JOT
JOURNAL OF
OBJECT TECHNOLOGY

Journal of Object Technology | **RESEARCH ARTICLE**

# A Quantitative SWOT-TOWS Analysis for the Adoption of Model-Based Software Engineering

**María-José Escalona[*], Nora Parcus de Koch[*], and Gustavo Rossi[‡]**
[*]Universidad de Sevilla, Spain
[‡]Universidad Nacional de La Plata and CONICET, Argentina

**ABSTRACT** Enterprises' trend to low-code development revives model-based software engineering (MBSE) since several low-code platforms are based on the principles of model-based design, automatic code generation, and visual programming. Changes in an enterprise's software development process, however, always require strategic planning. To find an appropriate strategy, we present an analytical tool for identifying and evaluating strengths, weaknesses, opportunities and threats factors for the adoption of MBSE. This tool provides a SWOT-TOWS analysis supplemented by a quantitative evaluation of strategies based on a multiple-criteria decision technique drawing on the knowledge of industry experts. Our analytical tool is general so it can be used in the industrial context for making other strategic decisions.

**KEYWORDS** Model-Based Software Engineering, Model-Driven Engineering, SWOT Analysis, TOWS strategies, Analytic Hierarchy Process.

## 1. Introduction

The increasing complexity of software, the social demand for higher software productivity, and the emergence of new domains such as embedded systems, cyber-physical systems and the Internet of Things have motivated engineers to (re)consider existing ways of tackling complexity and improving quality and productivity. As a state-of-the-art means of abstracting different aspects of systems, models are a good choice in this regard.

The term model-based systems engineering was introduced many years ago (Wymore 1993), mainly in reference to mathematical models. In software, numerous informal modelling approaches have been used for decades. These include Entity Relationship Diagrams, Flowcharts, formal languages such as Z, and others. The emergence of UML (Unified Modeling Language) opened up a new avenue of research focusing on the use of models as essential artefacts in software development (Brambilla et al. 2017). In around 2000, model-driven software engineering (MDSE or MDE for short) emerged as a new branch of software engineering and within a few years it had become very popular.

However, the use of models as first-class citizens throughout the software process (either to guide development or as an equivalent to programs), an approach known as model-based software engineering (MBSE), is still not a mainstream practice in the software industry. However, counter examples can be found in domains like railway systems, embedded systems and IoT (Internet of Things) (Morin et al. 2017; García-Borgoñon et al. 2013).

For years, researchers and practitioners have discussed the advantages and problems of using models during the development process, and there are numerous papers identifying the challenges that still have to be overcome on this matter (Vallecillo 2015; Bucchiarone et al. 2020). Besides, the scarcity of empirical evaluations regarding the use of models in industry is highlighted in (Liebel et al. 2018). Only a handful of papers indicate when and why organisations should start transitioning to MBSE, and there are no objective indicators of when the advantages outweigh the problems or possible threats.

What is changing? What is the future of MBSE? The current trend is for companies to turn to low-code development aimed mainly at solving the challenges of mass digitalisation and the shortage of information technology (IT) specialists (NTTDATA 2021). Several low-code development platforms are based, at

**JOT reference format:**
María-José Escalona, Nora Parcus de Koch, and Gustavo Rossi. *A Quantitative SWOT-TOWS Analysis for the Adoption:of Model-Based Software Engineering*. Journal of Object Technology. Vol. 21, No. 4, 2022.
 http://dx.doi.org/10.5381/jot.2022.21.4.a9

least partially, on conceptual and technological principles of model-based design such as meta-modelling, model-to-model and model-to-code transformations, automatic code generation and visual programming. They thus offer an implicit way of using models. They are also, in general, more intuitive and user-friendly environments than traditional modelling tools (e.g., EMF-based applications). Low-code and no-code platforms therefore represent a huge opportunity to bring modelling to new domains and developers communities (Cabot 2020). Given this opportunity, companies have to decide when and how to adopt new paradigms, technology platforms or processes.

*The contributions of this paper*. The final purpose of our work is to support decision makers with an analytical tool that provides stakeholders with information regarding the eventual adoption of a new paradigm, technology or process for software development. This paper validates our approach to MBSE adoption. Its intermediate aims are several. We first identify and analyse the factors that influence MBSE adoption. We then define different strategies to achieve that goal and finally we provide information with which to select the most appropriate strategy for doing so, based on the opinion of experts and using a multiple-criteria decision-making method.

To achieve our objective, we use the SWOT-TOWS approach (Gürel & Tat 2017; Weihrich 1982) in combination with the Analytic Hierarchy Process (AHP) (Saaty 1990). SWOT is an acronym for Strengths, Weaknesses, Opportunities and Threats. It is a tool commonly used by companies for analysing an environment's internal and external factors and thus providing systematic support for a decision situation (Gürel & Tat 2017). SWOT factors are used to define a set of alternative strategies (the so-called TOWS matrix)(Weihrich 1982). The TOWS matrix contains four groups of alternative strategies addressing combinations of external factors (Threats or Opportunities) and internal factors (Weaknesses or Strengths).

But factors and strategies alone make for a qualitative tool that is very often insufficient as explained in (Gürel & Tat 2017). We therefore decided to use AHP, a method that provides the quantitative values necessary to justify a strategic decision. AHP (Saaty 1990) is a multiple-criteria decision-making approach in which factors are arranged in a hierarchical structure and compared pairwise by experts. Weights are then calculated for them and used for their prioritisation.

Our work focuses on the strategic decision to adopt MBSE in the development of industrial software products. We do not intend to analyse the nature of MBSE. This has been extensively described in the literature, as we discuss in Section 2.

Note that our proposed SWOT-TOWS-AHP analytical tool can also be used for other kinds of strategic decision-making in the software engineering domain, such as selecting new development tools or crucial project decisions. This type of qualitative-quantitative analysis has to date been applied very little in the field of IT, although it has been used to aid other kinds of strategic decision-making by enterprises or public organisations in a variety of economic sectors, including e-government, manufacturing, environment, healthcare, energy and agriculture (see (Gottfried et al. 2018; Tang et al. 2013)).

