# What Are the Used UML Diagram Constructs? A Document and Tool Analysis Study covering Activity and Use Case Diagrams

Gianna Reggio, Maurizio Leotta, Filippo Ricca, Diego Clerissi

**Abstract:**

UML offers a very large set of constructs for each of its diagram types, however many of them seem scarcely used or even their existence is not known. Here, we decided to present a precise view of the usage levels of the constructs of activity and use case diagrams by means of a document and tool analysis study, covering preliminarily: books, courses, tutorials, and tools about UML. Results of the study show that, among the 47 activity diagrams constructs, a large majority of them seem to be scarcely used, while, only nine result widely used, whereas only two of the nine constructs of the use case diagrams seem scarcely used. This work is part of a larger project aimed at investigating the usage level of the UML diagrams and their constructs, also by means of a personal opinion survey intended for UML users.

**Digital Object Identifier (DOI):**

**http://dx.doi.org/10.1007/978-3-319-25156-1_5**

# What Are the Used UML Diagram Constructs? A Document and Tool Analysis Study covering Activity and Use Case Diagrams

Gianna Reggio, Maurizio Leotta, Filippo Ricca, Diego Clerissi

DIBRIS, Università di Genova, Italy
gianna.reggio@unige.it, maurizio.leotta@unige.it,
filippo.ricca@unige.it, diego.clerissi@gmail.com

**Abstract.** UML offers a very large set of constructs for each of its diagram types, however many of them seem scarcely used or even their existence is not known. Here, we decided to present a precise view of the usage levels of the constructs of activity and use case diagrams by means of a document and tool analysis study, covering preliminarily: books, courses, tutorials, and tools about UML. Results of the study show that, among the 47 activity diagrams constructs, a large majority of them seem to be scarcely used, while, only nine result widely used, whereas only two of the nine constructs of the use case diagrams seem scarcely used. This work is part of a larger project aimed at investigating the usage level of the UML diagrams and their constructs, also by means of a personal opinion survey intended for UML users.

## 1 Introduction

UML is a truly large notation offering many different diagrams, 14 in the last approved version [24], and for each diagram it provides a large set of constructs covering any possible need of any modeller for any possible task. As a result, the UML specification is a huge book, the UML metamodel is large and quite complex, and the definition and the understanding of its static and dynamic semantics is a truly difficult task, with also the consequence to make difficult to teach it both at the school/university level or in the industry [5]. Moreover, the large number of constructs and the consequent very large metamodel make complex and time consuming developing transformations of UML models and building tools for the UML. Clearly, these UML features have a negative impact on how the UML is perceived by the modellers hindering its adoption [13], and leading in some cases to replace the UML by ad-hoc Domain Specific Languages (DSLs).

On the other hand, users naturally tend to consider and use only a portion of its diagrams/constructs, and forgetting about some other ones. On his blog I. Jacobson states "For 80% of all software only 20% of UML is needed" [7]. Furthermore, few people try to learn the UML by reading its specification [24], instead the large majority of the users rely on books and courses/tutorials or just start to use some tools for drawing UML diagrams that in general do not cover the whole UML. For this reason, in many cases, the UML users will never become aware of the existence of many specific constructs

(e.g., how many of you do know the existence of the "Parameter Set" for activities or has even used this construct?).

We would like to asses by means of a document and tool analysis study which parts of the UML (diagrams and constructs) are the most used in practice and which are the less ones (and thus also indirectly which part of the UML is really known), using again the words of Jacobson trying to see if an "essential UML" [7] emerges. To discover how much a UML diagram/construct is used, we chose to preliminarily investigate objective sources: (1) the books about the UML, (2) the IT University courses covering also the UML, (3) the tutorials presenting the UML to the practitioners, and (4) the tools for producing UML models. Moreover, similarly to [3, 5], we are conducting a personal opinion survey [16] asking to UML users of different kinds (e.g., industrial practitioners and academics) which parts of the UML they know and which they have never used.

For a given UML diagram/construct, we have proceeded as follows. We have investigated the books to discover if they were citing such diagram/construct, and if they were giving an example of it. Similarly for the courses/tutorials, whereas for the tools we have tried to produce a model containing such diagram/construct (notice that we speak of a model not of a diagram, since there are several UML constructs that cannot be shown in a diagram). Finally, we have computed descriptive statistics to present the results. In this study, we analysed 30 books, 20 tools, 22 courses, and 18 tutorials.

The dimension of the UML prevented us to investigate the usage level of the constructs of all the 14 diagram types (for example a personal survey covering all diagrams would require at least one hour to be filled), thus we considered only the constructs of activity and use case diagrams, chosen because both are quite known and used, and the former has a large number of constructs, whereas the latter has only few constructs.

The results of this document and tool analysis study, and of the future personal opinion survey should be of help to many different categories of people:

– **teachers and instructors**: allowing to offer courses and/or tutorials concentrating only a smaller language made out of the most used UML diagrams/constructs;
– **tool builders/users**: obvious advantages since the tools covering the most used diagrams/constructs will be simpler to implement/use;
– **notation designers**: interested in discovering scarcely used constructs, and understanding for which reasons they have been added to the language. Moreover, other interesting questions arise: are the scarcely used constructs derived[1] or primitive? Can the scarcely used constructs be applied only in specific cases? It will be interesting to investigate whether the metamodel (and subsequently the UML specification book) may be easily simplified to cover only the most used constructs.

In our opinion, handling a notation with a large set of constructs where a portion of them are scarcely used, if not almost unknown, is problematic because it can cause a waste of effort and resources by who want/must use it (e.g., there are countries where some contracts with the public administration must be accompanied by a UML model). Indeed, trivially, to print the reference document requires 700 sheets, but also understanding the metamodel/preparing for the certification/deciding what to teach to the students/reading a UML book require a large number of hours; and we do not have

[1] A derived construct may be replaced by a combination of other constructs.

to forget that also maintaining the official specification and any related item requires a large amount of effort due to its size.

Also the OMG has recently recognised the need to simplify the UML with the initiative "UML Simplification" [21] which will result in the next UML version (2.5), but in this case the simplification concerns only the way UML is defined without any impact about its constructs.

In a previous phase of this document and tool analysis study [18], we analysed the usage level of the 14 types of UML diagrams; in this paper, we present the results of a following step of our work focusing on the usage level of the activity and use case diagram constructs covering books, courses/tutorials, and tools.

The remainder of the paper is organized as follows. In Sect. 2, we present related work literature regarding empirical study about the UML. In Sect. 3, we sketch the relevant aspects of the conducted document and tool analysis study such as: goals, research questions, followed process and analysis methodology. The results of the document and tool analysis study about activity and use case diagram's constructs are presented in Sect. 4, preceded by a summary of the results on the UML diagrams shown in [18], while threats to validity are discussed in Sect. 5. Finally, Sect. 6 concludes the paper.

