The Journal of Systems and Software 86 (2013) 997–1005

Contents lists available at SciVerse ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

# A Brazilian survey on UML and model-driven practices for embedded software development

Luciane Telinski Wiedermann Agner[a,b,*], Inali Wisniewski Soares[a,b], Paulo Cézar Stadzisz[b], Jean Marcelo Simão[b]

[a] Department of Computer Science, Mid-West State University (UNICENTRO), Rua Padre Salvador, 875, 85.015-430, Guarapuava, Parana, Brazil
[b] Graduate School of Electrical Engineering and Computer Science, Federal University of Technology - Paraná (UTFPR), Av. 7 de Setembro, 3165, 80.230-901, Curitiba, Parana, Brazil

ARTICLE INFO

*Article history:*
Received 18 January 2012
Received in revised form 9 May 2012
Accepted 12 November 2012
Available online 21 November 2012

*Keywords:*
Unified Modeling Language
Model Driven Engineering
Model Driven Architecture
Embedded Software
Survey

ABSTRACT

This paper brings statistical findings from a survey about the use of UML modeling and model-driven approaches for the design of embedded software in Brazil. The survey provides evidences regarding the maturity of use of UML and model-driven approaches, how they are employed, and which and where the professionals who use them are. Technical, organizational, and social aspects were investigated and documented by making use of a descriptive research method. Such aspects seemingly reflect the opinions of software engineers on how they perceive the impact of using UML and model-driven approaches on productivity and quality in embedded software development. Results show that most participants are clearly aware of the modeling approach value, even though they practice it only to a limited degree. Most respondents who make use of model-driven approaches attest that productivity and portability are the key advantages of their use.



## 1. Introduction

The term Model Driven Engineering (MDE) is used to describe model-driven software development approaches in which abstract models of the software systems are designed and systematically transformed into concrete implementations (France and Rumpe, 2007). Model Driven Architecture (MDA), a standard from the Object Management Group (OMG), is MDE's best known and currently adopted model-driven approach.

In MDA, the systematic use of models as engineering main artifacts occurs throughout the entire software development cycle (OMG, 2003). The primary technical advantages claimed by MDA consist of productivity, portability, reusability, and interoperability (García-Díaz et al., 2010; Kleppe et al., 2003). In this context, after the launch of the MDE approach, the role of the Unified Modeling Language (UML) became even more important, positioning the UML as the core part in the software development process. UML is a Software Engineering industry standard that stems from the principle of software development best practices (OMG, 2010).

Furthermore, several studies currently point out the importance of integrating model-driven approaches in the embedded software development process, aiming to minimize the effects of the heterogeneity of platforms and the increasing complexity inherent in this type of software (Jeon et al., 2009; Karsai et al., 2008). Embedded software comprises incorporated hardware features and is commonly subject to rigid constraints, which indeed represent a major challenge for the community of developers (Marwedel, 2006). The use of model-driven approaches for designing embedded software improves validation and verification as well as facilitating reuse and evolution (Espinoza et al., 2009).

This paper aims to identify gaps in the current comprehension of issues, such as knowledge of how exactly UML and MDE/MDA are used in industry; understanding of how social and organizational factors impact on UML and MDE/MDA use. More specifically, the following UML-related issues were investigated: UML diagrams more commonly used, UML complexity level under the view of the participants, and UML use maturity. Moreover, this paper emphasized the benefits of model-driven approaches in terms of productivity, portability, and quality with regard to the following activities: code generation, model transformations, documentation, and testing. In addition, MDE/MDA maturity of use in the organization was also investigated. Social and organizational aspects obtained were used in cross-tab reports so as to acquire new knowledge from the data collected. This paper compiles answers that illustrate aspects derived from the practical experience of

* Corresponding author at: Rua Padre Salvador, 875, 85.015-430, Guarapuava, Parana, Brazil. Tel.: +55 42 3621 1069; fax: +55 42 3621 1090.
*E-mail addresses:* lagner@unicentro.br, luagner@yahoo.com.br (L.T.W. Agner), inali@unicentro.br (I.W. Soares), stadzisz@utfpr.edu.br (P.C. Stadzisz), jeansimao@utfpr.edu.br (J.M. Simão).

0164-1212/$ – see front matter © 2012 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.jss.2012.11.023

software engineers who took part in a survey involving 209 Brazilian embedded software developers.

These aspects must be known because, despite the constant UML advancements, UML has been used in different ways and for different purposes in software development designs. Frequently, UML is used in the software development traditional context, not taking into account model-driven approaches. Therefore, models are used by programmers just as a reference for specifying and documenting software systems. In MDE/MDA approaches, however, models are used as the key artifacts in all software development stages. In this case, the development of software can be seen as a series of successive transformations from one model into another, including code generation from models.

The style and strictness in UML-based modeling also show a variation that relies on several aspects, such as the previous experience of the software engineers and the deadline defined for the software development. It is important to mention that UML is one of the most widely criticized modeling languages (France and Rumpe, 2007). As a reason for that, one could point out some of its deficiencies, for instance its lack of expressiveness (Pardillo and Cachero, 2010). Also, in many cases UML is not used, or its use is limited, on the grounds its complexity (Dori, 2002). In other cases, it is criticized due to the difficulty in modeling applications related to specific domains, e.g. embedded systems (Friedenthal et al., 2008).

The results obtained thus provide criteria for the identification of problems that hamper and impair the UML use for the development of embedded software in Brazil. The listed problems may be used by the academic community to suggest alternatives to the improvement of the UML usability, focusing on the future use of MDE/MDA approaches for embedded software design. Besides, the emphasis on embedded software is justified due to the wide range of existing applications such as industrial automation, electronic industry, telecommunications or military products. Currently, the growing complexity of embedded products is demanding the use of different software development approaches, such as the model-driven approach.

The paper is structured as follows. Section 2 brings the related works. Section 3 presents the details of the methodology used to perform the survey, including the purpose, instrument, and data collection. Section 4 presents the detailed results of the survey, including the background information of the respondents, the UML use, and the use of the MDE/MDA approaches. Section 5 discusses the research findings. Section 6 brings the conclusion, remarks, and future work.

## 2. Related work

Several related works investigate aspects of UML modeling and model-driven approaches. However, none of these surveys focused on the use of UML modeling and model-driven approaches in the development of embedded software, specifically. Moreover, the respondents of these surveys were basically European and North-American professionals. No similar surveys were found in South America, justifying the relevance of this work for the geographical area covered (Brazil), as well as its specific focus on embedded software development.

