# Development of Service-Oriented Architectures using Model-Driven Development: A Mapping Study

David Ameller[a], Xavier Burgués[a], Oriol Collell[a], Dolors Costal[a], Xavier Franch[a], Mike P. Papazoglou[b]

[a] *Universitat Politècnica de Catalunya, Barcelona, Spain.*
[b] *Tilburg University, Tilburg, The Netherlands.*

**Abstract**

*Context:* Model-Driven Development (MDD) and Service-Oriented Architecture (SOA) are two challenging research areas in software engineering. MDD is about improving software development whilst SOA is a service-based conceptual development style, therefore investigating the available proposals in the literature to use MDD when developing SOA may be insightful. However, no studies have been found with this purpose.

*Objective:* This work aims at assessing the state of the art in MDD for SOA systems. It mainly focuses on: what are the characteristics of MDD approaches that support SOA; what types of SOA are supported; how do they handle non-functional requirements.

*Method:* We conducted a mapping study following a rigorous protocol. We identified the representative set of venues that should be included in the study. We applied a search string over the set of selected venues. As result, 129 papers were selected and analysed (both frequency analysis and correlation analysis) with respect to the defined classification criteria derived from the research questions. Threats to validity were identified and mitigated whenever possible.

*Results:* The analysis allows us to answer the research questions. We highlight: 1) predominance of papers from Europe and written by researchers only; 2) predominance of top-down transformation in software development activities; 3) inexistence of consolidated methods; 4) significant percentage of works without tool support; 5) SOA systems and service compositions more targeted than single services and SOA enterprise systems; 6) limited use of metamodels; 7) very limited use of NFRs; 8) limited application in real cases.

*Conclusion:* This mapping study does not just provide the state of the art in the topic, but also identifies several issues that deserve investigation in the future, for instance the need of methods for activities other than software development (e.g., migration) or the need of conducting more real case studies.

*Keywords:* Service-Oriented Architecture, Model-Driven Development, SOA, MDD, State of the Art, Mapping Study, Survey

*Email addresses:* dameller@essi.upc.edu (David Ameller), diafebus@essi.upc.edu (Xavier Burgués), ocollell@essi.upc.edu (Oriol Collell), dolors@essi.upc.edu (Dolors Costal), franch@essi.upc.edu (Xavier Franch), m.p.papazoglou@uvt.nl (Mike P. Papazoglou)

## 1. Introduction

In the last decade we have witnessed the emergence of Service-Oriented Computing (SOC). Papazoglou and Georgakopoulos defined SOC as the computing paradigm that utilizes services as fundamental elements for developing applications [1]. The adoption of SOC impacted over the full application development lifecycle, from requirements engineering to implementation, maintenance and testing. Service-oriented applications are structured using a Service-Oriented Architecture (SOA) whose foundation includes basic services, their descriptions, and basic operations (publication, discovery, selection, and binding) that produce or utilize such descriptions [1].

The study of SOA has attracted a lot of attention from researchers and practitioners. Principles, metamodels, languages, technologies, methods, patterns, etc., for SOA have been and are being proposed continuously and have yield to an extensive body of knowledge. A particular issue has to do with the relationship of SOA with other orthogonal software engineering streams. One of such streams is Model-Driven Development (MDD). According to Mellor et al., MDD pushes the vision that we can construct a model of a system and then transform it into *the real thing* [2]. It is clear that the use of MDD in the context of SOC may deliver powerful software engineering methods. In fact, OMG released in 2012 a standard named SOA Modeling Language (SoaML [3]) which links SOA and MDD. A question that may arise naturally is: to what extent MDD is currently used for developing SOA-based applications? The goal of this paper is to answer this question in detail.

For attaining this goal, we search, analyse and discuss the different approaches that have been proposed in the scientific literature related to this question by performing a mapping study. A mapping study (MS) is a form of systematic literature review (SLR) that aims at identifying and categorizing the available research on a broad software engineering topic [4]. MS are intended to provide an overview of a topic area and identify whether there are sub-topics with sufficient primary studies to conduct conventional SLRs and also to identify sub-topics where more primary studies are needed. MS use the same basic methodology as SLRs.

The rest of the paper is structured as follows. In Section 2, we summarize the key points of both SOA and MDD. In Section 3, we describe the protocol used for developing the mapping study. Section 4 is the core of the paper, where we show the observations that emerge from conducting the study, and provide some interpretations of these observations. In Section 5 we include some discussion and, finally, Section 6 presents the conclusions and future work.

## 2. Background

### *2.1. Service-Oriented Architecture*

Service-Oriented Architecture (SOA) is a software architectural style that uses services as the main building component [5]. A service, as a software component, is a mechanism to enable access to one or more capabilities [6].

SOA applications are normally developed under the guidance of an SOA reference model. A reference model depicts the common characteristics or general structures that can be applied to numerous instances within a specific domain (vertical-industry). The purpose of an SOA reference model is to describe the important concepts and relationships in a specific domain (e.g., healthcare or logistics) and organize SOA functionality at the most appropriate level of detail, independent of the technologies, protocols, and products that are used to implement the domain. As such it provides best practices that are used as basis for sharing and agreeing on core processes allowing consistent process/service flows to be created that address application needs. It divides process flows into logical groupings called layers and defines a layered architecture. A layered SOA reference model inter-relates entities in an SOA development effort (such as processes, services, service implementation and component modules) and helps streamline, discipline and structure the work of software professionals who aim to create successful and widely used SOA-based applications.

Some of the most distinguishing characteristics of SOA are [5]:

**Reusability:** Since services are defined as autonomous software components that expose some generic, well-defined functionality, it should be easy to reuse them in a different context or project.

**Scalability:** Services can be deployed in more than one place. This characteristic brings many opportunities, for instance, the possibility of increasing the number of available services when there is more user demand.

**Flexibility:** SOA provides means to publish and discover new services. These facilities allow adapting the service compositions of the architecture in order to fulfil some goal at run-time, normally related to some non-functional requirement (NFR), e.g., performance. The flexibility of SOA is highly aligned with business principles (e.g., service provider and pay-per-use). This alignment with business is one of the keys to its success.

Service-Oriented Computing (SOC) is the development paradigm that uses SOA as the reference architectural style [1]. SOC promotes the use of dynamic service compositions of two kinds. In orchestration, a central service orchestrates several other services in a process flow. WS-BPEL is the common language to specify a process flow in the SOA context and also BPMN is used to specify orchestrations. In choreography, the interaction is not centralized and each service that takes part in the choreography has some protocol to follow. In this case the used languages are more varied: BPMN, WS-CDL, or the adaptation of BPEL for choreography (BPEL4Chor).

SOA is a technology-independent architectural style, although its most common implementation relies on Web Services (usually adhering to W3C or RESTful protocols). A Web Service developer has to cope with a huge amount of standard notations such as WSDL, UDDI, SOAP, XML, WS-Security, WS-BPEL, etc. In consequence, the learning curve of this technology is quite large. This is one of the main reasons that open the space for the adoption of MDD approaches.

### *2.2. Elements of SOA-based applications*

This section provides a brief overview of the essential elements involved in an SOA application. These are summarized and illustrated in Figure 1 and described below.

One of the most important points in Figure 1 is that we concern ourselves with two types of SOA applications: process-centric vs. data-centric. These two types of SOA have fundamentally different interfaces.

Process-centric SOA applications revolve around service aggregations. These are, in effect, orchestrations of multiple simpler services or business processes that can be located virtually anywhere and implemented in a variety of ways. The process-centric SOA interface provides functionality centred around business processes (as encapsulations of service orchestration) and roles and responsibilities, which govern business processes. Languages such as WSDL and BPEL usually typify this interface. Service orchestration is often vertical in nature, specific to a business function.

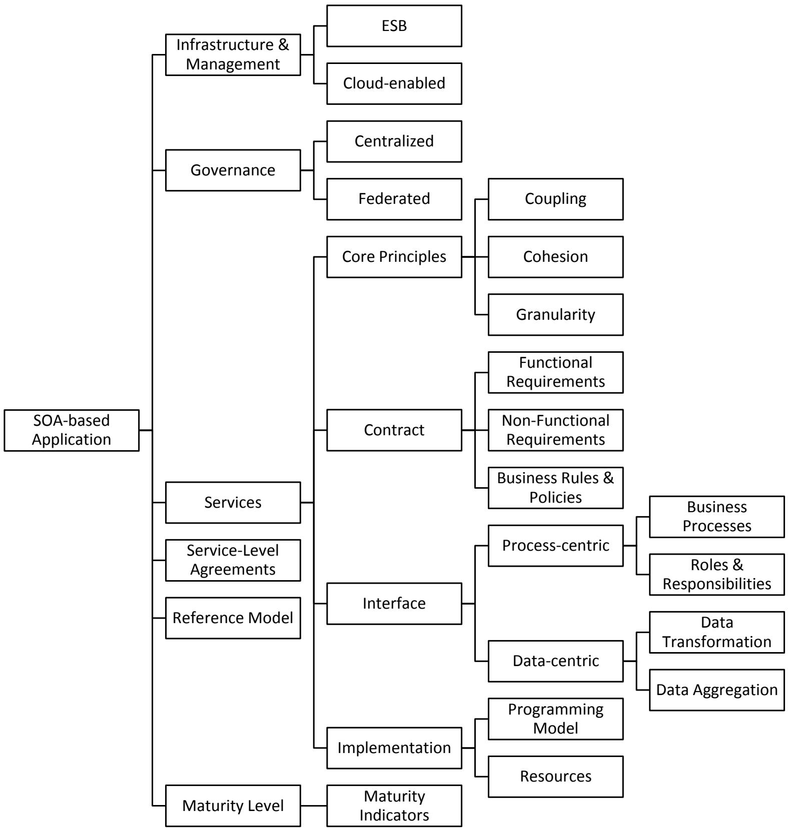


Figure 1: Essential elements involved in an SOA application [7]

The value obtained from service orchestration, however, hinges on the quality of the data that flows through the orchestrated services. To achieve the business benefits expected from SOA initiatives, enterprises must include a strong data-oriented perspective in their work. Enterprises must therefore take a horizontal view of the organization at a data level, to identify and implement data management functionality as services, and to store the data such that it meets broader corporate requirements. In data-centric SOAs provide a consistent, accurate and complete view of the most important data in an organization matters most. A data-centric SOA pushes the concepts of SOA into the data tier, creating reusable data services that can be leveraged in a business process orchestration or a composite application. Consequently, managing, in a single place, the uniqueness, integrity and interrelationships of organizational data becomes the most important mission of data-centric SOAs. This is typified by the data-centric SOA interface which supports data transformation and data integration functionality.

For developing and managing SOA applications, operational control and visibility becomes an important and complex issue [8]. This is due to the fact that SOA applications integrate multiple discrete services and/or applications that are incorporated into the SOA fabric. In an SOA application the notion of SLA is an instrument that makes it possible to exercise operational control and achieve visibility of composed end-to-end services [9]. It formally specifies the exact conditions (functional, non-functional, partner behaviour, regulatory requirements, sanctions, etc) under which the services are delivered to a client.

SOAs are typically implemented using an Enterprise Service Bus (ESB) and more recently cloud computing technology. An ESB provides the necessary infrastructure to create a SOA by providing a backbone for connecting and integrating an enterprises applications and services [10].

A maturity model is benchmark for both evaluating and assessing the current maturity state of an SOA. An SOA maturity model includes goals, characterization of the scope and business benefits of each level of SOA development, e.g., cost reduction or business responsiveness, the important industry standards, key practices, and critical success factors, both technological and organizational. It provides a procedure to create a roadmap for incremental adoption which maximizes business benefits at each stage and optimizes processes along the way.

The final element in Figure 1 is SOA governance. SOA governance ensures that the services and SOA solutions within an organization are adhering to the policies, guidelines and standards that are defined as a function of the objectives, strategies and regulations applied in the organization. The goal of SOA governance is to ensure consistency of the service and solution portfolio and the SOA development lifecycle processes. One of the key objectives of the SOA governance is to support the target SOA maturity level of the organization.

### *2.3. Model-Driven Development*

Model-Driven Development (MDD) can be defined as "the notion that we can construct a model of a system that we can then transform into the real thing" [2]. It is a generalization of the OMG's Model-Driven Architecture (MDA) initiative [11], which is proposing since late 1990's a comprehensive approach to develop, analyse and reason about software, specifications and related artefacts. Integration and interoperability are main targets of MDD and the use of models is the technical way to achieve these goals.

MDD goes through different levels of abstraction, from the very business model (the so-called CIM, Computation Independent Model) to the final executable code. In the middle, a PIM (Platform Independent Model) and a PSM (Platform Specific Model) are designed [11]. The CIM describes the context and requirements of the system but does not address its structure or processing. The PIM considers the operational capabilities of the application in a platform-neutral way showing only those parts that can be abstracted out of any platform. A PSM adds to a platform independent model the details relating to the use of a specific platform or set of platforms. Given the intensive use of models, there is general agreement that metamodeling is an essential foundation for model driven development [12].

Defining the corresponding transformation rules, it is possible to relate and to derive (semi-automatically) a target model from a source one. When performing a classical top-down development, the process starts with a CIM and transform it into a PIM and then into a PSM. Finally, the executable code can be generated from the PSM (though, in some cases, the intermediate models may also be executable to some degree), usually with some tool-support that allows automating partially this last step. There can also be contexts in which the interest is to derive a PIM from another PIM or even from a PSM. In fact, any kind of transformation may be useful, depending on the targeted activity (reverse engineering, integration of legacy systems, coordination of services, etc.).

Table 1: Research questions of the mapping study

| |
|---|
| RQ 1. — What are the characteristics of MDD approaches that support SOA? |
| RQ 1.1. — Which methods are applied by these approaches? |
| RQ 1.2. — What are the main characteristics of the MDD process? |
| RQ 1.3. — What are the input and the output of the MDD approach? |
| RQ 1.4. — Do the approaches report tools? |
| RQ 1.5. — Have the approaches been applied to real cases? |
| RQ 2. — What SOA types are supported? |
| RQ 2.1. — What types of SOA are generated by the MDD approach? |
| RQ 2.2. — What SOA elements are explicitly modelled in the MDD process? |
| RQ 3. — Do these approaches deal with NFRs? |
| RQ 3.1. — What types of NFRs are supported by the MDD approach? |
| RQ 3.2. — What notations are used to specify NFRs? |

## 3. Planning of the Mapping Study

In this section we describe the planning of the MS. We have adopted both the guidelines proposed by Kitchenham et al. for the development of SLR [13, 14] and those described in [15, 4] for the development of mapping studies.

### *3.1. Research Questions*

Table 1 presents the research questions. In this MS, we focus on MDD approaches related to SOA, so RQ1 is to determine how SOA is currently supported by MDD techniques. This RQ is operationalized through other, more concrete ones: first, we want to create a clear mapping of the methodologies used behind each approach (e.g., in many cases the approach is a specialization of a more generic one); second, we are interested in the type of transformation presented (e.g., from PIM to PSM); and third, we want to know exactly what inputs are required and what outputs are obtained (in particular the types of models involved). Also, we also differentiate between the theoretical framework and the implemented tool when it exists. Last, we analyse the application of the approach to real cases.

In RQ2 we are interested in getting more details of the SOA that is produced by the MDD process. As before, we decompose this RQ into: first, what type of SOA system is generated by the MDD process (e.g., single service, composition of services, full SOA system); second, what type of elements may be part of the generated system. As part of this last RQ, attention will be paid to see if the approaches incorporate some metamodel that articulates the MDD process.

SOA systems' success greatly depends on the fulfilment of the elicited NFRs (that eventually are operationalized into SLAs). But on the other hand, we observed that MDD approaches do not usually consider NFRs in the process [16]. We have considered then RQ3 exploring this issue. To answer this question we need to know: first, what types of NFRs are supported by the approaches found (e.g., security, usability, etc.); and second what notation do they use to specify the NFRs (e.g., some UML profile).

We also included Additional Questions (AQs) to show some aspects that, not being related to the main topic of the MS, are still relevant to get a picture of the addressed field (see Table 2). Adding this kind of questions is a common practice, and it is recommended by the MS theory [4]. With this questions we investigate the distribution of the studies on the topic concerning venues, countries, and academy vs. industry.

Table 2: Additional questions of the mapping study

| | |
|---|---|
| AQ 1. | — How publications in the topic are demographically distributed? |
| AQ 1.1. | — In which type of venue are articles mostly published? |
| AQ 1.2. | — How the number of publications has evolved over the years? |
| AQ 1.3. | — Which is the geographical distribution of published articles related to the topic? |
| AQ 1.4. | — How are publications distributed between academy and industry? |

### *3.2. Keyword selection and search string*

We have focused on the population dimension [13, 14] considering its *application area* vision. Since in fact we are interested in two areas altogether, the search string is clearly a conjunction of the two corresponding populations:

*search string = SOA population* AND *MDD population*

Concerning the *SOA population*, we have considered that *SOA* itself should be a recurrent keyword on the targeted approaches but, just to make sure, we have widened the search by including a more open keyword, with the prefix *service*:

*SOA population = service** OR *SOA*

As for the *MDD population*, the situation may be more diverse. First, *MDD* itself is not a universally-agreed string. In the OMG context (and even outside, sometimes unintendedly) we may find *MDA*, and even some consider *MDE*. Second, some authors focus on *language* aspects of the involved *models*. Third, some papers put the emphasis on *metamodels* and it becomes necessary to include explicitly this situation. As result, the string is:

*MDD population = MDA* OR *MDD* OR *MDE* OR *model driven* OR *model* language** OR *meta model** OR *metamodel**

### *3.3. Selection of digital libraries*

For the selection of digital libraries we determined a set of representative venues for SOA and MDD (Step 1) and then selected the digital libraries that cover these venues (Step 2). It is worth to remark that our scope is not limited to the list of venues identified in Step 1; these venues represent the minimal coverage of the literature that we can guarantee in this study.

