



**QUEEN'S
UNIVERSITY
BELFAST**

Classification and trend analysis of UML books (1997-2009)

Garousi, V. (2011). Classification and trend analysis of UML books (1997-2009). *Software and Systems Modeling*, 11(2), 273-285. <https://doi.org/10.1007/s10270-011-0189-9>

Published in:
Software and Systems Modeling

Document Version:
Peer reviewed version

Queen's University Belfast - Research Portal:
[Link to publication record in Queen's University Belfast Research Portal](#)

Publisher rights
Copyright 2013 Springer. This work is made available online in accordance with the publisher's policies. Please refer to any applicable terms of use of the publisher.

General rights
Copyright for the publications made accessible via the Queen's University Belfast Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The Research Portal is Queen's institutional repository that provides access to Queen's research output. Every effort has been made to ensure that content in the Research Portal does not infringe any person's rights, or applicable UK laws. If you discover content in the Research Portal that you believe breaches copyright or violates any law, please contact openaccess@qub.ac.uk.

Classification and Trend Analysis of UML Books (1997-2009)

Vahid Garousi

Software Quality Engineering Research Group (SoftQual)

www.softqual.ucalgary.ca

Department of Electrical and Computer Engineering, University of Calgary

2500 University Drive NW, Calgary, AB Canada T2N 1N4

vgarousi@ucalgary.ca

ABSTRACT

Technical books of each subject area denote the level of maturity and knowledge demand in that area. According to the Google Books database, about 208 UML books have been published from its inception in 1997 until 2009. While various book reviews are frequently published in various sources (e.g., IEEE Software Bookshelf), there are no studies to classify UML books into meaningful categories. Such a classification can help researchers in the area to identify trends and also reveal the level of activity in each sub-area of UML. The statistical survey reported in this article intends to be a first step in classification and trend analysis of the UML books published from 1997-2009. The study also sheds light on the quantity of books published in different focus areas (e.g., UML's core concepts, patterns, tool support, OCL and MDA) and also in different application domains (e.g., database modeling, web applications, and real-time systems). The trends of book publications in each sub-area of UML are also used to track the level of maturity, to identify possible *Hype* cycles and also to measure knowledge demand in each area.

KEYWORDS

Survey, statistical study, classification, trend analysis, UML, books.

TABLE OF CONTENTS

1 INTRODUCTION	2
2 RELATED REVIEWS/WORKS	2
3 GOAL AND RESEARCH QUESTIONS.....	3
4 DATA SOURCE AND DATA COLLECTION	4
4.1 Data Source	4
4.2 Search Method.....	4
4.3 Data Collection.....	5
5 RESULTS.....	5
5.1 RQ 1- Varying Number of UML Books Per Year	6
5.2 RQ 2- Classification and Book Trends by Programming Language focus.....	7
5.3 RQ 3- Classification and Book Trends by Phase of the Software Development Lifecycle	8
5.4 RQ 4- Classification and Book Trends by Application Domain.....	8
5.5 RQ 5- Classification and Book Trends by Focus Area	9
5.6 RQ 6- Relationship to Worldwide Interest on and Visibility of the UML	10
5.7 RQ 7- Other Findings	12
5.7.1 RQ 7.1- Focus Areas by Keywords.....	12
5.7.2 RQ 7.2- Authors Demographics.....	13
5.7.3 RQ 7.3- Towards Content Comparison and Identifying Possible Overlaps	15
5.7.4 RQ 7.4- More Successful UML Books	15
5.7.5 RQ 7.5- Notable Books, Observations and Quotes	16
5.7.6 RQ 7.6- Possible Future Outlooks	17
6 CONCLUSIONS AND FUTURE WORKS	17
ACKNOWLEDGEMENTS	17
REFERENCES	17

1 INTRODUCTION

Technical books of each subject area denote the level of maturity and knowledge demand in that topic. Usually a subject area needs to be mature and widely popular enough for books in that area to appear. Thus, the trends of book publications in a subject area can be used to track the level of maturity and also knowledge demand in that area.

Although research articles in a scientific area denote the research activity in that area, however they are books that indicate the high levels of maturity and knowledge demand (i.e., a book is only published if an author can convince a publisher that his/her planned book will have enough readers).

As of this writing, the Unified Modeling Language (UML) is 13 years old (1997-2010). Albeit UML's relatively young age, according to the Google Books database [1], about 210 UML books have been published from UML's inception in 1997 until 2009. Various book reviews are frequently published in various sources (e.g., IEEE Software Bookshelf). Also various online brief reviews/surveys (e.g., [2, 3]) report the pro's and con's of a small set of selected UML books. However, there is no systematic study to classify and analyze the trend of all the UML books since its inception. These are the goals of the study reported in this article.

The statistical survey reported in this article intends to be a first step in categorizing the UML books published from 1997-2009, and also shedding light on the quantity of books published in different focus areas (e.g., UML's core concepts, patterns, tool support, OCL and MDA) and also in different application domains (e.g., database modeling, web applications, and real-time systems). We have chosen the Google Books database [1] as our data source and mined statistical data about the UML books such as breakdown of books published each year in different focus areas of UML.

The results of this paper can help UML researchers to know which areas of the UML have been published more in terms of books compared to others, also what are the UML books' trends in terms of programming languages of choice, domains, focus areas, etc. Since most publishers publish books after conducting careful market demand analysis, the results presented in this paper would give researchers a high-level view of the level of penetration of UML subjects, and the areas for further work. Our analysis has also identified other findings, e.g., an interesting hype cycle for UML books, most successful books, and author demographics, which would be useful and interesting to the UML community.

The rest of this article is structured as follows. Section 2 discusses the related works and book reviews/surveys. Section 3 presents article's goal and research questions. The data source and data collection details of the study are discussed in Section 4. Section 5 answers the research questions and presents the results. Finally, Section 6 concludes the article and discusses the future work directions.

2 RELATED REVIEWS/WORKS

There are several online brief reviews/surveys on UML books (e.g., [2, 3]). Alleman reports in her online blog post [2] that she has used UML in a variety of settings, from analysis to Java code generation. She then reports on her favorite books on the topic. For example, for the following two books, she comments:

"Building Web Applications with UML [4]: This book describes the methods used to define and build web sites using an extension to UML. This notation extension can be easily implemented in any UML tools that support Stereotypes.

Use-Case Driven Object Modeling with UML: A Practical Approach [5]: Using another UML extension, a system development process is described for rapid definition and construction."

She also points out some frank comments, for example:

This is the post-print of a paper that has been published with the following DOI:

<http://dx.doi.org/10.1007/s10270-011-0189-9>

“There are too many books on UML, some good, some not so good, some a waste of money, some that are must haves that will never go out of style.... Simply buying a book or a product because of the [IBM] Rational logo may be a mistake.”

Another UML practitioner, Surveyer, has an online page about UML books [3]. He starts with the phrase that *“Ted Larman's book, Applying UML and Patterns [6], started a flood of good books on UML, Java and Object Oriented design. Jacquie Barkers's good Java overview, Java Objects followed and then a whole deluge.”* Surveyer then comments on the positive aspects of the above first book: *“... readers are introduced to agile and iterative development practices as some of the compelling reasons for OO design”.*

The book *“Applying UML and Patterns”* [6] has other online reviews such as the one on the IBM developerWorks [7]. The reviewer (Torpey) believes that the book has a new emphasis on the Agile methodology and is also *“good for beginners”*.

There are also several *“survey”* papers (e.g., [8]) and systematic literature reviews (e.g., [9]) focusing on UML, but these studies have a rather narrow focus, e.g., surveying practitioners for how they use UML [8], or systematically reviewing papers in the area of UML model consistency management [9]. Unlike the previous studies, the current article provides a bird-eye view on the 208 UML books published from 1997-2009.

Last but not least, the idea of books' classification has been widely adopted by librarians and researchers in bibliographic organization (for example, refer to the Cataloging & Classification Quarterly Journal). However, books in the focus area of UML have not yet been classified in details since precise classification of those books would require the human classifier to have knowledge-area (UML) expertise. This is one of the goals of this study.