The remainder of this paper is organised as follows. In Section 2, we present some basic information regarding models and MBSE and look at existing opinions about the impact of models and the extent to which they are used in industry. We also briefly discuss different ways of using models in the development process. Section 3 describes the methodology we used, and explains the concepts underlying SWOT analysis, the TOWS matrix and the Analytic Hierarchy Process. Section 4 shows the results of applying the SWOT-TOWS-AHP approach to the issue of MBSE adoption. The results of our analysis are discussed in Section 5. Section 6 offers an overview of related work, both on SWOT analysis and on quantitative studies into MBSE. Finally, Section 7 presents some conclusions and outlines our plans for future work.

## 2. Background

Here we present some basic concepts regarding models and model-based software engineering, we discuss the benefits of MBSE adoption and we briefly summarise the problems that have been addressed in the literature. A more thorough analysis of quantitative findings is included later in the related work section (Section 6).

### 2.1. Models, MBSE, MDE

Models are selectively reduced representations of a (software) system at an abstract level, intended to accurately capture those aspects of the system that are of interest for a given set of concerns, and a given audience (Selic 2016). Models can be useful for different reasons: (a) Their role can vary according to the style of the development process; (b) they can be used to help in understanding the features of a planned system and to facilitate communication among stakeholders (to improve understanding); (c) they can help in analysing and predicting the key properties of the system to be; and (d) they can serve as an (accurate) implementation of the system, either (d1) as a blueprint to be followed by developers or (d2) as a program to be executed (Gogolla & Selic 2020).

As discussed by (Brown 2004), the spectrum of modelling is broad (see Figure 1). While on the left-hand side of the figure we have only code, models take a more relevant role as we move towards the right. The model-centric view where the model is the code (also called MDE), is the most ambitious approach; whereas on the far right-hand side models are used "merely" as documentation artefacts; this latter use of models is not considered in this paper. MBSE covers the central part of the spectrum shown in Figure 1, where models represent core artefacts during the development process.

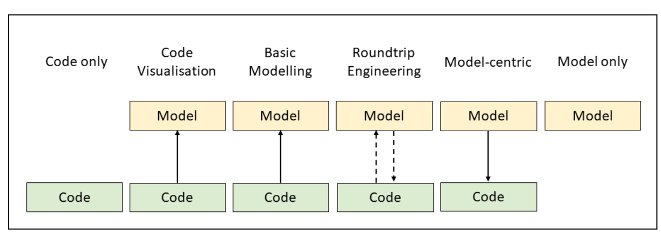


**Figure 1** The modelling spectrum.

(Bucchiarone et al. 2021) conclude that MBSE is a key area where a number of interesting modelling success stories can be found, particularly in "new" domains such as embedded systems (Liebel et al. 2018), Internet of Things (Morin et al. 2017), Digital Twins (Lopez & Akundi 2022) and avionics (Gregory et al. 2020).

### 2.2. Debates over Modelling, MBSE and MDE

The benefits of using models throughout the development process have been cited and discussed for more than 20 years. Rigorous experiments such as those of (Panach et al. 2021) have shown that software quality increases with the MBSE approach. Team productivity has also been found to improve with the systematical use of models (Vallecillo 2015).

As mentioned in the introduction, however, and as has been repeatedly reported over the last 15 years, MBSE is still not mainstream. In a survey presented in (Störrle 2017), more than 70 per cent of the respondents said they only used models as communication artefacts, mostly during the first stages of development (as depicted on the right of Figure 1). The multiple reasons for this are accurately summarised in (Vallecillo 2015). Some of them are listed in Section 3 as weaknesses and threats. Some of the problems that need to be solved are technical and cover foundation, theoretical issues and tools and implementation aspects (Bucchiarone et al. 2020), including the usability of tools (Abrahão et al. 2017). Other problems are more social, organisational or managerial (Hutchinson et al. 2014). The need to address community issues is also mentioned in (Bucchiarone et al. 2020) and (Luna et al. 2018) while the potential impact of educational aspects on the future of MBSE is discussed in (Burgueño et al. 2019; Ciccozzi et al. 2018). (Vallecillo 2015) mentions that the technological adoption of MDE has already passed through the trough of disillusionment in Gartner's Hype Cycle (see (Dedehayir & Steinert 2016)) and has now entered its plateau of productivity. Despite all these challenges, and as mentioned in the Introduction, new problem areas like the development of embedded systems in specific domains and new technologies like low-code platforms show that the concepts underlying MBSE are still perfectly relevant, see (Bucchiarone et al. 2020).

More specifically, (Cabot 2020) emphasises that low-code and no-code platforms represent an opportunity for new growth in MBSE, and (Di Ruscio et al. 2022) argues that while both approaches have points in common, there are still important differences.

In our opinion, however, the availability and popularity of certain technologies and application areas have proven to be good leverage for consolidated theories. This can be seen for example, in the resurgence of artificial intelligence due to machine learning in recent years.

## 3. Methodology

Our objective is to develop a quantitative analytical tool for decision makers in software companies. These kind of tools have been used for strategic decision-making in enterprises or public organisations of quite different economic sectors like e-government, manufacturing, environment, healthcare, energy and agriculture. Examples can be found in the literature reviews conducted by (Gottfried et al. 2018) and (Tang et al. 2013) which include each lists of more than twenty articles published in the periods 2007-2017 and 2000-2011, respectively. To the best of our knowledge only one of those studies addresses IT decision-making: The study in question (Tang et al. 2013) describes a systematic framework for the strategic planning of cloud computing adoption in an enterprise.

The analytical tool proposed in this article is based on qualitative SWOT-TOWS and quantitative AHP analysis – a so-called hybrid method (Kurttila et al. 2000). In the following subsections we briefly describe these techniques, discuss their pros and cons, and look at how their combined use makes it possible to determine the most appropriate strategy.

### 3.1. SWOT Analysis

SWOT (Gürel & Tat 2017) is a widely used technique for scanning the internal and external environment for positive and negative characteristics that should be taken into account by an organisation when formulating its strategic plans. The organisation's internal factors can be classified as strengths (S) or weaknesses (W); its external factors as opportunities (O) or threats (T). The different factors are usually arranged in a matrix.