For Step 1, we targeted four categories or sources. The first three categories are directly related to the topic of the study: 1) MDD (including Modelling Languages because of the metamodel topic); 2) SOA and SOC; 3) Requirements Engineering (since NFR is too narrow as category). A fourth category that includes these three is Software Engineering and Information System Engineering in general. To identify the appropriate sources on these categories, we complemented our own knowledge with experts' opinion from the involved fields and we also checked some other resources like interest groups (e.g., IFIP WG on Services-oriented Systems[1]). During this activity, we also concluded that 2003 would be the starting year of the review, since this was the year when the oldest SOC-specific conferences started. Table 3 shows the sources selected.

[1] `http://home.dei.polimi.it/baresi/ifip`

Table 3: Journals and conferences selected

| Topic | Journals | Conferences |
|---|---|---|
| MDD/Modelling | DKE, SoSyM | ECMFA, ECMT, ER, MoDELS |
| SOA/SOC | IJCIS, Internet Comp., SOCA, TSC, TWeb | ECOWS, ICSOC, SCC, ServiceWave, SOCA |
| RE | REJ | RE |
| SE/ISE | Computer, IBM Systems J., IST, JSS, Software, TOSEM, TSE | CAiSE, ESEC/FSE, FASE, ICEIS, ICSE |

For Step 2, we checked the availability in digital libraries of the papers of the selected venues. Almost all of them were indexed in the ISI Web of Science (WoS), so this one was our digital library of reference. However, we found three situations that required some additional work (see Appendix A for details on the selected venues):

- Some sources did not appear in ISI WoS. These ones were searched using the search engines provided by the publishers. For instance, ESEC/FSE was available in the ACM digital library.
- Some sources appeared in ISI WoS but some particular editions were missing. We used ISI WoS as primary digital library and used the publisher's engine for the missing editions. For instance, this was the case of the ICSE conference.
- One case (ICEIS conference) had 5 editions that did not appear in any digital library. We searched in the conference website.

### *3.4. Search strategy*

The search strategy was designed to consist of the following steps, which in conjunction to the definition of the protocol is an adaptation of the steps proposed in [14]:

Step 1. *Basic search.* Search of all potential candidates. This means:

- Limit the search to papers published from 2003 to 2013. The search in the ISI Web of Science was performed the 15$^{th}$ of January, 2014.
- Application of the search string over ISI WoS limiting the search to science databases. This search brings results not only from the sources listed in Table 3, but also from others venues with papers that match the search string (which are also considered as part of the MS).
- Application of the search string over Springer, ACM and IEEE digital libraries. These digital libraries were used to search the editions of the sources identified in Table 3 not indexed in ISI WoS. Some of these digital libraries had limitations that required a slight reformulation of the search string into an equivalent or including form.
- Manual search in ICEIS conference website (that contained title and abstract of presented papers).

Step 2. *Topic area considered.* In the case of ISI WoS, refinement of the resulting list considering only works related to computer science (i.e., the topic areas of Computer Science, Theory & Methods, Information Systems, Software Engineering, Interdisciplinary Applications, and Artificial Intelligence were selected). For the digital libraries, where we searched

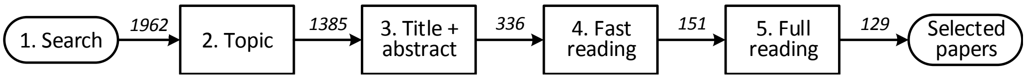


Figure 2: Results of applying the five steps of the mapping study

particular venues, we directly used the ISBN. The references of works found were saved as BibTex format. We used the JabRef reference manager software in order to ease the detection of repetitions when performing subsequent searches.

Step 3. *Title and abstract considered.* Appointment of one researcher of the team as responsible to read the title and the abstract and discard those papers not related to the topic of this study (e.g., we excluded a paper with the title "Educational challenge of health information systems interoperability"). The inclusion/exclusion criteria are shown in Table 4. The terms SOA and MDD mentioned in these criteria are defined in Section 2. Inclusion criteria were applied first and, over the selected works, exclusion criteria removed papers out of scope. We agreed that at this step, only papers clearly not relevant for the purposes of the study were to be removed, leaving them in the candidate list in case of doubt.

Step 4. *Fast reading.* Appointment of two researchers to each remaining paper, different from the one who made the assessment in Step 3. Each of these two researchers is required to classify the paper as *included* or *excluded* based on a fast reading of the complete text and using the same inclusion/exclusion criteria as for the abstracts. Those papers that had consensus were either included for the next step or definitively excluded. In case of conflict, the two reviewers had a face-to-face discussion where the inclusion/exclusion criteria were examined in depth and then a final joint decision made. Papers that required some payment subscription that our university does not cover, were requested directly to authors; some authors did not respond and then their work is not included in this study.

Step 5. *Full reading.* Appointment of a fourth researcher to each remaining paper, different from the three involved in steps 3 and 4. As a result we can say that every paper accepted for analysis was assessed to some extent by four researchers, which allowed insightful discussions in the last stages of the review.

Figure 2 shows the number of papers that was filtered after each step. The full list of selected papers in this mapping study is available in Appendix D.

### *3.5. Classification criteria*

In order to analyse the selected papers we defined some classification criteria, categorized into several values. The values of some criteria were fixed in advance, whilst other emerged in the analysis phase after reading the papers, this is explicitly declared for each criteria in Appendix B. In this subsection we introduce these classifications for the RQs and the additional question and we provide a precise definition to their values.

*AQ1. How publications in the topic are demographically distributed?*

*Venue*, *Year*, *Country* and *Continent* are characteristic of MS [4] to complete a thorough analysis of publication distribution, eventually crossing their values with others. The country was chosen as the country of affiliation of the first author. We went deeper on the venue by identifying its Topic, whose value assignment is (the one that applies most is selected):

- *Service-Oriented.* Focused on SOC. For instance, ICSOC and IEEE Internet Computing.

Table 4: Inclusion/Exclusion criteria

| Criteria | Description |
|---|---|
| Inclusion | The work is about the way to model SOA systems or a concrete aspect of SOA (e.g., security in SOA) |
| Inclusion | The work is about a MDD process for developing SOA systems, either in general or for a specific domain (e.g., only for mobile devices) |
| Inclusion | The work is about meta-models of SOA |
| Exclusion | The work is done using SOA but not to produce SOA systems |
| Exclusion | The work is about constructing SOA systems but there is no evident relation with MDD |
| Exclusion | The work is about other aspects of SOA systems that are off-topic |
| Exclusion | The work only claims that the approach could be implemented with MDD, but no real work is done in that direction |
| Exclusion | The topic of the paper is not focused on SOA or MDD but in other area (e.g., product lines) and SOA or MDD just appear incidentally |
| Exclusion | The work is a case study without much explanation about the MDD approach used |
| Exclusion | The work is not a research paper (e.g., an editorial for a special issue) |
| Exclusion | The paper is about the engineering of a particular element of a wider MDD framework that is not presented in the very paper |

- *Model-Oriented.* Focused on aspects of models. For instance, ER and SoSyM.
- *Software Engineering.* Covers other areas of Software Engineering, like Requirements Engineering, Testing, etc., or Software Engineering in general. For instance, IST and IEEE TSE.
- *General Computing.* Covers Computer Science in general. This includes, for instance, the IBM Systems Journal or the Conference on the Engineering of Computer Based Systems.
- *Other.* If the venue relates to other areas of Computer Science or even to areas that are not related. This includes, for instance, the Conference on Digital Information Management.

Also, AQ1.4 clearly demands a criterion, *Orientation*, to distinguish papers that come from academy and from industry. We decided to evaluate this criterion by looking at authors' affiliation, classifying a paper as *academy* if all authors come from academy and *industry* if there is at least one author from industry. To have an objective criterion, we established that industrial affiliations are those that specify a company (including research departments of a company), and academic affiliations are those that specify a university or research institute.

*RQ1. What are the characteristics of MDD approaches that support SOA?*

To investigate the nature of the method (RQ1.1), we introduced a main criterion, *Method*, whose value may be: *Proposes new method*, *Adapts existing method* or *No method*. This last value corresponds to papers that do not provide insightful information about the method used. In the case of *Adapts existing method*, a secondary criterion, *Method name*, records the name of the adapted method with the aim of finding out if there are consolidated methods in the field.

The two characteristics considered in the MDD process (RQ1.2) are the *type of transformation* and *type of activity*, each of them represented by a single classification criterion. The type of transformation focuses on the relationship of the input and output models' abstraction levels with values *top-down*, *bottom-up* or *horizontal*. The type of activity may take the following values:

- *Engineering.* Papers covering a wide spectrum of the software process.
- *Development.* Papers whose target is building software.
- *Specification.* Papers coping with the first stages of the software process.
- *Integration.* Papers willing to put existing software together.
- *Migration.* Papers giving solution to the change to SOA.
- *Others.* For instance, papers that address traceability support or reliability prediction of composite services.

For the characteristics of input and output (RQ1.3), we defined first two dual criteria, *Input model type* and *Output model type*. Their possible values are the types of artefacts involved in MDD processes, keeping an *Others* value for very low frequency cases. The use of the *Others* value in this question and others later on is almost mandatory to make the application of statistical analysis possible over this kind of outliers. Since the granularity of the values for these two criteria is coarse, we complemented them with two more dual criteria, *Input notation* and *Output notation*, to consider the particular notation of the models. In this case, the two criteria have different possible values, defined in advance according to the most usual scenarios in MDD processes. Remarkably, the input considers *Requirements description* as a single notation though it comprises several ones (mainly textual descriptions) due to the emphasis of our work in the investigation of NFRs. These criteria are defined as single-valued; for those approaches that presented more than one type of document, we chose the most dominant one.

The analysis of the proposed tools (RQ1.4) required up to seven classification criteria. First, *Tool* and *Tool platform* work similarly than *Method* and *Method name*. These criteria do not apply to papers that present a position or discuss about general concepts which are a subset of the papers that have been classified as *No method* according to the *Method* criterion. The rest of criteria collect information about the tool's input and output. As there is no predominant type of input, a single criterion, *Tool input*, was defined with a domain of 20 values (here we choose the most dominant input type, also note that the input type must be understood in their broader possible sense, e.g., UML includes MOF and Ecore). Contrarily to this, for the output, we uncovered three predominant values (*Java*, *WSDL* and *BPEL*) and we defined a fourth value for *Others*, with another criterion, *Tool output others*, that applies in that case for the concrete generated output. It is worth mentioning that for the statistical analysis stage, we translated *Tool output* into three boolean criterion, each for every predominant value (*Tool output Java*, *Tool output WSDL* and *Tool output BPEL*), whilst keeping *Tool output others*; this pre-treatment made the analysis simpler and in fact was applied over other criteria.

To answer about the application of the proposal to real cases (RQ1.5), we introduced one criterion *Example*, classified as *Case study* if the paper reports an application which is claimed to be based on a real project, or, otherwise, classified as *Academic*. Eventually, some papers had *No example* in their text.

*RQ2. What SOA types are supported?*

For the target SOA (RQ2.1) we defined one criterion, *SOA type*, whose values are defined according to the four major patterns presented in [5]:

- *Single service.* The target is a single service.
- *Service composition.* The target is the composition of services itself where the services already exist.
- *SOA system.* The target includes both the services and their composition.
- *SOA enterprise.* The target is a SOA system that is specially designed for its use in an enterprise, and somehow represents the application requirements of an enterprise.

*Single service* and *Service composition* can be seen as particular cases of *SOA system*, while *SOA system* and *SOA enterprise* differs in the focus of the services: *SOA system* is more technical and *SOA enterprise* is more oriented to cover the business needs.

Concerning the second question (RQ2.2), the elements that may appear in the SOA system are determined by the metamodel used, therefore we shifted our focus to metamodels. We analysed those papers that included one or more diagrams referred to as "metamodel" by the authors. The *Metamodel* criterion may take four values:

- *Adapted metamodel.* Metamodels that extend existing metamodels. The typical case in this category is that of UML profiles that are adapted to support new stereotypes.
- *Reused metamodel.* Papers that simply mention the use of one particular metamodel but whose description is not included in the paper. A particular case worth to mention are metamodels reused from previous proposals of the same authors.
- *Propose new metamodel.* Novel metamodels that do not reuse parts of other metamodels. UML profiles may fit in this category when they are only extensions of the UML metamodel.
- *No metamodel.* Papers that do not rely on the use of metamodels.

In the first two cases, we also registered the *metamodel name* (in case it is mentioned) with the goal of determining the most adapted and reused metamodels.

We extracted the most common *concepts* that appear in these metamodels. After an analysis and normalization process (e.g., "single service", "service" and "abstract service" represent barely the same concept) the most fundamental concepts were defined as: *Service*, *Port*, *Message*, *Operation*, *Behaviour*, *Context* and *NFR*.

*RQ3. Do these approaches deal with NFRs?*

For the types of NFRs (RQ3.1), we include a first criterion just to know if the approach *deals with NFRs* or not. Then, we observed in the primary studies that four NFR types were the most recurrent and we considered them in the value's domain: *Security*, *Reliability*, *Performance* and *Dependability*. We also included two extra categories: *Generic*, for those papers that deal with NFRs in a generic way; and *Other*, for those papers that deal with NFRs that appeared just occasionally in our study. This last value facilitates the subsequent data analysis. If the approach is *Generic*, then the other categories are considered included even if not explicitly marked.

For the *notations* (RQ3.2) we obtained the names of the languages that support NFRs.

### *3.6. Threats to validity*

There are a number of threats that might bias the validity of this MS. Our protocol tries to mitigate them with specific measures.

- *Definition of research questions.* The research questions may not cover all the relevant aspects that characterize the existing research in our area of interest. To minimize this risk, we used a brainstorming technique for defining them with the participation of all the authors of the study. It is worth to mention that the authors had knowledge and expertise on the three main topics of the study (SOA, MDD and NFR) prior undertaking the MS. Moreover, the initial set of research questions was refined, especially during the fast reading step, with the commonalities detected in the selected papers.

- *Identification of primary studies.* Since software engineering keywords are not standardized, the selected keywords and search string may not be able to capture all the relevant publications. To prevent the inadequacy of our search string, we added a reference analysis step to our search strategy to check if relevant papers referenced in our selected papers had been omitted. The papers that were subject of full reading were examined by a process of *snow-balling* (following up the references), looking for papers that could have evaded from our analysis. More precisely, the search strategy presented in Section 3.4 as linear went through a couple of iterations. The first search string that we applied was not the one presented in Section 3.2 but:

  (*meta-model* OR *metamodel* OR *model* language* OR *model driven* OR *MDA* OR *MDD* OR *MDE*) AND (*service oriented architecture* OR *SOA*)

  The *snow-balling* process revealed some relevant references that were not covered by our search string. A quick analysis of these referenced works showed that the search string had failed because it did not include all the variants of the term "SOA", and the only way to include them was to rebuild the search string and accept any paper with the term "service", which dramatically increased the number of papers to consider in the study (about 6 times the initial search). As a consequence, many papers captured by the new search string were discarded during the title and abstract consideration (only 20.6% of papers passed this step). Most of the work done with the papers selected in the first iteration was not lost since those papers were again part of the results. On the other hand, the search string covered adequately the works related to MDD according to the *snow-balling* process.

- *Identification of bibliographic sources.* The adequacy of the digital library was also validated through the above mentioned snow-balling step. In this respect, the analysis showed that we were not missing any relevant source. There is a risk of missing papers that are not indexed in the digital library used. To mitigate this risk, we checked year per year a relevant set of venues and found which editions were missing. For them, we performed a specific search using other digital libraries or even manually (the case of ICEIS). It could eventually happen that the set of venues could miss some relevant one. In fact, this happened with EDOC, which somehow surprisingly, was the venue with more selected works whilst it was not in that set. But even this case showed that the digital library is very complete, since all the editions were captured. Therefore, we think that it is very unlikely to have a major venue out of our list without a single relevant paper indexed in ISI WoS, again this is reinforced by the previously mentioned snow-balling step.

- *Selection of primary studies.* Since some steps of the selection of papers included in the search strategy may be based on personal judgments, there exists a risk of having a biased selection. Following the suggestions by [14], inclusion and exclusion criteria lists guided the selection and a multistage process involving several researchers for each paper was used to perform it (see Section 3.4).

- *Determination of ambiguous classification criteria.* To classify papers according to their nationality and orientation (academic or industrial), there exists the difficulty that a paper may have several authors with different nationalities and affiliations of different types (academic or industrial). We decided to take objective criteria for these classifications. Regarding orientation, we classified as academic those papers where all authors have an academic affiliation and as industrial those papers where at least one author is affiliated to a company or a non-academic institution (see Section 3.5). However, we must be aware that having an industrial author does not completely guarantee an industrial or practical orientation of a paper since, in some cases, companies have research positions. Regarding nationality, we decided to take the affiliation's country of the first author of the paper to disambiguate the classification but we are aware that other criteria could have been applied.