A survey reported in Nugroho and Chaudron (2008) inquired into the opinions of 80 software engineers with regard to the variety of styles and rigor in the use of UML modeling (i.e., completeness, level of detail, and correlation with implementation), as well as how they perceive them. Most survey respondents, originated mainly from European countries, agree that the impact of UML use on software development productivity is perceived mostly in the design, analysis, and implementation phases. Furthermore, according to the authors the impact on quality can be noticed mainly in understandability and modularity. Their study, however, focused only on aspects related to the improvement in software development quality and productivity. In addition, the link between UML use and model-driven approaches is not considered.

Dobing and Parsons (2006) conducted a survey that states which UML diagrams are more often used in practice. This survey involved about 171 professionals, being most of them members of the OMG. The survey demonstrated that *Class*, *Sequence*, and *Use Case* Diagrams are the most used. Also, this survey suggests that more training programs are needed to help developers increase their knowledge of UML. A similar survey, conducted by Grossman et al. (2005), founded that the respondents consider UML as fairly understandable. Further, many organizations use only a small set of UML diagrams, such as *Use Case* and *Class* diagrams. Organizations based in several countries took part in this survey, although no Brazilian organization was mentioned.

Cherubini et al. (2007) performed a survey that discussed the role of diagrams in software development practices, finding out that the use of diagrams plays an important role in understanding the code, fostering discussions, designing, and supporting documentation. Their study reveals how developers tend to use temporary and informal drawings and rarely use standard modeling languages such as UML to model software systems. Therefore, this survey is not specifically about issues regarding the use of UML diagrams.

It is worth pointing out that none of the previously mentioned studies associate the use of UML with model-driven approaches. In this context, the survey conducted by Hutchinson et al. (2011), Torchiano et al. (2011), Forward and Lethbridge (2008) stands out.

Hutchinson et al. (2011) collected experiences in the adoption and application of model-driven software development in industry, aiming to identify practices that lead to success or failure concerning model-driven approaches, but not focusing specifically on the development of embedded software nevertheless. According to the authors, for most software engineers MDE represents a need for new skills, including the UML modeling expertise. Also, the majority of the respondents considered the use of MDE on their projects to be beneficial in terms of productivity, maintainability, and portability (about 60%). However, a significant number of respondents disagreed (about 20%). The respondents were professionals with experience of using modeling in industry, given that their spatial location was not taken into account.

Torchiano et al. (2011) conducted a survey with 155 Italian software professionals that investigated modeling application in software development and in MDE. This survey also focused on mostly used modeling languages, processes and tools, and revealed that UML is the preferred language for modeling, although Domain Specific Languages (DSLs) are used as well. The main results of this survey show that approximately 68% of the respondents use models during software development, and among them, 44% generate codes stemming from models. Furthermore, according to the authors, the lack of usefulness and the large investments required are the two most frequent reasons that prevent modeling adoption. Another survey conducted by Forward and Lethbridge (2008) ask the opinion of 113 software developers about MDE, where most developers are originally from Canada and United States. The results of this survey show that model-driven approaches are not very popular as most participants work with code-centric approaches.

## 3. Methodology

A survey is a method for collecting data about features, behavior or opinions of a specific group of people, pointed out as representative from a target population (Pinsonneault and Kraemer, 1993). Thus, a survey aims to produce quantitative data on some aspects

**Table 1**
Questionnaire.

| |
|---|
| 1. Employment relationship. |
| 2. Work experience period. |
| 3. Educational background. |
| 4. Technical function. |
| 5. Line of business of the organization. |
| 6. Number of employees in the organization. |
| 7. Number of employees who develop software in the organization. |
| 8. Representative percentage of new systems. |
| 9. Software development team knows and uses the UML. |
| 10. Reason for making no or partial use of UML. |
| 11. UML complexity level. |
| 12. UML fulfills the respondent's needs. |
| 13. The enterprise uses UML modeling tools for the development of systems. |
| 14. UML models regularly used by the respondent. |
| 15. Maturity evaluation of the UML use in the organization. |
| 16. Software development team knows and uses the model-driven approach. |
| 17. Considering the MDE approach, do you agree with the following statements? |
| 18. In which way the MDE use affects the global productivity of projects developed by the enterprise. |
| 19. Use of models in different MDE activities. |
| 20. Code automatic generation x MDE-based productivity. |

of the population under study. This section details the purpose of this survey, the survey instrument, and the data collection mode.

### 3.1. Purpose of the survey

Actually, Brazil does not count on much expertise with respect to how UML is used in the development of embedded software. This survey firstly pointed out aspects regarding the effective use of UML modeling in this field, thereby aiming to untangle the factors that hamper or impair its use, such as

- developers' lack of knowledge on UML;
- developers belief that UML does not add value to the software development process;
- short deadline for the system development;
- lack of specialized UML tools in the organizations for the system modeling.

The survey also collected data about the use of model-driven approaches for embedded software development in Brazil, focusing on the MDE and MDA approaches. This paper makes use of the term MDE to indicate both MDE and MDA approaches. Some of the items covered are:

- The embedded software developer has the knowledge of model-driven approaches.
- The company in question makes use of model-driven approaches in the development of its projects.
- Contributions and advantages in using these approaches, according to embedded software developers.

### 3.2. Survey instrument

The survey instrument consisted of a questionnaire that aimed to find out how UML and model-driven approaches are used in embedded software development in Brazil. The questionnaire contained 20 questions as listed in Table 1 and grouped as follows: respondent background (4 questions), organization background (4 questions), UML modeling (7 questions), and MDE approaches (5 questions).

These questions allowed the investigation into UML and MDE aspects that were already present in the existing literature (Hutchinson et al., 2011; Dobing and Parsons, 2008; Grossman et al., 2005), as well as the indication and documentation of aspects from Brazilian experience, specifically.

A survey is a one-side interaction between a researcher and a respondent. In this way, the development of a questionnaire must observe several criteria (Perrien et al., 1986; Fowler, 1995). In this survey, the following criteria were considered: (a) questions were reliable; (b) questions were unbiased; (c) questions were clear and unambiguous in meaning; (d) question choices were exhaustively defined, in order to comprehend all possible answers; (e) limited number of questions, aiming to create a questionnaire to be answered in a short period of time; and (f) question sequencing was taken into consideration.