The starting point is the goal that a company or organisation may evaluate. Such a goal may be a new or changed business, project, method, process, etc. In this context:

– Strengths are factors that offer an advantage for that goal over others;
– Weaknesses are factors that place that goal at a disadvantage relative to others;
– Opportunities are elements in the environment that the goal could exploit to its advantage;
– Threats are elements in the environment that could be problematic for such a goal.

The advantage of the qualitative SWOT technique is that it is a good, simple basis for strategy formulation through the definition of what is known as TOWS matrix. The main disadvantage is that the factors are not measured quantitatively. To overcome the shortcomings, SWOT and TOWS can be combined with a multiple-criteria decision-making approach like the Analytic Hierarchical Process (AHP), which is used to systematically quantify the factors.

### 3.2. TOWS Matrix

The TOWS matrix introduced by (Weihrich 1982) is a tool built on top of a previous conducted SWOT analysis. It illustrates how external threats (T) and opportunities (O) can be matched with internal weaknesses (W) and strengths (S). The result is four groups of alternative strategies:

– **Maxi-Maxi** maximising both strengths and opportunities (SO). These strategies exploit company's internal strengths by taking advantage of external opportunities;

- **Maxi-Mini** maximising strengths and minimising threats (ST). These strategies exploit company's strengths by avoiding or reducing the impact of external threats;
- **Mini-Maxi** minimising weaknesses and maximising opportunities (WO). These strategies improve internal weaknesses by taking advantage of environmental opportunities;
- **Mini-Mini** minimising both weaknesses and threats (WT). These are defensive strategies aimed at reducing internal weaknesses and avoiding environmental threats.

Figure 2 shows the hierarchical structure of the SWOT and TOWS factors, the relationships between the different levels, and an example of relationships between strategies and SWOT factors.

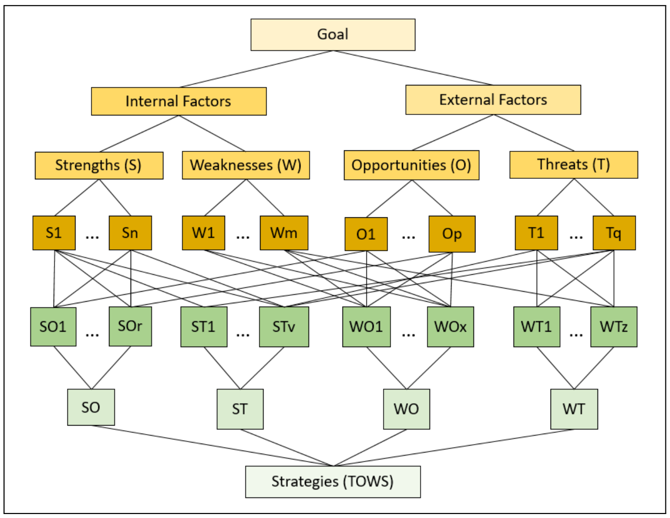


**Figure 2** SWOT-TOWS hierarchy.

The strength of the TOWS approach lies in the pairwise relationship between strengths or weaknesses with opportunities or threats in the SWOT matrix for defining alternative strategies. Although the strategies are called alternative strategies, sometimes more than one of them can be applied simultaneously. One disadvantage of this analytical technique is that the combinations SW and OT are not considered, although these are clearly less relevant because they focus only on either internal or external factors. As with the SWOT matrix the biggest disadvantage of the TOWS matrix is the lack of quantification and the consequent non-provision of any criteria for selecting one or more of the strategies.

### 3.3. Analytic Hierarchy Process

The Analytic Hierarchy Process (AHP) developed by T. Saaty (Saaty 1990) was a means of resolving a complex decision problem by decomposing it into a hierarchy, with the goal at the top, criteria and sub-criteria levels in the middle, and decision alternatives at the bottom. The calculations are not too complicated and can be performed by using spreadsheet-based tools. Thanks to this ease of use, AHP is a widely used method for strategic planning in a variety of sectors, including manufacturing, energy, healthcare, e-government, environmental management and agriculture. It is very seldom used, however, in the field of IT (Gottfried et al. 2018; Tang et al. 2013).

The central characteristics of the AHP are the use of pairwise comparison and matrix algebra to weight criteria. The weight of influential factors is determined more reliably when using pairwise comparisons than by obtaining them directly; as it is easier to compare two criteria than to assign overall weight to all criteria simultaneously.

Factors can be compared by an expert or a group of experts, who for each pair decide, which criteria are more important and how much more important one is than the other. Saaty and Aczel proved in (Saaty & Vargas 2012) that the weighted geometric mean satisfies the unanimity and homogeneity properties required for the aggregation of the experts' individual judgements ($x_i$). The formula for calculating the geometric mean is shown in Equation 1.

$$X = \left(\prod_{i=1}^{m} x_i\right)^{\frac{1}{m}} = \sqrt[m]{x_1 x_2 \cdots x_m} \tag{1}$$

For evaluation Saaty proposed a nine-point scale with the following semantics:

| Intensity | Definition |
|---|---|
| 1 | Equal importance of both criteria |
| 3 | Moderate importance of one over another |
| 5 | Essential or strong importance of one over another |
| 7 | Very strong importance of one over another |
| 9 | Extreme importance of one over another |
| 2,4,6,8 | Intermediate values between adjacent odd numbers |

**Table 1** Nine-point scale (Saaty 1990).

The results of the pairwise comparison are placed in a matrix A of n rows and columns, where n is the total number of criteria. $a_{ij} = 5$ means that the value of criteria i is equal to 5 times the value of criteria j. The matrix is completed automatically with $a_{ii} = 1$ and the reciprocal ratios $a_{ji} = 1/a_{ij}$.

$$A = [a_{ij}] = \begin{vmatrix} 1 & w_1/w_2 & \dots & w_1/w_n \\ w_2/w_1 & 1 & \dots & w_2/w_n \\ \dots & \dots & \dots & \dots \\ w_n/w_1 & w_n/w_2 & \dots & 1 \end{vmatrix} \tag{2}$$

Once the comparison matrix is built it needs to be normalised before the eigenvector technique can be applied. This involves using the following eigenvector formula (see Equation 3) to calculate the vector of priorities $[w_j]$.

$$A\ w = \lambda_{max}\ w \tag{3}$$

where A is the comparison matrix, w is the eigenvector of weights and $\lambda_{max}$ is the largest eigenvalue of the matrix A of size n. Saaty demonstrated that $\lambda_{max} = n$ is a necessary and sufficient condition for consistency and that $\lambda_{max} \geq n$ is always satisfied (Saaty 1990). This way the inconsistency throughout the matrix can be captured by the difference $\lambda_{max} - n$, which measures the deviation of the judgements from the consistent approximation.