3 GOAL AND RESEARCH QUESTIONS

The approach we have used in our study is the Goal, Question, Metric (GQM) methodology [10]. Using the GQM's goal template [10], the goal of this survey is to mine statistical data from the collection of all UML books and to categorize them into different focus areas for the purpose of identifying the trends, and also to provide a view on the growth of field, as measured by the number and types of books being published. Based on the above goal, we raise the following research questions.

1. What is the trend of number of books published per year since 1997?
2. What are the classification and trend of UML books by programming language focus?
3. What are the classification and trend of UML books by phase of the software development lifecycle?
4. What are the classification and trend of UML books by application domain?
5. What are the classification and trend of UML books by focus area?
6. What is relationship (if any) between the UML books' publication trend to worldwide interest on the UML (as measured by Google search volumes)?
7. Can any other types of finding be extracted from the analysis?

To conduct a classification of the UML books, we classified them according to the four dimensions posed in RQ 2-5. These dimensions were defined after comprehensive and careful reviewing of the UML books' titles and their contents.

For the purpose of classification, the author reviewed each book's title and its table of contents (as provided by Google Books) and determined, according to his UML knowledge and expertise, the category that each book would fall w.r.t. each dimension. The entire spreadsheet file that we prepared for this classification is provided as Online Resource 1, along with this paper in the SoSyM paper repository, for replicability purposes and also for further studies.

This is the post-print of a paper that has been published with the following DOI:
<http://dx.doi.org/10.1007/s10270-011-0189-9>

4 DATA SOURCE AND DATA COLLECTION

4.1 DATA SOURCE

As our data source, we used the Google Books database [1]. To provide a measure on the massive size of this database, as of October 2008, Google stated that they had seven million books searchable through Google Books [11].

4.2 SEARCH METHOD

To search for the published UML books, we conducted several experimental searches to find the best search criteria and used the list of books we knew of to verify and refine our search method. Our experiments showed that one suitable search method was to use the “Advanced Search” feature and then enter the “UML” or the “Unified Modeling Language” phrase in the book title field.

We also found that searching in the Google Books for the phrases “object-oriented” and “software design” would bring books which sometimes have major focus on UML as well, e.g., [12]. In the list of these books, we carefully inspected the table of contents and often the book contents (offered for free review by the Google Books or Amazon.com) to determine if UML was a major focus of books having “object-oriented” in their titles, but not “UML” exactly. For example, considering the book “The object-oriented thought process” [12], although it does not have the word “UML” in its title, almost all of the OO concepts discussed in this book are represented by UML diagram and Java code examples. Many UML concepts (e.g., metamodels) are also discussed in this book. There are many books that merely have a short discussion on UML and reference it. The distinction to include or exclude a given OO book in our study pool was not an easy choice. The author thus consulted with a few colleagues and their opinions were considered in the decisions. As a general rule, we chose to include the OO books which, in addition to “using” UML, also discuss UML concepts, and chose to exclude the OO book which only used some of the UML (e.g., class diagrams) to present the OO concepts and also those that covered only a subset (but not all) of the UML diagrams (e.g., [13, 14]). As another example, in a book entitled “Object-Oriented Programming and Java” [15], a search in the contents for UML returns no results, meaning that there is no discussion of UML in this book.

From the general software modeling perspective, since this study intends to classify and conduct a trend analysis of only UML books and not all the software modeling books, books covering only the concepts of “software modeling” and not UML in enough detail were excluded from our pool of books.

To get the list of books published in each individual year, the publication year was filtered. As an example, two screenshots from the Google Books search tool (searching for UML books published in 2009) and part of the corresponding result page are shown in Figure 1.

Although Google Books is supposed to only show the actual books, it often shows in the search results non-book items as well (e.g., theses, ISO standards, conference proceedings, and tutorials). Thus, we had to conduct manual data filtering to extract only the actual books. We are suggesting to the Google Books team to improve its search tool in future to only show the book results.

This is the post-print of a paper that has been published with the following DOI:
<http://dx.doi.org/10.1007/s10270-011-0189-9>

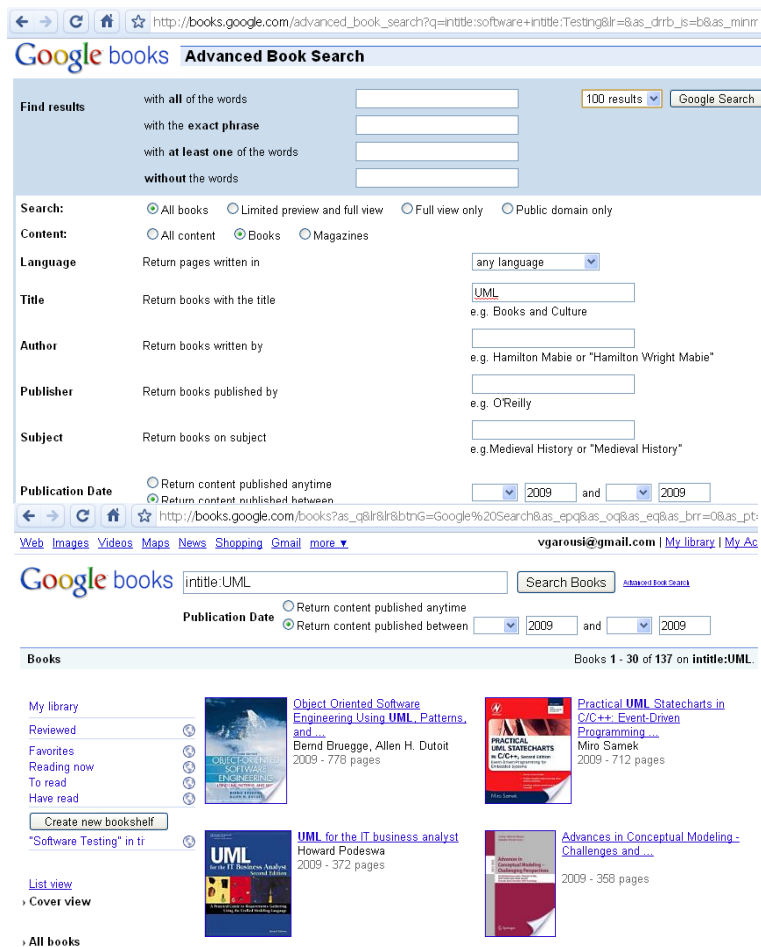


Figure 1-Screenshots from the Google Books search tool and its example search result page.

We also wanted to assess the completeness and accuracy of the Google Books database in terms of including all the published books. Our first intent was to search for systematic studies on this topic. However we were not able to find any resource providing a formal basis for the completeness of Google Books. The best source we could rely on was the support page of the Google Books that says: "our books are provided either by publishers and authors, through the Google Books Partner Program, or by our library partners, through the Library Project." We also saw that major publishers of software engineering books such as Springer, Addison-Wesley, Prentice Hall, Wiley, Cambridge University Press, and many others participate in Google Books. Also, from an informal perspective, we did not find any UML books that we knew about, missing in the pool. Thus, we could rely to some extent on the completeness of the Google Books database. Nevertheless, formal studies in this area are needed.

4.3 DATA COLLECTION

After extracting the records for each year from Google Books, they were filtered as discussed above and were then entered into an Excel spreadsheet which is provided as Online Resource 1. All the data we analyze in this paper were collected during April of 2010.

5 RESULTS

Detailed statistical results from our survey are presented in this section.

5.1 RQ 1- VARYING NUMBER OF UML BOOKS PER YEAR

The yearly histogram of the number of UML books, as indexed by Google Books, is shown in Figure 2. The first four books on the topic [16-19] were published in 1997, the official birth year of UML. The total number of UML books in the area from 1997-2009 is 210. As of April 2010, our search for year 2010 returned one UML book published in this year, but since 2010 was in progress as of this writing, we decided to not report the (partial) results of 2010 in this article.

There is no clear (increasing or decreasing) trend in Figure 2. However, interesting up and down trends can be observed in various periods. The trend of the first half of the period under study reminds the author of the *Hype cycle* [20, 21]. (A Hype cycle is a graphic representation of the maturity, adoption and business applications of specific technologies).