Data were collected through an on-line questionnaire created by means of the Survey Monkey tool (www.surveymonkey.com). Web-based questionnaires allow easier information input from the respondent perspective, and more efficient data collection from the researcher perspective. Jelitshka et al. (2007) have observed that web-based questionnaires guarantee high feedback rates.

In order to evaluate the survey, a pilot process was developed in partnership with two software organizations, providing feedback for the improvement of the survey interface. The feedback was incorporated into the survey questionnaire.

### 3.3. Data collection

The sampling process used for defining the target population of this survey was the non-probability process, consisting of a particular group of typical cases. A non-probabilistic sample is obtained from a particular criterion, so not all elements of the population have the same opportunity to be selected. When the features targeted by the survey are the same both in the object population and in the sample, then non-probabilistic samples become equivalent to probabilistic samples. In spite of its limitations, this kind of sampling is indicated when the respondents belong to groups with specific features (Fink, 1995). The survey under study is an example of this kind of sampling, given that the chosen participants represent a group of embedded software developers acting in Brazil. The sampling rate was 209 developers, considered to be the necessary quantity of respondents to obtain reliable results.

Given the lack of any defined population of Brazilian embedded software developers to obtain a random sample, respondents were selected to participate in the survey as follows:

- A list of organizations to be contacted was defined, based on data from Brazilian institutions like the Brazilian National Confederation of Industry (CNI) and its associated federations (www.cni.org.br). After that, organizations were contacted through telephone calls and emails in order to obtain the contact data of the developers and then invite them to join in the survey.
- Invitations were sent through online forums on embedded systems development, such as: sis_embarcados Group (www.br.groups.yahoo.com/group/sis_embarcados) and *Embarcados Portal* (www.embarcados.com.br).
- Invitation emails were sent to embedded software development researchers associated with Brazilian research institutes on this field of study, such as: National Science and Technology Institute (INCT – http://www.inct-sec.org/en/o-inct-sec/o-instituto), GMicro (Microelectronics Group of the Santa Maria Federal University – http://w3.ufsm.br/gmicro/), and Computer Systems Laboratory (LSC – http://www.lsc.ic.unicamp.br/).

The participants invited to take part in the survey received an introduction text containing: (1) a formal invitation; (2) the survey objectives; (3) the survey identification; and (4) instructions on how to fill in the questionnaire. The invitation sent to the respondents included a privacy statement and the estimated time for answering the questionnaire, which is approximately 6 min. The questionnaire was sent to Brazilian embedded software developers from several important software organizations. In the questionnaire preamble it was stressed that the target community was professionals with experience in developing embedded software in the industry. The data collected were organized in a common format, aiming the later application of statistical methods oriented to a descriptive analysis of such data. It is worth mentioning that other related researchers (Dobing and Parsons, 2008; Grossman et al., 2005) have used similar methods of data collection due to the difficulty in finding more representative samples.

By means of such contact procedure, it was possible to select 1740 embedded software developers and researchers: 640 through email contact and 1110 through invitations posted on online forums. The survey was conducted from November 2010 until May 2011. Thus, after a 6-month surveying period, a total of 275 people responded by accessing the online survey, while complete data were submitted by 209 respondents. That is to say, an approximate 16% feedback rate was achieved with the invitations sent. Such result is relevant to the purpose of this survey, once the employed sample is stratified. Considering that no money incentive was offered to entice respondents to complete this survey, this rate was considered as a reasonable outcome. To be included in this survey the respondent had to be directly involved with embedded software development. Aside from that requirement, there were no other criteria for inclusion in the sample.

Responses from all regions of the country were obtained, as follows: South (41.15%), Southeast (46.41%), Central-West (2.87%), North (0.96%), and Northeast (5.74%). Only 2.87% of the respondents did not inform their geographical location. Most respondents reside and work either in the South or in the Southeast, i.e., a total of 87.57% of the participants. Such data reflect the Brazilian current economic scenario, given that most industry plants are established in those regions, as well as most technology produced (IBGE, 2008; Silva et al., 2009; Amato Neto, 2007). Despite Brazil's huge territory, most companies of the productive sector are concentrated both in the South and in the Southeast regions, mainly in the following states: São Paulo, Rio de Janeiro, Minas Gerais, Rio Grande do Sul, and Paraná (Silva et al., 2009). In its turn, the response participation rate in each of these states was achieved as follows: São Paulo (33.49%), Paraná (16.27%), Rio Grande do Sul (12.92%), Minas Gerais (6.7%), and finally Rio de Janeiro (5.26%).

Considering that in all the questions the non-response rate represents less than 5% of the population, the estimated reliability margin of the survey reaches 95%. That can be observed in the Figures and Tables presented in Section 4 of this paper. Also, the error margin is insignificant, once the sample is non-probabilistic and the survey is descriptive, which means that the population universe is known.

## 4. Survey findings

The results discussed in this section are organized around four issues: respondent background, organization background, UML Questions, and MDE Questions. Next, the findings related to these issues will be presented.

### 4.1. Respondent background

Data concerning the respondent's employment relationship status (Question 1) are presented in Table 2. The majority of the respondents (78.5%) has a continued employment relationship with enterprises that, among other activities, develop embedded software. The rest either works for education/research institutions (14.8%) or is self-employed (6.7%).

**Table 2**
Employment relationship status.

| Employment relationship | Respondents (%) |
|---|---|
| Enterprise | 78.5 |
| Education/Research Institutions | 14.8 |
| Self-employed | 6.7 |

**Table 3**
Work experience.

| Years | Respondents (%) |
|---|---|
| 0-2 | 18.2 |
| 2–5 | 31.1 |
| 5–10 | 38.8 |
| >10 | 11.0 |
| No response | 1.0 |

**Table 4**
Educational background.

| Academic area | Respondents (%) |
|---|---|
| Computer Science or similar | 57.9 |
| Electronic Engineering | 26.8 |
| Other areas | 13.9 |
| No academic degree | 1.4 |

Considering Question 2 (Table 3), the majority of the respondents (38.8%) has 5–10 years of work experience, while 31.1% have 2–5 years experience, and 18.2% have up to 2 years. That means, the background of most respondents is characterized by professionals with a continued employment relationship and with 5- to 10-year work experience.

In terms of educational background (Question 3), most respondents (57.9%) hold a bachelor's degree in Informatics, Computer Science, Computer Engineering or similar degrees. In their turn, about 26.8% of them hold a bachelor's degree in Electronic Engineering, followed by 13.9% who hold a degree in other areas, such as Civil Engineering, Mechanical Engineering, Electrical Engineering, Mechatronics Engineering, Mathematics, and Physics. Only 1.4% stated that they hold no academic degree. Table 4 shows the results obtained in Question 3.