- *Identification of values for classification criteria.* For some of the criteria to classify the papers, the possible values were not obvious. Regarding the SOA type criteria, we defined the possible values according to the four major patterns presented by Erl [5] in order to use a consolidated framework. For the NFR type criterion, we took the possible values from the software product quality characteristics proposed by ISO/IEC 25010 [17] with a single addition: we added *dependability* because it emerged in several papers and corresponds to a combination of software quality characteristics. Regarding the metamodel concept criterion, a difficulty for identifying the possible values is the high diversity of concepts that may be proposed. Therefore, in this case, we took the terms for metamodel concepts that emerged from the data extraction process and, then, we unified those referring to similar metamodel concepts.

- *Classification of venues according to the venue topic.* The venues of the primary studies were classified according to their topic as *Service-oriented*, *Model-oriented*, *Software Engineering*, *General computing* or *Others* (see Section 3.5). This was done with the participation of all the authors of the study to mitigate the risk of venue misclassification.

- *Data extraction.* The data extraction process consisted on filling out a form for each of the selected papers with the most relevant information. Having a big picture of all the selected papers helped us analyse the results afterwards. For each of the fields of the form, its meaning and the way it is measured was completely described as Kitchenham suggests in [14] so that all the reviewers used the same criteria and, also, it is clear to the readers of the study. We must take into account that some papers might report their underlying research work in an incomplete or poor way. In particular, this may affect the data extraction process of our study (e.g., a paper does not mention that a proposed approach was applied in a real case while it was). This risk could be mitigated by attempting to obtain more information from the authors of the papers. However, the number of publications managed in our study made this solution impractical. Thus, we must be aware that our MS assesses the state of the research as it has been published and not as it actually is.

## 4. Analysis of the Results

For the analysis of the results we performed a frequency analysis and a correlation analysis between all the criteria used to answer the research questions. Each analysis is followed by our interpretation of the result, whenever possible contrasted with results of other sources.

- *Frequency analysis.* For the frequency analysis we show the total number of papers for each category and the percentage of the valid amount of papers (e.g., for the criteria related to metamodel the percentage is calculated over the 46 papers with metamodels).
- *Correlation analysis.* For the correlations we show the result of the idependency test, the $p$-value[2]. The independency test used in this paper is the Fisher's Exact Test for Count Data. In our study all the contingency tables were small enough to run the Fisher's test in a reasonable time.

*AQ1. How publications in the topic are demographically distributed?*

*AQ1.1. In which type of venue are articles mostly published?*

**Frequency 1.** *Venue.* 98 papers (76.0%) were published in conferences and workshops (90 and 8 papers respectively) while the other 31 (24.0%) were published in journals.

In order to know whether this ratio is representative, we compared it with a sample of 14 systematic mappings that explicitly provide this information (see Appendix C). In these studies we found that the percentage of journals oscillates among 4.7% and 51.7% with an average of 25.9% and median being 20.0%, whilst the percentage of conferences[3] oscillates between 44.7% and 89.2% with an average of 71% and median being 72.7%. Therefore, we may argue that the percentage found in our domain of interest falls into the general trend in the software engineering discipline. In fact, if we take the SM with the closest topic to us, in this case aspect-oriented code generation [18], the percentages are very similar (27.9% journals and 72.1% conferences).

We have analysed the journals where the works were published using the ISI WoS JCR index (2012 edition). Only 4 out of the 31 journal papers are not indexed in JCR. The value of the 5-year impact factor of these journals (excluding 3 of them that entered recently) ranges from 2.374 down to 0.268 with an average of 1.351 (1.395 if weighted by the number of papers). The journals with more publications are *IST*, *IBM Systems Journal* and *SoSyM* with 4 papers each.

**Frequency 2.** *Topic.* 46 papers (35.7%) were published in SOC-related venues, while only 15 (11.6%) were published in MDD-related venues. The other papers were published in engineering (36, 27.9%), computing (29, 22.5%) and general science (3, 2.3%) venues.

The unbalance between the first two categories is a bit surprising, since this study encompasses both the service and the modelling areas, and the number of venues for services is by no means significantly greater than those for modelling. One possible interpretation is that the SOC perspective is prevalent over the MDD perspective when both are combined in a research work or in other words, MDD is seen as a subordinate instrument to achieve a research goal in the SOC field.

[2]In this study correlations are considered statistically significant when $p$-value $< 0.05$. But we keep some results that we considered interesting for the topic of study even when they have a $p$-value $\geq 0.05$.

[3]In the following, we will talk about *conferences* as shorthand of *conferences and workshops*.

Table 5: Contingency table for *Venue* and *Topic*

| | Conference | Journal | Sum |
|---|---|---|---|
| Computing | 18 (62.1%) | 11 (37.9%) | 29 (100.0%) |
| Engineering | 26 (72.2%) | 10 (27.8%) | 36 (100.0%) |
| Modeling | 9 (60.0%) | 6 (40.0%) | 15 (100.0%) |
| Science | 3 (100.0%) | 0 (0.0%) | 3 (100.0%) |
| Services | 42 (91.3%) | 4 (8.7%) | 46 (100.0%) |
| Sum | 98 (76.0%) | 31 (24.0%) | 129 (100.0%) |

More generally, it is interesting to remark that the proportion of papers published in venues specific of the field compared to general venues (engineering, computing, and science) is close to fifty-fifty. Also, the percentages differ depending on the topic (see Correlation 1).

**Correlation 1.** *Venue and Topic.* Only 8.7% of papers published in venues for services are journals, and 40.0% of papers published in venues for modelling are journals ($p$=0.009). The contingency table is presented in Table 5.

A possible explanation is that most of the journals for services are still young and the community has not yet adopted them widely in their publishing habits (e.g., because they took some time to have a JCR index).

We observed that journal papers in the topic of this MS align with the general belief that journal papers discussing on a software engineering methodological field should include a concrete method of work and some kind of tool support as indicator of maturity (See Correlation 2 and 3).

**Correlation 2.** *Method and Venue.* Only the 3.2% of papers that do not present a method are journal papers ($p$=0.004).

**Correlation 3.** *Tool and Venue.* Only the 16.7% of papers that do not provide a tool are journal papers ($p$=0.002).

*AQ1.2. How the number of publications has evolved over the years?*

**Frequency 3.** *Year.* The papers were distributed as shown in Figure 3, with a maximum of 22 papers (17.1%) in 2008 and a minimum of 3 (2.3%) in 2003.

We observe that most of the articles have been published between 2006 and 2008. There is no clear tendency to make assumptions about the evolution of the number of publications on the research topic in the future. Special care has to be taken with the apparent drop of papers in 2013 since it may be partly due to the fact that some papers were not available online in the ISI WoS at the time the search was performed. Nevertheless, this may be an indication that other research topics related to SOA are taking the lead (e.g., cloud computing), or that MDD for SOA is no longer a very active topic of research.

*AQ1.3. Which is the geographical distribution of published articles related to the topic?*

**Frequency 4.** *Country.* The distribution of papers by country is the following: Spain (17), Germany (16), United States (14), China (10), and France (9) (see Figure 4-left).

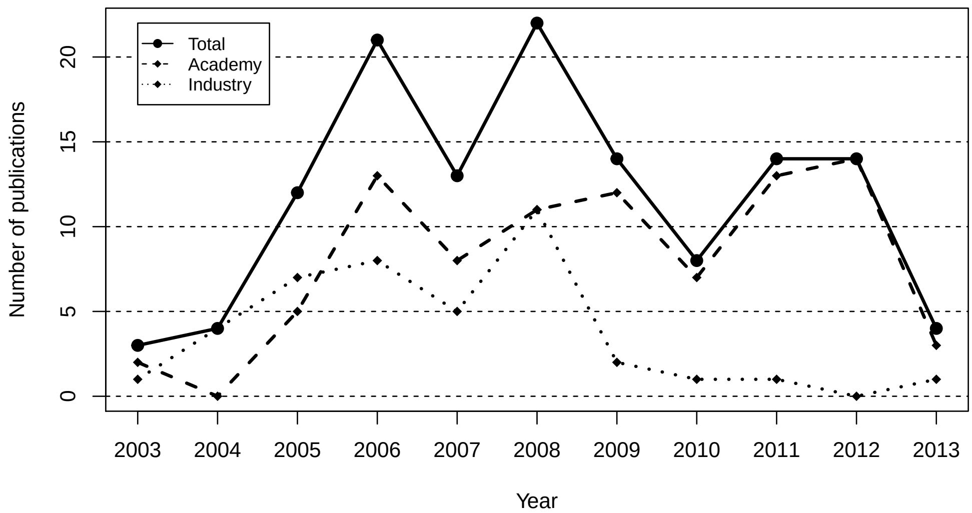


Figure 3: Evolution of publications over the years (and orientation)

It is interesting to note that a small number of countries concentrate the majority of the publications. In fact, the 5 listed countries have the 51.2% of all the analysed articles. From all of these, Spain stands out having 17 (13.2%) publications, followed by Germany with 16 (12.4%).

We have put these numbers into the general context of publications in Computing Science (see Figure 4-right) in the period 2006-2010 as presented in [19], also based in data from WoS. We can see that the 5 identified countries appear in the top 9 of this classification being probably the most remarkable change the higher position held by Spain (from 4.7% to 13.2% in our study). However if we take a look not to individual papers, but to research groups we observe that 10 out of 17 Spanish papers come from the same group. In fact, considering groups the ranking changes to: United States (10 groups), Germany (9), China (9), United Kingdom (7), and Spain and France (6).

When it comes to continents, it is worth to mention the high amount of papers published in Europe.

**Frequency 5.** *Continent.* The distribution of papers by continent is the following: Europe (81, 62.8%), Asia (21, 16.3%), North America (15, 11.6%), Australia (5, 3.9%), Africa (4, 3.1%), and South America (3, 2.3%).

These numbers diverge from the ones presented in [19] in the period 2006-2010, remarkably North America (31.64% in that study).

*AQ1.4. How are publications distributed between academy and industry?*

To answer this question we analysed whether at least one of the authors in each paper came from a non-academic institution. We made the assumption, thus, that the presence of one single

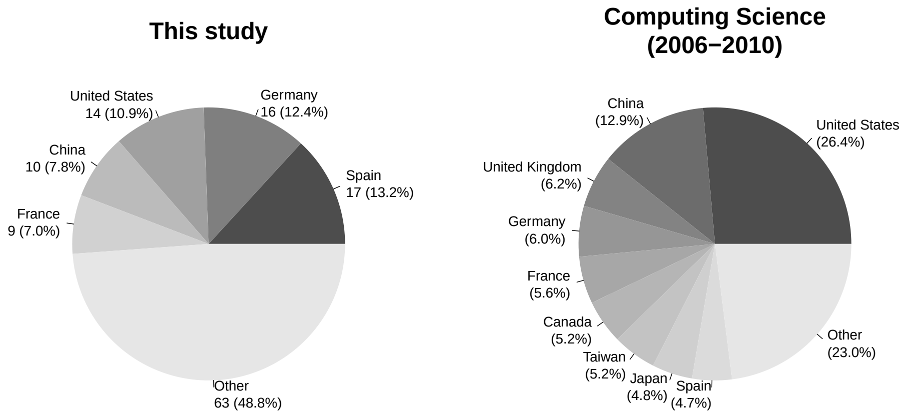


Figure 4: Classification of publications with respect to origin by country

industry author ensures at some extent that the paper will consider the practical implications of the proposal made.

**Frequency 6.** *Orientation.* 88 papers (68.2%) come from academy while 41 (31.8%) come from industry.

We observed two interesting correlations of orientation with country and year.

**Correlation 4.** *Orientation and Country.* We observed that, within the top countries, none of the European and Asiatic ones have more industrial papers than academic ones (with the extreme case of Spain with all the 17 papers being from academy), while papers from United States have far more papers from industry (see Figure 5). ($p$<0.001)

**Correlation 5.** *Orientation and Year.* We observed that the contributions from the academia were mostly published between 2006 and 2012 (88.6%), while contributions from the industry are concentrated between 2004 and 2008 (85.4%) (see Figure 3). ($p$<0.001)

As we can see in the Figure 3, starting from 2008, the great majority of the papers come from the academia, being the role of industry contributions marginal.

*RQ1. What are the characteristics of MDD approaches that support SOA?*

*RQ1.1. Which methods are applied by these approaches?*

**Frequency 7.** *Method.* From the 129 papers considered, 77 (59.7%) propose a new method (the paper defines and details its own method) while 28 papers (21.7%) adapt an existing method (the paper cites a method described in another paper and enhances it). The remaining 24 papers (18.6%) do not provide information about the method used.

**Frequency 8.** *Method name.* The 28 papers that adapt an existing method have a big dispersion on the original method that they adapt; for instance, the most mentioned ones (MDA, OOWS, SOD-M, VbDMF) are used in just 3 papers.

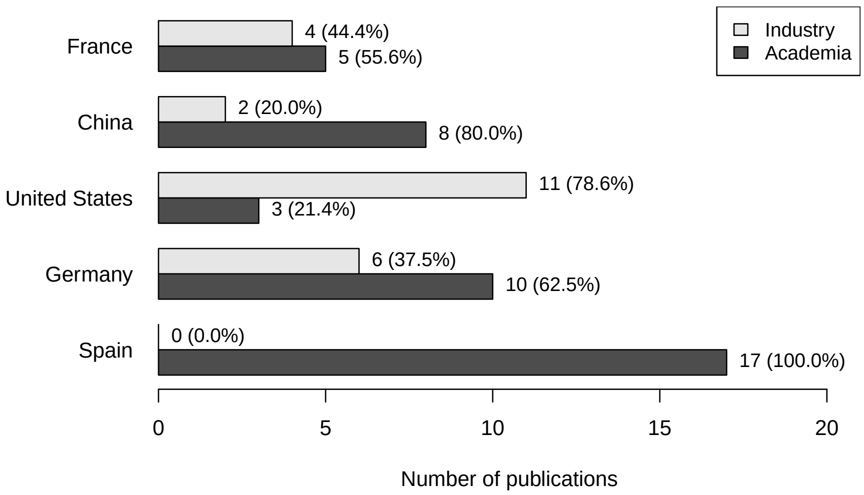


Figure 5: Classification of publications between academy and industry

At least three possible explanations to this low adoption of existing MDD methods exist: 1) this is a similar percentage as for any other software architecture domain (embedded systems, cloud architectures, etc.); 2) some SOA targets are still under consolidation so trying to apply MDD practices would be too premature; 3) there is some specific reason that prevents this adoption (inherent complexity of SOA systems, accentuated importance of quality requirements, etc.). Correlations 6 and 7 show that papers adapting/reusing methods use in a greater percentage the more consolidated types of SOA (single service) and output notations (web service description), which seems to support the second hypothesis.

**Correlation 6.** *Method and SOA type.* The 44.2% of papers that propose a new method have as target service composition. Similarly, when comparing to SOA types, we found that only the 17.9% of papers that adapt a method belong to the service composition SOA type (papers that belong to the service composition SOA type are the 34.9% of the total). ($p$=0.015)

**Correlation 7.** *Method and Output notation.* If we narrow the comparison to concrete notations, similar percentages and significance are found: 28.6% of papers adapting a method have a well-known technology as output, namely web service descriptions (although only the 17.1% of the total of papers have this output). On the other hand, the 35.1% of papers that propose a method have a service orchestration as output (papers with this output are only the 24.8% of the total of papers). ($p$=0.001)

*RQ1.2. What are the main characteristics of the MDD process?*
**Frequency 9.** *Transformation.* As illustrated in Figure 6-left, a great majority of papers present a top-down transformation direction (106, 82.2%).

**Frequency 10.** *Activity.* As illustrated in Figure 6-right, nearly all papers present a development activity (116, 89.9%) while the rest of categories altogether have 13 papers: Other (5), Specification (4), Integration (2), Engineering (1), and Migration (1).

This situation aligns with the classical setting in which software engineering methods are formulated and, thus, is not surprising. It clearly points out to the need of having more research for horizontal and bottom-up transformations. In fact, bottom-up transformation is directly related to reverse MDD which have been often identified as one challenge in the MDD state of the art and practice. Concerning activities, it is clear that in the context of SOC, aspects like service integration and application migration are fundamental and our study shows that they have been neglected by most of current MDD approaches.

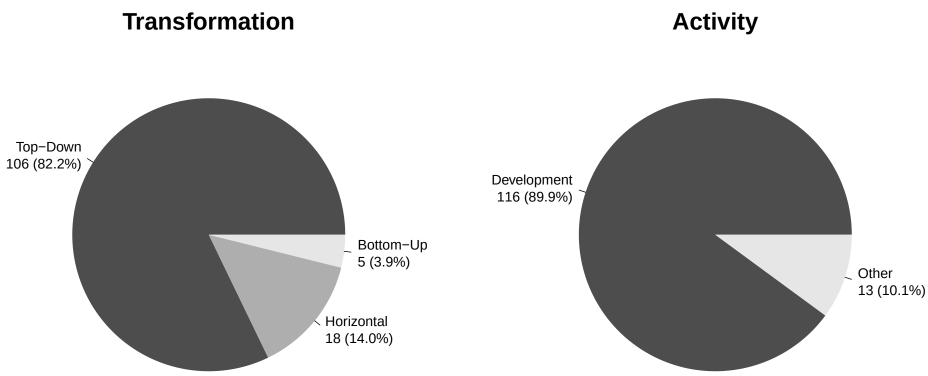


Figure 6: Number of publications in relation to *transformation* and *activity*

*RQ1.3. What are the input and the output of the MDD approach?*

**Frequency 11.** *Input Model Type.* 110 papers (85.3%) start with CIM or PIM (see Figure 7 for details). There are *other* input model types (3 papers, 2.3%) not shown in Figure 7.