Calculation of priorities therefore requires a consistency check using the consistency index (CI) defined by (Saaty 1990) with the following equation:

$$CI = (\lambda_{max} - n)/(n-1) \tag{4}$$

The CI value is compared with the same index obtained as an average over a large number of matrices of the same order with random entries. The values for the random index (RI) are shown in Table 2.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0.58 | 0.90 | 1.12 | 1.24 | 1.32 | 1.41 | 1.45 | 1.49 |

**Table 2** Random indexes for n = 1 to 10 (Saaty 1990).

This relationship between the consistency index (CI) and the random index (RI) is defined as the consistency ratio (CR) (see Equation 5).

$$CR = CI/RI \tag{5}$$

(Saaty 1990) recommended accepting the estimation of the weights if the consistency ratio (CR) from the matrix is significantly small (10% or less). Otherwise consistency has to be improved.

Advantages of AHP include its mathematical foundation (Saaty 1990), its intuitive approach and the fact that the calculations needed to obtain the best alternative are not too complicated. One disadvantage of AHP is that it assume that all criteria are independent and determines the priority of strategy alternatives based on this assumption, which is not always true. For the purposes of our paper, however, this technique is good enough and offers a suitable global view of the situation that concurs with the experience of our experts from the industry.

### 3.4. Step by Step

The steps of the methodology (shown in Figure 3) presents our proposal for conducting a quantitative strategy analysis to apply SWOT-TOWS-AHP process. It is summarised as follows:

- *Step 1* - Clearly define the goal for which the qualitative-quantitative analysis is to be performed.
- *Step 2* - Determine the internal and external factors, i.e., strengths, weaknesses, opportunities and threats by conducting interviews with company stakeholders or external experts, and perhaps also performing a literature review.
- *Step 3* - Adjust the SWOT matrix using the experts judgement to identify the similarity and relevance of factors.
- *Step 4* - Define a set of strategies that address the factors selected in the previous step by constructing the TOWS matrix. Strategies are again defined by external experts or company stakeholders.
- *Step 5* - Subject strategies and strategy groups to pairwise comparison by experts.
- *Step 6* - Calculate the local and global weightings (priorities) of the strategies.
- *Step 7* - Calculate the consistency ratios of the AHP matrices.
- *Step 8* - Rank the strategies according to the weights obtained in step 6.

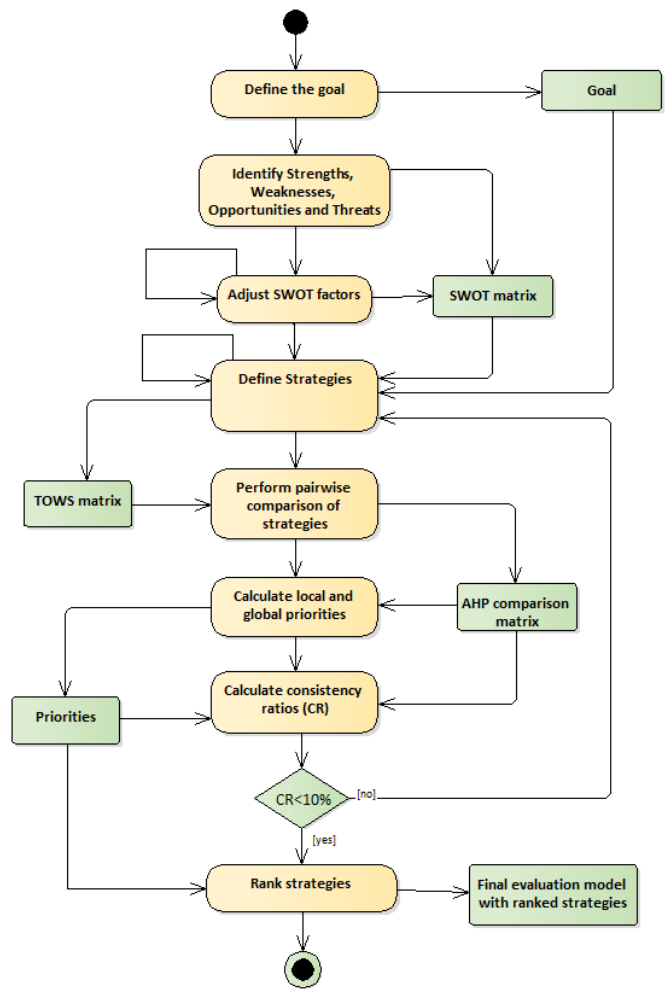


**Figure 3** The SWOT-TOWS-AHP process.

To validate our proposal, we conducted a survey among software project managers. The validation goal was the adoption of model-based software engineering and the result was a ranking of strategies for such adoption. For more information about the validation see Section 4.

Note that the use of our analytical tool is not restricted to the specific goal pursued during validation, The tool is just as applicable for other strategic decisions in the IT domain, such as those related to architecture, agile development, DevOps, etc. Usually, stakeholders from the same enterprise or organisation will be involved in such a strategic survey.

## 4. Applying SWOT-TOWS-AHP Methodology

Many enterprises are now beginning to evaluate their possibilities of adopting the MBSE paradigm. Although there is sufficient information and discussion about the pros and cons, each individual company has to analyse its own benefits and hurdles, and based on their conclusions decide whether or not to adopt MBSE in one, some or all projects, and how and when to do so (Vallecillo 2015). The "how" is the most challenging aspect, so much so that enterprises actually need support to help them move from making a list of MBSE benefits and obstacles to defining the best strategy and making the best decisions. Such support whether in the form of guidelines, frameworks or processes is difficult to find in the literature. In this section, we therefore present an example of how SWOT-TOWS-AHP can be used to evaluate factors and find an enterprise's best MBSE adoption strategy.