Concerning Question 4 (Table 5), the majority of the respondents (67.5%) works as programmer. Other technical functions with a significant number of responses were: software engineers (47.8%), design managers (32.5%), team leaders (26.3%), and researchers (25.4%). In addition, it was observed that team leaders and design managers were among the participants with "5- to 10-year" and "more than 10-year" work experience. The findings confirmed that nowadays more and more companies are counting on the expertise of design managers and team leaders to coordinate their projects, as stated in Lewis (2007).

**Table 5**
Technical function.

| Technical function | Respondents (%) |
|---|---|
| Software engineer | 47.8 |
| Programmer | 67.5 |
| Team leader | 26.3 |
| Design manager | 32.5 |
| Domain expert | 12.0 |
| Researcher | 25.4 |
| Test chief | 13.9 |
| Requisite chief | 11.5 |
| Quality chief | 7.2 |
| No response | 1.0 |

**Table 6**
Organization line of business.

| Line of business | Respondents (%) |
|---|---|
| Industrial automation | 38.8 |
| Information technology | 23.9 |
| Telecommunications | 23.0 |
| Eletronic industry | 20.6 |
| Research | 16.3 |
| Security | 14.8 |
| Transport | 13.9 |
| Commercial automation | 13.4 |
| Other | 12.4 |
| Energy | 10.5 |
| Aerospace | 4.8 |
| Medical | 3.8 |
| Agriculture | 2.4 |
| Defense | 1.4 |
| No response | 0.5 |

**Table 7**
Number of employees in the organization.

| Number of employees | Respondents (%) |
|---|---|
| 1–10 | 14.2 |
| 10–100 | 39.2 |
| 100–1000 | 24.5 |
| 1000–10,000 | 11.8 |
| > 10,000 | 3.9 |
| No response | 6.4 |

### *4.2. Organization background*

The next questions refer to the organizational background. Their goal was to collect and evaluate data regarding the activities carried out by the enterprises/institutions, the number of employees, the percentage of employees working specifically with software development, as well as the representative percentage of new software systems being developed.

Initially, the line of business of the enterprises/institutions was determined. Table 6 presents the responses obtained in Question 5 (multiple choice question). The top five lines of business according to the number of responses were: industrial automation (38.8%), information technology (23.9%), telecommunications (23%), electronic industry (20.6%), and research (16.3%).

Table 7 illustrates the number of employees in the organizations (Question 6). The organizations with a larger number of responses were the ones that have: 10-100 employees (39.2%) and 100-1000 employees (24.5%). Table 8 brings data concerning the number of employees working with software development in the respective organization (Question 7). The top three options with the larger number of responses were: 10-100 employees (39.7%), 1-10 employees (32.8%), and 100-1000 employees (16.2%). That is to say, the findings of this survey stem mainly from respondents who work for medium and small-sized enterprises. Indeed, the majority of software development organizations are small and medium enterprise. Consequently, a larger number of employees working with software development are found in such organizations (Anacleto et al., 2004; Mishra and Mishra, 2009).

**Table 8**
Software developers in the organization.

| Number of software developers | Respondents (%) |
|---|---|
| 1–10 | 32.8 |
| 10–100 | 39.7 |
| 100–1000 | 16.2 |
| 1000–10,000 | 2.0 |
| > 10,000 | 0.5 |
| No response | 8.8 |

**Table 9**
Representative percentage of new systems.

| Percentage of new systems | Respondents (%) |
|---|---|
| 0–25 | 20.6 |
| 25–50 | 26.3 |
| 50–75 | 19.1 |
| 75–100 | 18.2 |
| None | 15.8 |

Question 8 refers to the representative percentage of new systems developed by the enterprise/institution for which the respondent works. In this case, it should be borne in mind that the development of new software systems is driven by organizational demands. For example, new products, new technologies, or new functional requirements may lead to the development of new software systems. The responses are presented in Table 9.

### *4.3. UML questions*

Firstly, the focus of these questions was on the awareness and the use of UML by the software development team in which the respondent takes part. As a result, 45% of the respondents claimed to be using UML in their development projects. A cross-tab report was then designed from the follow-up data to investigate the relationship between the professionals who know and make use of UML in the embedded software development (Question 9) and the period of involvement in software development activities (Question 2). It was observed that the most experienced professionals are the ones who better know and make more use of UML. In addition, another cross-tab attested that respondents who work for organizations with 100-1000 employees (Question 6) are the ones who know and make use of UML at the most.

Question 10 focuses on getting to know the main reason why the software development team, of which the respondent is part, does not make use or makes partial use of the UML diagrams (Table 10). A total of 115 responses was obtained, being that the top three results were: short lead-time for the software development (39.1%), lack of understanding or knowledge of UML models (22.6%), and existence of few people in the company who have deep knowledge of UML (13.9%). Similar results regarding the development of general purpose software were found in Nugroho and Chaudron (2008). The findings indeed confirm that the pressure on project resources, e.g. time, tends to restrain the use of UML, since the development time would increase as system models are conceived. In addition, the survey developed by Hutchinson et al. (2011) also points out the need of a longer training period so as to overcome the lack of UML expertise. Only 0.9% of the respondents confirmed the use of another modeling language, thus attesting the fact that UML is the de facto standard modeling language (Booch et al., 2005).

A total of 94 respondents took part in the following UML questions. Question 11 aimed to find out the respondents' opinion on the complexity level of the UML modeling language (Fig. 1). Most responses corresponded to a medium level of complexity (58.5%), followed by 25.5% corresponding to a considerable level of

**Table 10**
Reason for not making use or making partial use of the UML.

| The reason | Respondents (%) |
|---|---|
| Do not know UML | 22.6 |
| UML does not add value to justify its use | 11.3 |
| The UML is not useful for most projects | 9.6 |
| Only a few people in the enterprise know the UML | 13.9 |
| The captured information is redundant with the system code | 2.6 |
| Short lead-time for the software development | 39.1 |
| Use another modeling language | 0.9 |