**Frequency 12.** *Output Model Type.* 112 papers (86.8%) are about processes ending with code or PSM (see Figure 7 for details). There are *other* output model types (8 papers, 6.2%) not shown in Figure 7.

**Frequency 13.** *Input Notation.* 75 papers (58.1%) consider as input notation requirements, class diagrams or activity diagrams (see Figure 8 for details). There are *other* input notations (35 papers, 27.1%) not shown in Figure 8.

**Frequency 14.** *Output Notation.* 89 papers (69.0%) have an output expressed using a programming language, a service orchestration description or a web service description (see Figure 8 for details). There are *other* output notations (35 papers, 27.1%) not shown in Figure 8.

If we observe the distribution of *Input Model Type* and *Input Notation*, we see that most papers describe approaches that start at the upper abstraction levels and finish at the lower ones (see evolution of the lines in Figure 7). Figure 8 shows input bars located mainly at the bottom area (corresponding to notations oriented to abstract documents) and output bars at the top.

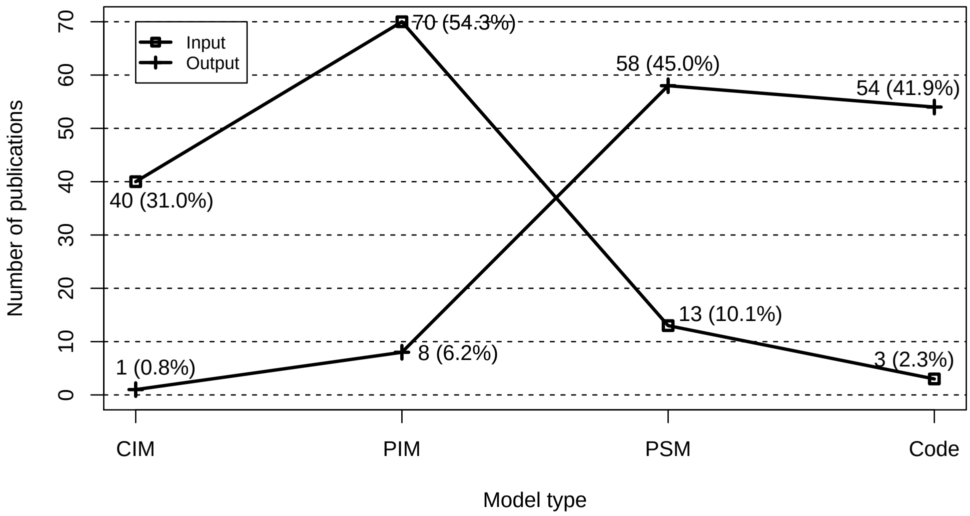


Figure 7: Number of publications in relation to input and output model types

There is certainly a correlation between models and notations, see Correlation 8 and 9.

**Correlation 8.** *Input model type and Input Notation.* 27.5% of papers starting at CIM use activity diagrams and 27.5% use requirements while 48.6% of papers starting at PIM use class diagrams and 14.3% use activity diagrams. ($p$<0.001)

**Correlation 9.** *Output model type and Output Notation.* 64.8% of papers ending at *code* use a programming language and 16.7% use web service description while 44.8% of papers ending at PSM use service orchestration and 19.0% use web service description. ($p$<0.001)

The predominance of top-down development (thus, starting at an abstract level and ending at a concrete one) is consistently observed in Correlation 10. This observation aligns with the observations made in the previous RQ and consequently, the lines of future research that we have identified in it have a counterpart here (e.g., if we talk about reverse engineering MDD support, this means having more approaches going from PSM/Code to CIM/PIM).

**Correlation 10.** *Input Model Type and Output Model Type.* 55.0% of papers starting at CIM end at PSM and 52.9% of papers starting at PIM end at *code*. Table 6 shows the contingency table for this correlation. ($p$<0.001)

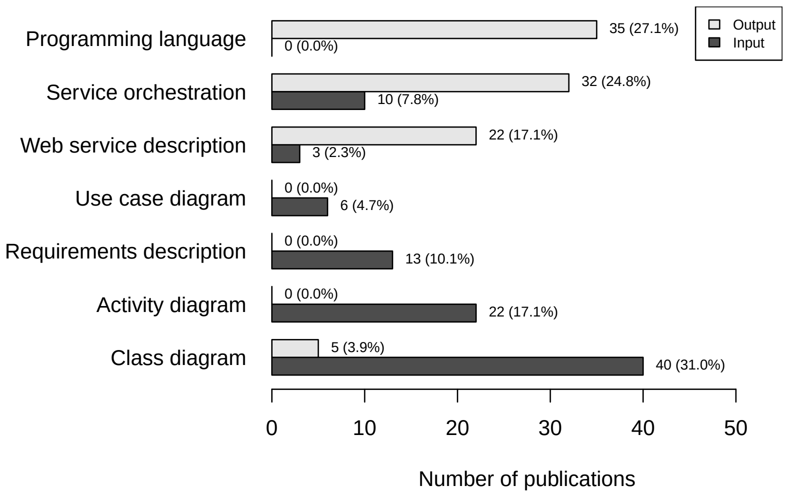


Figure 8: Number of publications in relation to input and output notations

Table 6: Contingency table for *Input Model Type* (rows) and *Output Model Type* (cols)

| | CIM | PIM | PSM | Code | Other | Sum |
|---|---|---|---|---|---|---|
| CIM | 0 (0.0%) | 4 (10.0%) | 22 (55.0%) | 11 (27.5%) | 3 (7.5%) | 40 (100.0%) |
| PIM | 0 (0.0%) | 2 (2.9%) | 29 (41.4%) | 37 (52.9%) | 2 (2.9%) | 70 (100.0%) |
| PSM | 0 (0.0%) | 2 (15.4%) | 7 (53.8%) | 4 (30.8%) | 0 (0.0%) | 13 (100.0%) |
| Code | 1 (33.3%) | 0 (0.0%) | 0 (0.0%) | 2 (66.7%) | 0 (0.0%) | 3 (100.0%) |
| Other | 0 (0.0%) | 0 (0.0%) | 0 (0.0%) | 0 (0.0%) | 3 (100.0%) | 3 (100.0%) |
| Sum | 1 (0.8%) | 8 (6.2%) | 58 (45.0%) | 54 (41.9%) | 8 (6.2%) | 129 (100.0%) |

*RQ1.4. Do the approaches report tools?*

**Frequency 15.** *Tool.* 50 papers (40.3%) propose a new tool, 24 (19.4%) adapt an existing one, and 50 (40.3%) are not proposing any tool support.

**Frequency 16.** *Tool platform.* Considering only the adapted tools, there are two predominant platforms: Eclipse is used in 10 works (41.7%) and IBM Rational tools in 5 works (20.8%). The rest of adapted tools are just used in one platform or were not mentioned.

We do not know of any study in the field to compare with, but the percentage of papers not using tool support seems a bit too high for the field of study.

**Frequency 17.** *Tool Input.* The total number of works using UML as input is 30 (41.7% of the papers reporting a tool). The rest of categories for tool input reach percentages under 6.9%.

**Frequency 18.** *Tool Output.* When considering the output coming from the tools we found a high diversity. To deal with this diversity we have codified the criterion as multivalued (i.e., one tool may have more than one output), and then selected the most recurrent ones. The most recurrent were: 22 BPEL (30.6% of the papers reporting a tool), 16 WSDL (22.2%), 17 Java (23.6%).

These results confirm that the predominance of UML models as input for the MDD processes, still holds when it comes to services. The diversity in the output can be explained because of its lower level of abstraction, being then necessary to match to the platform being addressed by the approach.

We observed that United States' proposals tend to put their work in practice by materializing it into a tool. Also, they tend to reuse or adapt tools instead of creating new ones (see Correlation 11).

**Correlation 11.** *Tool and Country.* The 29.2% of papers that adapt an existing tool come from the United States while papers from that country are only the 10.9% of the total. Only the 4.0% of papers that do not provide a tool come from North America while the 62.0% of them come from Europe. ($p$<0.001)

*RQ1.5. Have the approaches been applied to industrial cases?*

The frequency shows how industrial cases are still scarce in this topic.

**Frequency 19.** *Example.* From the 129 selected papers, 21 (16.3%) report a case study, 83 (64.3%) present an academic example and 25 (19.4%) do not present any example.

An observation to be made is that papers with at least one author from industry present case studies more frequently than papers from academy (see Correlation 12).

**Correlation 12.** *Example and Orientation.* While the total percentage of papers that present a case study is 16.3%, the percentage of papers that present a case study with authors from industry is 38.1%. ($p$=0.627, please note that this correlation is not statistically significant, it may have occurred by chance.)

*RQ2. What SOA systems are supported?*

*RQ2.1. What types of SOA are generated by the MDD approach?*

**Frequency 20.** *SOA Type.* Figure 10-left shows how the two most used types of SOA are targeted almost the same: SOA system in 51 papers (39.5%) and Service composition in 45 papers (34.9%), then comes Single service with 23 papers (17.8%) and the last one, SOA enterprise, is quite neglected with 10 papers (7.8%).

It is remarkable to find this low percentage of papers targeting SOA enterprises, considering that in fact this type of SOA is the preferred one by practitioners in percentage (see Correlation 13). The two facts together seem clearly to point out the need of future research for SOA enterprises.

**Correlation 13.** *SOA Type and Orientation.* 50.0% of papers that have SOA Enterprise as target were produced by papers authored by at least one industrial author. Figure 10-right give more details of this correlation. ($p$=0.530, please note that this correlation is not statistically significant, it may have occurred by chance.)

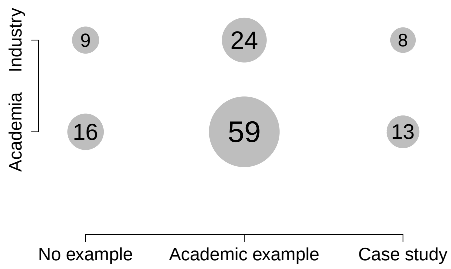


Figure 9: Relationship between example and orientation

We also studied the chronological perspective of the issue in Correlation 14.

**Correlation 14.** *SOA Type and Year.* We have observed that MDD for a single service had more publications in 2006 (26.1%) while service composition have been most published in 2008 and 2011 (15.6%, 15.6%), and SOA enterprise in 2008 (40.0%). The SOA system as the target of MDD process has been actively published in 2006, 2008, and 2012 (17.6%, 17.6%, 19.6%). See Figure 11 for more details. ($p$=0.024)

We can observe that research on single services is not a timely research topic at least in the context of MDD: there are more papers published in the 4-year interval 2003-2006 than in the 7-year interval 2007-2013. On the other hand SOA systems and service composition do not show any significant trend, with peaks in the last couple of years which show that they still attract interest.

We have also explored the relation of type of SOA with the inputs and outputs that were reported in RQ1.3. See Correlation 15, 16, 17 and 18, and Figure 12 and 13 (categories with low occurrences have been omitted in these figures).

**Correlation 15.** *SOA Type and Input Model Type.* PIM is the predominant type of input for single service (17, 73.9%), service composition (25, 55.6%), and SOA system (28, 54.9%) while CIM is more predominant for SOA enterprise (8, 80.0%). It is also relevant to remark that 20 (39.2%) papers with SOA system as target use CIM as input for the MDD process. See Figure 12-left. ($p$<0.001)

**Correlation 16.** *SOA Type and Output Model Type.* Code is the predominant type for single service (12, 52.2%) and SOA system (27, 52.9%), for service composition it is PSM (29, 64.4%), whilst SOA enterprise does not have any dominant type of model. See Figure 12-right. ($p$=0.002)

We can observe that single service and SOA system tend to transform from PIM to code, while service composition seems to work in a more abstract level by ending in the PSM. Despite of the low number of publications, we observe that SOA enterprise always starts from the CIM.

If we look into the details of notations, we find some interesting correlations.

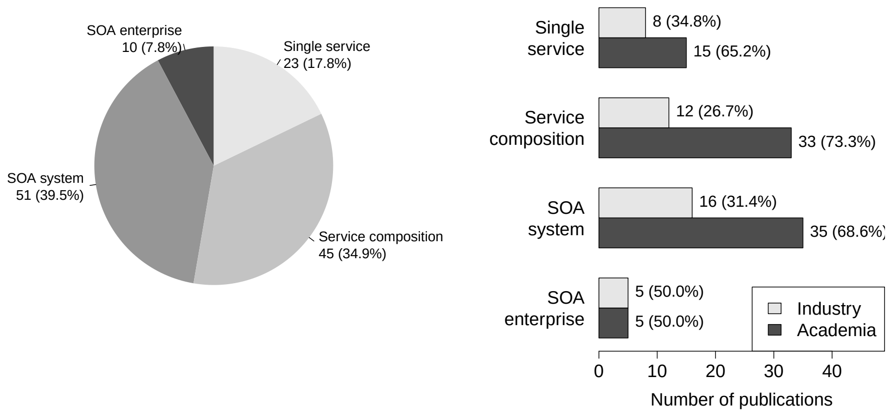


Figure 10: SOA types (left) and SOA types classified by *Orientation* (right)

**Correlation 17.** *SOA Type and Input Notation.* Class diagrams are predominant for single service (11, 47.8%) and SOA system (19, 37.3%), while activity diagrams are for more predominant for service composition (11, 24.4%). See Figure 13-left. ($p$=0.037)

The numbers in Correlation 17 are aligned with the nature of the SOA types, since single service and SOA system do not focus on behavioural aspects, whilst service composition does and therefore an activity diagram is appropriate.

**Correlation 18.** *SOA Type and Output Notation.* Web service descriptions (7, 30.4%) and programming languages (9, 39.1%) are predominant for single service, whilst programming languages is for SOA system (16, 31.4%). Remarkably, but also expected, service orchestration is predominant for Service composition (23, 51.1%). See Figure 13-right. ($p$<0.001)

Afterwards, we took a close look considering the different possibilities for the *method* criteria and found that the type of SOA and the type of method are highly dependent (see Correlation 19).

**Correlation 19.** *SOA Type and Method.* The approaches that target service composition mostly propose new methods (34, 75.6%), while single service and SOA enterprise tend to adapt existing methods more than the other two types of SOA (11, 47.8%; 4, 40.0%). Figure 14 gives more detail about this correlation. ($p$=0.015)

Last, for tool support, we found out that single services seem easier to implement than the other types of SOA (see Correlation 20).

**Correlation 20.** *SOA Type and Tool.* Whilst the most frequent option found in the study is to present manual approaches (without tool), in the case of papers with single service as target of the MDD process, it is more common for them to have tool support (15, 65.2%). ($p$=0.066, please note that this correlation is not statistically significant, it may have occurred by chance.)

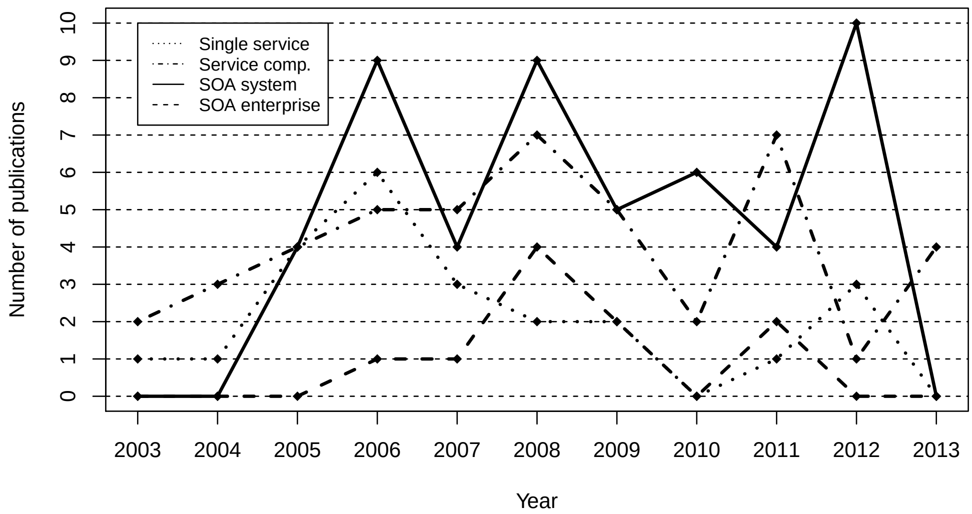


Figure 11: Number of publications in relation to type of SOA over the years

*RQ2.2. What SOA elements are explicitly modelled in the MDD process?*

**Frequency 21.** *Metamodel.* Up to 83 papers (64.3%) did not present metamodels for SOA (i.e., the metamodel used was not explained in the paper and/or was not specific for SOA). For the 46 (35.7%) that provided a description of a metamodel specific for SOA, the majority (26, 56.5% of these 46) defined a new metamodel, 12 (26.1%) adapted an existing one and 8 (17.4%) explained the metamodel but it was not theirs, it was reused.