In analysis of Hype cycles (e.g., [22]), the x-axis is timeline (maturity) of a technology and y-axis is its perceived visibility. We take the number of UML books in each year as a suitable measure of UML's visibility, and thus analyze the Hype cycle of UML according to its books as follows.

As per the Hype cycle definition, it seems that the UML initiative was well received by book authors when it was *triggered*. This phase is labeled by *TT* (Technology Trigger) in Figure 2, as per the Hype cycle terminology [20, 21]. Afterwards, there seem to be a peak of "inflated expectations" (*PIE*) from 1999-2000. In 2000, there seems to be a "trough of disillusionment" (*TOD*), followed by a "slope of enlightenment" (*SOE*). In 2002-2004, we see a slightly increasing "plateau of productivity" (*POP*) for UML as perceived by the number of books in the area.

A few other researchers have analyzed and commented on the trending and Hype cycle of the UML and its successors (e.g., SysML) [23]. Willard believes [23] that SysML has the liability of a software-centric language in its heritage (UML). He believes that SysML may not experience the same sharp height of interest that typically occurs in the first two phases of the Hype cycle.

Research firms such as Gartner Group and Forrester Research have also conducted trend prediction for object-oriented modeling. Gartner Group studied in [24] the popularity trend of the Model-driven architecture (MDA), the design approach which encourages the use of UML. In a 2006 report [24], Gartner Group identified MDA as an "on the rise" technology.

Forrester Research has a conflicting view, as it refers to MDA as being "Dead On Arrival" (DOA) in its 2006 report [25]. An excerpt from the report says that: "Although several companies have contacted Forrester to talk about MDA, very few have wholeheartedly adopted it, especially once they understand its full implications."

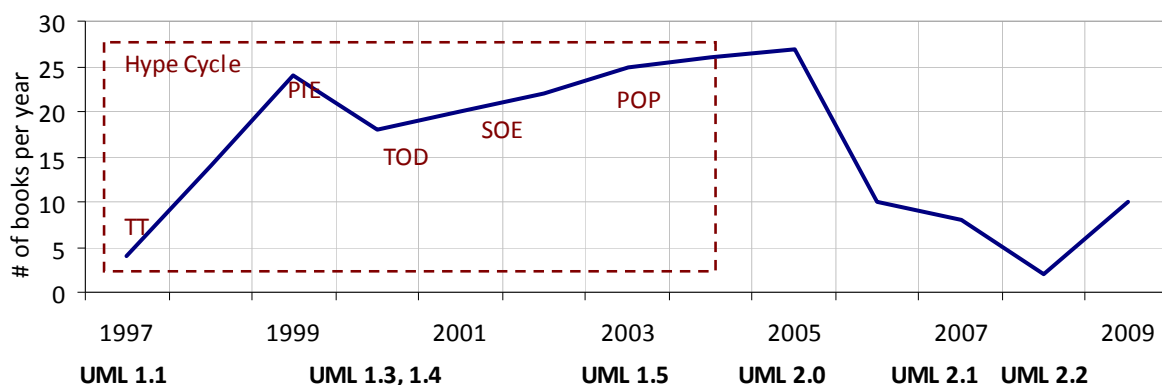


Figure 2-Histogram of the number of UML books from 1997-2009.

According to Figure 2, introduction of UML 2.0 in 2005 seems to have led to a light peak in 2005 when authors wrote new books on UML 2.0. But from 2005 onwards, we see a severe reduction in the quantity of UML books, reaching to only 2 books in 2008. 2009 shows a sign of increase (10 books in that year). The

This is the post-print of a paper that has been published with the following DOI:
<http://dx.doi.org/10.1007/s10270-011-0189-9>

reduction in the quantity of books can be perhaps explained by relative maturity of UML itself in recent years (thus not needing many new books to say the same things) and also due to branching-out of several similar modeling languages from it, e.g., the Systems Modeling Language (SysML) and various Domain-Specific Languages (DSL) in recent years. This naturally takes the main focus off the UML and the community (including book authors) start to focus on other similar notations and languages.

5.2 RQ 2- CLASSIFICATION AND BOOK TRENDS BY PROGRAMMING LANGUAGE FOCUS

Some UML books are neutral in terms of their programming language focus, while others target specific programming languages, such as C++ [26], Visual Basic [27], Java [28], or .Net family of languages [29].

The aggregate results of our classification analysis are reported in this and the next few sections. For detailed results, i.e., to see more clearly which books fit into each classification, the reader is referred to the Excel data file (Online Resource 1).

The histogram of the UML books in our study pool based on programming language focus is depicted as a stack-chart in Figure 3. About 81% (171 out of 208) of the books are neutral in terms of their programming language focus (shown as Generic in the histogram). Java (21 books), Visual Basic (7 books), and C++ (6 books) had the highest programming language focus, in order. 3 books focused on using UML for developing programs in the .Net framework and one book [30] on the UML and SystemC programming language.

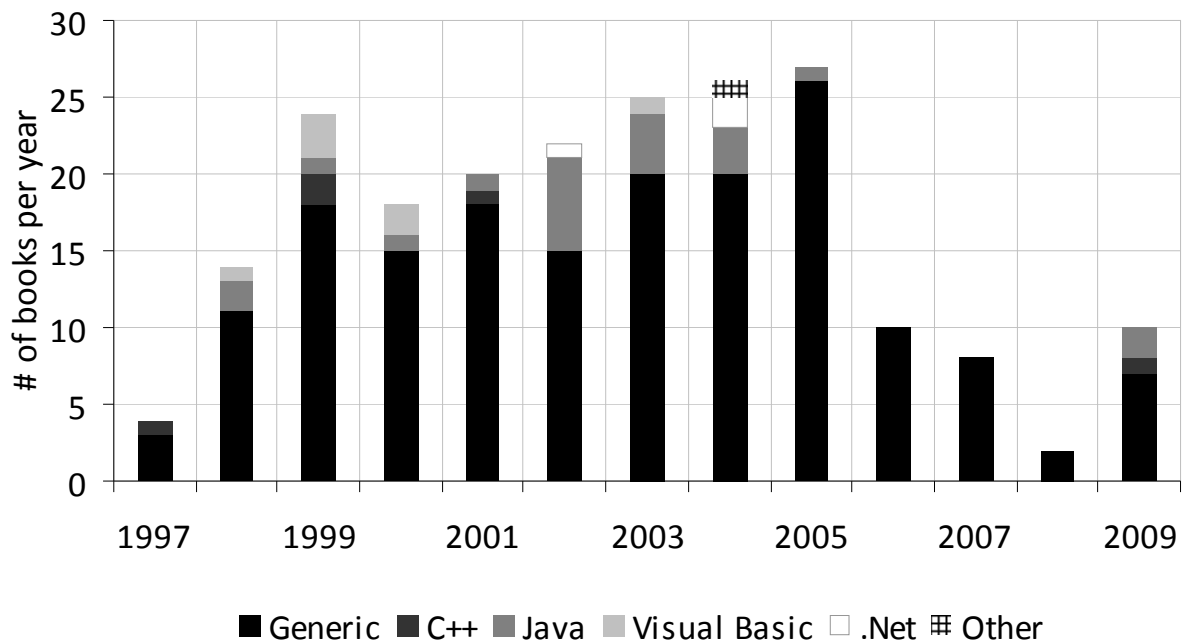


Figure 3-Histogram of UML books published based on programming language focus.

In terms of timeline, it seems that quickly after the UML's inception in 1997, books have started to focus on the practical and code-related mappings of the UML concepts (e.g., generating source code from class diagrams). Java took a big portion of the books' focus from 2001-2003. After 2005, it is interesting and somewhat odd to see that books are putting less focus on programming languages, which is perhaps due to the inception of UML 2.0 and that books have tried to cover this new version in detail than focusing on UML's mapping and its relationship to programming languages. Visual Basic was popular in UML books in early years, but new books are not focusing much on it anymore.

5.3 RQ 3- CLASSIFICATION AND BOOK TRENDS BY PHASE OF THE SOFTWARE DEVELOPMENT LIFECYCLE

While most books seem to focus on the object-oriented analysis and design (OOAD) phases, some UML books focus exclusively on other phases such as: testing [31].