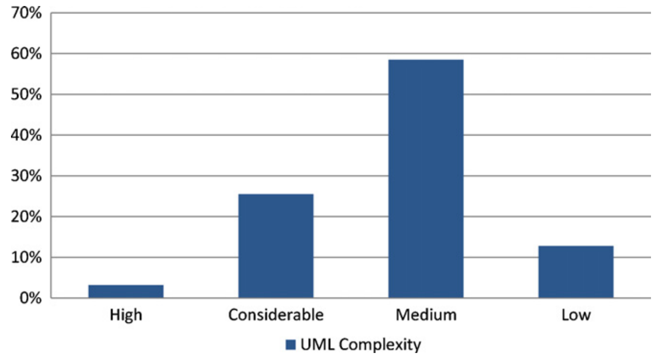


**Fig. 1.** UML complexity level.

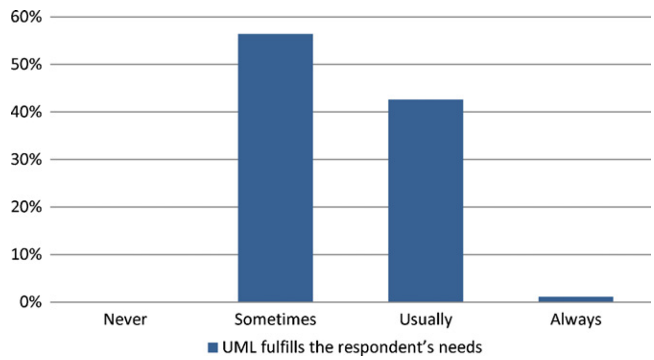


**Fig. 2.** UML fulfills the respondent's needs.

complexity. Only 3.2% considered the level of UML complexity as high. Complexity is also cited as an impediment to the use of UML in development of general purpose software (Grossman et al., 2005). Such findings show that education/research institutions need to invest more in studies and training programs oriented to the software development.

Concerning Question 12 (UML fulfills the respondent's needs), most responses were: sometimes (56.4%), and usually (42.6%), as illustrated in Fig. 2. The professionals involved with embedded software development for more than 10 years were the most satisfied respondents with the UML diagrams, representing 83.3% in their respective category. Question 13 inquired into the use of UML modeling tools, finding out that the large majority (93.6%) of the organizations make use and only 6.4% do not make use of UML tools. A future study could investigate which UML tools are being used in practice.

Fig. 3 shows the UML diagrams commonly used (Question 14). The four most cited diagrams were: *Class* (99%), *Sequence* (93.6%), *Use Case* (78.7%), and *State Machine* (91.5%). It is important to mention that such results were similarly found in other surveys (Dobing and Parsons, 2006; Grossman et al., 2005), as demonstrated in Table 11. *Class* diagrams were the most frequently used in the three surveys being compared, while *Communication* diagrams have the highest non-usage rate. In this comparison, a significant difference was found with regard to the use of *State Machine* diagrams. A possible reason for that lies in the fact that this survey focuses on the development of embedded software, in which *State Machine* diagrams commonly describe the internal behavior of the system components.

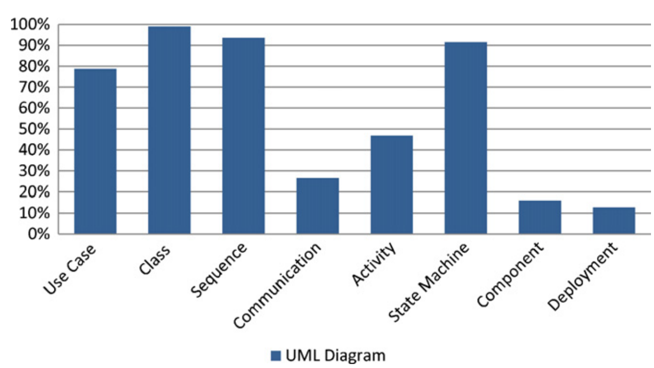


**Fig. 3.** UML diagrams most frequently used.

**Table 11**
Comparative – UML diagram usage.

| UML diagram | Respondents (%) | *1 | *2 |
|---|---|---|---|
| Use Case | 79 | 72% | NA% |
| Class | 99 | 87% | 93% |
| Sequence | 94 | 75% | 89% |
| Communication*3 | 27 | 42% | 50% |
| Activity | 47 | 55% | 60% |
| State Machine | 91 | 53% | 63% |

*1 Dobing and Parsons (2006), diagrams used into two-thirds or more of the projects.
*2 Grossman et al. (2005).
*3 Communication and collaboration diagrams.

**Table 12**
Maturity evaluation of the UML use in the organization.

| UML maturity evaluation | Respondents % |
|---|---|
| Initial exploration | 47.9 |
| First experimental use | 12.8 |
| First significant Project | 17.0 |
| Several projects completed | 21.3 |
| Vast experience | 0.0 |
| No response | 1.0 |

Maturity evaluation of the UML use in the enterprise/institution is demonstrated in Table 12 (Question 15). Only 21.3% of the respondents declared the development of several complete projects using UML, whereas the others confirmed its use as an initial exploration (47.9%), first experimental use (12.8%), and first significant project (17.0%). Furthermore, the respondents who developed the largest number of complete projects are the most experienced ones, i.e., the ones with 5- to 10-year (60%) and more than 10-year (30%) experience in embedded software development. None of the participants declared to have a vast experience in the UML use.

### *4.4. Model-driven questions*

Concerning the questions about MDE, the initial purpose was to identify whether the respondents know and make use of MDE approaches. Next, it consisted in understanding how such approaches are used, as well as their benefits, extent, and maturity of use. Indeed, MDE questions were answered by 45.4% (95 out of 209) of the survey participants, once only the participants who confirmed the use of a modeling language in the development of embedded software had access to such questions.

Question 16 identified whether the respondent knows and uses MDE approaches (Table 13). Most respondents neither know nor

**Table 13**
The respondent knows model-driven approaches.

| Knowledge of MDE | Respondents (%) |
|---|---|
| Knows and makes use | 7.3 |
| Knows and makes partial use | 15.8 |
| Knows but makes no use | 17.9 |
| Neither knows nor makes use | 56.8 |
| No response | 2.1 |

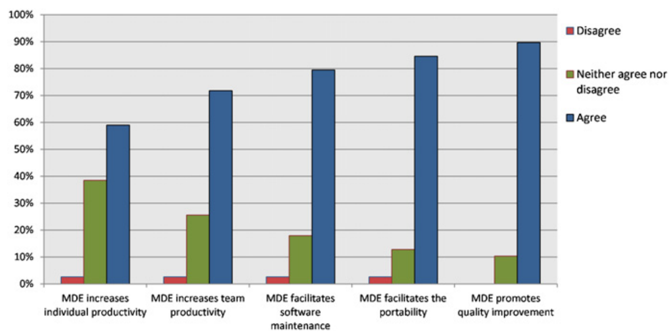


**Fig. 4.** Considering the MDE approach, do you agree with the following statements?

make use of MDE (56.8%), followed by 17.9% who know but make no use. Only 15.8% know MDE and use it, either completely or partially.