The majority of contributions did not propose or describe a metamodel for SOA. The extent to what it is possible to apply a MDD method in the SOA domain without a metamodel specific for SOA remains open. For those that proposed a metamodel, the majority opted by defining their own proposal instead of adapting or reusing existing metamodels, being papers with industry authors the exception to the rule (see Correlation 21).

**Correlation 21.** *Metamodel and Orientation.* The 62.5% (5 out of 8) of papers that reused a metamodel had some author from industry, whilst the 73.1% (19 out of 26) of papers that defined a new metamodel are from researchers. ($p$=0.209, please note that this correlation is not statistically significant, it may have occurred by chance.)

If we consider the 20 papers that reuse or adapt existing metamodels, we can analyse whether some existing metamodel is widely adopted by the community:

**Frequency 22.** *Metamodel Name.* The WSDL metamodel (including “MVWSDL” and “Q-WSDL”) is the most adapted or reused (6, 30.0%) and PIM4SOA is the second one (3, 15.0%). The remaining adapted or reused metamodels only appear in one work each.

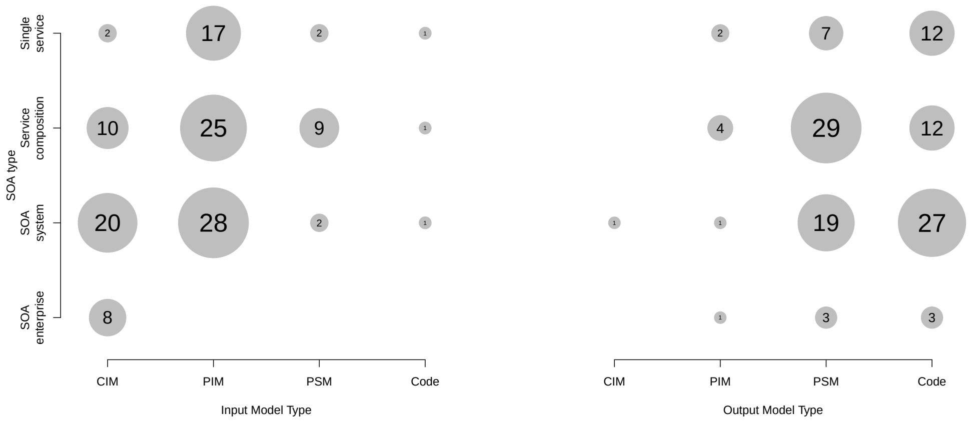


Figure 12: Relationship of type of SOA target with input/output model type

Table 7: Correlations considering papers that propose metamodel for SOA

| Correlation | Observation | Compared to | Test |
|---|---|---|---|
| *Input Model Type* | The 60.9% of papers with metamodel for SOA start at PIM | The total starting at PIM is 54.3% | $p$=0.330 |
| *Output Model Type* | The 52.2% of papers with metamodel for SOA end at code | The 47.0% of papers without metamodel end at PSM | $p$=0.047 |
| *Input Notation* | The 34.8% of papers with metamodel for SOA use class diagrams as input | The total using class diagrams is 31.0% | $p$=0.609 |
| *Output Notation* | The 37.0% of papers with metamodel for SOA end producing code in some programing language | The 31.3% of papers without metamodel produce a service orchestration | $p$=0.118 |

It is clear that contrary to what happens with other criterion (models, tools, etc.), metamodels for SOA (at least the ones used for MDD) are very diverse and no particular metamodel is dominant. Even in the case of PIM4SOA, it happens that the 3 papers share common authors.

The adoption of a metamodel for SOA creates some trends on other criterion (see Table 7, please note that most correlations are not statistically significant): the process tends to start at the PIM level using class diagrams as input and producing code as output written in some programming language. It is remarkable that the adoption of metamodel does not directly impact on the automation of the MDD process with more tool support.

If we consider the type of metamodel (reused, adapted or new), just a few correlations deserve some mention (see Table 8, please note that these correlations are not statistically significant). Remarkably, the first correlation in Table 8 could point out a benefit of working with adapted/reused metamodels, namely improved ability of generating code.

As mentioned in Section 3.5, we consolidated the concepts presented in the papers so that the values of this criterion have to be considered categories of terms rather than concrete terms. For example, “service” includes “single service”, “abstract service”, and other similar terms; and “port” includes “interface”, “connection”, etc.


Figure 13: Relationship of type of SOA target with input/output notations

Table 8: Correlations considering the type of metamodel for SOA

| Correlation | Observation | Compared to | Test |
|---|---|---|---|
| *Output Model Type* | The 50.0% (13 out of 26) of papers that propose new metamodel end at code level | The 55.0% (11 out of 20) of papers with adapted/reused metamodel end at code level | $p$=1.000 |
| *Input Notation* | The 26.9% (7 out of 26) of papers that propose new metamodel start with an activity diagram | The 5.0% (1 out of 20) of adapted/reused metamodels start with an activity diagram | $p$=0.140 |
| *Output Notation* | Papers that propose new metamodels have diverse output notation: WSDL (5, 19.2%), programming language (9, 34.6%), service orchestration (3, 11.5%), and other (9, 34.6%) | Papers with adapted/reused metamodel have programming language as predominant output (8 out of 20, 40.0%) | $p$=0.386 |

**Frequency 23.** *Metamodel Concepts.* Figure 15 shows the most recurrent concepts in the papers that present a SOA metamodel. It is worth to mention that even being in the context of MDD for SOA, the "service" concept does not appear in 8 of the papers that include a SOA metamodel (82.6% of the papers include this concept).

The typical situations where "service" concept is missing are the ones in which the metamodel does not pretend to model the whole SOA architecture but a particular aspect (e.g., the service features, the activities of the services, etc.). We found few remarkable correlations related to the concepts that appear in the metamodels (see Table 9, please note that these correlations are not statistically significant).

*RQ3. Do these approaches deal with NFRs?*

**Frequency 24.** *NFR.* From the total 129 studied papers, only 31 (24.0%) offer some support to NFRs. The rest do not deal with NFRs in the MDD process.

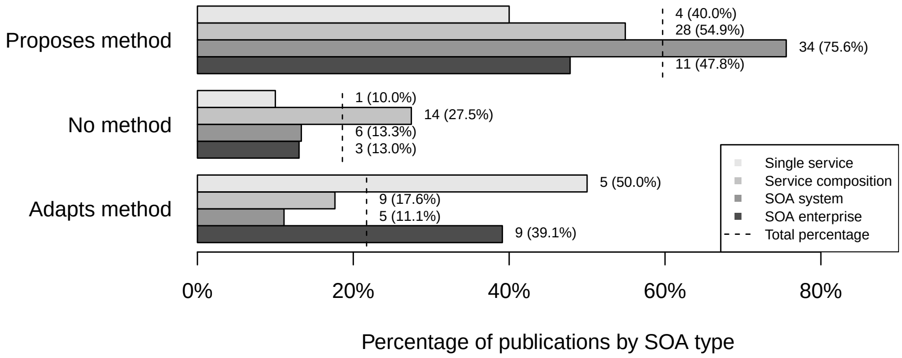


Figure 14: SOA types classified by *Method*

Table 9: Correlations considering the metamodel for SOA concepts

| Correlation | Observation | Test |
|---|---|---|
| *NFR concept and Input Model Type* | We already reported that in most cases the MDD starts with PIM, but we notice a tendency to start at the CIM when the metamodel has the "NFR" concept (40.0%), which is not surprising taking into account that requirements are normally considered part of CIM | $p$=0.090 |
| *NFR concept and Output Model Type* | Large majority (80.0%) of the approaches that have the "NFR" concept in their metamodel finish at code level. This could be an indicator that researchers interested in NFRs want to explore their effect in code | $p$=0.066 |
| *Behaviour concepts and SOA Type* | Behaviour concepts does not exist at all when the target is single service. This is normal, since the behaviour concepts are mainly used in composition of services | $p$=0.113 |

This may seem a low percentage but it is worthwhile to take into account that generally MDD approaches do not deal with NFRs [16]. Considering the great influence of NFRs in software systems, this indicator reflects the need to research more in the topic. A previous step could be to investigate the reason of this situation. Difficulty of dealing with NFRs may be a reason, as pointed out by the use of examples simpler than usual in the publications found.

*RQ3.1. What types of NFRs are supported by the MDD approach?*

**Frequency 25.** *NFR type.* As shown in Figure 16, 19 papers (61.3% of the 31 papers dealing with NFRs) deal with security, then the other three more recurrent types are reliability (10 papers, 32.3%), performance (7, 22.6%) and dependability (3, 9.7%). 4 papers (12.9%) introduce other type of NFR (e.g., accuracy, or non-technical NFRs like cost), and 5 papers (16.1%) present a generic approach to NFRs. Including the papers with generic approach, 10 papers supported more than one type of NFR.

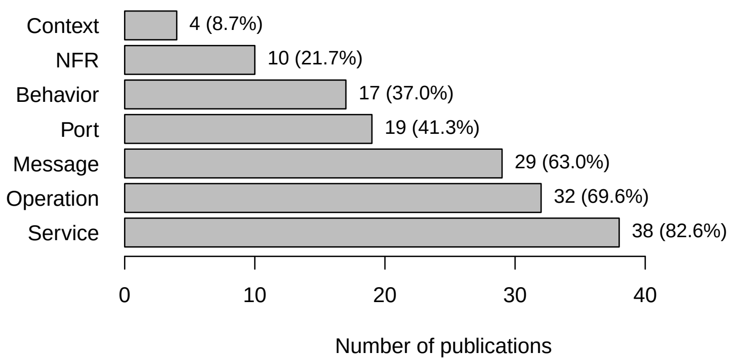


Figure 15: Recurrent metamodel for SOA concepts

The NFRs managed by these 31 papers are shown in Figure 16. Papers that handle Availability are included in Reliability (following the definition of the ISO/IEC 25010 [17]). Not many papers that presented a generic approach to deal with NFRs, and remarkably none of them had industry authors, which seems to point out that industry prefers to deal with more concrete situations that they find in their daily work. Security has been the most predominant type of NFR that was handled in the MDD for SOA, as most of these papers say, security is one of the major problems in SOA. For example: *"the involvement of independent trust domains constitutes the key aspect regarding security in service-oriented architectures"* [20] *"One of the fundamental characteristics of SOA is the usage and orchestration of services in different application contexts. However, this extensibility comes along with new security risks and threats that require an individual protection of each orchestrated service"* [21]. Comparing these results with the current state of practice as described in [22], practitioners are not so worried about SOA security which may be an indicator of miscommunication between academia and industry.

*RQ3.2. What notations are used to specify NFRs?*

Contrary to other parts of this study, notations for NFR show a dominant language:

**Frequency 26.** *NFR notation.* The most used notation for specifying NFRs was UML profile (14, 45.2%). Other notations such as OCL, WS-*, and Q-WSDL were used in 2 papers at most. In 7 papers (22.6%) the notation for NFRs was not described in the paper (even if the paper claimed its support) or the support for the NFR was handled by an NFR specific transformation.

## 5. Discussion

From the results analysis we can conclude that the research area of this mapping study is still not mature. There are four reasons for this assertion:

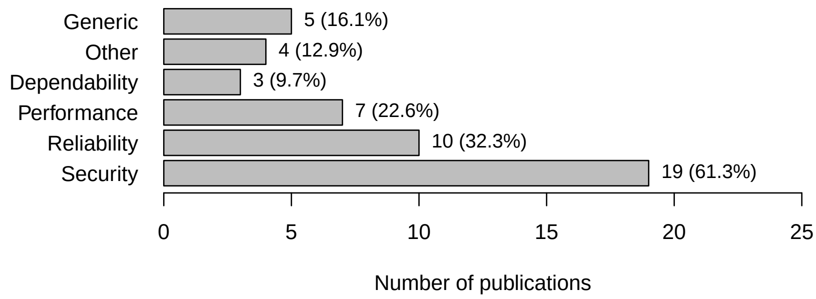


Figure 16: Types of NFRs handled in the studied papers

1. *There is a lack of consensus and standards.* First, the percentage of papers that adapt an existing method is not high and, more significantly, there is a high diversity on the methods that they take as a basis. Second, the percentage of papers that have their proposals implemented by means of an existing tool is not high. Finally, if we focus on those papers that present a metamodel for SOA, we find a lack of consensus on the metamodels used: most of them are created by the authors themselves and, in fewer cases, adapted or reused. In other words, the define-as-new vs. adapt/reuse option has consistently leaned towards definition in all possible cases (new methods, new tools, new metamodels). The absence of such standards and specifically of a commonly agreed metamodel definitively hampers the consolidation of the approaches which remain as isolated pieces that can hardly be combined for providing more holistic solutions. This situation could improve in the future if SoaML [3] finally gets consolidated.
2. *Many primary studies didn't present consolidated work.* Some observations indicate that the individual works still do not present work consolidate enough. Many of the papers present proposals which must be manually applied since they do not provide a tool and the number of papers that have applied their proposals to case studies is low. A remarkable exception are industry papers that represent a greater percentage of case studies hinting that maturity is greater in this particular category. Finally, we remark that most of the journals publishing these papers are not top-ranked in the ISI WoS JCR index, with just three journals (IST, IBM Systems Journals and Computers in Industry) appearing in the first quartile in their respective categories.
3. *Some research aspects are not sufficiently covered.* The scope of the research in this area should be widened since some important aspects have not received enough contributions. Although this could eventually happen because these aspects do not attract enough attention, we argue below our position that they should. First, the number of works that have a SOA enterprise target is low compared to other types of SOA systems. A possible reason is that the SOA enterprise target is the most ambitious and difficult SOA type since it tackles the whole (or a significant part of the) life cycle of the SOA adoption by an enterprise based on a service-oriented abstract model (or a service-oriented business model) for the enterprise.

Second, we have also found that most papers that have as target a service composition present proposals that produce as output a PSM but do not tackle the production of code. This situation is the consequence of the current theoretical focus of the papers addressing this type of SOA; more practical work following these theories should emerge in the near future. Third, if we focus on those papers that include a metamodel in their proposals, we find that a significant part of the papers with an own-made metamodel produce as output of their proposals a PSM and again do not tackle the production of code. By contrast, this situation does not appear in papers with reused metamodels since most of them address code production. It appears that the lack of consensus or standards on the metamodels to use damages the coverage of the implementation stage. Finally, the percentage of works that support NFRs is low in spite of the relevant influence of NFRs in software systems. In general, MDD approaches do not deal with NFRs and the reasons for this situation should be investigated.

4. *The work done outside the United States needs some more transfer to industry.* We observe that Europe concentrates the majority of the papers. However, if we compare Europe's academic and industry papers, there is a big difference of proportion with the United States where industry papers are majority. This unbalance between academy and industry also holds for the rest of the world. Moreover, the United States is the country with the greatest number of papers that include a metamodel, arguably due to the fact that they need to formalize their methods in order to be able to use them with tools. We can conclude that the research methodology applied in the US is more practical than in other countries and should serve as lighthouse for the upcoming research trend.

## 6. Conclusions

In this mapping study we have surveyed the state of the art in MDD processes used for developing SOA-based applications. We designed and followed a rigorous protocol which uncovered up to 129 papers to answer the different research questions that we identified. We may consider the answer to these questions as the main outcome of this paper. We summarize these answers below:

*AQ1. How publications in the topic are demographically distributed?*

*AQ1.1. In which type of venue are articles mostly published?* The approximate ratio of publication in journals with respect to conferences is 1 to 3. Works are published equitably in general-purpose venues and topic-related venues and from these latter, more in SOC-related venues than in modelling venues.

*AQ1.2. How the number of publications has evolved over the years?* Publication peaks occurred in 2006 and 2008. We cannot conclude that there is a general declining of the topic, but there is a clear tendency that confirms a pronounced declining of interest from the industry.

*AQ1.3. Which is the geographical distribution of published articles related to the topic?* Europe has a significantly greater presence than the other continents, being Spain, Germany, United States, China, and France the countries with more publications (although United States is the country with more different research groups).

*AQ1.4. How are publications distributed between academy and industry?* The publications written only by academic authors rule are by far more than the number of papers having at least one industry author, except in the particular of the United States.

*RQ1. What are the characteristics of MDD approaches that support SOA?*

*RQ1.1. Which methods are applied by these approaches?* No consolidated methods are applied in MDD approaches for SOA.

*RQ1.2. What are the main characteristics of the MDD process?* A great majority of MDD methods adopt a top-down transformation approach and are applied in software development activities.

*RQ1.3. What are the input and the output of the MDD approach?* A great majority of MDD methods use PIM as input model types, and in particular CIM for SOA enterprise. For output model types, PSM and code, and in particular PSM for service compositions and code for SOA systems. Class diagrams and activity diagrams are the most used input notations, and in particular class diagrams for single services. For output notations, programming languages and service orchestrations, and in particular programming languages for SOA system and service orchestrations for service composition.

*RQ1.4. Do the approaches report tools?* Although there are more approaches that report tool support that do not, the percentage of papers not reporting tools is significant (except when targeting single services). Eclipse and IBM Rational are the most adapted platforms, and UML the most usual input of the tools (especially for IBM Rational), whilst for the output of the tools there is more diversity.