## 2 Related Work

The systematic literature review by Mohagheghi et al. [11] about model-based software development states that "the UML is currently the most widely used modelling language". A similar result has also been obtained in [23] where a personal opinion survey with 155 Italian professionals has been conducted, while in [20] emerges that UML is often employed by companies in the software analysis and design phases.

Another personal opinion survey about UML (171 professionals in total), by Dobing and Parsons [3], points out another strong statement: "regular usage of UML was lower than expected". The authors of [3] suggest that the difficulty of understanding many of the notations supports the argument that the UML may be too complex. The same claim, in more or less different forms, is present in several blogs, where several proposals of UML simplification are arising[2]. Maybe, the most authoritative is the one of Ivar Jacobson entitled "Taking the temperature of UML" [7], where he wrote: "Still, UML has become complex and clumsy. For 80% of all software only 20% of UML is needed. However, it is not easy to find the subset of UML which we would call the 'Essential' UML. We must make UML smarter to use". The need to simplify the UML is also shown by the recently released OMG draft proposal about this topic [21]. Moreover, the complexity of the UML seems to be one of the factors that limit its diffusion and usage in the industry [13]: "UML is considered unnecessarily complex" and understanding its notation could require a considerable overhead.

In the tentative to find the "essential UML", Erickson and Siau [4] have conducted a Delphi study[3] with the goal of identifying a UML kernel for three well-known UML application areas: Real-Time, Web-based, and Enterprise systems. The participants to

[2] e.g., www.devx.com/architect/Article/45694 and blogs.msdn.com/b/sonuarora/archive/2009/11/02/simplify-uml.aspx

[3] It attempts to form a reliable consensus of a group of experts in specialized areas.

the study (44 experts in total) were asked to rate the relative importance of the various UML diagrams in building systems. UML overall results (i.e., non-domain specific) were: 100% for Class and Statechart diagrams, 95.5% for Sequence diagrams, 90.9% for Use Case diagrams. All the others diagrams received a percentage lesser than 50% (e.g., 27.3% for Activity diagrams). Another personal opinion survey about UML [5] with 131 subjects confirms the results of Erickson and Siau. Results indicate that the three most important diagrams are Use Case diagrams, Class diagrams and Sequence diagrams.

The main conclusions from another systematic literature review by Budgen et al. [1] about empirical evidence of the UML are two:

– while there are many studies that use the UML in some way, including to assess other topics, there are relatively few for which the UML is itself the object of study, and hence that assess the UML in some way (e.g., UML studies of adoption and use in the field).
– there is a need to study the UML and its elements much more rigorously and to identify which features are valuable, and which could be discarded.

Our preliminary empirical work, much in the spirit of the Erickson and Siau's work but using a different approach, tries to add a small tile to the empirical knowledge about the UML as requested in the first conclusion of the Budgen's systematic literature review. We could say that our work tries to identify which UML features (diagrams and constructs) are valuable, and which could be discarded — as requested in the second point of the Budgen's work — equating the word "valuable" with "used in practice" and the concept "could be discarded" with "not used in practice".

## 3 Study Definition

The instrument we selected to take a snapshot of the state of the practice concerning UML usage is that of a document and tool analysis study [14]. In the work's design and execution phases we followed as much as possible the guidelines provided in [6] and used the same presentation format of [22].

The document and tool analysis study has been conducted through the following steps: (1) goals selection, (2) goals transformation into research questions, (3) identification of the population, sample and process, (4) data extraction, and (5) analysis of results and packaging.

We conceived and designed the document and tool analysis study with the **goal** of understanding *which are the less/most used parts of the UML in practice*.

Within the scope of this work, in this paper we aim at addressing two **research questions** related to the above described goal:

**RQ1:** Which UML constructs for the activity diagrams are the most/less used in practice?

**RQ2:** Which UML constructs for the use case diagrams are the most/less used in practice?

### 3.1 Population Identification

The first step to conduct a document and tool analysis study is defining a target population. The target population of our study consists of *sources* concerning UML. In particular,

in this study we considered the following four kinds of objective sources: books, tools, courses and tutorials. Currently, we are conducting a personal opinion survey [16] with industrial practitioners and academics to understand which parts of the UML they know, which they use, and which they have never used.

To sample the population and select the sources to consider in our study we: (1) conducted a systematic search performed using Internet resources, Web search engines and electronic databases and (2) used non-probabilistic (convenience sampling) methods [8]. Moreover, in making decisions about whether (or not) to include a source in the study, we adopted some well-defined inclusion/exclusion criteria (see below).

**Inclusion and Exclusion Criteria** The inclusion/exclusion criteria can be common for all the kind of sources or specific. For all the sources we adopted the following inclusion criterion: only sources concerning UML versions $\geq 2.0$.

Concerning books, in case of different editions of the same book we opted (when possible) for the last one. Moreover, we excluded elements of "grey" literature, i.e., books without ISBN.

Concerning tools, we included only UML modelling tools (both commercial and non-commercial) and excluded: (1) general graphics editor (e.g., Inkscape), (2) tools providing only a specific type of diagram (e.g., class diagrams), (3) really unstable, not complete or preliminary tools (e.g., tools in beta version).

About courses, we considered only university courses concerning IT studies. We considered courses offered also in languages different from English but known or understood by the authors (e.g., French, Italian and Spanish).

Concerning tutorials, we considered only tutorials provided on Internet as written documents (either on-line or downloadable) and video (where a person gives instructions on how to do something) but we have excluded tutorials taking the form of a screen recording (screencast) and interactive tutorials. For selecting a document of this kind we used the common meaning/perception of tutorials: a tutorial is more interactive and specific than a book or a lecture; a tutorial seeks to teach by example.

### 3.2 The Process

The process followed to conduct a document and tool analysis study should be as much as possible well defined in order to ensure that such a study can be objective and repeatable. For each category of sources, we followed a different process to collect them.