By assuming that the use of the MDE approach provides a set of different benefits for different users, it is important to understand the impact of this approach on aspects such as: increase in individual and team productivity, maintenance improvement, portability to new platforms, and quality of software. The histogram in Fig. 4 clearly illustrates that in all cases the majority of the respondents agrees that the use of MDE improves the previously mentioned aspects. In addition, the most significant benefits are associated with quality and portability of the developed software system. According to Mohagheghi and Dehlen (2008), among software engineers discussing improvements in software quality due to the use of model-driven approaches, the key experienced benefit consists of a substantial reduction in the number of software defects. A total of 39 respondents took part in Question 18.

Question 18 identified the impact of models on several software development activities. The activities were selected so as to represent the various interpretations regarding MDE in its different knowledge levels. Respondents were able to assert that an activity improved or reduced productivity, as well as indicating that they did not use a particular activity, or were indifferent to it. Fig. 5 illustrates the percentage of respondents who stated that each activity improved productivity, as well as those who did not use that particular activity. The results show that considering the use of models in some activities, e.g. communication within the team, better understanding of a problem in an abstract level, and the design and documentation of projects – most respondents agree on the fact that the use of MDE improves productivity.

**Table 14**
Maturity evaluation of the MDE use in the organization.

| UML maturity evaluation | Respondents (%) |
|---|---|
| Initial exploration | 46.2 |
| First experimental use | 12.8 |
| First significant project | 2.6 |
| Several projects completed | 7.7 |
| Vast experience | 2.6 |
| No response | 28.2 |

That means, productivity improvements were attributed mainly to design- and documentation-related activities. On the other hand, in activities such as model transformation, tests, code automatic generation, and simulation of models/executable models – most respondents denied the use of the MDE approach. The lack of skilled professionals in Brazil, professionals with a vast experience in UML and MDE, reflect the results obtained. Furthermore, another factor assumed as a restraint on the use of code generation and model transformations is the short supply of existing user friendly tools to support these processes. According to Mohagheghi and Dehlen (2008), in MDE the users must have access to appropriate tools, in a way that integrating a tool suite that meets requirements such as modeling, transformations, and code generation into a consistent environment becomes a challenge.

Such findings differ a little from Hutchinson et al. (2011), to whom activities like code generation, transformation models, and executable models are more used in practice. That is because Hutchinson et al. (2011) focuses on the development of systems in general, not specifically on embedded software. Indeed, the use of the MDE approach is more recent than the development of general purpose software (France and Rumpe, 2007). In addition, typical embedded software comprises some specific features, e.g. concurrency, real-time processing, and limited resources.

Question 19 investigates whether code generation is an important aspect to improve productivity in MDE. The responses obtained show that most respondents agree (82.1%), but only 17.9% really make use of the code generation. Finally, Question 21 focused on evaluating the use maturity of the MDE approach in the enterprise/institution for which the respondent works (Table 14). A large majority of the respondents assumed an initial exploration of such approach (46.2%), whereas only 10.3% declared to have sev-

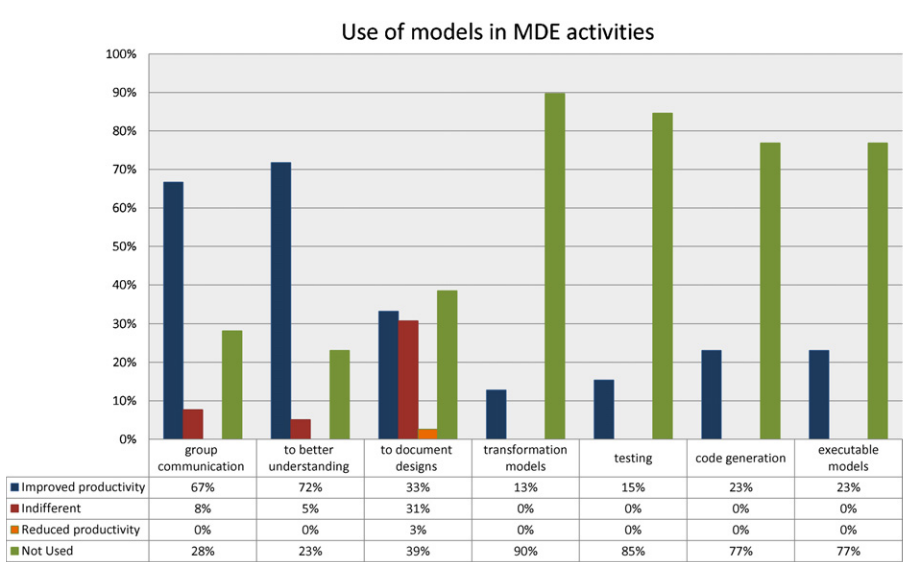


**Fig. 5.** Use of models in MDE activities.

eral projects completed or a vast experience in MDE projects. It is important to point out that all respondents who opted for "several projects completed" and "vast experience" are researchers from education institutions and/or research centers. The number of non-respondents is far larger if compared to the other questions, as a consequence of the little expertise in MDE mentioned throughout the paper.

## 5. Discussion of findings

The findings of this survey provided information on the use of UML and model-driven approaches for embedded software development in Brazil. According to the survey, 45% of the embedded software developers responded that they know and make use of the UML modeling, either completely or partially. Further studies on such finding were conducted so as to find out who those respondents are. As a result, it was verified that they are the most experienced developers, i.e., the ones with 5- to 10-year work experience and those with more than 10-year experience. This emphasizes the benefits of the UML use, once experienced developers can better assess the benefits of the UML employment for the development of embedded software. In addition, it could be attested that most of these professionals work for enterprises with 100–1000 employees, namely medium-sized enterprises. Usually, large-sized enterprises count on more resources and, consequently, are able to invest more in the education of professionals and in the acquisition of software tools.

The paper also shows an analysis of the key factors that impair the use of UML for embedded software development, being the most relevant results: short lead-time for the software development, lack of understanding or knowledge of UML models, and reduced number of employees with deep knowledge of UML. The survey findings indicate a lack of professionals who are specialist in UML and model-driven approaches in the enterprises and institutions under study.

