**: The primary study consists of a compilation of work, for instance, from a conference or workshop.

### 3.2 Step 2: Conduction of the Search

In this step, the search by primary studies is conducted according to previously established plan. This search is done by looking for all primary studies that match with the search string in the search sources. This can be automatically conducted if these sources provide an efficient search engine. For this, we established the keywords and the search strings.
Keywords: To keep the scope of the systematic mapping, the keywords must be simple and well chosen. Thus, they must be simple enough to bring many results and also rigorous enough to cover only the desired primary studies. The keywords chosen to our systematic mapping are: UML, aspect-based, aspect-oriented, and aspect-oriented modeling.

Search Strings: Based on the keywords, the search string is built (through connections AND/OR) to incorporate the semantic expected to the search. The search string used in our systematic mapping was:

```
(UML AND ("aspect-based" OR "aspect oriented" OR "aspect-oriented" OR AOM OR "aspect oriented modeling" OR "aspect-oriented modeling")
```

We have also built the strings and/or settings for each search engine, since each engine accepts a specific formatted. In Table 2, it is presented these strings and settings.

As result of this step, we obtained a total of 460 primary studies: 103 from ACM Digital Library, 52 from IEEE Xplore, 10 from ScienceDirect, 185 from Scopus, and 110 from ISI Web of Science.

3.3 Step 3: Selection of the Primary Studies

In this step, the selection criteria (i.e., inclusion and exclusion criteria) are applied to select the relevant primary studies. To support the organization and manipulation of the primary studies, we used JabRef\(^2\), an open source reference manager system. It made possible to store information on the primary studies (including, for instance, title, authors, and abstract), as well as the inclusion/exclusion criteria applied to select each primary study.

From 460 primary studies, 231 were repeated. Thus, we found 229 different studies. Applying our selection criteria, we included then 74 primary studies. Table 3 summarizes the total of primary studies obtained in each database, the number of studies included (or selected), the precision rate\(^3\), the rate index\(^4\), and the search date. For instance, from ACM Digital Library, we obtained 103 studies; using the selection criteria, 46 studies were selected; precision rate is 0.446 (i.e., 46/103) and rate index is 0.622 (i.e., 46/229). It is important to observe that Scopus was the most efficient source, since 72.9% of all included papers were obtained in this source. Otherwise, ScienceDirect contributed with only 2.7% of primary studies which were also indexed by other sources. Therefore, we can point out that if there is an interest in searching work that involve the research topic of this work (i.e., UML extensions to model aspect-oriented software systems), searches using Scopus and ACM Digital Library could cover almost all

\(^2\) http://jabref.sourceforge.net/

\(^3\) Ratio between the total of included studies of a database and the total of primary studies obtained by this database

\(^4\) Ratio between the total of included studies of a database and the total of primary studies selected.
work of this topic. In other words, Scopus obtained 72.9% and ACM Digital Library obtained 62.2%. Since Scopus indexes also studies of other databases, such as IEEE Xplore and ACM Digital Library, if the search was concentrated only in these two database (Scopus and ACM Digital Library), we would find 93.3% of the primary studies that we have found.

Figure 2 presents the amount of primary studies included as relevant and separated by years. It is observed a strong growth of publications in the last years, mainly in Scopus and ACM Digital Library. Other sources (IEEE Xplore and ISI Web of Science) have what might be called an expected growth, considering the other two most numerous sources (Scopus and ACM Digital Library). It is important to note that only studies published until begin of February, 2010 were considered in our systematic mapping, since this systematic mapping was conducted in that period. Furthermore, we believe that some work published in