**Books.** We started by the Amazon website and used the search form to find UML related books. We selected the "Computers & Technology" category in the books section. Then, we experimented with several different search criteria using different combinations of strings. Finally, the one that retrieved the highest number of useful items was the simple string "UML 2". Starting from this long list of books ordered by relevance (2.726 books on July 20, 2013) we filtered out books not satisfying the inclusion criteria explained above. Then, we tried to recover them using the electronic facilities provided by our library. Finally, we collected and analysed 30 books. Note that, 18 of them are in the top 24 books ordering the Amazon list by relevance. The list of the selected books is shown in Fig. 1. More in detail, we have selected two kinds of books. UML guides

| | Title | Edition | Author(s) | Year | Publisher |
|---|---|---|---|---|---|
| UML Notation Guides | UML 2.0 in a Nutshell | 1st | Pilone, Pitman | 2005 | O'Reilly Media Inc. |
| | The Elements of UML 2.0 Style | 1st | Ambler | 2005 | Cambridge University Press |
| | Sams Teach Yourself UML in 24 Hours | 3rd | Schmuller | 2004 | Sams Publishing |
| | UML 2 Certification Guide: Fundamental & Intermediate Exams | 1st | Welkiens, Oestereich | 2006 | Morgan Kaufmann Publishers |
| | UML Distilled: A Brief Guide to the Standard Object Modeling Language | 3rd | Fowler | 2003 | Addison-Wesley |
| | Learning UML 2.0 | 1st | Miles, Hamilton | 2006 | O'Reilly Media Inc. |
| | UML 2 for Dummies | 1st | Chonoles, Schardt | 2003 | Wiley Publishing Inc. |
| | UML 2 Toolkit | 2nd | Eriksson, Penker, Lyons, Fado | 2004 | Wiley Publishing Inc. |
| | UML 2.0 in Action | 1st | Grassle, Baumann, Baumann | 2005 | Packt Publishing Ltd |
| | UML Bible | 1st | Pender | 2003 | Wiley Publishing Inc. |
| | UML Demystified | 1st | Kimmel | 2005 | McGraw-Hill |
| | UML for the IT Business Analyst | 1st | Podeswa | 2005 | Muska & Lipman Pub |
| | Verification and Validation for Quality of UML 2.0 Models | 1st | Unhelkar | 2005 | John Wiley & Sons |
| | The Unified Modeling Language Reference Manual | 2nd | Rumbaugh, Jacobson, Booch | 2005 | Addison-Wesley |
| | The Unified Modeling Language User Guide | 2nd | Booch, Rumbaugh, Jacobson | 2005 | Addison-Wesley |
| Software Engineering books based on UML | Object-Oriented Software Engineering Using UML, Patterns and Java | 3rd | Bruegge, Dutoit | 2010 | Prentice Hall |
| | System Analysis & Design with UML version 2.0: An Object-Oriented Approach | 3rd | Dennis, Wixom, Tegarden | 2009 | John Wiley & Sons |
| | UML 2 and the Unified Process: Practical Object-Oriented Analysis & Design | 2nd | Arlow, Neustadt | 2005 | Addison-Wesley |
| | UML 2 Semantics and Applications | 1st | Lano | 2009 | John Wiley & Sons |
| | Object-Oriented Analysis & Design: Understanding System Development with UML 2.0 | 1st | O'Docherty | 2005 | John Wiley & Sons |
| | Using UML: Software Engineering with Objects and Components | 2nd | Stevens, Pooley | 2006 | Addison-Wesley |
| | UML 2 Pour les bases de donnees | 1st | Soutou | 2007 | Éditions Eyrolles |
| | Fast Track UML 2.0 | 1st | Scott | 2004 | Apress Media LLC |
| | Model-Driven Development with Executable UML | 1st | Milicev | 2009 | Wiley Publishing Inc. |
| | Professional Application Lifecycle Management with Visual Studio 2010 | 1st | Gousset, Keller, Krishnamoorthy, Woodward | 2010 | Wiley Publishing Inc. |
| | Software Modeling and Design | 1st | Gomaa | 2011 | Cambridge University Press |
| | Systems Engineering with SysML UML: Modeling, Analysis, Design | 1st | Weilkiens | 2006 | Morgan Kaufmann Publishers |
| | Use Case Driven Object Modeling with UML: Theory and Practice | 1st | Rosenberg, Stephens | 2007 | Apress Media LLC |
| | Management of The Object-Oriented Development Process | 1st | Liu, Roussev | 2006 | Idea Group Inc. |
| | Real-Time Object Uniform Design Methodology with UML | 1st | Duc | 2007 | Springer |

**Fig. 1.** UML Books Considered

(*UML Notation Guides*) and books using UML as a notation, i.e. books where UML is not the primary subject (*Software Engineering books based on UML*).

**Tools.** We started by the "List of Unified Modeling Language tools" Wikipedia page[4] containing 49 UML tools. Then, we considered also the UML-tools website[5]. A full Internet search was also carried out using Google. Also in this case, we experimented with several different search criteria using different combinations of strings to provide to Google ("UML tools", "UML tools list" and "UML free tools").

For each tool of our list, we found the official website and checked whether it was satisfying the inclusion criteria explained above. Then, we downloaded and installed the most recent version of all the selected tools. In case of commercial tools, we selected a "free for not commercial use" version or a version with university licence or a trial version. At the end, we collected and analysed 20 different tools. The complete list of tools is shown in Fig. 2. Finally, we tried to produce a model containing the diagrams and constructs of interest for our study (for each tool we produced the same model with the same diagrams and the same UML constructs). Notice that Argo UML, one of the most known UML tool, was not included in our document and tool analysis study since it does support UML 1.x only.

[4] en.wikipedia.org/wiki/List_of_Unified_Modeling_Language_tools

[5] www.uml-tools.com

| Name | Release | Year | Licence | Web Site |
|---|---|---|---|---|
| Altova Umodel | | 2012 | Commercial (Enterprise – Trial) | www.altova.com/umodel.html |
| Artisan Studio | 7.4 | 2012 | Commercial (Trial) | www.atego.com/products/artisan-studio/ |
| Astah | 6.6 | 2012 | Commercial (Community Edition) | astah.net/ |
| Borland Together | 12.0 | 2012 | Commercial (Trial) | www.borland.com/products/together/ |
| BOUML | 6.4.3 | 2013 | Commercial (Viewer - Limited) | www.bouml.fr/ |
| Enterprise Architect | 10 | 2013 | Commercial (Trial 30 days) | www.sparxsystems.eu/enterprisearchitect/ |
| IBM Rational Rhapsody Modeler | 7.5 | 2009 | Free | www-01.ibm.com/software/awdtools/modeler/ |
| IBM Rational SW Architect | 8.5.1 | 2012 | Commercial (Trial 30 days) | www-01.ibm.com/software/awdtools/swarchitect/ |
| MagicDraw | 17.0.3 | 2012 | Commercial (Enterprise – Trial) | https://www.magicdraw.com/ |
| Metamill | 6.1 | 2012 | Commercial (Trial) | www.metamill.com/ |
| Modelio | 2.2.1 | 2012 | Free | sourceforge.net/projects/modeliouml/ |
| Open Modelsphere | 3.2 | 2012 | Free | www.modelsphere.org/ |
| Papyrus | 0.9.1 | 2012 | Free (Eclipse Plug in) | www.eclipse.org/papyrus/ |
| Poseidon for UML | 8 | 2009 | Commercial (Community Edition) | www.gentleware.com/ |
| Power Designer | 16.1 | 2012 | Commercial (Trial) | www.sybase.com/products/ |
| RedKoda | 3.0.7 | 2012 | Commercial (Community Edition) | www.redkoda.com/ |
| Software Ideas Modeler | 5.82 | 2013 | Free | www.softwareideas.net/ |
| StarUML | 5.0.2.1570 | 2006 | Free | staruml.sourceforge.net/ |
| Violet | 0.21.1 | 2007 | Free | sourceforge.net/projects/violet/ |
| Visual Paradigm | 10.1 | 2013 | Commercial (Community Edition) | www.visual-paradigm.com/product/vpuml/ |