The histogram of UML books based on explicit focus on phases of the software development lifecycle is shown as a stack-chart in Figure 4. As one would expect, most of the books (76%) are focusing on requirements/analysis/design as they are the major focus of UML.

Three books [31-33] discuss testing, verification and validations based on and related to UML models. Eight books discuss the “entire” development process based on UML and focus on concepts such as the Unified Process. Two books [34, 35] explicitly have the word “architecture” in their title and focus heavily on the topic. Note that, in this work, the concept of “architecture” has been considered a sub-area of “design” in the context of UML, which is a reasonable consideration. Thus, books which heavily discuss architecture would fall in the “design” category in books in Figure 4.

Several books also focus mainly on development aspects and UML, e.g, the book in [36] presents iterative UML development using Visual Basic 5.0, and discusses a lot of development details.

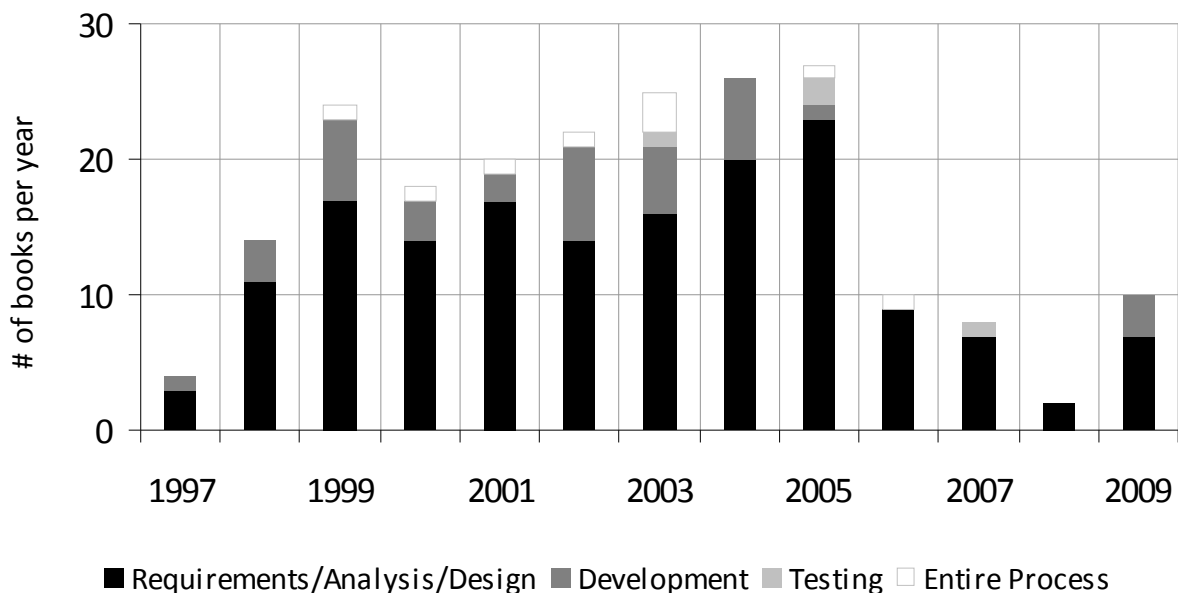


Figure 4- Histogram of UML books with an explicit focus on phases of the software development lifecycle.

5.4 RQ 4- CLASSIFICATION AND BOOK TRENDS BY APPLICATION DOMAIN

This dimension represents the application domain under focus of each book which include: Generic (i.e., all application domains) [37], web applications [4], real-time systems [38], databases and data modeling [39], system-on-a-chip (hardware) design [40], and agricultural systems [41].

The histogram of UML books based on their application domain of focus is shown as a stack-chart in Figure 5. In total, 87% of books (183 of 208) are generic in terms of their application domain, i.e., they present UML and OO analysis/design from a domain-natural perspective.

From the domain-focused books, those on real-time systems (also called real-time UML) had the highest single-domain focus (9 books). 3 and 2 books focused on using UML for designing and building web applications, and database/data models, respectively. There have been 11 other domain-focused UML

books on topics such as system-on-a-chip [40], secure systems [42], mobile computing [43], and agricultural systems [41].

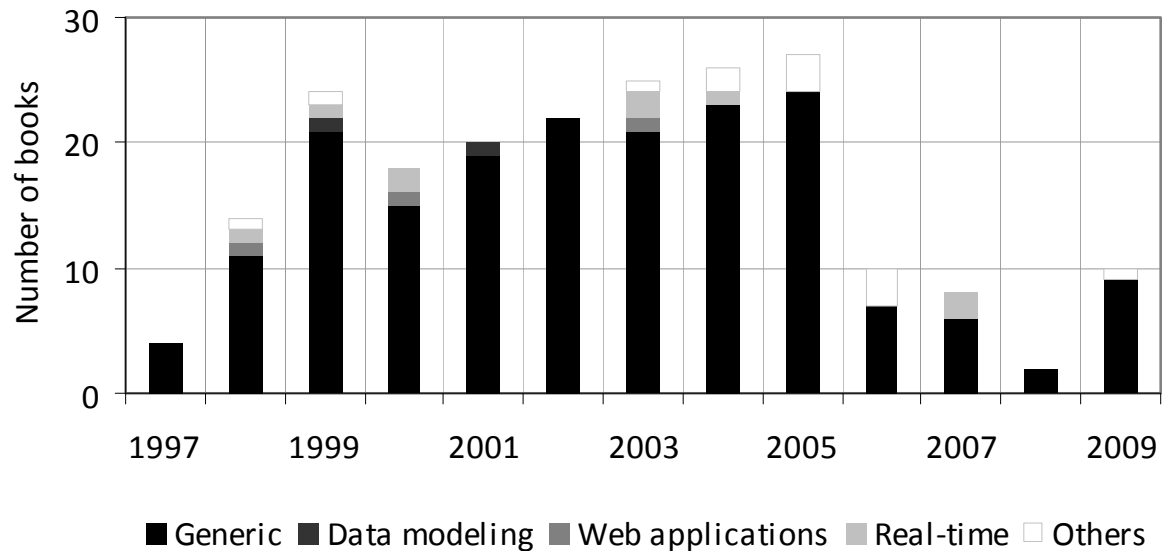


Figure 5- Trend of UML books based on domain focus.

Several data modeling UML books were published around year 2000, but no new editions or new books have come out after that time. This might be due to relatively stable, mature and up-to-date concepts of those books or due to low demand for new books on those subjects.

More books on other non-conventional domains have been published in recent years which seem to indicate that books readers (researchers and practitioners) in other areas such as agricultural system development [41] are using UML for their analysis, design and modeling needs.

5.5 RQ 5- CLASSIFICATION AND BOOK TRENDS BY FOCUS AREA

This dimension is the main focus area of each book. For example, there are books covering only the “core” of the UML [37], i.e., its diagrams and notation without presenting any tool support or focusing on any particular programming language (e.g., Java). There are other books which focus on tools [44] (such as Rational Rose and IBM Rational Software Architect), UML profiles [31], process aspects [45] (e.g., the Unified Process), Object Constraint Language (OCL) [46], patterns [6], or Model-Driven Architecture (MDA) [47].

Figure 6 depicts the ratio of UML books published in each of the focus areas (as defined in Section 4.2). Some of the main observations from these ratios are discussed next.

- The “core” UML topics (its diagram types and notations) dominate most of the focus areas in the books (79.8% of the books).
- After the “core” UML focus area, books focusing on patterns (19 books) and tools (8 books) are the next two top areas.
- There was one book published in 2004 on Agile UML modeling [48].
- In the entire pool, only two books had a major focus on MDA (published in 2002 and 2004), which seems to denote that the MDA initiative has not received a good attention from book authors.