### 4.1. Conducting the SWOT Analysis

Building a SWOT is not an easy task. For the construction of our SWOT, we followed the process described in Section 3 (Steps 1, 2 and 3). Firstly, we built a SWOT based on an analysis of the literature, in particular, those papers mentioned in the description of the background and related work (Sections 2 and 6), e.g. (Amorim et al. 2019; Störrle 2017; Vallecillo 2015; Whittle et al. 2013; Hutchinson et al. 2011; Teppola et al. 2009; Forward & Lethbridge 2008). This preliminary version of the SWOT was presented to four experts in MBSE in industry, who reviewed our approach, and suggested some improvements. An overview of the resulting SWOT is shown in Table 3. The corresponding strengths, weaknesses, opportunities and threats are briefly described below.

***Strengths*** *(S)*

- *S1 - Improvement in software quality characteristics*. Some specific features of quality, like robustness, traceability, adaptability and modularity, all of which are required in software (ISO / IEC 25010 2011) could be improved with the use of MBSE (Panach et al. 2021).
- *S2 - Reduction of software development and maintenance efforts and costs*. MBSE can increase automatism and help to detect errors in early phases of software development. Traceability mechanisms can also help in reducing software maintenance costs and increasing the "time-to-market" (Vallecillo 2015).
- *S3 - Adaptability to different software environments*. Higher level models are platform independent artefacts. That means that such models can be more easily adapted to different platforms, languages or interfaces, thus improving the versatility of the software production (Brown 2004).
- *S4 - Suitable MBSE tools reduce the learning curve of the software team*. Software teams are constantly changing their environments to assimilate new technologies, new software development platforms, etc. The adoption of MBSE can introduce a certain automatism, supported by widgets, predefined artefacts, patterns, and other elements, which can help software engineers to adapt to these new environments more quickly (Khandoker et al. 2022).
- *S5 - Increased capacity to manage software development complexity*. Since models describe certain aspects of a system under construction in an abstract way, MBSE makes it possible to avoid unnecessary complexity while at all times retaining a high-level vision of the system-to-be (Brown 2004).
- *S6 - Improvement of team communication*. Models can be "a common language" for all the members of a team allowing developers, analysts, managers, or even users to discuss application features (Störrle 2017).

***Weaknesses*** *(W)*

- *W1 - Lack of guidance in the modelling and use of MBSE*. In comparison with examples of code repositories, there are fewer detailed reports regarding well-known successful cases of MBSE application (Vallecillo 2015; Luna et al. 2018).
- *W2 - Lack of appropriate, readily available tools for model development and integration*. MBSE software tools are sometimes unstable or of little use in productive settings. They are not often scalable, efficient or fully operable. This limits the use of models in industry (Whittle et al. 2013).
- *W3 - Lack of integration with development frameworks*. MBSE seems to be too disconnected from existing popular infrastructures. Its tools are seldom compatible with existing code-based frameworks (Luna et al. 2018).
- *W4 - Lack of expressiveness communicating non-functional requirements*. As discussed in (Ameller et al. 2019), modelling approaches, particularly those supporting MDE do not include primitives to deal with non-functional requirements such as usability, scalability, etc. These are usually dealt with at coding level.

***Opportunities*** *(O)*:

- *O1 - Increased popularity of low-code or no-code platforms*. The use of these platforms which ultimately use some kind of modelling approach could stimulate and motivate the use of models (Cabot 2020).
- *O2 - Increased social demand for time-to-market software production*. Present-day society revolves around software. Development teams work under pressure to satisfy the huge demand of software and deliver it as fast as possible. This is an ideal context to transit to MBSE (Störrle 2017).
- *O3 - Potential synergy with other engineering disciplines*. The need to interact with different engineering endeavours, such as embedded or cyber-physical systems could also motivate the use of models in software, since other engineers are already using them (Broy et al. 2012; Liebel et al. 2018; Morin et al. 2017).
- *O4 - Need for even more decoupling of business and technical issues in software development*. Good definitions or analyses of a software solution are very often limited to the future technology that is going to be used for their development. Analysts frequently end up thinking more about the technological solution than about the problem.

| Strengths | Weaknesses |
|---|---|
| S1 Improvement in software quality characteristics | W1 Lack of guidance in the modelling and use of MBSE |
| S2 Reduction of software development and maintenance efforts and costs | W2 Lack of appropriate, readily available tools for model development and integration |
| S3 Adaptability to different software environments | W3 Lack of integration with development frameworks |
| S4 Suitable MBSE tools reduce the learning curve of the software team | W4 Lack of expressiveness communicating non-functional requirements |
| S5 Increased capacity to manage software development complexity | |
| S6 Improvement of team communication | |
| **Opportunities** | **Threats** |
| O1 Increased popularity of low-code or no-code platforms | T1 Difficulty of quantifying investments due to paradigm change |
| O2 Increased social demand for time-to-market software production | T2 Additional efforts required for maintenance of existing developments |
| O3 Potential synergy with other engineering disciplines | T3 Lack of MBSE experts in industry. |
| O4 Need for even more decoupling of business and technical issues in software development | T4 Absence of an enterprise modellers' community |
| O5 Use of Domain Specific Languages (DSL) in certain areas | T5 Coding-culture preventing mindset change |

**Table 3** SWOT matrix for MBSE adoption.

Again, the need to think independently of the technology could stimulate the use of models (Vallecillo 2015).

– *O5 - Use of Domain Specific Languages (DSL) in certain areas*. In some fields, like aeronautics or healthcare, there is an increasing use of domain specific languages (García-Borgoñon et al. 2013). This could awaken interest in modelling.

***Threats (T)***

– *T1 - Difficulty of quantifying investments due to paradigm change*. The application of MBSE is a global change of paradigm that can be very expensive to assume (Vallecillo 2015).
– *T2 - Additional efforts required for maintenance of existing developments*. The change of paradigm to MBSE cannot be automatic and all-embracing, so MBSE will have to coexist with other classic paradigms (Teppola et al. 2009; Hutchinson et al. 2014).
– *T3 - Lack of MBSE experts in industry*. There are relatively few experts in MBSE available to industry. Academia currently does not generate enough MBSE experts and (in part due to job offers) software engineers tend to be attracted to more technological solutions (Vallecillo 2015; Teppola et al. 2009).
– *T4 - Absence of an enterprise modellers' community*. MBSE is poorly employed in industry and no community exists there to support and promote its application. There is certainly no equivalent to the popular code repositories where developers discuss problems and solutions (Luna et al. 2018; Hutchinson et al. 2014).
– *T5 - Coding-culture preventing mindset change*. Software and system engineers are closer to coding culture. Most industries and therefore most job opportunities are focused on coding and most of software studies in universities are oriented towards code and technology (Selic 2008).