**Fig. 2.** UML Tools Considered

**Courses.** We started carrying out a search using Google. The combinations of strings used were: “UML course”, “UML lecture” and “UML university course”. We found several university courses satisfying the inclusion criteria stated above, but in several cases it was difficult, if not impossible, to recover the slides of the lectures, and in general the material. Often, the material was not publicly available; only the content of the lessons was present on the website. For this reason, we resort also to convenience sampling, asking to our colleagues the slides of UML courses they teach. At the end, we collected and analysed 22 different University courses. The complete list of lectures is shown in Fig. 3. Convenience sampling was also useful to balance a little the geographic origin of the UML courses (e.g., before convenience sampling we had three courses from France and zero from USA).

| Lecturer | Country | Title | Year |
|---|---|---|---|
| Afsarmanesh | Netherlands | Project Analysis | 2012 |
| De Angelis | Italy | Lab. Ingegneria del SW | 2012/13 |
| Ciancarini, Iorio | Italy | Lab. Ingegneria del SW | 2012/13 |
| Vincent | Australia | System Analysis and Modeling | 2012/13 |
| Casalicchio | Italy | Progettazione SW | 2009/10 |
| Gérard | France | UML | |
| Prié | France | Systèmes d'information méthodes avancées | 2011/12 |
| Felici | UK | Software Engineering with Objects and Components | 2011/12 |
| Siebers | UK | Object Oriented Systems | 2012/13 |
| Varrò | Hungary | Modellalapú szoftvertervezés | 2012 |
| Lehre | Germany | Softwaretechnik | 2012/13 |
| Rumpe | Germany | Modellbasierte Softwareentwicklung | 2011/12 |
| Correo, Rossi | Argentina | Uml Basico | |
| Brambilla | Italy | Ingegneria del SW | 2012/13 |
| Alkan | Turkey | Object Oriented Software Engineering | 2012/13 |
| Farrow | UK | Software Engineering | 2012/13 |
| Easterbrook | Canada | Engineering Large SW Systems | 2012 |
| Negre | France | Ingéniérie des Systèmes d'Information | 2012/13 |
| Sellares | Spain | Enginyeria del Software | 2008/09 |
| Jezequel | France | Approche objet pour le développement de logiciels par objets avec UML | |
| Turgut | US | Software Engineering I | 2009 |
| Cheng | US | Advanced Software Engineering | 2013 |

**Fig. 3.** UML Courses Considered

| Author / Source | Title | Web Site |
|---|---|---|
| Allen Holub | Allen Holub's UML Quick Reference | www.holub.com/goodies/uml/index.html |
| Analisi-disegno | Introduzione a UML | www.analisi-disegno.com/uml/uml.htm |
| Crag Systems | A UML Tutorial Introduction | www.cragsystems.co.uk/uml_tutorial/index.htm |
| devmentor | UML Guide v2.1 | devmentor.org/references/uml/uml.php |
| Dumke | UML Tutorial | www-ivs.cs.uni-magdeburg.de/~dumke/UML/index.htm |
| Embarcadero | Practical UML: A Hands-On Introduction for Developers | edn.embarcadero.com/article/31863 |
| HTML.it | Guida UML | www.html.it/guide/guida-uml/ |
| John Deacon | Developer's Guide to UML 2: A UML Tutorial | www.johndeacon.net/UML/UML_Appendix/Generated/UML_Appendix.asp |
| lemiffe | Reference Guide for UML 2.0 | www.lemiffe.com/wp-content/uploads/2008/12/uml2.pdf |
| New Think Tank | Video Tutorials | www.newthinktank.com/2012/11/ |
| Online Teach | UML Training | www.online-teach.com/u-m-l.php |
| Parlezuml | UML Tutorial | www.codemanship.co.uk/parlezuml/ |
| Richard Botting | A Beginners Guide to The Unified Modeling Language (UML) | www.csci.csusb.edu/dick/cs201/uml.html |
| SmartDraw | What is UML? | www.smartdraw.com/resources/tutorials/uml-diagrams/ |
| Sparx Systems | UML 2 Tutorial | www.sparxsystems.com.au/resources/uml2_tutorial/index.html |
| Storrle & Knapp | Unified Modeling Language 2.0 | www.pst.ifi.lmu.de/veroeffentlichungen/UML-2.0-Tutorial.pdf |
| Uml.free | UML, le langage de modélisation objet unifié | uml.free.fr/index-cours.html |
| uml-diagrams | UML 2.5 Diagrams Overview | www.uml-diagrams.org/uml-25-diagrams.html |

**Fig. 4.** UML Tutorials Considered

**Tutorials.** We started with the tutorial lists present in three websites[6]. Then, we integrated the obtained results with other tutorials recovered using Google (the research was conducted using the strings: "UML tutorials" and "UML guide"). Finally, we collected and analysed 18 tutorials. The complete list of tutorials is shown in Fig. 4.

## 4 Results of the Document and Tool Analysis Study

After having collected the sources, we extracted the data of interest for our research questions and finally we performed the analysis. Given the nature of this document and tool analysis study, that is mainly descriptive (it describes some condition or factor found in a population in terms of its frequency and impact) and exploratory, we mainly applied descriptive statistics and showed our findings by means of charts.

We preliminarily decided to interpret the results of our survey assuming that a diagram/construct is *widely used* if it is present in the 60% or more of the sources, similarly it is *scarcely used* if it is present in the 40% or less of the sources, having also some non-defined cases (*grey zone*). In the following subsections we briefly summarize the results concerning UML diagrams, see [18] for the details, and present the ones for the activity and the use case diagram constructs.

### 4.1 Level of Usage of the UML Diagrams

The level of usage of the various UML diagrams in books, courses, tutorials, tools, and in the totality of the sources respectively is summarized in Fig. 5.