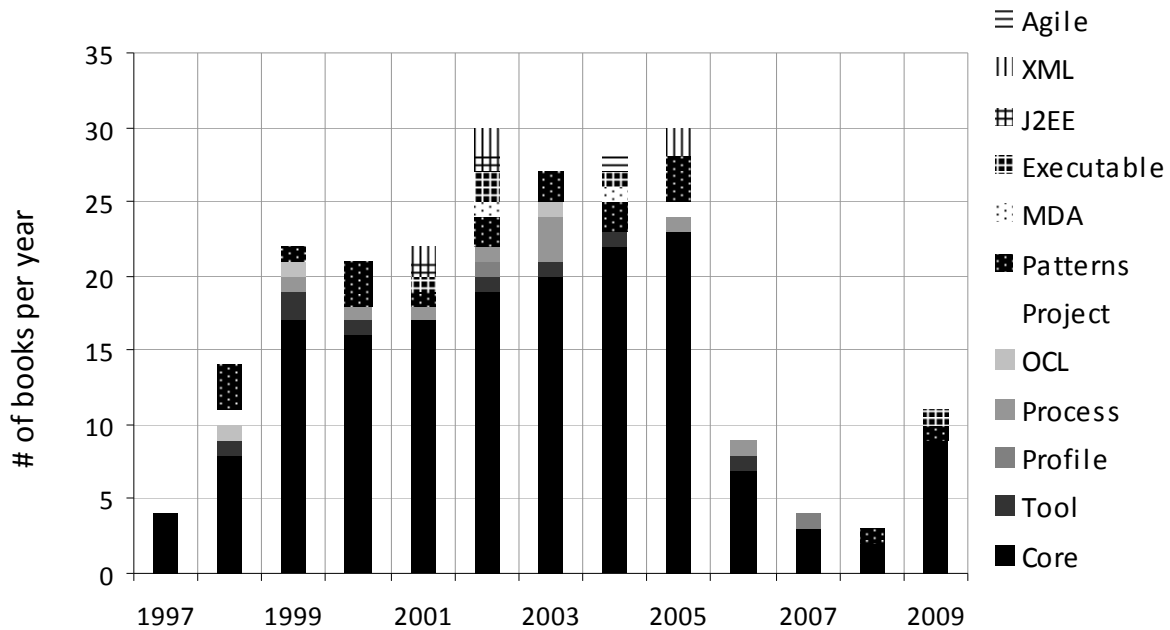


Figure 6- Ratio of UML books based on focus areas.

5.6 RQ 6- RELATIONSHIP TO WORLDWIDE INTEREST ON AND VISIBILITY OF THE UML

The purpose of this analysis is to see whether there is any correlation between the UML books trend and that of UML's online popularity.

Internet search statistics have been used in a variety of studies and in various domains, e.g., to detect influenza epidemics [49], to mine business intelligence [50], or to identify public interest in science [51].

Recently, software engineers have also started to use Internet search statistics, to mine/discover trends in their area. For example, Rech suggests [52] that Google Trends statistics can be used for software engineering in several scenarios:

- *Shape research activities*, e.g., by identifying new or increasing interest by the public or the media
- *Investigate technology maturity*, e.g., by analyzing the long-term search behavior for CMMI
- *Investigate potential markets*, e.g., through an analysis of the news articles associated with peaks for searches or news
- *Investigate market penetration*, e.g., through an analysis of the news articles that start or lead to rising search interest in a technology

Rech also suggests [52] that Google Trends can be used as a tool to identify a potential hypes in software engineering, i.e., finding topics that have a steep search or news curve. Examples for these hypes can be seen for topics such as "Web 2.0", "AJAX", or "Wikis".

To find historical worldwide interest on UML versus "software design" and "software modeling", we used the Google's Insights for Search tool (www.google.com/insights/search) by searching for the above keywords. Note that although we only consider the UML books in our analysis in Section 5.1 to 5.5, we still would like to compare internet search popularity of keyword "UML" versus "software design" and "software modeling" since these are similar concepts to UML in the software engineering community.

This is the post-print of a paper that has been published with the following DOI:
<http://dx.doi.org/10.1007/s10270-011-0189-9>

The purpose of this analysis is to see whether there is any correlation between the two trends, i.e., Does books' trend change as the UML's online popularity change?

Also note that both tools Google Trends and Google Insights for Search are quite similar in functionality, but the latter provides more detailed results. Thus, we used that one. But both tools have only started to store the statistical data after 2004.

For "modeling", we used both the American and British English pronunciations ("modeling" and "modelling") to ensure that all the data are included. Results are shown in Figure 7. Note that the y-axis in this curve is on a relative logarithmic scale as absolute values are not provided by Google. The measures denote the relative worldwide search volume for each keyword.

We can see that the trend of the "UML" curve is decreasing, denoting that people worldwide are searching less and less for "UML". The exact reason(s) is (are) not clear, but several contributing factors can exist: (1) on a most naïve reasoning, one might say that UML is losing its popularity, (2) another possible contributing factor is that more and more people are learning UML from other sources (such as books), and thus are searching less and less online to learn about it.

The term "Unified Modeling Language" has not been searched for much (almost negligible compared to the other four terms). "Software design" is only marginally less searched for than "UML". "Software modeling" has also been low in search volumes, and its searches have been decreasing slowly. The search results for the American versus British spelling of term "modeling/modelling" have slightly different trends, with "UML modeling" having slightly more search requests (denoting that more people are searching using the American spelling).

By comparing Figure 7 to Figure 2, we cannot observe any similarity in trends. meaning that the trend of book publications do not correlate with that of world-wide search for information on UML.

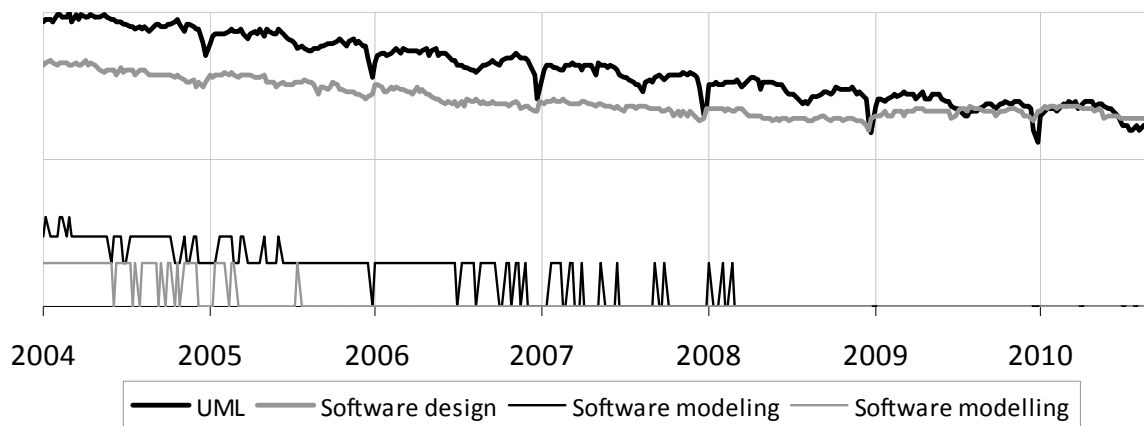


Figure 7-Worldwide interest on UML, "software design" and "software modeling" over time according to Google search volume.

Another trending tool provided by Google for search terms is its timeline trends of appearance in online resources (web pages), which is different from the search volumes. This metric somewhat denotes the visibility of a technology measured by the frequency of appearance in online pages, documents, etc. For this purpose, we used the "timeline" mode of the conventional Google search tool, and generated the graph in Figure 8. Unlike Google Trends and Google Insights for Search (with data since only 2004), the timeline range of this tool is much longer as it supports data back to many years ago and depending on the search query, the date can be as old as in 1800's (in old English texts for example).

On this timeline curve, the release dates of different UML versions are shown. Only the first version 1.1 seems to have caused a major peak in terms of visibility (appearance) of UML in online resources. This is

This is the post-print of a paper that has been published with the following DOI:
<http://dx.doi.org/10.1007/s10270-011-0189-9>

perhaps due to the fact that since version 1.1 was UML's first version, many online pages and resources dedicated more discussions to explain its features and capabilities. After a slight Hype cycle from 1997-2004, the "UML" keyword seems to be currently staying in its "plateau of productivity" [21] in online resources. There is a slight peak around 2004 and 2005 which is most probably due to the introduction of UML 2.0. There are some similarities between this trend line and the books trend line in Figure 2. Compared to Google Insights, Google timeline is showing better the hype cycle of the UML because it includes data before 2004.