### Table 2. Search Strings by Source

<table>
<thead>
<tr>
<th>Source</th>
<th>Search String</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACM Digital Library</td>
<td>(Title:(uml AND (aspect-based OR aspect-oriented OR “aspect oriented” OR “aspect oriented modeling” OR aom OR “aspect-oriented modeling”)) OR Keywords:(uml AND (aspect-based OR aspect-oriented OR “aspect oriented” OR “aspect oriented modeling” OR aom OR “aspect-oriented modeling”)) OR Abstract:(uml AND (aspect-based OR aspect-oriented OR “aspect oriented” OR “aspect oriented modeling” OR aom OR “aspect-oriented modeling”)))</td>
</tr>
<tr>
<td>IEEE Xplore</td>
<td>((uml &lt;and&gt; (aspect-based &lt;or&gt; aspect-oriented &lt;or&gt; ‘aspect oriented’ &lt;or&gt; ‘aspect oriented modeling’ &lt;or&gt; aom &lt;or&gt; ‘aspect-oriented modeling’)) &lt;in&gt; ti) &lt;or&gt; ((uml &lt;and&gt; (aspect-based &lt;or&gt; aspect-oriented &lt;or&gt; ‘aspect oriented’ &lt;or&gt; ‘aspect oriented modeling’ &lt;or&gt; aom &lt;or&gt; ‘aspect-oriented modeling’)) &lt;in&gt; ab)</td>
</tr>
<tr>
<td>ScienceDirect</td>
<td>TITLE-ABSTR-KEY((uml AND (aspect-based OR aspect-oriented OR aspect oriented OR aspect oriented modeling OR aom OR aspect-oriented modeling)))</td>
</tr>
<tr>
<td>Scopus</td>
<td>TITLE-ABS-KEY(uml AND (aspect-based OR aspect-oriented OR “aspect oriented” OR “aspect oriented modeling” OR aom OR “aspect-oriented modeling”)) AND (LIMITTO(SUBJAREA, “COMP”) OR LIMITTO(SUBJAREA, “MULT”))</td>
</tr>
<tr>
<td>ISI Web of Science</td>
<td>TI=((uml AND (aspect-based OR aspect-oriented OR “aspect oriented” OR “aspect oriented modeling” OR aom OR “aspect-oriented modeling”)) OR TS=((uml AND (aspect-based OR aspect-oriented OR “aspect oriented” OR “aspect oriented modeling” OR aom OR “aspect-oriented modeling”)))</td>
</tr>
</tbody>
</table>
2009 were not still available in the databases when we conducted the search. Thus, we can observe a decrease of work in 2009.

3.4 Step 4: Analysis, Classification and Map Building

While in previous section we present a more quantitative analysis of our systematic mapping, in this section, we present the categorization of the primary studies. For this, two tasks were conducted: (i) search by keywords; and (ii) grouping of the primary studies into categories. Thus, primary studies were firstly carefully read (i.e., title, keywords, and abstract). Next task was the creation of categories. For this, research questions were considered. In the case of our systematic mapping, we were interested in identifying ways of applying UML techniques to model aspect-oriented systems. Thus, four categories were created:

UML Diagrams to Represent Aspect-oriented Software Systems: The keywords used in this category refer to the UML techniques (version 2.2): Use Case Diagram, State Machine Diagram, Timing Diagram, Interaction Overview Diagram, Communication Diagram, Sequence Diagram, Activity Diagram, Profile Diagram, Composite Structure Diagram, Package Diagram, Deployment Diagram, Component Diagram, Class Diagram, and Object Diagram. We are interested in knowing which UML techniques have been more investigated to model aspect-oriented software systems;

Main Keywords: Keywords of this category refer to the most relevant keywords obtained from the selected primary studies. To do this, we firstly
identified all the keywords from studies included. Then, these keywords were
classified according to their frequency of occurrence. Finally, we selected
the keywords that seemed more relevant or referred to software engineering
research areas;  

**Main Events and Journals:** Keywords of this category refer to name of the
events (such as conference and workshop) and journals in which the included
primary studies were published; and

**Main Research Groups:** Keywords of this category refer to the research groups
that have published work related to UML extension to model aspect-oriented
software systems. For this, we firstly identified all research groups related to
included primary studies identified in our mapping. Then, research groups
that had two or more primary studies (that were included in our mapping)
were chosen as keywords of this category.

Following, we classified the primary studies into keywords in the categories.
It is observed that a study can be classified sometimes into more than one
keyword per category. Moreover, for category “UML Diagrams to Represent
Aspect-oriented Systems”, we read all relevant primary studies and separated
them among the keywords. The same strategy was conducted to other three
category: “Main Keywords”, “Main Events and Journal”, and “Main Research
Groups”. As result, we build four maps, one for each category.

Figure 3 shows the map to the category “UML Diagrams to Represent
Aspect-oriented Systems”. It is observed that from first studies in 2001, there
is an increasing interest in extending UML diagrams to model aspect-oriented
software systems. It is important to notice that there is an overlap between Class
Diagram and Profile Diagram\(^5\), since primary studies extend the Class Diagram
using new stereotypes, such as \texttt{<<aspect>>} and \texttt{<<crosscuts>>}, that are
specific to aspect-oriented systems. Among the behavioral diagrams, Sequence
Diagram and State Machine Diagram have been more investigated to represent
aspect-oriented systems. These diagrams make possible to describe new behav-
iors introduced by the aspects and represent more easily when and how these
aspects will be executed. Otherwise, diagrams that are not so disseminated and
used (for example, Timing Diagram, Interaction Overview Diagram, and Com-
posite Structure Diagram) if compared with, for instance, Class Diagram, have
not been considered to model aspect-oriented systems.