If we consider the totality of the sources, disregarding their kind, we have that the scarcely used diagrams are timing, interaction overview and profile, listed starting from

[6] www.uml.org/#Links-Tutorials, stackoverflow.com/questions/1661961/recommended-uml-tutorials, and www.jeckle.de/umllinks.htm#tutorials

the most used; all of them were not present in UML 1.x, and the profile diagram appeared only in version 2.2. The last position of the profile diagram is not very surprising due both to the late appearance and to the fact that this kind of diagram has a very restrict scope (indeed it is used only to present a profile) and that, it is essentially a variant of the package diagram. Also timing diagrams have a restrict scope, and UML offers other ways to model time related aspects (e.g., timed events may be used in state machines and activity diagrams; durations and time intervals may appear in sequence diagrams), and this may be the motivation for their low usage. Finally, interaction overview diagrams are quite complex and in many cases may be replaced by sequence diagrams and/or a combination of sequence and activity diagrams, and perhaps this is the reason for not being so considered.

The widely used diagrams, when considering the totality of the sources, are instead, listed again starting from the most used ones, class (100% in any kind of sources), activity, sequence, state machine, use cases, communication, deployment, component, object and package diagrams. The first position of class diagrams is not surprising; it is indeed the main building block of the UML, while the fact that activity diagrams are the second is relevant and is due, in our opinion, to the fact that they are used also for business process modelling [17] and for describing SOA based systems [10, 9]. All the widely used diagrams, except the package diagram, were already present in the UML 1.x versions (although the communication diagrams were before called collaboration diagrams).

The only diagram in the grey area (i.e., above 40% and below 60%) is the composite structure, which allows to represent both structured classes and collaborations; again it is a new diagram appearing in the UML 2.0 and this may be a reason for its low usage. However, the result is surprising because structured classes were completely absent in UML 1.x, and this was a perceived problem, and the new collaborations are truly useful (see for example the big role that they have in representing service oriented architectures in the SoaML OMG standard profile [12]). For a more complete analysis of the results on UML diagrams and a deeper discussion see [18].

| UML Diagram | Book Guide | Book Spec | Book Tot | Tool | Course | Tutorial | All Sources |
|---|---|---|---|---|---|---|---|
| Class | 100% | 100% | 100% | 100% | 100% | 100% | **100%** |
| Activity | 100% | 93% | 97% | 100% | 95% | 100% | **98%** |
| Sequence | 100% | 93% | 97% | 100% | 100% | 89% | **97%** |
| Use Case | 100% | 93% | 97% | 100% | 95% | 89% | **96%** |
| State Machine | 100% | 93% | 97% | 100% | 95% | 89% | **96%** |
| Communication | 100% | 80% | 90% | 90% | 59% | 89% | **82%** |
| Component | 93% | 80% | 87% | 85% | 59% | 89% | **80%** |
| Deployment | 93% | 80% | 87% | 90% | 55% | 89% | **80%** |
| Object | 93% | 80% | 87% | 70% | 55% | 67% | **71%** |
| Package | 100% | 79% | 89% | 65% | 52% | 67% | **70%** |
| Composite Structure | 87% | 60% | 73% | 80% | 14% | 33% | **52%** |
| Timing | 87% | 53% | 70% | 40% | 5% | 33% | **40%** |
| Interaction Overview | 80% | 53% | 67% | 45% | 5% | 28% | **39%** |
| Profile | 7% | 13% | 10% | 30% | 0% | 6% | **11%** |

Book Guide = UML Notation Guides, Book Spec = Software Engineering books based on UML, Book Tot = All books Dashed lines represent the 40% and 60% thresholds

**Fig. 5.** Usage levels of UML diagrams

### 4.2 Level of Usage of the Activity Diagram Constructs

The number of available constructs in activity diagrams is 47, a truly large number, and as it is shown in Fig. 8 a lot of them are scarcely used (their description and some examples may be found in the UML specification [24]).

If we consider the totality of our sources disregarding the fact that they are of different kind, we have that the widely used constructs are only nine, precisely Action and Control Flow Edge (both 100%), Initial and Final Node (99%), Decision/Merge Nodes (98%), Fork/Join Nodes (95%), Activity Partition, i.e., swimlanes (83%), Object Node (80%), and Object Flow Edge (76%)[7].

Those scarcely used, i.e., with a percentage less than 40% are **31**, and in 8 cases the percentage is lower than 10%. One of the least used constructs Parameter Set is shown in Fig. 6, it is used to provide alternative sets of inputs or outputs that a behaviour, and thus an activity, may use.

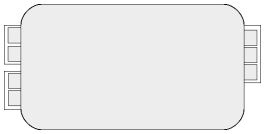
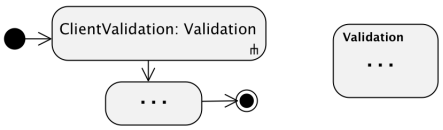


**Fig. 6.** Activity parameter set, usage level 10% in the totality of the sources

The fact that Activity is in the grey area is a little surprising since Activity is the construct that, together with Call Behaviour Action (rake), allows to structure complex activity diagrams, also if we suspect that in some cases Activity is considered but not precisely indicated with its proper name. See an example of Call Behaviour Action (rake) in Fig. 7.




*(The first action of the activity diagram on the left is a call to the behaviour defined elsewhere by the activity Validation)*

**Fig. 7.** Call Behaviour Action (Rake) construct, usage level 39% in the totality of the sources

We briefly discuss two of the least used constructs (resulting used only in 3% of the sources): Input and Output Effects for Object Node and Decision Node with Input Flow. The semantics of the first one is not very clear "*Specifying the effect that the behaviour of actions has on the objects passed in and out of their parameters can be*

[7] This surprising result, since it is not possible to connect an object node to other nodes without using the object flow, is due to the fact that in some courses the classification of the arcs in activity diagrams in control and object flow was not mentioned.