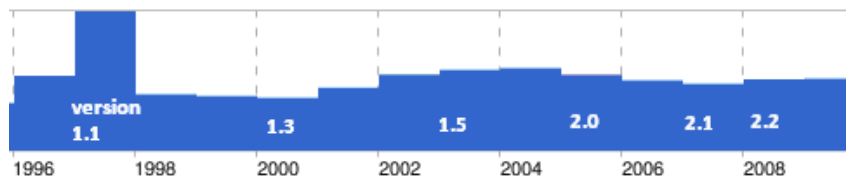


Figure 8- UML's trend of popularity (appearance) in online resources (derived from Google timeline search).

5.7 RQ 7- OTHER FINDINGS

5.7.1 RQ 7.1- Focus Areas by Keywords

To get a birds-eye understanding of major topics addressed by the books in our pool, we applied a text visualization technique on the books' titles. For this purpose, we used a powerful free online visualization tool, called *Many Eyes* [53], developed by IBM.

Since the tables-of-contents of all books were not available in a machine-readable format (e.g., text), we decided to limit our text visualization to the book titles in our pool. Figure 9 shows the "word cloud" of all the book titles given as the input text. Since the words "UML", "software", "object-oriented", and "systems" are very common in this context, we excluded them from the input text. This "word cloud" shows that phrases such as design, modeling, analysis, and object are among the most widely used terms in book titles. The phrase "2.0" (referring to UML 2.0) is also a top hit. We can see that Figure 9 provides a birds-eye understanding of major topics addressed by the books in the pool.

Java seems to be the most popular programming language in the book titles compared to others (e.g., C++). This confirms the finding we had based on manual classification of books based on programming languages in Section 5.2.

If the reader is interested in finding which books have in their title the top keywords, the Excel data file (Online Resource 1) can be easily searched for this purpose. The visualization file including the data used to create Figure 9 is also available online [54] and can be explored for other types of analysis by interested readers.

This is the post-print of a paper that has been published with the following DOI:
<http://dx.doi.org/10.1007/s10270-011-0189-9>



Figure 9- “Word Cloud” of book titles (generated using *Many Eyes* [53]). Since the words “UML”, “software”, “object-oriented”, and “systems” are very common in this context, we excluded them from the input text.

As discussed above, we were interested to conduct the above “word cloud” analysis and visualization on the books tables of contents, however unfortunately, the tables of contents for only a small fraction of the books were available online in a machine-readable format (e.g., text). Furthermore, the effort involved with manually typing all those tables of contents from graphical files (provided by Google Books) was beyond the capacity of our project. We decided not to conduct that analysis on only partial set of data (tables of contents).

Also, it should be mentioned that several books cover UML in less than half of their chapters, and discuss other matters in the remainder. For example, the main focus of the book in [55] is patterns in Java and the UML is chosen as the illustration notation. As another example, the book in [36] presents iterative UML development using Visual Basic 5.0.

5.7.2 RQ 7.2- Authors Demographics

In this RQ, we wanted to see which authors have authored more than others. The list of authors which have authored at least two UML books in our pool is shown in Figure 10. Kendall Scott, with participation in the authorship of 7 books, stands first.

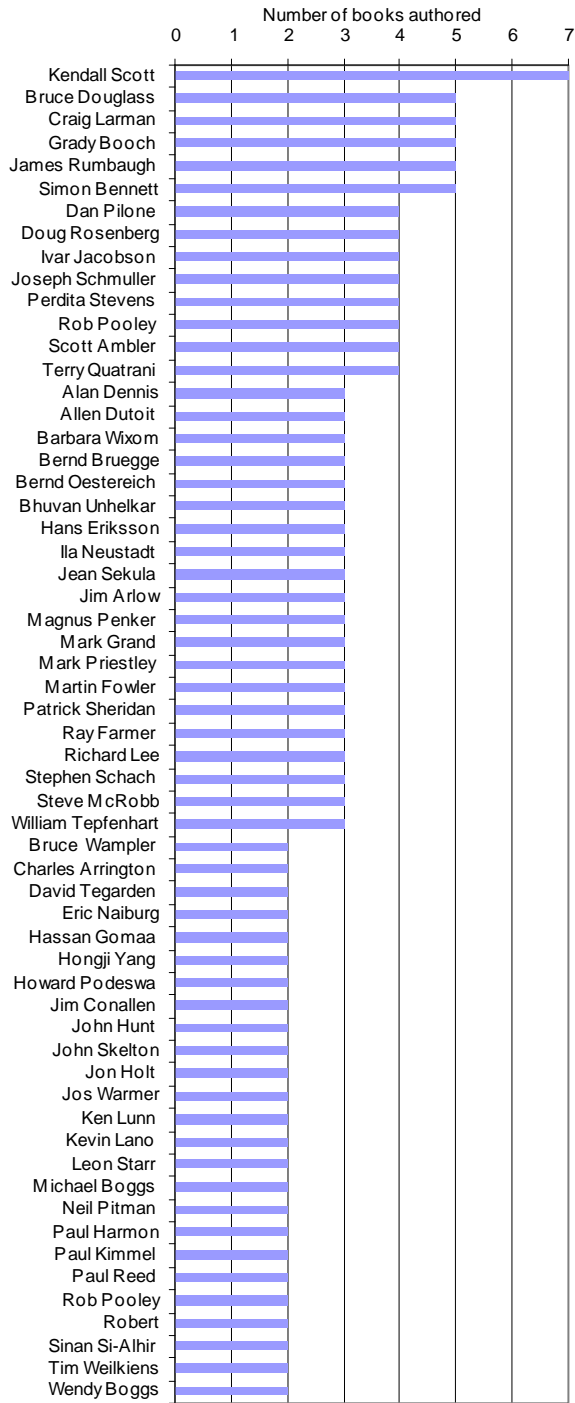


Figure 10- Authors which have authored at least two UML books.

We also wanted to derive the statistics of the number of authors for books. The results are shown in Figure 11. 87 books were written by single authors. 62, 21, and 1 books had two, three and four authors, respectively. Two books [29, 31] were written by six authors each.

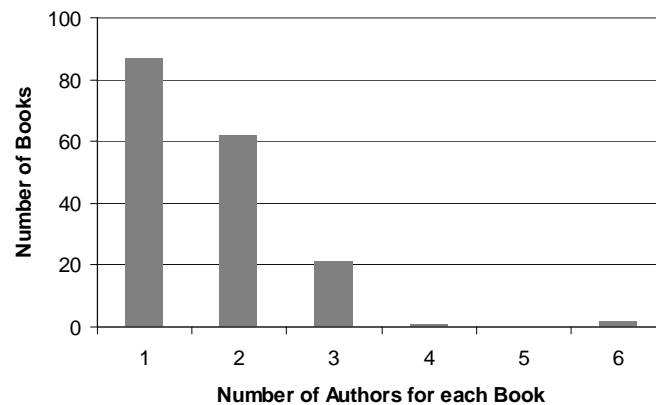


Figure 11- Size of book authors' teams.

5.7.3 RQ 7.3- Towards Content Comparison and Identifying Possible Overlaps

Many people including the author believe that there is considerable overlap among the software engineering books, including those on UML. Clearly, more than 200 books all with new materials on UML would be hard to imagine. Most people believe that each software engineering book does not necessarily propose many new ideas. For example, in a recent blog post by Adam Goucher, a Canadian software tester and blogger, he mentions that: *“When I’m reading testing books, I am looking not for (completely) new techniques, but clarifications of, new insights on and new ways of explaining techniques. HWTSAM [the book titled: How We Test Software At Microsoft] ably met all that criteria”* [56].

We did not have adequate resources to conduct detailed analysis on the extents of overlap (content similarity) among books, but we can mention some high-level examples and insights. For example, the author has five of the UML-Java-combination books and, by reviewing them, one can easily find noticeable overlap among them. The reader is recommended to carefully review the book reviews and other resources to choose a suitable book for her/his needs on UML. Future studies on this topic are needed.