The majority of the respondents found UML complex at a medium, considerable, or high level. It was clear that *Class*, *Sequence*, *Use Case*, and *State Machine* diagrams are the most popularly used in embedded software development in Brazil. Similar results were achieved in North-American and European countries by surveys oriented toward the development of general purpose software (Dobing and Parsons, 2006; Grossman et al., 2005).

The survey findings show that the respondents who are best satisfied with the UML are those involved with software development activities for more than 10 years, and they represent 83.3% of the respondents in their category. In this way, it was possible to state that respondents who were more experienced in the development of embedded software seek alternatives for the development of embedded software, such as the use of UML and model-driven approaches.

Most respondents that make use of model-driven approaches attest the key advantages of their use, for instance, individual and team productivity increase, quality improvement, facilitation concerning software maintenance and portability to new platforms. On the other hand, it was observed that most respondents still do not take part in MDE-related activities that demand deep knowledge of such approach, such as model transformation, code generation, and the use of models in tests. In this manner, one may conclude that software engineers are still not mature enough with regard to the use of MDE approaches.

## 6. Conclusion and future work

This paper reports findings based on a survey that deals with the lack of knowledge on how exactly UML and model-driven approaches are used for the design of embedded software in Brazil. The survey was online-based and, from November 2010 until June 2011, a total of 209 applicable responses were received. Although the survey was exploratory in its nature, it clearly captured important issues related to the use of modeling by Brazilian software engineers. The results achieved match the results obtained in other countries, what was demonstrated through a comparison with other related works. Analysis of questionnaire data has involved some enumeration and statistical calculations in order to bring some overall awareness to the UML and MDE use.

In general, most participants clearly perceive the value of the modeling approach, even though they practice it only to a limited degree. UML is the dominant language for modeling, and the use of modeling tools is widespread.

The major problems encountered in the adoption of UML refer to the lack of skills, the lack of coherent tools, and the strict time requisites applicable to software development projects. Key findings mainly include: modeling is primarily used for documentation and design with little code generation, model-centric approaches are currently not very popular as most participants work in code-centric environments, and software engineers who work extensively on models are the more experienced ones.

The survey has also explored real-life experiences in using model-driven techniques in embedded software development. As software engineers are continuously looking for better ways to improve their software development cycles, MDE has been regarded as an approach able to support it. This survey has inquired into how organizations are currently using model-driven technologies in product development and how they perceive the benefits of this use. The results show that 23.1% of the respondents know and use model-driven approaches, partially or not. Among them, the majority believes that the use of model-driven approaches is beneficial in terms of productivity and portability, mainly. However, MDE is currently far from achieving a mature level, considering its recent introduction in the software development (OMG, 2003).

Based on the survey findings, it may be concluded that there are many challenges in the area of model-driven development processes and practices. Also, although model-driven approaches claim many potential benefits, it has been developed without empirical support for these claims. At this evolution stage, it may be that developers using UML modeling are still not aware enough of how the model-driven approach fits with the tasks (goals) they are trying to perform. In addition, supporting MDE with an integrated tool environment is crucial, as many of the MDE requirements strongly rely on appropriate tool support.

Future research is needed to better understand UML use so as to gain insight into how it can be effectively used to support model-driven development. An aspect not covered in this survey, but that may be subject of study in future works, concerns the benefits obtained in terms of interoperability and reusability with the use of MDE approaches. Also, understanding the needs of MDE users is clearly a question that requires further research.

## References

Amato Neto, J., 2007. As políticas industriais e tecnológicas e as pequenas e médias empresas: iniciativas na história recente da economia brasileira. Revista Gestão Industrial 3 (1), 87–102.

Anacleto, A., von Wangenheim, C.G., Salviano, C.F., Savi, R., 2004. Experiences gained from applying ISO/IEC 15504 to small software companies in Brazil. In: 4th International SPICE Conference on Process Assessment and Improvement, Lisbon, Portugal, pp. 33–37.

Booch, G., Rumbaugh, J., Jacobson, I., 2005. Unified Modeling Language User Guide, 2nd edition. Addison-Wesley Professional, Massachusetts, USA.

Cherubini, M., Venolia, G., DeLine, R., Ko, A.J.,2007. Let's go to the whiteboard: how and why software developers use drawings. In: SIGCHI Conference on Human Factors in Computing Systems (CHI 2007). ACM Press, New York, NY, USA, pp. 557–566.

Dobing, B., Parsons, J., 2006. How UML is used. Communications of the ACM 49 (5), 109–113.

Dobing, B., Parsons, J., 2008. Dimensions of UML diagram use: a survey of practitioners. Journal of Database Management (JDM) 19 (1), 1–18.

Dori, D., 2002. What UML should be: why significant UML change is unlikely. Communications of the ACM 45 (11), 82–85.

Espinoza, H., Selic, B., Cancila, D., Gérard, S., 2009. Challenges in combining SysML and MARTE for model-based design of embedded systems. European Conference on Model Driven-Architecture Foundations and Applications (ECMDA 2009). Lecture Notes in Computer Science (Springer) 5562, 98–113.

Fink, A., 1995. How to sample in surveys. Sage, Thousand Oaks, pp. 1995–1999.

Forward, A., Lethbridge, T.C., 2008. Problems and opportunities for model-centric versus code-centric software development: a survey of software professionals. In: International Workshop on Models in Software Engineering (MiSE 2008). ACM Press, New York, NY, USA, pp. 27–32.

Fowler, F.J., 1995. Improving Survey Questions: Design and Evaluation. Sage Publications, Thousand Oaks, CA.

France, R., Rumpe, B., 2007. Model-driven development of complex software: a research roadmap. In: Future of Software Engineering (FOSE 2007). IEEE Computer Society, Washington, DC, USA, pp. 37–54.

Friedenthal, S., Moore, A., Steiner, R., 2008. A Practical Guide to SysML: The Systems Modeling Language. Morgan Kaufmann Publishers Inc., Elsevier, San Francisco, USA.

García-Díaz, V., Fernández-Fernández, H., Palacios-González, E., G-Bustelo, B.C.P., Sanjuán-Martínez, O., Lovelle, J.M.C., 2010. TALISMAN MDE: mixing MDE principles. Journal of Systems and Software 83, 1179–1191.

Grossman, M., Aronson, J.E., McCarthy, R.V., 2005. Does UML make the grade? Insights from the software development community. Information and Software Technology 47 (6), 383–397.