### 4.2. Constructing the Strategies Matrix

This section explains how we built the strategies matrix (TOWS) for the adoption of MBSE. It corresponds to the result of Step 4 of our methodology (see Section 3). The aim was to find strategies that address the strengths, opportunities, weaknesses and threats identified and described in the previous step (Section 4.1), and to classify them following the TOWS schemata SO, ST, WO and WT (see Figure 2).

In industry, managers, employees and experts who are aware of the company's and environmental situation often define strategies, similar to the process of collecting the SWOT factors (Gürel & Tat 2017). Usually, strategy planning is mainly based on situational analysis, brainstorming, forecast preparation, etc. (Weihrich 1982).

In our case, we selected input from the literature review and used it in successive discussion rounds with experts, in order to obtain a set of alternative strategies. The main input source from the literature was the analysis performed in (Vallecillo 2015). In that article, Vallecillo discusses hurdles for the adoption of

MDE in a company and includes twelve questions that could be helpful to make decisions in such adoption. We re-formulated these questions as alternative strategies; the overview is shown in Table 4. For example: Vallecillo's question regarding the company's commitment (# 7) results in ST1; the question related to the criteria to use and metrics needed (# 9) is summed up in WT2; Vallecillo's question related to the cost and the duration of a project (# 8) is our SO2; the question on understanding implications of the MDE adoption (# 1) is formulated as SO1 and WT3, etc.

The discussion with the experts was centred on how these strategies addressed the selected SWOT factors and whether names or descriptions needed some adjustments. The resulting relationship between factors and strategies is shown in Table 5. For example, the strategy that proposes to import expertise (WO1) aims to mitigate the weakness due to the lack of guidance using MBSE techniques (W1), and to benefit from the opportunities of using know-how of other engineering disciplines (O3) and the decoupling of business and technical issues (O4). Even more intuitive is the relationship between the strategy suggesting training (ST3) and the strengths S4 and S5 as well the threats T3 and T4. Training (ST3) increases know-how and abilities on MBSE tools (S4) and management of software development complexity (S5), and reduces the lack of expertise in industry (T3) and the lack of an enterprise modellers' community (T4).

| **Maxi-Maxi** | **Maxi-Mini** |
|---|---|
| SO1 Motivation and innovation | ST1 Commitment and support |
| SO2 Costs and duration | ST2 Process evolution |
| SO3 Integration | ST3 Training |
| **Mini-Maxi** | **Mini-Mini** |
| WO1 Importing expertise | WT1 Outsourcing |
| WO2 Investment in mature tools | WT2 Measurement of success/-failure |
| WO3 Use of DSL | WT3 Migration and redesign |

**Table 4** TOWS matrix for MBSE adoption.

***Maxi-Maxi Strategies***

- *SO1 - Motivation and innovation*. This strategy represents the idea of analysing a company's motivation and innovation needs with a view to adopting MBSE as a solution to new commercial and organisational challenges.
- *SO2 - Costs and duration*. This strategy involves estimating project costs and duration using MBSE in comparison with current development costs and duration. It also takes into account the calculation of return on investment (ROI) considering the additional costs incurred in the implementation of a pilot project.
- *SO3 - Integration*. This strategy focuses on integrating the MBSE approach with the other notations and methods used in the company.

***Maxi-Mini Strategies***

- *ST1 - Commitment and support*. This strategy is aimed at obtaining the fully commitment and support of the enterprise management; i.e., the willingness of the key players in the project/company.
- *ST2 - Process evolution*. This strategy supports the need to identify the changes that are required in current processes in order to use MBSE in a project.
- *ST3 - Training*. This strategy addresses the need to train development teams in the MBSE concepts, tools and processes that are required to carry out a project.

***Mini-Maxi Strategies***

- *WO1 - Importing expertise*. This strategy exploits the opportunity to work together with MBSE experts from other branches of engineering.
- *WO2 - Investment in mature tools*. This strategy considers investing in MBSE tools that are mature enough for the project, and also giving feedback to tool providers.
- *WO3 - Use of DSL*. This strategy consists of adopting or defining domain specific modelling languages, as is done in aeronautics and healthcare.

***Mini-Mini Strategies***

- *WT1 - Outsourcing*. This strategy consists of outsourcing the project to another company with expertise in MBSE and planning the transfer of external know-how to development teams.
- *WT2 - Measurement of success/failure*. This strategy consists of selecting the criteria and metrics needed to measure the success/failure of the project. It will allow results to be communicated to customers as a means of increasing endorser participation in software design.
- *WT3 - Migration and redesign*. This strategy focuses on migrating and redesigning existing software developments.

As mentioned, Table 5 shows the relationship between strategies and SWOT factors, identifying which strengths, weaknesses, opportunities and threats are addressed by each strategy with the SWOT factors in columns and the strategies in rows. By implementing strategy SO1, for example, a company can make use of opportunities O1 and O5 and benefit from strengths S1, S2, S3, S4, S5 and S6.

### 4.3. Comparing Strategies Pairwise

We used the TOWS matrix and the description of the SO, ST, WO and WT strategies to create a questionnaire for an online survey. The link to the survey was sent via email to the selected group of experts (Step 5 of our methodology described in Section 3). The experts' answers provided the information needed to carry out the AHP quantitative evaluation.