5.7.4 RQ 7.4- More Successful UML Books

We found out that 11 of the books in our study pool have been published in several editions. The data are shown in Table 1. Considering these 11 book titles and their multi editions, the number of unique book titles on UML drops from 210 (size of our pool) to 193.

Although relating the number of editions to success of a UML book might be seen as a simplistic judgment, but it certainly is a factor in which publishers agree to proceed with upcoming editions of a book. Thus, some of the most successful UML books can be found in Table 1.

Book title	Editions		
	1	2	3
Applying UML and patterns: an introduction to object-oriented analysis and design	1998	2002	2005
Developing software with UML: object-oriented analysis and design in practice?	1999	2002	
Object-oriented systems analysis and design using UML?	1999	2002	2006
Real-time UML: developing efficient objects for embedded systems?	1998	2000	2004
Sams teach yourself UML in 24 hours	2001	2004	
Systems Analysis and Design with UML	2003	2005	2008
The object constraint language: precise modeling with UML?	1999	2003	

This is the post-print of a paper that has been published with the following DOI:
<http://dx.doi.org/10.1007/s10270-011-0189-9>

The unified modeling language user guide	1995	2005	
UML and C++: A Practical Guide to Object-Oriented Development	1997	2001	
UML distilled: applying the standard object modeling language?	1997	2000	2004
UML for systems engineering: watching the wheels?	2001	2004	
Using UML: software engineering with objects and components?	1999	2006	

Table 1-Multi-edition UML books

5.7.5 RQ 7.5- Notable Books, Observations and Quotes

We found several interesting books which are worth noting here. Tanaka presents how the UML has been utilized in the Japanese industry in his 2004 book *“The Contextualized UML: Practical Use of Unified Modeling Language in Japanese Industry”* [57].

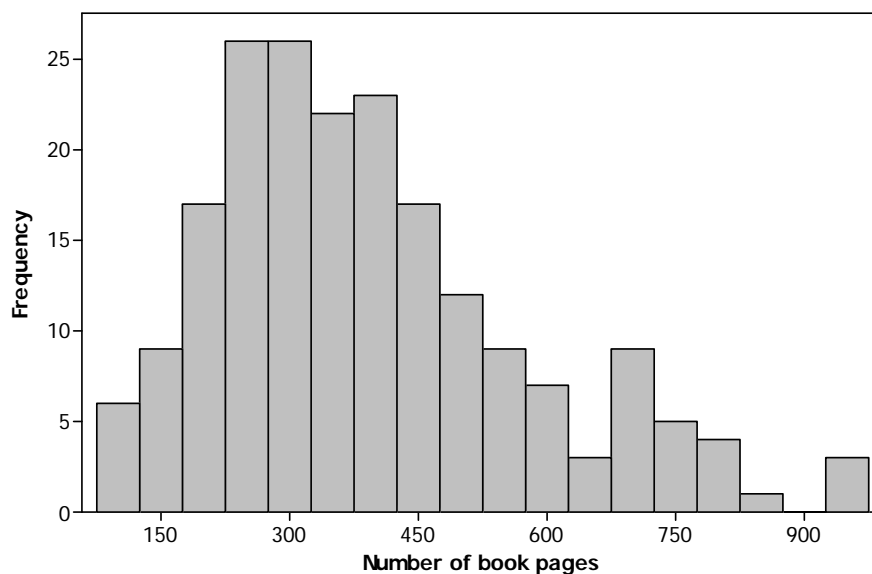
Martin and Müller present a book on using UML for system-on-a-chip (hardware) design [40]. The book includes interesting chapters on how hardware and software can be co-designed using UML.

Papajorgji and Pardalos have authored a book on applying software engineering techniques (mainly UML) to agricultural software systems [41]. The authors present the complete UML diagrams of a several real-world software systems in the agricultural domain, e.g., a soil-water balance and irrigation scheduling system. Weilkiens and Oestereich have co-authored a guide book for a UML 2 certification, referred to as OMG Certified UML Professional (OCUP) [58].

Also a major book series on UML and object-orientation is published by Addison-Wesley which is called the *object technology series*. As of April 2010, that book series had 66 titles [59] which is perhaps the single book series/publisher with a heavy focus on UML.

We also came across interesting quotes in some books, e.g., in the book *“The object-oriented thought process”* by Matt Weisfeld, the author points out that: *“Although learning a modeling language is an important step, it is much more important to learn OO skills first. Learning UML before OO concepts is similar to learning to read an electrical diagram without first knowing anything about electricity”* [12].

As a notable observation, we measured the histogram of the book sizes in number of pages. The results are shown in Figure 12. Most of the books have between 250 and 350 pages. There were three books over 900 pages [60-62].



This is the post-print of a paper that has been published with the following DOI:
<http://dx.doi.org/10.1007/s10270-011-0189-9>

Figure 12- Histogram of book sizes in number of pages.

5.7.6 RQ 7.6- Possible Future Outlooks

The future of UML and UML books is not too clear. With the wider popularity of lean software development practices (e.g., Agile and Scrum), some practitioners advocate against building heavy-up-front design documents which sometimes include far too comprehensive UML models. However, there are also many in the Agile community who encourage light use of UML. Some practitioners and researchers such as Carl Zetie [25] believe that UML's popularity might decrease by time. This is also partly the author's experience after having many informal discussions with many practitioners and researchers.

The numbers for the latest years in the histogram of the number of UML books in Figure 2 are somewhat supporting the above prediction. However, the new modeling notations branched from the UML, e.g., SysML, are also starting to gain popularity and we need to wait several years to see what occurs.

6 CONCLUSIONS AND FUTURE WORKS

The classification, trend analysis and statistical survey reported in this article provided a first attempt towards categorizing the UML books published during the 12 years after its inception (1997-2009). The study also shed light on the quantity of books published in each subject areas of UML. We were able to mine interesting findings from the data provided by Google Books database [1], such as the Hype cycle analysis (Section 5.1).

As a future work direction, analysis of overlap (commonalities) and differences among different UML books would be interesting and useful. It is also important to find out which sub-areas of the UML body of knowledge need further new books. This would help authors and publishers in planning for new books on the topic. The author further believes that conducting surveys with students, researchers and practitioners on existing UML books and also inquiring about needs for future UML books would be a useful idea.

Another natural future work would be to extend the scope of the study so that it is useful for students and practitioners in choosing suitable UML books. Also the initial classification reported in this paper should be further extended in future works. Last but not least, the online Excel data file could be turned into an online searchable guide, for instance, with fields to guide the analysis of its data, maybe with an extensible database behind, which could accept the inclusion of new titles by online users.

ACKNOWLEDGEMENTS

This project was supported by the Discovery Grant no. 341511-07 from the Natural Sciences and Engineering Research Council of Canada (NSERC) and also by the Alberta Ingenuity New Faculty Award no. 200600673.

REFERENCES

- [1] Google, "Google Books database," *books.google.com*, Last accessed: April 2010.
- [2] G. Alleman, "UML Topics," <http://www.niwotridge.com/Books/UMLBooks.htm>, Last accessed: April 2010.
- [3] J. B. Surveyer, "UML Book Reviews," <http://www.theopensource.com/osbkuml.htm>, Last accessed: April 2010.
- [4] J. Conallen, *Building Web applications with UML*, 2nd ed.: Addison-Wesley Professional, 2002.
- [5] D. Rosenberg and K. Scott, *Use-Case Driven Object Modeling with UML: A Practical Approach*: Addison-Wesley Professional, 1999.