*RQ1.5. Have the approaches been applied to real cases?* The majority just used academic examples or even no example at all, and just a few approaches were applied to real cases.

*RQ2. What SOA types are supported?*

*RQ2.1. What types of SOA are generated by the MDD approach?* Using Erl's classification [5], SOA systems and service compositions are the two most targeted types of SOA, whilst SOA enterprise the least targeted, but it seems that practitioners are more interested than researchers in SOA enterprise.

*RQ2.2. What SOA elements can be modelled in the MDD process?* One third of the approaches are based on metamodels. A majority of those, tend to define a new metamodel instead of adapting or reusing an existing one. The most recurrent concepts in the metamodels are *Service*, *Operation* and *Message*.

*RQ3. Do these approaches deal with NFRs?*

*RQ3.1. What types of NFRs are supported by the MDD approach?* Less than one fourth of the approaches deal with NFRs. A majority of those address a few specific types instead of providing a general solution. Security is by far the most targeted type, with reliability, performance and dependability also standing out from the rest.

*RQ3.2. What notations are used to specify NFRs?* The most used notation by far is the use of specific UML profiles.

### *6.1. Future Work*

Concerning future work, we are designing a plan to update the results in a yearly basis. This is a significant work, since statistical analysis needs to be redone when new publications are selected. Given the time that digital libraries take to get completely updated, it seems unavoidable to allow for a one-year gap. With this observation in mind, an appropriate update plan may be:

- End December 20nn: apply the search string over ISI WoS for the year 20(nn-1). For any selected venue whose 20(nn-1) papers are not incorporated in this database, use other search engines or even a manual search. Of course, the set of venues itself needs some constant monitoring (e.g., some new journal may arise).
- January 20(nn+1): process the selected papers.
- February 20(nn+1): update the statistical analysis.
- March 20(nn+1): update the written report.

As a first step towards this direction, we have automatized the calculation of the frequencies and correlations (and the related figures and tables) appearing in this paper using R and Latex together. With this integration we can regenerate the numbers that appear in the paper, which is the starting point to update the qualitative analysis.

Other future work involves the issues identified in the discussion section:

- Design and conduct systematic literature reviews for the research areas outlined in Section 5. In particular, further research may investigate whether the trend in defining new methods, tools and metamodels instead of reusing or adapting existing ones is particular of the topic of this study (and in this case, why) or it is a general situation in software engineering.
- Perform research to find out the bibliographic data needed to contextualize the results obtained whilst investigating AQ1. For instance, to know to what extent the percentage of industry papers or high-ranked JCR journals are different than in systematic mappings conducted over other domains.

## Acknowledgments

We thank Jordi Cabot, Antonio Vallecillo, and the anonymous reviewers of the IST journal for their valuable comments and suggestions. This work has been partially supported by the Spanish MICINN project TIN2013-44641-P.

## References

[1] M. P. Papazoglou, D. Georgakopoulos, Introduction: Service-oriented computing, Commun. ACM 46 (10) (2003) 24–28.

[2] S. Mellor, A. Clark, T. Futagami, Model-driven development - guest editor's introduction, Software, IEEE 20 (5) (2003) 14–18.

[3] OMG, Service oriented architecture modeling language (soaml), version 1.0.1 (2012).
URL `http://www.omg.org/spec/SoaML/1.0.1/PDF`

[4] B. A. Kitchenham, D. Budgen, O. Pearl Brereton, Using mapping studies as the basis for further research - a participant-observer case study, Inf. Softw. Technol. 53 (6) (2011) 638–651.

[5] T. Erl, Service-Oriented Architecture: Concepts, Technology, and Design, Prentice Hall PTR, Upper Saddle River, NJ, USA, 2005.

[6] C. M. MacKenzie, K. Laskey, F. McCabe, P. F. Brown, R. Metz, Reference model for service oriented architecture 1.0, oASIS Standard (2006).
URL `http://docs.oasis-open.org/soa-rm/v1.0/`

[7] M. Papazoglou, Web Services and SOA: Principles and Technology, 2nd Edition, Prentice-Hall, 2012.

[8] H. Müller, Soa governance, in: Proceedings of the 2009 Conference of the Center for Advanced Studies on Collaborative Research, CASCON '09, IBM Corp., Riverton, NJ, USA, 2009, pp. 339–340.

[9] A. Keller, G. Kar, H. Ludwig, A. Dan, J. Hellerstein, Managing dynamic services: a contract based approach to a conceptual architecture, in: Network Operations and Management Symposium, 2002. NOMS 2002. 2002 IEEE/IFIP, 2002, pp. 513–528.
[10] M. Papazoglou, W.-J. van den Heuvel, Service oriented architectures: Approaches, technologies and research issues, The VLDB Journal 16 (3) (2007) 389–415.
[11] OMG, MDA Guide Version 1.0.1 (Jun. 2003).
URL `http://www.omg.org/cgi-bin/doc?omg/03-06-01.pdf`
[12] C. Atkinson, T. Kühne, Model-driven development: A metamodeling foundation, IEEE Software 20 (5) (2003) 36–41.
[13] B. A. Kitchenham, S. L. Pfleeger, L. M. Pickard, P. W. Jones, D. C. Hoaglin, K. E. Emam, J. Rosenberg, Preliminary guidelines for empirical research in software engineering, IEEE Trans. Softw. Eng. 28 (8) (2002) 721–734.
[14] B. Kitchenham, S. Charters, Guidelines for performing Systematic Literature Reviews in Software Engineering, Tech. Rep. EBSE 2007-001, Keele University and Durham University Joint Report (2007).
URL `http://www.dur.ac.uk/ebse/resources/Systematic-reviews-5-8.pdf`
[15] K. Petersen, R. Feldt, S. Mujtaba, M. Mattsson, Systematic mapping studies in software engineering, in: Proceedings of the 12$^{th}$ International Conference on Evaluation and Assessment in Software Engineering, EASE'08, British Computer Society, Swinton, UK, UK, 2008, pp. 68–77.
[16] D. Ameller, X. Franch, J. Cabot, Dealing with non-functional requirements in model-driven development, in: Requirements Engineering Conference (RE), 2010 18$^{th}$ IEEE International, 2010, pp. 189–198.
[17] I. F. 25010, System and software engineering systems and software quality requirements and evaluation (square): System and software quality (2010).
[18] A. Mehmood, D. N. A. Jawawi, Aspect-oriented model-driven code generation: A systematic mapping study, Inf. Softw. Technol. 55 (2) (2013) 395–411.
[19] F. Ruiz, La investigación en informática en españa. análisis bibliométrico., Novatica 215 (2012) 54–58.
[20] M. Menzel, I. Thomas, C. Meinel, Security requirements specification in service-oriented business process management, in: International Conference on Availability, Reliability and Security (ARES), 2009, pp. 41–48.
[21] M. Menzel, R. Warschofsky, C. Meinel, A pattern-driven generation of security policies for service-oriented architectures, in: IEEE International Conference on Web Services (ICWS), 2010, pp. 243–250.
[22] D. Ameller, M. Galster, P. Avgeriou, X. Franch, The role of quality attributes in service-based systems architecting: A survey, in: Proceedings of the 7$^{th}$ European Conference on Software Architecture, ECSA'13, Springer-Verlag, Berlin, Heidelberg, 2013, pp. 200–207.

## Appendix A: Sources and databases

Table 10: Journals

| Venue | Database | Years |
|---|---|---|
| Computer (IEEE) | WoS | 2003-2013 |
| DKE | WoS | 2003-2013 |
| IBM Systems Journal[a] | WoS | 2003-2008 |
| IBM J. of Res and Dev. | IEEE | 2003-2010 |
| | WoS | 2011-2013 |
| IJCIS | WoS | 2003-2013 |
| Internet Comp. (IEEE) | WoS | 2003-2013 |
| IST | WoS | 2003-2013 |
| JSS | WoS | 2003-2013 |
| REJ | Springer | 2003 |
| | WoS | 2004-2013 |
| SOCA (Springer) | Springer | 2007-2013 (2007 1st ed.) |
| Software (IEEE) | WoS | 2003-2013 |
| SoSyM | Springer | 2003-2007 |
| | WoS | 2008-2013 |
| TOSEM (ACM) | WoS | 2003-2013 |
| TSC (IEEE) | IEEE | 2008 (2008 1st ed.) |
| | WoS | 2009-2013 |
| TSE (IEEE) | WoS | 2003-2013 |
| TWeb (ACM) | WoS | 2007-2013 (2007 1st ed.) |

[a] In 2008 merged with IBM J. of Res and Dev.

Table 11: Conferences

| Venue | Database | Years |
|---|---|---|
| CAiSE | WoS | 2003-2010 |
| | Springer | 2011-2013 |
| ECMFA[a] | WoS | 2005-2006 (2005 1st ed.) |
| | Springer | 2007 |
| | WoS | 2008-2011 |
| | Springer | 2012-2013 |
| ECOWS[b] | WoS | 2003-2009 |
| | IEEE | 2010-2011 |
| ESOCC[c] | Springer | 2012-2013 (2012 1st ed.) |
| ER | WoS | 2003-2011 |
| | Springer | 2012-2013 |
| ESEC/FSE | ACM | 2003-2013 (biannual) |
| FASE | WoS | 2003-2011 |
| | Springer | 2012-2013 |

| | | |
|---|---|---|
| ICEIS | SCITEPRESS | 2003-2005 |
| | WoS | 2006-2009 |
| | SCITEPRESS | 2010-2013 |
| ICMT | ISI WoS | 2008-2010 (2008 1st ed.) |
| | Springer | 2011-2013 |
| ICSE | WoS | 2003-2005 |
| | ACM | 2006 |
| | WoS | 2007-2009 |
| | ACM | 2010 |
| | WoS | 2011-2013 |
| ICSOC | WoS | 2003 |
| | ACM | 2004 |
| | WoS | 2005-2010 |
| | Springer | 2011-2013 |
| ICWS | WoS | 2003-2007 |
| | IEEE | 2008 |
| | WoS | 2009 |
| | IEEE | 2010-2013 |
| MoDELS[d] | WoS | 2003-2010 |
| | Springer | 2011-2012 |
| | WoS | 2013 |
| OOPSLA | ACM | 2003-2006 |
| | WoS | 2007-2009 |
| | ACM | 2010 |
| | WoS | 2011-2013 |
| RE | WoS | 2003-2009 |
| | IEEE | 2010 |
| | WoS | 2011 |
| | IEEE | 2012-2013 |
| SCC | WoS | 2004-2009 (2004 1st ed.) |
| | IEEE | 2010-2013 |
| ServiceWave | WoS | 2008-2010 (2008 1st ed.) |
| | Springer | 2011 (2011 last ed.) |
| Services (IEEE) | WoS | 2008-2009 (2008 1st ed.) |
| | IEEE | 2010-2013 |
| SOCA[e] | WoS | 2007 (2007 1st ed.) |
| | IEEE | 2019-2011 |
| | WoS | 2012 |
| | IEEE | 2013 |

[a] Previously known as ECMDA-FA.
[b] In 2003 the name was ICWS-Europe.
[c] Previously known as ECOWS.
[d] Till 2004 known as UML.
[e] In 2008 it was not held.

**Appendix B: Criteria definition**

| Criterion | Description | Nature |
|---|---|---|
| Venue | Bibliographic source, i.e. journal or event (e.g., conference, workshop) proceedings, where the work has been published | Emergent |
| Topic | One of: Service-oriented, Model-oriented, Software engineering, General computing, Others | Predefined |
| Year | Year of publication | Predefined |
| Country | Affiliations country of first author | Predefined |
| Continent | Derived from the criterion above | Predefined |
| Orientation | Academy if all authors are affiliated to research institutions; Industry otherwise | Predefined |
| Method | One of: Proposes new method, Adapts existing method, No method | Predefined |
| Method name | Applies only if the method is Adapts existing method. One of: ARIS, SOMA, OOWS, RUP, CAIDE, WebML, MDA, MIDAS, SOD-M, VbDMF, Not mentioned | Emergent |
| Transformation | In terms of the relationship between the models level. One of: Top-down, Bottom-up, Horizontal | Predefined |
| Activity | Type of MDD process described in the paper. One of: Engineering, Development, Specification, Integration, Migration, Others | Predefined |
| Input model type | Type of model of the MDD process input. One of: CIM, PIM, PSM, Code, Other | Predefined |
| Output model type | Type of model of the MDD process output. Values as above | Predefined |
| Input notation | Notation used to express the input. One of: Class diagram, Service orchestration, Web service description, Activity diagram, Use case diagram, Requirements, Other | Emergent |
| Output notation | Notation used to express the output. One of: Class diagram, Service orchestration, Web service description, Programming language, Other | Emergent |
| Tool | One of: Proposes new tool, Adapts existing tool, No tool | Predefined |
| Tool platform | Applies only if the method is Adapts existing tool. One of: Eclipse, IBM Rational tools, AndroMDA, Beans, ArgoUML, Not mentioned | Emergent |
| Tool input | One of: ADL, BPEL, BPMN, BPMN with extension, CAMLE, COM, ContextUML, DSL, DSLM, EPC, Excel, HL7-DICOM, i*, p-ADL, UML, UML4SPM, WebML, WSDL, XMI, XML, Not mentioned | Emergent |

| | | |
|---|---|---|
| Tool output | One or more of: Java, WSDL, BPEL, Others | Emergent |
| Tool output other | Applies only if the output is Others. One of: None, .Net, ARL, AspectJ, Code, EJB, Hibernate, J2EE, Jini, Jolie, MuleESB-GridFTP, No name, OWL, PPC, SQL, SRML, UML, WorkSCo, WS-SecurityPolice, WSFL, XML, XSD | Emergent |
| Example | One of: Case study, Academic example, No Example | Predefined |
| SOA type | One of: Single service, Service composition, SOA system, SOA enterprise | Predefined |
| Metamodel | One of: Propose new metamodel, Adapt existing metamodel, Reuse existing metamodel, No metamodel | Predefined |
| Metamodel name | Applies only if the method is Adapt / Reuse existing metamodel. Values are: UML4SOA, SOFM, SD, UP-SNFRs, PIM4SOA, UML, OOWS, WS-Security, Migration path metamodel, Q-WSDL, SOD-M Metamodel, SECTET, MVWSDL, WSFL, EDOC, UML Profile, CAMLE, SecureSOA, No name | Emergent |
| Metamodel concept | One or more of: Service, Port, Message, Operation, Behavior, Context, NFR | Emergent |
| NFR | One of: Has NFRs, No NFRs | Predefined |
| NFR type | Applies only if Has NFRs. One or more of: Security, Performance, Reliability, Dependability, Generic, Other | Emergent |
| NFR notation | UML Profile, Feature model, WS-*, OCL, Q-WSDL metamodel, SECTET-PL, Class diagram, and Not mentioned | Emergent |

### Appendix C: Other mapping studies

| Ref | Topic | Journals | Conferences | Others[a] |
|---|---|---|---|---|
| [1] | OO design | 68 (51.5%) | 59 (44.7%) | 5 (3.8%) |
| [2] | Software quality trade-offs | 38 (22.6%) | 130 (77.4%) | |
| [3] | Distances in RE | 19 (34.5%) | 34 (65.5%) | |
| [4] | AO code generation | 19 (27.9%) | 49 (72.1%) | |
| [5] | UML models | 16 (48.5%) | 17 (51.5%) | |
| [6] | Quality assurance | 10 (20.0%) | 40 (80.0%) | |
| [7] | Requirements modeling | 9 (19.6%) | 37 (80.4%) | |
| [8] | Software ecosystems | 8 (18.2%) | 32 (72.7%) | 4 (9.1%) |
| [9] | SPL evolution | 8 (10.8%) | 66 (89.2%) | |
| [10] | SOA testing | 6 (16.7%) | 25 (69.5%) | 5 (13.8%) |
| [11] | Formal methods in cross cutting | 4 (40.0%) | 6 (60.0%) | |
| [12] | SPL testing | 3 (6.7%) | 36 (80.0%) | 6 (13.4%) |
| [13] | SPL testing | 3 (4.7%) | 57 (89.1%) | 4 (6.2%) |

[a] This column includes book chapters, PhD thesis and technical reports.

### References of Appendix C


[1] J. Bailey, D. Budgen, M. Turner, B. Kitchenham, P. Brereton, S. Linkman, Evidence relating to object-oriented software design: A survey, in: Proceedings of the First International Symposium on Empirical Software Engineering and Measurement, 2007, pp. 482–484.

[2] S. Barney, K. Petersen, M. Svahnberg, A. Aurum, H. Barney, Software quality trade-offs: A systematic map, Inf. Softw. Technol. 54 (7) (2012) 651–662.

[3] E. Bjarnason, Distances between requirements engineering and later software development activities: A systematic map, in: Proceedings of the 19th International Conference on Requirements Engineering: Foundation for Software Quality, 2013, pp. 292–307.

[4] A. Mehmood, D. N. A. Jawawi, Aspect-oriented model-driven code generation: A systematic mapping study, Inf. Softw. Technol. 55 (2) (2013) 395–411.

[5] R. Pretorius, D. Budgen, A mapping study on empirical evidence related to the models and forms used in the uml, in: Proceedings of the Second ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, 2008, pp. 342–344.