| Activity Diagram Construct | Book Guide | Book Spec | Book Tot | Tool | Course | Tutorial | All Sources |
|---|---|---|---|---|---|---|---|
| Action | 100% | 100% | 100% | 100% | 100% | 100% | **100%** |
| Control Flow Edge | 100% | 100% | 100% | 100% | 100% | 100% | **100%** |
| Final Node | 100% | 100% | 100% | 100% | 100% | 94% | **99%** |
| Initial Node | 100% | 100% | 100% | 100% | 100% | 94% | **99%** |
| Decision/Merge Nodes | 100% | 93% | 97% | 100% | 100% | 94% | **98%** |
| Fork/Join Nodes | 100% | 100% | 100% | 100% | 100% | 78% | **95%** |
| Activity Partition | 100% | 71% | 86% | 85% | 81% | 78% | **83%** |
| Object Node | 100% | 79% | 90% | 95% | 67% | 61% | **80%** |
| Object Flow Edge | 100% | 64% | 83% | 95% | 62% | 61% | **76%** |
| Accept Event Action | 60% | 43% | 52% | 95% | 38% | 28% | **53%** |
| Activity | 93% | 64% | 79% | 55% | 38% | 28% | **53%** |
| Flow Final Node | 80% | 43% | 62% | 85% | 29% | 28% | **52%** |
| Send Signal Node | 60% | 50% | 55% | 95% | 33% | 17% | **51%** |
| Action Pin | 67% | 43% | 55% | 75% | 19% | 28% | **45%** |
| Named Activity Edge | 27% | 21% | 24% | 95% | 33% | 28% | **43%** |
| Object Node for Objects in Specific States | 53% | 29% | 41% | 50% | 38% | 39% | **42%** |
| Call Behaviour Action (rake) | 33% | 43% | 38% | 65% | 29% | 22% | **39%** |
| Activity Parameters | 67% | 36% | 52% | 50% | 19% | 22% | **38%** |
| Accept Timed Event | 53% | 29% | 41% | 55% | 24% | 11% | **34%** |
| Expansion Region | 53% | 36% | 45% | 50% | 10% | 22% | **33%** |
| Exception Handler | 47% | 29% | 38% | 50% | 10% | 22% | **31%** |
| Interruptible Activity Region | 33% | 29% | 31% | 50% | 14% | 22% | **30%** |
| Activity Edge Connector | 73% | 29% | 52% | 10% | 14% | 22% | **27%** |
| Data Store | 33% | 14% | 24% | 55% | 5% | 17% | **25%** |
| Central Buffer Node | 27% | 21% | 24% | 55% | 5% | 6% | **23%** |
| Behaviour Specification for Object Flows | 33% | 14% | 24% | 40% | 10% | 11% | **22%** |
| Local Pre-Post Condition | 53% | 21% | 38% | 30% | 0% | 11% | **22%** |
| Weighted Activity Edge | 20% | 14% | 17% | 55% | 5% | 11% | **22%** |
| Decision Node with Behaviour | 33% | 29% | 31% | 25% | 5% | 6% | **18%** |
| Object Node Ordering Kind | 13% | 14% | 14% | 55% | 0% | 6% | **18%** |
| Structured Nodes/Activities | 27% | 21% | 24% | 40% | 5% | 0% | **18%** |
| Activity Partition Textually Presented | 47% | 14% | 31% | 0% | 19% | 11% | **17%** |
| Object Node with a Limited Upper Bound | 13% | 21% | 17% | 45% | 0% | 6% | **17%** |
| Exception Pin | 27% | 14% | 21% | 20% | 5% | 11% | **15%** |
| Selection Specification for Object Flow | 13% | 7% | 10% | 35% | 5% | 6% | **14%** |
| Join Specification | 20% | 7% | 14% | 30% | 5% | 0% | **13%** |
| Stream Pin | 20% | 21% | 21% | 15% | 0% | 11% | **13%** |
| Value Pin | 13% | 7% | 10% | 40% | 0% | 0% | **13%** |
| Parameter Set (Activity Parameters) | 33% | 14% | 24% | 5% | 0% | 6% | **10%** |
| Selection Specification for Object Node | 7% | 7% | 7% | 30% | 0% | 0% | **9%** |
| Object Node for Tokens with Signal as Type | 33% | 14% | 24% | 0% | 0% | 0% | **8%** |
| Parameter Effect Kind | 7% | 7% | 7% | 20% | 0% | 0% | **7%** |
| Activity Exception Parameters | 13% | 0% | 7% | 15% | 0% | 0% | **6%** |
| Activity Stream Parameters | 7% | 7% | 7% | 15% | 0% | 0% | **6%** |
| Stream Edge | 13% | 7% | 10% | 5% | 0% | 0% | **5%** |
| Decision Node with Input Flow | 0% | 0% | 0% | 15% | 0% | 0% | **3%** |
| Input and Output Effects for Object Node | 0% | 7% | 3% | 10% | 0% | 0% | **3%** |

Book Guide = UML Notation Guides, Book Spec = Software Engineering books based on UML, Book Tot = All books Dashed lines represent the 40% and 60% thresholds
1 book out of 30 considers only Class Diagrams and thus it was not included in this analysis.

**Fig. 8.** Usage levels of Activity Diagram constructs

*represented by placing the effect in braces near the edge leading to or from the pin for the parameter*" see [24]. Indeed, it may be interpreted as a kind of constraint (the preceding/following action should be such that to produce the depicted effects) and so it is not clear the difference with the local pre-post conditions. Otherwise, it may be

considered as a kind of comment making explicit some effects already defined by the behaviour of the preceding/following action. In the latter case it may be replaced by a comment. Also the semantics of Decision Node with Input Flow is quite complex, see [24][371-372], but this construct may be considered as derived (it may be replaced by combining the normal and the extra input tokens of the decision node by means of a new data structure). We think that however in these two cases one of the reasons of the fact that they are neglected is the complexity of their definition.

One of the possible reasons, for the quite surprising result summarized in Fig. 8, is that in UML 1.x there were few constructs for the activity diagrams, more or less those resulting widely used in our document and tool analysis study, and so those resulting scarcely used are quite new. Another reason for these constructs being so unpopular is that many of them are not depicted visually by the tools on the activity diagrams (for example, the Object Node Ordering Kind construct may be defined using the Visual Paradigm tool for an Object Node by means of the property panel but cannot be visualized).

In our opinion, some of the constructs classified by our document and tool analysis study as scarcely used or in the grey area may be useful for modelling workflows and business processes [2, 10, 17, 19, 15], e.g., Flow Final Node, Accept Event Action, Send Signal Node, Activity and the constructs related with exception handling; whereas many others are just a kind of derived constructs, i.e., they may be replaced by an equivalent activity diagram fragment (e.g., Value Pin).

If we examine the usage data considering separately the various kind of sources, we have that the tools and the books are "using" a greater number of constructs, whereas courses and tutorials are considering less constructs. Precisely the numbers of scarcely used/widely used constructs for the various kinds are:

– books guide: scarcely used 25 and widely used 16
– books spec: scarcely used 32 and widely used 10
– books tot (i.e., all books): scarcely used 28 and widely used 11
– tools: scarcely used 20 and widely used 15
– courses: scarcely used 38 and widely used 9
– tutorials: scarcely used 38 and widely used 9

Moreover, three constructs Accept Event Action, Action Pin and Flow Final Node result widely used for both books guide and tools. The percentages relative to courses and tutorials are really striking; the considered activity diagrams constructs are just 1/5 of those available. Finally, looking at Fig. 8, it is interesting to note that the usage levels among different sources, even if different in magnitude, have similar trends.