The questionnaire comprised 36 questions for the pairwise comparisons and five questions on professional experience. Questions took the form: a) "Which strategy is in your opinion

| Strategies | Strengths | | | | | | Weaknesses | | | | Opportunities | | | | | Threats | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | S1 | S2 | S3 | S4 | S5 | S6 | W1 | W2 | W3 | W4 | O1 | O2 | O3 | O4 | O5 | T1 | T2 | T3 | T4 | T5 |
| SO1 | X | X | X | X | X | X | | | | | X | | | | X | | | | | |
| SO2 | | X | | | | | | | | | | X | | | | | | | | |
| SO3 | | | X | | X | X | | | | | | | X | X | X | | | | | |
| ST1 | X | X | X | X | X | X | | | | | | | | | | X | | | | X |
| ST2 | X | | X | | | | | | | | | | | | | X | X | | | |
| ST3 | | | | X | X | | | | | | | | | | | | | X | X | |
| WO1 | | | | | | | X | | | | | | X | X | | | | | | |
| WO2 | | | | | | | | X | X | | X | X | | X | X | | | | | |
| WO3 | | | | | | | X | | | X | | X | | X | X | | | | | |
| WT1 | | | | | | | X | X | | | | | | | | | | X | X | X |
| WT2 | | | | | | | | | X | | | | | | | X | | | | |
| WT3 | | | | | | | X | X | | | | | | | | | X | X | X | |

**Table 5** Relationship between SWOT factors and MBSE adoption strategies.

more important (select both if equal)?" followed by descriptions of the two strategies, e.g., WT1 and WT2. b) "In your opinion, how much more important is WT1 than WT2 or WT2 than WT1?; and c) "Are you working in the private or public sector?" "How many years on software development experience do you have?". Figure 4 shows an example of questions for comparing two strategies, the options for selecting either one or both strategies, and the nine-point scale of importance.

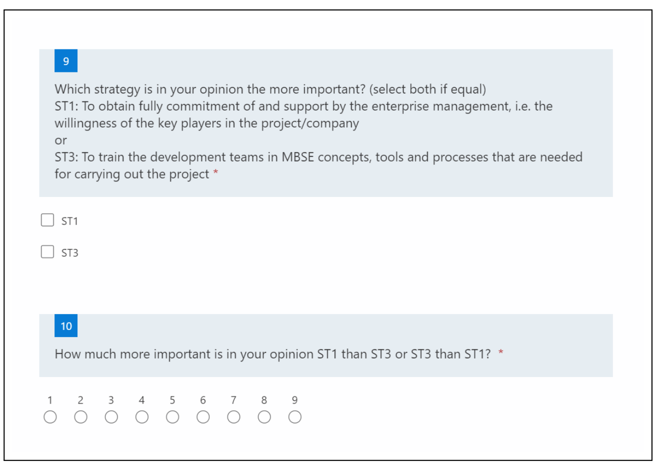


**Figure 4** Survey excerpt.

The online survey was sent to 27 experts (17 men and 10 women) in two iterations to ensure its comprehensibility. In the first iteration six experts were asked to answer the questions and provide feedback for improvement; the other experts were consulted in the second iteration. The survey was completed anonymously by 21 respondents (six of the first and 15 of the second iteration). 14 experts were men and seven women, whom eight worked in public organisations and 13 worked in the private sector (10 in large corporations and three in SMEs). All experts had extensive experience in MBSE; 18 of them more than 10 years. The range of MBSE projects they had been involved encompassed as many as 50 projects, with an average of 13, and their experience in MBSE adoption ranged from cero to 34 times with an average number of six. The questionnaire and the raw survey data have been published online (Escalona et al. 2022a,b).

The answers obtained from the experts were (a) complemented with the calculated reciprocal value, and (b) aggregated using Equation 1 for the geometric mean as explained in Section 3 (see explanation in the next subsection).

### 4.4. Using AHP to Calculate Priorities

In this section, the results of the pairwise comparisons of factors in the TOWS groups (strategy types) and for the strategies within each group are presented. We explain how AHP reciprocal matrices are constructed, how priorities calculated using Equation 3 of Section 3, and a ranking of weights (priorities) for those strategies obtained.

For each answer by an expert, as mentioned, reciprocal values were calculated for a pair of strategies. For example, if SO2 was weighted as five time more important than SO1, then the pair (SO2,SO1) had the weight five and the weight assigned to the pair (SO1, SO2) was 1/5, i.e., 0.20. If both strategies were evaluated as being of equal importance, the weight was one; and the elements of the diagonal of the matrix, e.g., pair (SO1,SO1), would always be equal to one. The aggregation of all experts' judgements for all pairs of strategies $(X_i; X_j)$ was then calculated using the geometric mean, where X refers to a strategy type, i.e., SO, ST, WO or WT (For further details on the geometric mean see formula 1 in Section 3).

Since in our case each group of strategies (Maxi-Maxi, Maxi-Mini, Mini-Maxi and Mini-Mini) contained three strategies, the starting point for the AHP priorities calculations (Step 6, see Section 3) was four 3 x 3 matrices filled in with the weights obtained using the geometric means, as shown in Tables 6, 7, 8 and 9. Although the AHP technique does not require an equal number of factors or strategies for each group, a non-substantially different number is recommended.

The priorities of the last column in Tables 6, 7, 8 and 9 were calculated as follows: (a) the given matrix was normalised, (b) the eigenvector formula presented in Equation 3 was applied to this normalised matrix with n = 3. Note that the sum of priorities is always equal to one.

| **Maxi-Maxi Strategies** | **SO1** | **SO2** | **SO3** | **Priority** |
|---|---|---|---|---|
| SO1 Motivation and innovation | 1 | 0.76 | 0.88 | 0.286 |
| SO2 Costs and duration | 1.32 | 1 | 1.72 | 0.429 |
| SO3 Integration | 1.13 | 0.58 | 1 | 0.284 |
| | $\lambda max = 3.017 \quad CR = 0.014$ | | | |

**Table 6** Weights and priorities for strategies maximising strengths and opportunities.

| **Maxi-Mini Strategies** | **ST1** | **ST2** | **ST3** | **Priority** |
|---|---|---|---|---|
| ST1 Commitment and support | 1 | 1.49 | 0.81 | 0.352 |
| ST2 Identification of changes | 0.67 | 1 | 0.87 | 0.276 |
| ST3 Training | 1.23 | 1.16 | 1 | 0.371 |
| | $\lambda max = 3.023 \quad CR = 0.020$ | | | |

**Table 7** Weights and priorities for strategies maximising strengths and minimising threats.

| **Mini-Maxi Strategies** | **WO1** | **WO2** | **WO3** | **Priority** |
|---|---|---|---|---|
| WO1 Importing expertise | 1 | 1.86 | 2.00 | 0.489 |
| WO2 Investment in mature tools | 0.54 | 1 | 1.39 | 0.287 |
| WO3 Use of DSML | 1 | 0.72 | 1 | 0.225 |
| | $\lambda max = 3.008 \quad CR = 0.006$ | | | |

**Table 8** Weights and priorities for strategies minimising weaknesses and maximising opportunities.

Using pairwise comparison, the experts also determine the relative importance of the strategy groups SO, ST, WO and WT. The resulting priorities express the weight of the groups with respect to the goal of MBSE adoption either in the company as a whole or for a specific project (see Table 10).