Figure 4 shows the map to the category “Main Keywords”. The keywords of
this category are: Aspect Composition, Case Study, Component, Composition
Mechanism, Framework, Join Point Model, Model-driven Architecture, Model-
driven Development, Requirement, Reverse Engineering, Software Architecture,
Tool Support, UML Based, UML Extension, UML Profile, and UML Standard.
This map shows which are the keywords that are more directly related to work
that extend UML to model aspect-oriented systems. Thus, for instance, the
keywords “UML extension” and “UML Based” are two important keywords that

\(^5\) Profile Diagram allows to define custom stereotypes, tag definitions, and constraints
in a UML diagram.
could be used in the future searches for primary studies. Otherwise, only few primary studies on UML extension to aspect-orientation domain are related to “Requirements” and “Reverse Engineering”.

Figure 5 shows the map to the category “Main Events and Journals”. The keywords of this category are events and journals in which the 74 primary studies selected in our mapping were published. It is important to notice that the primary studies were published in a diversity of events and journal (42 different events and 7 journals). Furthermore, these studies are not only concentrate in well-known events or those specific of aspect-orientation, such as ICSE (ACM/IEEE International Conference on Software Engineering) and AOSD (International Conference on Aspect-Oriented Software Development). Therefore, if the search space was limited to those more known events and journals, important and relevant primary studies could not be found. For instance, AOSD published three primary studies and ICSE published only one among our 74 studies. Thus, systematic mapping was a fundamental and important technique to conduct our search and achieve positive and more extensive result.

Figure 6 shows the map to the category “Main Research Groups”. For the sake of space, only research groups that published two or more studies were considered. Therefore, 13 research groups were considered in this map. For instance,  

---

6 The meaning of the abbreviators can be found in http://www.icmc.usp.br/~buenolro/clei2010.
7 http://www.aosd.net/
research group of the Hebei University of Engineering (in China) published six primary studies and each one of them deals with one UML diagram: Sequence Diagram, State Machine Diagram, Class Diagram, and in the design of connectors to represent aspect-oriented components. There is also an effort in applying
4 Discussion

In this section, we discuss some issues related to the conducted systematic mapping, what also includes lessons learned and limitations we have noticed.

Regarding research questions established for our systematic mapping, it is observed that all of them were successfully answered. This suggests that, the general knowledge about representing aspect-oriented systems using UML techniques has been mapped. Besides that, we believe that results presented in this work are representative of the whole universe of existing approaches to model aspect-oriented systems, since systematic mapping has provide mechanism to achieve it.

Considering knowledge arisen from this work, it is possible to identify interesting and important research lines that can be investigated in future work. For instance, the establishment of a unified UML notation to aspect-oriented modeling and extension of UML diagram that have not been explored in aspect-orientation domain, such as Timing Diagram.

Regarding limitation of this work, other categories could have beeb established and, as a consequence, related maps could have been designed. In another perspective, other research questions could be established, considering specific
topics of research. Moreover, it is worth highlighting that systematic mapping conduction is not a trivial task because the amount of papers that need to be manipulated. Besides that, both paper reading and category analysis are manually conducted. Many of the studies provided none substantial information in the abstract or representative keywords to facilitate the analysis phase of the mapping. In those cases, it was also necessary to read the introduction and conclusion sections and, sometimes, the entire text of the primary study to determine the context and contributions of the study. Sometimes it was necessary to infer a conclusion to make some decisions. Besides that, during the conduction of our systematic mapping, more specifically after the primary studies had already been selected for our mapping, IEEE Xplore changed its search engine. Thus, the results presented here might be different from those obtained if a new search is conducted in that database.

5 Conclusions

The main contribution of this work was to present an overview about UML extensions to represent aspect-oriented software systems. To establish this overview, it was essential to systematically apply the set of steps provided by systematic mapping. As results, we found that there are so many initiatives extending UML for aspect-orientation domain. Furthermore, there is not a consensus about extensions widely disseminated and used. Thus, we believe that this overview can contribute to aspect-oriented software development area, getting new lines of research and mainly showing the need of initiatives to achieve a consensus to aspect-oriented software modeling.

Motivated by the promising results, in the future work, we intend to conduct systematic mappings on specific topics, for instance, UML extensions for Class Diagram. These mappings could contribute to decide which extension could be used to the aspect-orientation domain, what can also contribute to the establishment of a unified solution to aspect-oriented modeling.

Acknowledgments: This work is supported by Brazilian funding agencies: FAPESP, CAPES, and CNPq.

References