Hutchinson, J., Whittle, J., Rouncefield, M., Kristoffersen, S.,2011. Empirical assessment of MDE in industry. In: 33rd International Conference on Software Engineering (ICSE 2011). ACM Press, pp. 471–480.

IBGE, 2008. Table 2.1 – Selected variables of mining and quarrying and manufacturing industrial companies, by selected Major Regions and Federation Units – 2006–2008. Survey of Technological Innovation, Brazilian Institute of Geography and Statistics (IBGE). http://www.ibge.gov.br/home/estatistica/economia/industria/pintec/2008/tabelas_pdf/tab_2_01.pdf (accessed 29.3.11) (in portuguese).

Jelitshka, A., Ciolkowski, M., Denger, C., Freimut, B., Schlichting, A., 2007. Relevant information sources for successful technology transfer: a survey using inspections as an example. In: First International Symposium on Empirical Software Engineering and Measurement (ESEM 2007). IEEE Computer Society, pp. 31–40.

Jeon, S.U., Hong, J.E., Song, I.G., Bae, D.H., 2009. Developing platform specific model for MPSoC architecture from UML-based embedded software models. Journal of Systems and Software 82 (10), 1695–1708.

Karsai, G., Neema, S., Sharp, D., 2008. Model-driven architecture for embedded software: a synopsis and an example. Science of Computer Programming 73 (1), 26–38.

Kleppe, A.G., Warmer, J., Bast, W., 2003. MDA Explained: The Model Driven Architecture: Practice and Promise. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Lewis, J.P., 2007. Fundamentals of Project Management, 3rd ed. AMACOM Books, New York, NY, USA.

Marwedel, P., 2006. Embedded System Design. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Mishra, D., Mishra, A., 2009. Software process improvement in SMEs: a comparative view. Computer Science and Information Systems (ComSIS) 6 (1), 111–140.

Mohagheghi, P., Dehlen, V., 2008. Where is the proof? A review of experiences from applying MDE in industry. European Conference on Model Driven-Architecture Foundations and Applications (ECMDA-FA) 2008. Lecture Notes in Computer Science (Springer) 5095, 432–443.

Nugroho, A., Chaudron, M.R.V., 2008. A survey into the rigor of UML use and its perceived impact on quality and productivity. In: 2nd International Symposium of Empirical Software Engineering and Measurement (ESEM 2008). IEEE Computer Society, New York, NY, USA, pp. 90–99.

OMG, 2003. MDA Guide, Version 1.0.1. Object Management Group, Document: omg/03-06-01, Retrieved from: http://www.omg.org/cgi-bin/doc?omg/03-06-01

OMG, 2010. Unified Modeling Language (UML) Infrastructure, Version 2.3. Object Management Group, Document: formal/2010-05-03, Retrieved from: http://www.omg.org/spec/UML/2.4/Infrastructure

Pardillo, J., Cachero, C., 2010. Domain-specific language modelling with UML profiles by decoupling abstract and concrete syntaxes. Journal of Systems and Software 83, 2591–2606.

Perrien, J., Chéron, E.J., Zins, M., 1986. Recherche en marketing: méthodes et décisions. Gaëtan Morin Editeur, Montreal.

Pinsonneault, A., Kraemer, K.L., 1993. Survey research methodology in management information systems: an assessment. Journal of Management Information Systems 10 (2), 75–105.

Silva, M.V.B., Silveira Neto, R.M., 2009. Dinâmica da concentração da atividade industrial no Brasil entre 1994 e 2004: uma análise a partir de economias de aglomeração e da nova geografia econômica. Economia Aplicada 13 (2), 299–331.

Torchiano, M., Tomassetti, F., Ricca, F., Tiso, A., Reggio, G., 2011. Preliminary findings from a survey on the MD state of the practice. In: International Symposium on Empirical Software Engineering and Measurement (ESEM 2011). IEEE Computer Society, Washington, DC, USA, pp. 372–375.

**Luciane Telinski Wiedermann Agner** is a Ph.D. student of the Graduate Program in Electrical Engineering and Industrial Computer Science at the Paraná Federal University of Technology (UTFPR), Brazil. Also, she is a Professor at the Department of Computer Science of the Mid-West State University (UNICENTRO), Brazil. She got her B.Sc. degree in Computer Science from the Pontifical Catholic University of Paraná (PUC-PR) in 1991. In 2000, she received the M.Sc. degree in Computer Science from the Federal University of Paraná (UFPR). Her main research interests are software/system engineering, embedded software, and computer science.

**Inali Wisniewski Soares** obtained her B.Sc. degree in Computer Science from UEPG, Brazil, in 1990 and in 2000 the M.Sc. degree in Computer Science from the Federal University of Paraná (UFPR), Brazil. Currently, she is a Ph.D. student of the Graduate Program in Electrical Engineering and Industrial Computer Science (CPGEI) at the Paraná Federal University of Technology (UTFPR), Brazil. Also, she is a Professor at the Department of Computer Science (DECOMP) of the Mid-West State University (UNICENTRO), Brazil. Her publications and researches include embedded software, software/system engineering and computer science.

**Paulo Cézar Stadzisz** obtained the Diploma in Data Processing Technologies from the Federal University of Paraná, Brazil, in 1987 and in 1990 the M.Sc. degree in Industrial Computer Science from the Graduate Program in Electrical Engineering and Industrial Computer Science (CPGEI) of the Federal Center of Technological Education, Brazil. He received the Ph.D. degree from the Franche-Comté University (France) in 1997. He is currently a Professor at the Paraná Federal University of Technology (UTFPR). He teaches and coordinates researches in the CPGEI/UTFPR doctoral program. He coordinates the Laboratory of Innovation in Technology at UTFPR. His publications and researches include manufacturing modeling and analysis, simulation, and software engineering.

**Jean M. Simão** obtained his B.Sc. degree in Computer Science from UEPG in 1998 and in 2001 the M.Sc. degree from CPGEI/CEFET-PR/UTFPR, Brazil. In 2005, he obtained the Ph.D. degree in Computing and Automatic Domains from CPGEI/UTFPR and the Research Center for Automatic Control of Nancy (CRAN) – Henry Poincaré University (UHP) France. In 2006, he worked for the UHP/CRAN in a post-doctoral position. Nowadays, he is a Professor at UTFPR in Departments (Electronics, Informatics, CPGEI) and Laboratories (Innovation in Technology, Intelligent Systems). His teaching activities concern computer science and his researches/publications include intelligent systems and programming paradigms.