This is the post-print of a paper that has been published with the following DOI:
<http://dx.doi.org/10.1007/s10270-011-0189-9>

-
- [6] C. Larman, *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and the Unified Process*: Prentice Hall, 2001.
 - [7] D. Torpey, "Book review: Applying UML and Patterns, Third Edition," <http://www.ibm.com/developerworks/rational/library/nov05/reader/torpey.html>, Last accessed: April 2010.
 - [8] B. Dobing and J. Parsons, "How UML is used," *Communications of the ACM*, vol. 49, no. 5, pp. 109-113, 2006.
 - [9] F. J. Lucas, F. Molina, and Ambrosio Toval (organizers), "A systematic review of UML model consistency management," *Information and Software Technology*, vol. 51, no. 12, pp. 1631-1645, 2009.
 - [10] V. R. Basili, "Software modeling and measurement: the Goal/Question/Metric paradigm," Technical Report, University of Maryland at College Park 1992.
 - [11] Google, "New chapter for Google Book Search," <http://googleblog.blogspot.com/2008/10/new-chapter-for-google-book-search.html>, Last accessed: April 2010.
 - [12] M. A. Weisfeld, *The Object-oriented Thought Process*: Sams Publishing, 2004.
 - [13] B. McLaughlin, G. Pollice, and D. West, *Head first Object-oriented Analysis and Design*: O'Reilly, 2006.
 - [14] P. Lavin, *Object-oriented PHP: Concepts, Techniques, and Code*: No Starch Press, 2006.
 - [15] D. Poo, D. Kiong, and S. Ashok, *Object-Oriented Programming and Java*: Springer, 2007.
 - [16] M. Scott and K. Fowler, *UML distilled: applying the standard object modeling language*: Addison-Wesley, 1997.
 - [17] P.-A. Muller, *Instant UML*: Wrox Press, 1997.
 - [18] R. C. Lee and W. M. Tepfenhart, *UML and C++: a practical guide to object-oriented development*: Prentice Hall, 1997.
 - [19] P. P. Texel and C. B. Williams, *Use cases combined with BOOCH/OMT/UML: process and products*: Prentice Hall, 1997.
 - [20] J. Fenn, "When to Leap on the Hype Cycle," *Technical Report*, Gartner Inc., 1995.
 - [21] Gartner Inc., "Understanding Hype Cycles," <http://www.gartner.com/pages/story.php.id.8795.s.8.jsp>, Last accessed: April 2010.
 - [22] Gartner Inc., "Gartner's 2009 Hype Cycle Special Report Evaluates Maturity of 1,650 Technologies, Press Release," <http://www.gartner.com/it/page.jsp?id=1124212>, Last accessed: April 2010.
 - [23] B. Willard, "UML for systems engineering," *Computer Standards & Interfaces*, vol. 29, no. 1, pp. 69-81, 2007.
 - [24] Gartner Group, "Hype Cycle for Emerging Technologies, 2006," <http://www.gartner.com/DisplayDocument?id=494180>, Last accessed: April 2010.
 - [25] C. Zetie, "MDA Is DOA, Partly Thanks To SOA," http://www.forrester.com/rb/Research/mda_is_doa_partly_thanks_to_soa/q/id/39156/t/2, Last accessed: April 2010.
 - [26] S. R. Schach, *Classical and Object-Oriented Software Engineering with UML & C++*: McGraw-Hill, 1999.
 - [27] J. Sturm, *VB6 UML Design and Development*: Peer Information Inc., 1999.
 - [28] C. T. Arrington and S. H. Rayhan, *Enterprise Java with UML*, 2nd ed.: Wiley, 2003.
 - [29] A. Filev, T. Loton, K. McNeish, B. Schoellmann, J. Slater, and C. G. Wu, *Professional UML Using Visual Studio.Net: Unmasking Visio for Enterprise Architects* John Wiley & Sons, 2002.

This is the post-print of a paper that has been published with the following DOI:
<http://dx.doi.org/10.1007/s10270-011-0189-9>

- [30] C. Grimm, *Languages for system specification: selected contributions on UML, SystemC*: Springer, 1996.
- [31] P. Baker, Z. R. Dai, J. Grabowski, Ø. Haugen, I. Schieferdecker, and C. Williams, *Model-Driven Testing: Using the UML Testing Profile*: Springer, 2007.
- [32] H.-G. Gross, *Component-based software testing with UML*: Springer, 2004.
- [33] B. Unhelkar, *Verification and validation for quality of UML 2.0 models*: Wiley, 2005.
- [34] D. J. Duffy, *Domain architectures: models and architectures for UML applications*: Wiley, 2004.
- [35] J. Garland and R. Anthony, *Large-scale software architecture: a practical guide using UML*: Wiley, 2002.
- [36] J. M. Sekula and P. W. Sheridan, *Iterative UML development using Visual Basic 5.0*: Wordware Publishing, 1998.
- [37] B. Oestereich, *Developing software with UML: object-oriented analysis and design in practice*, 2nd ed.: Addison-Wesley Professional, 2002.
- [38] B. P. Douglass, *Real time UML: advances in the UML for real-time systems*, 3rd ed.: Addison-Wesley Professional, 2004.
- [39] E. J. Naiburg and R. A. Maksimchuk, *UML for database design*: Addison-Wesley Professional, 2001.
- [40] G. Martin and W. Müller, *UML for SOC design*: Springer, 2005.
- [41] P. J. Papajorgji and P. M. Pardalos, *Software Engineering Techniques Applied to Agricultural Systems: An Object-Oriented and UML Approach*: Springer, 2005.
- [42] J. Jürjens, *Secure Systems Development with UML*: Springer, 2004.
- [43] R. B'Far and R. T. Fielding, *Mobile Computing Principles: Designing and Developing Mobile Applications with UML and XML*: Cambridge University Press, 2004.
- [44] T. Quatrani and J. Palistrant, *Visual Modeling with IBM Rational Software Architect and UML*: IBM Press, 2006.
- [45] J. Arlow and I. Neustadt, *UML 2 and the unified process: practical object-oriented analysis and design*: Addison-Wesley Professional, 2005.
- [46] J. B. Warmer and A. G. Kleppe, *The object constraint language: precise modeling with UML*: Addison-Wesley Professional, 1998.
- [47] J. Arlow and I. Neustadt, *Enterprise Patterns and MDA: Building Better Software with Archetype Patterns and UML*: Addison-Wesley Professional, 2004.
- [48] S. W. Ambler, *The object primer: agile model-driven development with UML 2.0*: Cambridge University Press, 2004.
- [49] J. Ginsberg, M. H. Mohebbi, R. S. Patel, L. Brammer, M. S. Smolinski, and L. Brilliant, "Detecting influenza epidemics using search engine query data," *Nature*, vol. 457, pp. 1012-1014, 2009.
- [50] G. K. Webb, "Internet Search Statistics as a Source of Business Intelligence," *Issues in Information Systems*, no. 2, pp. 82-87, 2009.
- [51] A. Baram-Tsabari and E. Segev, "Exploring New Web-based Tools to Identify Public Interest in Science," *Public Understanding of Science*, no. 1, pp. 1-14, 2009.
- [52] J. Rech, "Discovering Trends in Software Engineering with Google Trend," *ACM SIGSOFT Software Engineering Notes*, vol. 32, no. 2, pp. 1-2, 2007.
- [53] IBM Collaborative User Experience Research Group, "About Many Eyes," <http://manyeyes.alphaworks.ibm.com/manyeyes/page/About.html>, Last accessed: April 2010.

This is the post-print of a paper that has been published with the following DOI:
<http://dx.doi.org/10.1007/s10270-011-0189-9>

- [54] V. Garousi, "Visualization data sets on the ManyEyes online system," <http://manyeyes.alphaworks.ibm.com/manyeyes/users/Garousi>, Last accessed: April 2010.
- [55] M. Grand, *Patterns in Java: a catalog of reusable design patterns illustrated with UML*: Wiley, 1998.
- [56] A. Goucher, "How They Test Software At Microsoft," <http://adam.goucher.ca/?p=1202>, Last accessed: April 2010.
- [57] A. Tanaka, *The Contextualized UML: Practical Use of Unified Modeling Language in Japanese Industry*: Authorhouse, 2004.
- [58] T. Weilkiens and B. Oestereich, *UML 2 certification guide: fundamental and intermediate exams*: Morgan Kaufmann, 2006.
- [59] Various Authors, "Object Technology Book Series " *Addison-Wesley*, http://www.informit.com/imprint/series_detail.aspx?ser=335183, Last accessed: April 2010.
- [60] W. Boggs and M. Boggs, *Mastering UML with Rational Rose*: Sybex, 1999.
- [61] J. E. Hansen and C. Thomsen, *Enterprise development with Visual Studio .NET, UML, and MSF*: Apress, 2004.
- [62] T. Pender, *UML Bible*: Wiley, 2003.