### 4.3 Level of Usage of the Use Case Diagram Constructs

Fig. 9 presents the level of usage of the constructs relative to the use case diagram in the different kinds of sources. As before, we consider a construct widely used if the percentage is $\geq 60\%$, and scarcely used if such percentage is $\leq 40\%$.

Use case diagrams are quite simple, and the results are straightforward and not surprising. All constructs except Actor User Defined Icon and Extend with Condition are widely used, and Use Case and Actor have a percentage of 100%. The use cases

| Use Case Diagram Construct | Book Guide | Book Spec | Book Tot | Tool | Course | Tutorial | All Sources |
|---|---|---|---|---|---|---|---|
| Actor | 100% | 100% | 100% | 100% | 100% | 100% | **100%** |
| Use Case | 100% | 100% | 100% | 100% | 100% | 100% | **100%** |
| Include | 100% | 93% | 97% | 100% | 100% | 75% | **94%** |
| Extend | 93% | 93% | 93% | 100% | 100% | 75% | **93%** |
| Subject (System) Box | 87% | 100% | 93% | 70% | 95% | 75% | **85%** |
| Use Case Specialization | 87% | 79% | 83% | 95% | 81% | 44% | **78%** |
| Actor Specialization | 80% | 71% | 76% | 100% | 76% | 38% | **74%** |
| Extend with Condition | 53% | 50% | 52% | 35% | 38% | 13% | **37%** |
| Actor User Defined Icon | 53% | 50% | 52% | 30% | 10% | 0% | **27%** |

Book Guide = UML Notation Guides, Book Spec = Software Engineering books based on UML, Book Tot = All books Dashed line represents the 40% and 60% thresholds
1 book out of 30 considers only Class Diagrams and thus it was not included in this analysis.

**Fig. 9.** Usage levels of Use Case Diagram Constructs

relationships (Include and Extend) are over 90%, the Actor Specialization and Use Case Specialization are over 70%, and the Subject (System) Box is over 80%.

In this case the tools are not those providing the larger set of constructs, for Extend with Condition, User Defined Actor Icon and Subject (System) Box the percentage of tools proving them is lower than that of the books mentioning them, perhaps because they cannot be drawn reusing other visual constructs. For the courses also the Include and the Extend reach a 100% percentage, and the Subject (System) Box reaches 95%.

The results seem quite clear in this case, the fundamental constructs for use case diagrams are Use Case, Actor, Extend and Include; Subject (System) Box, and Actor/Use Case Specialization are quite useful, whereas Extend with Condition and Actor User Defined Icon are perceived as less relevant.

## 5 Threats to Validity

In our study a possible threat is the fact that when examining books, courses and tutorials we have decided to assume that a construct is used in that source if it is mentioned without making a deeper analysis to measure how much detailed was presented or taught. Unfortunately, it is really difficult to devise a better metric; for example, trying to distinguish if a diagram is just mentioned, shortly presented, presented, and presented with all the details may be too much depending on the personal judgement of who examines the textual sources; also counting the occurrence of the name of a construct is in our opinion too dependent on the way the texts are written, e.g., more or less verbose. We have also tried to distinguish the case of a simple mention of a construct in the text and the presence of an example of such construct, without detecting a relevant difference.

To avoid to bias the results of our document and tool analysis study, we have considered only sources concerned with the use of the UML, avoiding those with different aims, for example drawing tools suitable to produce pictures of UML diagrams, or books presenting a survey on the current visual notations have been excluded; whereas instead books covering specific use of the UML or courses about software engineering where the UML was taught were included.

Concerning the textual sources, we have considered only books/courses/tutorials presented using languages understood by the authors (i.e., English, Italian, French and

Spanish). For the courses the limitation on the language is less problematic, indeed many of the courses presented using the English language are taught in countries where the English is not the mother tongue (e.g., Hungary and Turkey).

For the tools, instead, we are quite confident to have examined almost all the available ones; we think that a UML tool cannot exist without being presented somewhere on the Web. Notice that Argo UML, one of the most known UML tool was not included in our document and tool analysis study since it supports only UML 1.x.

We have considered here only four kinds of sources (books, courses, tutorials, sources) and we are aware that these are not the only ones; indeed there are also the UML users, and we are now running a personal survey to investigate which constructs they know and which they use. Instead, we do not plan to make a literature survey examining which constructs are used in scientific papers in the area of modelling or of Model Driven Development, since this will be surely biased: e.g., there are few scientific papers about class diagrams, whereas the newest and the most problematic constructs will appear in many of them. However, literature survey covering applicative areas for modelling, for example concerning SOA (Service Oriented Architecture) or requirement engineering may give valuable insights.

The various diagrams/constructs of the UML have different ages, i.e., they were introduced in the UML at different times, some appeared in the UML 1.x, others in UML 2.0 and someone is still more recent (e.g., the profile diagram appeared only in UML 2.2). To mitigate this threat we have careful considered only sources that explicitly stated that were considering at least UML 2.0. A more refined analysis made considering as a source for a given construct only those dated after the official time of appearance of the same is unfeasible, also because we have the official date of appearance of a constructs, i.e., the official approval of the OMG document presenting it, but such document was already available to the community and so the construct was already know and used.

Finally, we have decided to define widely used (scarcely used) when a construct was considered in the $\geq 60\%$ ($\leq 40\%$) of the sources, resulting also in a grey area. We think that this a sensible choice, using a threshold lower than 60%, e.g., 50%, should have led to have that a construct is either widely used or scarcely used without any doubt cases, and this does not sound realistic. On the other hand, a higher threshold, e.g., 80%, should have led to a quite large number of inconclusive cases.

We have also computed the widely/scarcely used on the totality of the sources, disregarding the fact that they are of very different kinds, e.g., books and courses, and so assigning them different weights would have been more realistic. Again, we had the problem to compute these weights in an unbiased way: is it sensible to say that a book is three times more relevant than a course, or that a tool is two times more relevant than a tutorial? To avoid to make our results too dependent on our personal judgement we have preferred to assume that all the sources have the same weight.

## 6 Conclusions

We have investigated, by means of a document and tool analysis study, the level of usage of the UML activity and use case diagram constructs, considering in this paper four kinds of sources: books, tools, courses, and tutorials. The results of our document and

tool analysis study show that, the level of usage varies considerably among the different constructs, and in some cases it is really very low. However, it is important to note that, a low level of usage of a construct does not mean that it is useless. This could be caused by different factors, for example: (1) a construct may be replaced by a combination of other constructs (i.e., it is derived), (2) its existence may be not widely known, (3) its definition could be too complex and not very clear, and thus discouraging potential users, or finally, (4) it could be useful only in very specific and rare cases.