[6] F. Elberzhager, J. Münch, V. T. N. Nha, A systematic mapping study on the combination of static and dynamic quality assurance techniques, Inf. Softw. Technol. 54 (1) (2012) 1–15.

[7] N. Condori-Fernandez, M. Daneva, K. Sikkel, R. Wieringa, O. Dieste, O. Pastor, A systematic mapping study on empirical evaluation of software requirements specifications techniques, in: Proceedings of the 2009 3rd International Symposium on Empirical Software Engineering and Measurement, 2009, pp. 502–505.

[8] O. Barbosa, C. Alves, A systematic mapping study on software ecosystems, in: Proceedings of the 3rd International Workshop on Software Ecosystems, 2011, pp. 15–26.

[9] M. A. Laguna, Y. Crespo, A systematic mapping study on software product line evolution: From legacy system reengineering to product line refactoring, Sci. Comput. Program. 78 (8) (2013) 1010–1034.

[10] M. Palacios, J. García-Fanjul, J. Tuya, Testing in service oriented architectures with dynamic binding: A mapping study, Inf. Softw. Technol. 53 (3) (2011) 171–189.

[11] R. Durelli, V. Durelli, A systematic mapping study on formal methods applied to crosscutting concerns mining, in: Proceedings of the ESELAW, 2012.

[12] P. A. Da Mota Silveira Neto, I. D. Carmo Machado, J. D. Mcgregor, E. S. De Almeida, S. R. De Lemos Meira, A systematic mapping study of software product lines testing, Inf. Softw. Technol. 53 (5) (2011) 407–423.

[13] E. Engström, P. Runeson, Software product line testing - a systematic mapping study, Inf. Softw. Technol. 53 (1) (2011) 2–13.

## Appendix D: Papers selected for the mapping study


[1] D. B. Abeywickrama, S. Ramakrishnan, Context-Aware Services Engineering: Models, Transformations, and Verification, ACM Transactions on Internet Technology 11 (3).

[2] M. A. Aboulsamh, Towards a model-driven approach for planning a standard-based migration of enterprise applications to SOA, in: Proceedings of the IEEE Congress on Services (SERVICES), 2009, pp. 471–474.

[3] R. Acerbis, A. Bongio, M. Brambilla, S. Butti, S. Ceri, P. Fraternali, Web applications design and development with WebML and WebRatio 5.0, in: Proceedings of the 46th International Conference on Tools in Europe, 2008, pp. 392–411.

[4] A. Achilleos, N. Paspallis, G. A. Papadopoulos, Automating the Development of Device-Aware Web Services: A Model-Driven Approach, in: Proceedings of the 35TH IEEE International Computer Software and Applications Conference (COMPSAC), 2011, pp. 535–540.

[5] R. Akkiraju, T. Mitra, P. Dhoolia, W. Zhao, S. Fu, M. Bhandar, N. Ghosh, D. Saha, Toward the Development of Contextually Aware Business Applications via Model-Driven Transformations, in: Proceedings of the IEEE International Conference on Web Services (ICWS), 2008, pp. 569–576.

[6] S. Alahmari, E. Zaluska, D. D. Roure, A Service Identification Framework for Legacy System Migration into SOA, IEEE International Conference on Services Computing (2010) 614–617.

[7] M. Alam, R. Breu, M. Breu, Model driven security for web services (MDS4WS), in: Proceedings of the 8th International Multitopic Conference (INMIC), 2004, pp. 498–505.

[8] J. P. A. Almeida, M.-E. Iacob, H. Jonkers, D. Quartel, Model-driven development of context-aware services, in: Proceedings of the 6th International Conference on Distributed Applications and Interoperable Systems, 2006, pp. 213–227.

[9] M. Alodib, B. Bordbar, B. Majeed, A model driven approach to the design and implementing of fault tolerant Service oriented Architectures, in: Proceedings of the 3rd International Conference on Digital Information Management, 2008, pp. 473–478.

[10] M. Angel Sanchez, A. Fermoso, L. Joyanes, An On Demand Business context to improve software development process based on business knowledge, in: Proceedings of the 1st World Summit on the Knowledge Society (WSKS), 2008, pp. 433–440.

[11] R. Anzbock, S. Dustdar, Semi-automatic generation of web services and BPEL processes - A model-driven approach, in: Proceedings of the 3rd International Conference on Business Process Management, 2005, pp. 64–79.

[12] M. Autili, D. Ruscio, A. Salle, P. Inverardi, M. Tivoli, A Model-Based Synthesis Process for Choreography Realizability Enforcement, in: Proceedings of the 16th International Conference on Fundamental Approaches to Software Engineering (FASE), 2013, pp. 37–52.

[13] L. Bahler, F. Caruso, J. Micallef, A practical method and tool for systems engineering of service-oriented applications, in: Proceedings of the 8th International Conference on Web Information Systems Engineering (WISE), 2007, pp. 472–483.

[14] L. Bahler, F. Caruso, J. Micallef, Experience with a model-driven approach for enterprise-wide interface specification and XML schema generation, in: Proceedings of the 7th IEEE International Enterprise Distributed Object Computing Conference, 2003, pp. 288–295.

[15] R. Barrett, C. Pahl, Model driven design of distribution patterns for Web service compositions, in: Proceedings of the IEEE International Conference on Web Services (ICWS), 2006, pp. 887–888.

[16] B. Bauer, J. Muller, MDA applied: From sequence diagrams to Web Service choreography, in: Proceedings of the 4th International Workshop on Web-Oriented Software Technologies/4th International Conference on Web Engineering, 2004, pp. 136–148.

[17] M. Belaunde, P. Falcarin, Realizing an MDA and SOA marriage for the development of mobile services, in: Proceedings of the 4th European Conference on Model Driven Architecture - Foundations and Applications, 2008, pp. 393–405.

[18] R. Bendraou, A. Sadovykh, M.-P. Gervais, X. Blanc, Software process modeling and execution: The UML4SPM to WS-BPEL approach, in: Proceedings of the 33rd EUROMICRO Conference on Software Engineering and Advanced Applications, 2007, pp. 314–321.

[19] A. Bercovici, F. Fournier, A. J. Wecker, From business architecture to SOA realization using MDD, in: Proceedings of the 4th European Conference on Model Driven Architecture - Foundations and Applications, 2008, pp. 381–392.

[20] J. Bezivin, S. Hammoudi, D. Lopes, F. Jouault, Applying MDA approach for Web service platform, in: Proceedings of the 8th IEEE International Enterprise Distributed Object Computing Conference, 2004, pp. 58–70.

[21] L. Bocchi, Y. Hong, A. Lopes, J. L. Fiadeiro, From BPEL to SRML: A formal transformational approach, in: Proceedings of the 4th International Workshop on Web Services and Formal Methods, 2008, pp. 92–107.

[22] P. Bocciarelli, A. D'Ambrogio, A model-driven method for describing and predicting the reliability of composite services, Software and Systems Modeling 10 (2) (2011) 265–280.

[23] M. Borek, N. Moebius, K. Stenzel, W. Reif, Model-Driven Development of Secure Service Applications, in: Proceedings of the 35th IEEE Software Engineering Workshop (SEW), 2012, pp. 62–71.

[24] M. Brambilla, M. Dosmi, P. Fraternali, Model-Driven Engineering of Service Orchestrations, in: Proceedings of the 4th IEEE Congress on Services, 2009, pp. 562–569.

[25] A. Brown, M. Delbaere, P. Eeles, S. Johnston, R. Weaver, Realizing service-oriented solutions with the IBM Rational Software Development Platform, IBM Systems Journal 44 (4) (2005) 727–752.

[26] V. de Castro, J. Manuel Vara, E. Marcos, Service-Oriented Development of Web Information Systems, Journal of Universal Computer Science 18 (17) (2012) 2474–2492.

[27] V. de Castro, E. Marcos, M. L. Sanz, Service composition modeling: A case study, in: Proceedings of the 7th Mexican International Conference on Computer Science, 2006, pp. 101–108.

[28] B. Chakir, M. Fredj, M. Nassar, Promoting reuse in web services by managing variability, in: Proceedings of the International Conference on Multimedia Computing and Systems (ICMCS), 2012, pp. 712–717.

[29] K. Chan, I. Poernomo, QoS-aware model driven architecture through the UML and CIM, Information Systems Frontiers 9 (2-3) (2007) 209–224.

[30] A. D'Ambrogio, A model-driven WSDL extension for describing the QoS of web services, in: Proceedings of the IEEE International Conference on Web Services (ICWS 2006), 2006, pp. 789–796.

[31] K. Dahman, F. Charoy, C. Godart, Generation of Component Based Architecture from Business Processes: Model Driven Engineering for SOA, in: Proceedings of the ECOWS, 2010, pp. 155–162.

[32] V. De Castro, E. Marcos, J. Manuel Vara, Applying CIM-to-PIM model transformations for the service-oriented development of information systems, Information and Software Technology 53 (1) (2011) 87–105.

[33] V. De Castro, E. Marcos, R. Wieringa, Towards a Service-oriented MDA-based Approach to the Alignment of Business Processes with it Systems: From the Business Model to a Web Service Composition Model, International Journal of Cooperative Information Systems 18 (2) (2009) 225–260.

[34] N. A. Delessy, E. B. Fernandez, A pattern-driven security process for SOA applications, in: Proceedings of the 3rd International Conference on Availability, Reliability and Security (ARES), 2008, pp. 416–421.

[35] J. Fabra, V. De Castro, P. Alvarez, E. Marcos, Automatic execution of business process models: Exploiting the benefits of Model-driven Engineering approaches, Journal of Systems and Software 85 (3) (2012) 607–625.

[36] M. Felderer, J. Chimiak-Opoka, R. Breu, Model-Driven System Testing of Service Oriented Systems - A Standard-aligned Approach based on Independent System and Test Models, in: Proceedings of the 12th International Conference on Enterprise Information Systems, 2010, pp. 428–435.

[37] R. Garcia-Carmona, J. C. Duenas, F. Cuadrado, J. Luis Ruiz, Lessons Learned on the Development of an Enterprise Service Management System Using Model-Driven Engineering, in: Cordeiro, J and Ranchordas, AK and Shishkov, B (Ed.), Proceedings of the 4th International Conference on Software and Data Technologies, 2011, pp. 59–68.

[38] S. Gilmore, L. Goenczy, N. Koch, P. Mayer, M. Tribastone, D. Varro, Non-functional properties in the model-driven development of service-oriented systems, Software and Systems Modeling 10 (3) (2011) 287–311.

[39] R. Gronmo, M. Jaeger, Model-driven methodology for building QoS-optimised Web service compositions, in: Proceedings of the 5th IFIP International Conference on Distributed Applications and Interoperable Systems, 2005, pp. 68–82.

[40] T. Haase, Model-driven service development for A-posteriori application integration, in: Proceedings of the IEEE International Conference on e-Business Engineering, 2007, pp. 649–656.

[41] M. Hafner, M. Alam, R. Breu, Towards a MOF/QVT-based domain architecture for model driven security, in: Proceedings of the 9th International Conference on Model Driven Engineering Languages and Systems, 2006, pp. 275–290.

[42] C. Hahn, K. Fischer, Service composition in holonic multiagent systems: Model-driven choreography and orchestration, in: Proceedings of the 3rd International Conference on Industrial Application of Holonic and Multi-Agent Systems, 2007, pp. 47–58.

[43] C. Hahn, C. Madrigal-Mora, K. Fischer, B. Elvesaeter, A.-J. Berre, I. Zinnikus, Meta-models, models, and model transformations: Towards interoperable agents, in: Proceedings of the 4th German Conference on Multiagent Systems Technologies, 2006, pp. 123–134.

[44] P. Hoyer, M. Gebhart, I. Pansa, S. Link, A. Dikanski, S. Abeck, A Model-Driven Development Approach for Service-Oriented Integration Scenarios, in: Proceedings of the Computation World - Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns Conference, 2009, pp. 353–358.

[45] S. Huang, Y. Fan, Model driven and service oriented enterprise integration - The method, framework and platform, in: Proceedings of the 6th International Conference on Advanced Language Processing and Web Information Technology (ALPIT), 2007, pp. 504–509.

[46] W. Hummer, P. Gaubatz, M. Strembeck, U. Zdun, S. Dustdar, Enforcement of entailment constraints in distributed service-based business processes, Information and Software Technology 55 (11) (2013) 1884–1903.

[47] M.-E. Iacob, H. Jonkers, A model-driven perspective on the rule-based specification and analysis of service-based

applications, Enterprise Information Systems 3 (3) (2009) 279–298.
[48] Z. Jaroucheh, X. Liu, S. Smith, Apto: A MDD-Based Generic Framework for Context-Aware Deeply Adaptive Service-Based Processes, in: Proceedings of the IEEE International Conference on Web Services (ICWS), 2010, pp. 219–226.
[49] M. Jensen, S. Feja, A Security Modeling Approach for Web-Service-based Business Processes, in: Proceedings of the 16th IEEE International Conference and Workshop on the Engineering of Computer-Based Systems, 2009, pp. 340–347.
[50] M. Jiang, Z. Yang, A model-driven approach for dependable software systems, in: Proceedings of the 7th International Conference on Quality Software, 2007, pp. 100–106.
[51] S. K. Johnson, A. W. Brown, A model-driven development approach to creating service-oriented solutions, in: Proceedings of the 4th International Conference on Service-Oriented Computing (ICSOC), 2006, pp. 624–636.
[52] B. Karakostas, Y. Zorgios, C. Alevizos, Automatic derivation of BPEL4WS from IDEF0 process models, Software and Systems Modeling 5 (2006) 208–218.
[53] A. Kenzi, B. E. Asri, M. Nassar, A. Kriouile, Engineering adaptable service oriented systems: A model driven approach, in: Proceedings of the Service-Oriented Computing and Applications (SOCA), 2009, pp. 1–8.
[54] A. Kenzi, B. El Asri, M. Nassar, A. Kriouile, A model driven framework for multiview service oriented system development, in: Proceedings of the 7th ACS/IEEE International Conference on Computer Systems and Applications (AICCSA), 2009, pp. 404–411.
[55] R. Khadka, B. Sapkota, L. F. Pires, M. van Sinderen, S. Jansen, Model-driven approach to enterprise interoperability at the technical service level, Computers in Industry 64 (8) (2013) 951–965.
[56] R. Khadka, B. Sapkota, L. F. Pires, M. van Sinderen, S. Jansen, Model-Driven Development of Service Compositions for Enterprise Interoperability, in: Proceedings of the 3rd International IFIP Working Conference on Enterprise Interoperability, 2011, pp. 177–190.
[57] H.-K. Kim, Mobile Applications Development with Combine on MDA and SOA, in: Proceedings of the Conference on Education and Learning/Int Conf on Database Theory and Application/Int Conf on u-Service and e-Service, Science and Technology, 2012, pp. 58–71.
[58] J. Koehler, R. Hauser, S. Sendall, M. Wahler, Declarative techniques for model-driven business process integration, IBM Systems Journal 44 (1) (2005) 47–65.
[59] M. Leotta, G. Reggio, F. Ricca, E. Astesiano, Towards a Lightweight Model Driven Method for Developing SOA Systems Using Existing Assets, in: Proceedings of the 14th IEEE International Symposium on Web Systems Evolution (WSE), 2012, pp. 51–60.
[60] W.-j. Li, C.-w. Huang, Q.-c. Chen, H. Bian, A model-driven aspect framework for grid service development, in: Proceedings of the IEEE Asia-Pacific Services Computing Conference (APSCC), 2006, pp. 82–89.
[61] C.-M. Lo, S.-J. Huang, Applying Model-Driven Approach to Building Rapid Distributed Data Services, IEICE Transactions on Information and Systems E95D (12) (2012) 2796–2809.
[62] D. Lopes, S. Hammoudi, Web Services in the context of MDA, in: Proceedings of the International Conference on Web Services (ICWS), 2003, pp. 424–427.
[63] D. Lopes, S. Hammoudi, J. Bezivin, F. Jouault, Generating transformation definition from mapping specification: Application to web service platform, in: Proceedings of the 17th International Conference on Advanced Information Systems Engineering, 2005, pp. 309–325.
[64] M. Lopez-Sanz, C. J. Acuna, C. E. Cuesta, E. Marcos, Defining service-oriented software architecture models for a MDA-based development process at the PIM level, in: Proceedings of the 7th Working IEEE/IFIP Conference on Software Architecture (WISCA), 2008, pp. 309–312.
[65] M. Lopez-Sanz, J. Manuel Vara, E. Marcos, C. E. Cuesta, A model-driven approach to weave architectural styles into service-oriented architectures, International Journal of Cooperative Information Systems 20 (2) (2011) 201–220.
[66] M. Lopez-Sanz, E. Marcos, ArchiMeDeS: A model-driven framework for the specification of service-oriented architectures, Information Systems 37 (3) (2012) 257–268.
[67] M. Lopez-Sanz, Z. Qayyum, C. E. Cuesta, E. Marcos, F. Oquendo, Representing Service-Oriented Architectural Models Using pi-ADL, in: Proceedings of the 2nd European Conference on Software Architecture, 2008, pp. 273–280.
[68] D. Manset, H. Verjus, R. McClatchey, F. Oquendo, A formal architecture-centric model-driven approach for the automatic generation of Grid applications, in: Proceedings of the 8th International Conference on Enterprise Information Systems (ICEIS), 2006, pp. 322–330.
[69] Mardiana, K. Araki, Y. Omori, MDA and SOA Approach to Development of Web Application Interface, in: Proceedings of the IEEE Region 10 Conference on TENCON, 2011, pp. 226–231.
[70] F. P. Marzullo, J. M. de Souza, J. R. Blaschek, A Domain-Driven Development Approach for Enterprise Applications, using MDA, SOA and Web Services, in: Proceedings of the 10th IEEE Conference on E-Commerce Technology/5th IEEE Conference on Enterprise Computing, E-Commerce and E-Services (CEC/EEE), 2008, pp.