Results show that, a large majority of the 47 activity diagrams constructs seem to be scarcely used. More precisely, 31 activity diagram constructs result scarcely used (in some cases the percentage of usage is less than 10%), while, only nine constructs result widely used by our document and tool analysis study: Action, Control Flow Edge, Initial/Final Node, Decision/Merge Nodes, Fork/Join Nodes, Activity Partition (i.e., swimlane), Object Node, and Object Flow Edge. Instead, only two of the nine constructs of the use case diagram (Extend with Condition and Actor User Defined Icon) result scarcely used, with percentages 37% and 27% respectively.

In this paper, we have considered only unbiased and objective sources and examined them for checking if some UML constructs are used in an objective way (e.g., can a tool produce a model including such constructs?, is a course/tutorial teaching the fact that UML has such constructs?). For this reason, we believe that the results of this document and tool analysis study are not biased by any personal opinion (neither ours nor of any human being taking part in the examination of the sources). We are now investigating the usage of the other UML diagrams/constructs and performing a personal survey [16] to investigate which UML diagrams/constructs are known and used by UML users trying to cover different categories of them, and different applicative fields. The combined results of this work and of the ongoing personal opinion survey should lead to finally sketch an "essential" UML.

## References


1. D. Budgen, A. J. Burn, O. P. Brereton, B. A. Kitchenham, and R. Pretorius. Empirical evidence about the UML: a systematic literature review. *Software Practice and Experience*, 41(4):363–392, Apr. 2011.
2. F. Di Cerbo, G. Dodero, G. Reggio, F. Ricca, and G. Scanniello. Precise vs. ultra-light activity diagrams - an experimental assessment in the context of business process modelling. In *Proceedings of 12th International Conference on Product Focused Software Development and Process Improvement (PROFES 2011)*, volume 6759 of *LNBIP*, pages 291–305. Springer, 2011.
3. B. Dobing and J. Parsons. How UML is used. *Comm. of the ACM*, 49(5):109–113, May 2006.
4. J. Erickson and K. Siau. Can UML be simplified? Practitioner use of UML in separate domains. In *Proceedings of 12th International Workshop on Exploring Modeling Methods for Systems Analysis and Design (EMMSAD 2007)*, volume 365, pages 81–90. CEUR Workshop Proceedings, 2007.
5. M. Grossman, J. E. Aronson, and R. V. McCarthy. Does UML make the grade? Insights from the software development community. *Information and Software Technology*, 47(6):383–397, Apr. 2005.
6. R. M. Groves, F. J. J. Fowler, M. P. Couper, J. M. Lepkowski, E. Singer, and R. Tourangeau. *Survey Methodology*. John Wiley and Sons, 2009.

7. I. Jacobson. Taking the temperature of UML. Web site: http://blog.ivarjacobson.com/taking-the-temperature-of-uml/, 2009.
8. B. Kitchenham and S. Pfleeger. Personal opinion surveys. In F. Shull and Singer, editors, *Guide to Advanced Empirical Software Engineering*, pages 63–92. Springer London, 2008.
9. M. Leotta, G. Reggio, F. Ricca, and E. Astesiano. Building VECM-based systems with a model driven approach: an experience report. In *Proceedings of 1st International Workshop on Experiences and Empirical Studies in Software Modelling (EESSMod 2011)*, volume 785, pages 38–47. CEUR Workshop Proceedings, 2011.
10. M. Leotta, G. Reggio, F. Ricca, and E. Astesiano. Towards a lightweight model driven method for developing SOA systems using existing assets. In *Proceedings of 14th International Symposium on Web Systems Evolution*, WSE 2012, pages 51–60. IEEE, 2012.
11. P. Mohagheghi, V. Dehlen, and T. Neple. Definitions and approaches to model quality in model-based software development - a review of literature. *Information and Software Technology*, 51(12):1646–1669, Dec. 2009.
12. OMG. *Service oriented architecture Modeling Language (SoaML) Specification Version 1.0.1*, 2012. Available at www.omg.org/spec/SoaML/1.0.1/PDF.
13. M. Petre. UML in practice. In *Proceedings of 35th International Conference on Software Engineering*, ICSE 2013, pages 722–731. IEEE, 2013.
14. L. F. Prior. Document analysis : The sage encyclopedia of qualitative research methods. *SAGE Publications*, 2008.
15. G. Reggio, M. Leotta, and F. Ricca. "Precise is better than light" A document analysis study about quality of business process models. In *Proceedings of 1st International Workshop on Empirical Requirements Engineering*, EmpiRE 2011, pages 61–68. IEEE, 2011.
16. G. Reggio, M. Leotta, and F. Ricca. Who knows/uses what of the UML: A personal opinion survey. Submitted to *17th International Conference on Model Driven Engineering Languages and Systems (MoDELS 2014)*, 2014.
17. G. Reggio, M. Leotta, F. Ricca, and E. Astesiano. Business process modelling: Five styles and a method to choose the most suitable one. In *Proceedings of 2nd International Workshop on Experiences and Empirical Studies in Software Modelling*, EESSMod 2012, pages 8:1–8:6. ACM, 2012.
18. G. Reggio, M. Leotta, F. Ricca, and D. Clerissi. What are the used UML diagrams? A preliminary survey. In *Proceedings of 3rd International Workshop on Experiences and Empirical Studies in Software Modelling (EESSMod 2013)*, volume 1078, pages 3–12. CEUR Workshop Proceedings, 2013.
19. G. Reggio, F. Ricca, G. Scanniello, F. Di Cerbo, and G. Dodero. A precise style for business process modelling: Results from two controlled experiments. In *Proceedings of 14th International Conference on Model Driven Engineering Languages and Systems (MoDELS 2011)*, volume 6981 of *LNCS*, pages 138–152. Springer, 2011.
20. G. Scanniello, C. Gravino, and G. Tortora. Investigating the role of UML in the software modeling and maintenance - A preliminary industrial survey. In J. Filipe and J. Cordeiro, editors, *ICEIS (3)*, pages 141–148. SciTePress, 2010.
21. E. Seidewitz. UML 2.5: Specification simplification. Presented at "Third Biannual Workshop on Eclipse Open Source Software and OMG Open Specifications", May 2012.
22. M. Torchiano, M. D. Penta, F. Ricca, A. D. Lucia, and F. Lanubile. Migration of information systems in the italian industry: A state of the practice survey. *Information and Software Technology*, 53(1):71 – 86, 2011.
23. M. Torchiano, F. Tomassetti, F. Ricca, A. Tiso, and G. Reggio. Relevance, benefits, and problems of software modelling and model driven techniques: A survey in the Italian industry. *Journal of Systems and Software*, 86(8):2110–2126, 2013.
24. UML Revision Task Force. *OMG Unified Modeling Language (OMG UML), Superstructure, V2.4.1*, 2011.