432–437.
[71] P. Mayer, A. Schroeder, N. Koch, A Model-Driven Approach to Service Orchestration, in: Proceedings of the IEEE International Conference on Services Computing, 2008, pp. 533–536.
[72] C. Mayr, U. Zdun, S. Dustdar, View-based model-driven architecture for enhancing maintainability of data access services, Data and Knowledge Engineering 70 (9) (2011) 794–819.
[73] T. D. Meijler, G. Kruithof, N. van Beest, Top down versus bottom up in service-oriented integration: An MDA-based solution for minimizing technology coupling, in: Proceedings of the 4th International Conference on Service-Oriented Computing (ICSOC), 2006, pp. 484–489.
[74] M. Menzel, C. Meinel, A Security Meta-Model for Service-oriented Architectures, in: Proceedings of the IEEE International Conference on Services Computing, 2009, pp. 251–259.
[75] M. Menzel, C. Meinel, SecureSOA Modelling Security Requirements for Service-Oriented Architectures, in: Proceedings of the IEEE International Conference on Services Computing (SCC), 2010, pp. 146–153.
[76] M. Menzel, I. Thomas, C. Meinel, Security Requirements Specification in Service-oriented Business Process Management, in: Proceedings of the 4th International Conference on Availability, Reliability and Security (ARES), 2009, pp. 41–48.
[77] M. Menzel, R. Warschofsky, C. Meinel, A Pattern-Driven Generation of Security Policies for Service-Oriented Architectures, in: Proceedings of the 19th IEEE International Conference on Web Services, 2010, pp. 243–250.
[78] D. Moreno-Garcia, J. Estublier, Model-Driven Design, Development, Execution and Management of Service-Based Applications, in: Proceedings of the IEEE 9th International Conference on Services Computing (SCC), 2012, pp. 470–477.
[79] N. Nagaratnam, A. Nadalin, M. Hondo, M. McIntosh, P. Austel, Business-driven application security: From modeling to managing secure applications, IBM Systems Journal 44 (4) (2005) 847–867.
[80] Y. Nakamura, M. Tatsubori, T. Imamura, K. Ono, Model-driven security based on a Web services security architecture, in: Proceedings of the IEEE International Conference on Services Computing, 2005, pp. 7–15.
[81] T. Nguyen, A. Colman, J. Han, Modeling and Managing Variability in Process-Based Service Compositions, in: Proceedings of the 9th International Conference on Service-Oriented Computing (ICSOC), 2011, pp. 404–420.
[82] B. Orriens, J. Yang, M. Papazoglou, Model driven service composition, in: Proceedings of the 1st International Conference on Service-Oriented Computing (ICSOC), 2003, pp. 75–90.
[83] G. Ortiz, J. Hernandez, F. Sanchez, Model driven extra-functional properties for Web services, in: Proceedings of the IEEE Services Computing Workshops, 2006, pp. 113–120.
[84] C. Pahl, Semantic model-driven architecting of service-based software systems, Information and Software Technology 49 (8) (2007) 838–850.
[85] C. Pahl, Ontology-based composition and transformation for model-driven service architecture, in: Proceedings of the 2nd European Conference on Model Driven Architecture Foundations and Applications, 2006, pp. 198–212.
[86] C. Pahl, Layered ontological modelling for web service-oriented model-driven architecture, in: Proceedings of the 1st European Conference on Model Driven Architecture - Foundations and Applications, 2005, pp. 88–102.
[87] R. Perez-Castillo, I. Garcia-Rodriguez de Guzman, I. Caballero, M. Piattini, Software modernization by recovering Web services from legacy databases, Journal of Software-Evolution and Process 25 (5) (2013) 507–533.
[88] R. R. Quintero Meza, L. Z. Zepeda Sanchez, L. Vega Zazueta, Model Driven Software Development of Applications Based on Web Services, in: Proceedings of the International Conference on Computational Science and Its Applications, 2010, pp. 241–256.
[89] A. T. Rahmani, V. Rafe, S. Sedighian, A. Abbaspour, An MDA-based modeling and design of service oriented architecture, in: Proceedings of the 6th International Conference on Computational Science (ICCS), 2006, pp. 578–585.
[90] J. Ricken, Top-down modeling methodology for model-driven SOA construction, in: Proceedings of the OTM Confederated International Conference and Workshop, 2007, pp. 323–332.
[91] M. Ruiz, V. Pelechano, Model driven design of web service operations using web engineering practices, in: Proceedings of the Workshop on Emerging Web Services Technology held in conjunction with the 4th European Conference on Web Services, 2007, pp. 83–100.
[92] M. Ruiz, V. Pelechano, O. Pastor, Designing Web Services for supporting user tasks: A model driven approach, in: Proceedings of the 25th International Conference on Conceptual Modeling, 2006, pp. 193–202.
[93] M. Ruiz, P. Valderas, V. Pelechano, Applying a web engineering method to design web services, in: Proceedings of the 3rd International Conference on Service-Oriented Computing (ICSOC), 2005, pp. 576–581.
[94] F. Satoh, Y. Nakamura, K. Ono, Adding authentication to model driven security, in: Proceedings of the IEEE International Conference on Web Services (ICWS), 2006, pp. 585–592.
[95] F. Satoh, M. Tatsubori, Y. Nakamura, N. K. Mukhi, K. Ono, Methodology and Tools for End-to-End SOA Security Configurations, in: Proceedings of the IEEE Congress on Services, 2008, pp. 307–314.
[96] R. Sharma, M. Sood, Cloud SaaS: Models and Transformation, in: Proceedings of the 1st International Conference on Digital Image Processing and Pattern Recognition (DPPR), 2011, pp. 305–314.

[97] Q. Z. Sheng, S. Pohlenz, J. Yu, H. S. Wong, A. H. H. Ngu, Z. Maamar, ContextServ: A Platform for Rapid and Flexible Development of Context-Aware Web Services, in: Proceedings of the 31st International Conference on Software Engineering (ICSE), 2009, pp. 619–622.
[98] R. Sindhgatta, Interleaving Execution into Model Driven Service Design, in: IEEE 20th International Conference on Proceedings of the Web Services (ICWS), 2013, pp. 364–371.
[99] D. Skogan, R. Gronmo, I. Solheim, Web service composition in UML, in: Proceedings of the 8th IEEE International Enterprise Distributed Object Computing Conference, 2004, pp. 47–57.
[100] S. Stein, J. Lauer, K. Ivanov, ARIS Method Extension for Business-Driven SOA, Wirtschaftsinformatik 50 (6) (2008) 436+.
[101] W. Sun, S. Li, D. Zhang, Y. Yan, A Model-driven Reverse Engineering Approach for Semantic Web Services Composition, in: Proceedings of the World Congress on Software Engineering, 2009, pp. 101–105.
[102] F. Taghizadeh, S. R. Taghizadeh, A Graph Transformation-Based Approach for applying MDA to SOA, in: Proceedings of the 4th International Conference on Frontier of Computer Science and Technology, 2009, pp. 446–451.
[103] W. Tan, L. Li, W. Xu, F. Yang, C. Jiang, L. Yang, J. Choi, A role-oriented service system architecture for enterprise process collaboration, Computers and Operations Research 39 (8, SI) (2012) 1893–1900.
[104] R. Thiagarajan, M. Stumptner, W. Mayer, Semantic web service composition by consistency-based model refinement, in: Proceedings of the 2nd IEEE Asia-Pacific Service Computing Conference (APSCC), 2007, pp. 336–343.
[105] H. Tran, U. Zdun, S. Dustdar, VbTrace: using view-based and model-driven development to support traceability in process-driven SOAs, Software and Systems Modeling 10 (1) (2011) 5–29.
[106] H. Tran, U. Zdun, T. Holmes, E. Oberortner, E. Mulo, S. Dustdar, Compliance in service-oriented architectures: A model-driven and view-based approach, Information and Software Technology 54 (6) (2012) 531–552.
[107] W.-T. Tsai, X. Wei, R. Paul, J.-Y. Chung, Q. Huang, Y. Chen, Service-oriented system engineering (SOSE) and its applications to embedded system development, Service Oriented Computing and Applications 1 (2007) 3–17.
[108] S. Vale, S. Hammoudi, Model Driven Development of Context-aware Service Oriented Architecture, in: Proceedings of the 11th IEEE International Conference on Computational Science and Engineering, 2008, pp. 412–418.
[109] J. Vara, V. de Castro, E. Marcos, WSDL automatic generation from UML models in a MDA framework, in: Proceedings of the International Conference on Next Generation Web Services Practices, 2005, pp. 319–324.
[110] A. Vemulapalli, N. Subramanian, Transforming Functional Requirements from UML into BPEL to Efficiently Develop SOA-Based Systems, in: Proceedings of the On the Move Confederated International Conference and Workshops, 2009, pp. 337–349.
[111] P. D. Villarreal, E. Salomone, O. Chiotti, Transforming collaborative business process models into web services choreography specifications, in: Proceedings of the 2nd International Workshop on Data Engineering Issues in E-Commerce and Services, 2006, pp. 50–65.
[112] H. Wada, J. Suzuki, K. Oba, Leveraging Early Aspects in End-to-End Model Driven Development for Non-Functional Properties in Service Oriented Architecture, Journal of Database Management 22 (2) (2011) 93–123.
[113] H. Wada, J. Suzuki, K. Oba, A model-driven development framework for non-functional aspects in service oriented architecture, International Journal of Web Services Research 5 (4) (2008) 1–31.
[114] H. Wada, J. Suzuki, K. Oba, A feature modeling support for non-functional constraints in service oriented architecture, in: Proceedings of the IEEE International Conference on Services Computing, 2007, pp. 187–195.
[115] S. Walderhaug, E. Stav, M. Mikalsen, Experiences from Model-Driven Development of Homecare Services: UML Profiles and Domain Models, in: Proceedings of the Workshop and Symposia on Models in Software Engineering, 2009, pp. 199–212.
[116] M. Wirsing, M. Hoelzl, N. Koch, P. Mayer, A. Schroeder, Service engineering: The sensoria model driven approach, in: Proceedings of the 6th ACIS International Conference on Software Engineering Research, Management and Applications (SERA), 2008, pp. 14–16.
[117] C. Xiao-Xia, M. Huai-Kou, X. Qing-Guo, Modeling and refining the service-oriented requirement, in: Proceedings of the 2nd IFIP/IEEE International Symposium on Theoretical Aspects of Software Engineering (TASE), 2008, pp. 159–165.
[118] Y. Yang, Q. Tan, Y. Xiao, Transformation-driven development of composite Web services, in: Proceedings of the ISPA Workshops AEPP, ASTED, BIOS, GCIC, IASD, MASN, SGCA and WISA, 2005, pp. 637–646.
[119] B. Yu, C. Zhang, Y. Zhao, Transform from models to service description based on MDA, in: Proceedings of the IEEE Asia-Pacific Services Computing Conference (APSCC), 2006, pp. 605–608.
[120] J. Yu, Q. Sheng, J. Swee, Model-Driven Development of Adaptive Service-Based Systems with Aspects and Rules, in: Proceedings of the 11th International Conference on Web Information Systems Engineering (WISE), 2010, pp. 548–563.
[121] X. Yu, Y. Zhang, T. Zhang, L. Wang, J. Hu, J. Zhao, X. Li, A model-driven development framework for enterprise Web services, in: Proceedings of the 10th IEEE International Enterprise Distributed Object Computing Conference (EDOC), 2006, pp. 75–84.
[122] L. J. Zhang, N. Zhou, Y. M. Chee, A. Jalaldeen, K. Ponnalagu, R. R. Sindhgatta, A. Arsanjani, F. Bernardini,

SOMA-ME: A platform for the model-driven design of SOA solutions, IBM Systems Journal 47 (3) (2008) 397–413.
[123] M. Zhang, Z. Duan, From Business Process Models to Web Services Orchestration: The Case of UML 2.0 Activity Diagram to BPEL, in: Proceedings of the 6th International Conference on Service-Oriented Computing (ICSOC), 2008, pp. 505–510.
[124] X. Zhang, K. He, J. Wang, J. Liu, C. Wang, H. Lu, On-Demand Service-Oriented MDA Approach for SaaS and Enterprise Mashup Application Development, in: Proceedings of the International Conference on Cloud Computing and Service Computing (CSC), 2012, pp. 96–103.
[125] C. Zhao, Z. Duan, M. Zhang, A Model-Driven Approach for Dynamic Web Service Composition, in: Proceedings of the World Congress on Software Engineering, 2009, pp. 273–277.
[126] H. Zhu, L. Shan, Agent-oriented modelling and specification of web services, in: Proceedings of the 10th IEEE International Workshop on Object-Oriented Real-Time Dependable Systems (WORDS), 2005, pp. 152–159.
[127] O. Zimmermann, J. Grundler, S. Tai, F. Leymann, Architectural decisions and patterns for transactional workflows in SOA, in: Proceedings of the 5th International Conference on Service-Oriented Computing (ICSOC), 2007, pp. 81–93.
[128] I. Zinnikus, G. Benguria, B. Elvesaeter, K. Fischer, J. Vayssiere, A model driven approach to agent-based service-oriented architectures, in: Proceedings of the 4th German Conference on Multiagent Systems Technologies, 2006, pp. 110–122.
[129] I. Zinnikus, C. Hahn, M. Klein, K. Fischer, An agent-based, model-driven approach for enabling interoperability in the area of multi-brand vehicle configuration, in: Proceedings of the 5th International Conference on Service-Oriented Computing (ICSOC), 2007, pp. 330–341.

**Appendix E: Significant correlations (Chi$^2$)**

| | V1 | V2 | p |
|---|---|---|---|
| 1 | Venue topic | Venue type | 0.015 |
| 2 | Venue topic | Year | 0.031 |
| 36 | Venue type | Year | 0.007 |
| 40 | Venue type | Method | 0.005 |
| 41 | Venue type | Tool | 0.002 |
| 65 | Venue type | NFR: Reliability | 0.039 |
| 72 | Year | Industry | 0.001 |
| 74 | Year | Tool | 0.030 |
| 77 | Year | Input level | 0.024 |
| 78 | Year | Input notation | 0.001 |
| 85 | Year | Transformation | 0.003 |
| 87 | Year | SOA metamodel type | 0.031 |
| 103 | Country | Continent | 0.000 |
| 104 | Country | Industry | 0.001 |
| 110 | Country | Input notation | 0.006 |
| 117 | Country | Transformation | 0.005 |
| 135 | Continent | Industry | 0.004 |
| 137 | Continent | Tool | 0.022 |
| 196 | Method | Tool | 0.016 |
| 202 | Method | Output notation | 0.003 |
| 203 | Method | Tool input | 0.008 |
| 210 | Method | Concept: Service | 0.019 |
| 217 | Method | SOA type | 0.011 |
| 219 | Method | NFR: Security | 0.006 |
| 237 | Tool | SOA metamodel type | 0.039 |
| 244 | Tool | Concept: NFR | 0.004 |
| 247 | Tool | NFR: Security | 0.021 |
| 280 | Approach type | Input level | 0.000 |
| 282 | Approach type | Output level | 0.039 |
| 288 | Approach type | Transformation | 0.000 |
| 306 | Input level | Input notation | 0.000 |
| 307 | Input level | Output level | 0.000 |
| 313 | Input level | Transformation | 0.000 |
| 323 | Input level | SOA type | 0.000 |
| 332 | Input notation | Output notation | 0.015 |
| 337 | Input notation | Transformation | 0.000 |
| 339 | Input notation | SOA metamodel type | 0.044 |
| 343 | Input notation | Concept: Operation | 0.008 |
| 347 | Input notation | SOA type | 0.048 |
| 355 | Output level | Output notation | 0.000 |
| 360 | Output level | Transformation | 0.000 |
| 370 | Output level | SOA type | 0.007 |
| 379 | Output notation | Tool output (Java) | 0.000 |

| | | | |
|---|---|---|---|
| 381 | Output notation | Tool output (BPEL) | 0.000 |
| 382 | Output notation | Transformation | 0.004 |
| 385 | Output notation | Concept: Service | 0.003 |
| 388 | Output notation | Concept: Operation | 0.008 |
| 392 | Output notation | SOA type | 0.001 |
| 403 | Tool input | Transformation | 0.002 |
| 480 | Transformation | Concept: Service | 0.010 |
| 482 | Transformation | Concept: Message | 0.021 |
| 513 | SOA metamodel type | Concept: Message | 0.003 |
| 514 | SOA metamodel type | Concept: Operation | 0.036 |
| 526 | Concept: Service | Concept: Port | 0.027 |
| 596 | Concept: NFR | NFR support | 0.002 |