Consistency ratios (CRs) were calculated for Tables 6 to 10 based on formula given in Equation 5 (Saaty 1990) explained

| **Mini-Mini Strategies** | **WT1** | **WT2** | **WT3** | **Priority** |
|---|---|---|---|---|
| WT1 Outsourcing | 1 | 0.39 | 0.69 | 0.196 |
| WT2 Measurement of success/-failure | 2.55 | 1 | 2.33 | 0.547 |
| WT3 Migration and redesign | 1.44 | 0.43 | 1 | 0.257 |
| | $\lambda max = 3.008 \quad CR = 0.007$ | | | |

**Table 9** Weights and priorities for strategies minimising weaknesses and threats.

| **Strategy Groups** | **SO** | **ST** | **WO** | **WT** | **Priority** |
|---|---|---|---|---|---|
| SO Maxi-Maxi | 1 | 0.82 | 2.02 | 2.73 | **0.333** |
| ST Maxi-Mini | 1.22 | 1 | 2.14 | 2.50 | **0.365** |
| WO Mini-Maxi | 0.50 | 0.47 | 1 | 1.37 | **0.171** |
| WT Mini-Mini | 0.37 | 0.40 | 0.73 | 1 | **0.131** |
| | $\lambda max = 4.008 \quad CR = 0.003$ | | | | |

**Table 10** Weights and priorities for strategy groups.

in Section 3. The $\lambda$ value and the CR are listed at the bottom of each table. All $\lambda$ values were $\geq$ n, whit $n = 3$ or $n = 4$ depending on the size of the matrix. All consistency ratios are $\leq 10\%$, satisfying this way the AHP consistency requirement.

In the next step (Step 7 of our methodology), the overall priority scores of the TOWS strategy matrix were calculated by multiplying the local priorities of the strategies in each group by the group priority (see Table 10). Local priorities are those shown in Tables 6, 7, 8 and 9. Global priorities are those shown in Table 11. Finally, the global priorities could be ranked based on their values (Step 8). The ranking is shown in the last column of Table 11. Figure 5 provides a visual representation of the relative impact of the MBSE adoption strategies.

## 5. Discussion of the Results

In this work, we applied the SWOT-TOWS-AHP methodology defined in Section 3 to select the best strategy for adopting model-based software development in an enterprise.

The findings ranked the TOWS groups of strategies as follows: Maxi-Mini 36.5%, Maxi-Maxi 33.3%, Mini-Maxi 17.1% and Mini-Mini 13.1%. This means that, according to the answers obtained in this survey, emphasis should be placed on maximising strengths and at the same time minimising threats. Note that consistency ratios for all matrices were below 3% (AHP requires below 10%).

According to the quantitative analysis the strategy with the highest priority (14.3%) was SO2, which consists of estimating the costs and duration of a project when adopting MBSE and comparing them to existing software development costs and project duration in the company or public organisation. The fact that this strategy scored highest in the ranking confirms

| Strategy Group | Group Priority | Strategy | Local Priority | Global Priority | Ranking |
|---|---|---|---|---|---|
| Maxi-Maxi | 0.333 | SO1 Motivation and innovation | 0.286 | 0.095 | 6 |
| | | SO2 Costs and duration | 0.429 | 0.143 | 1 |
| | | SO3 Integration | 0.284 | 0.095 | 5 |
| Maxi-Mini | 0.365 | ST1 Commitment and support | 0.352 | 0.129 | 3 |
| | | ST2 Identification of changes | 0.276 | 0.101 | 4 |
| | | ST3 Training | 0.371 | 0.136 | 2 |
| Mini-Maxi | 0.171 | WO1 Importing expertise | 0.489 | 0.084 | 7 |
| | | WO2 Investment in mature tools | 0.287 | 0.049 | 9 |
| | | WO3 Using DSML | 0.225 | 0.038 | 10 |
| Mini-Mini | 0.131 | WT1 Outsourcing | 0.196 | 0.026 | 12 |
| | | WT2 Measurement of success/failure | 0.547 | 0.072 | 8 |
| | | WT3 Migration and redesign | 0.257 | 0.034 | 11 |

**Table 11** Overall priority scores and ranking of TOWS MBSE adoption strategies.

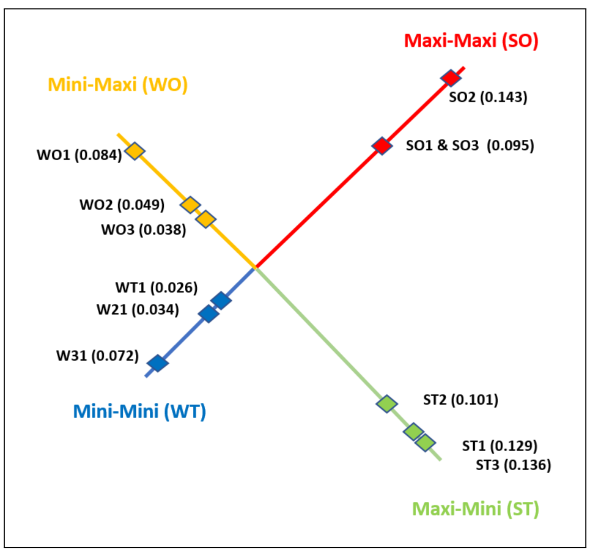


**Figure 5** Relative impact of MBSE adoption strategies based on the global priorities shown in Table 11.

the qualitative statement "Therefore, each company, depending on the particular project, on its perceived value and ROI, and on its capability to be resolved using MDE, should decide whether to adopt MDE or not, for which projects, and with which development team." (Vallecillo 2015).

Strategy ST3 was second in the ranking with a priority of 13.6% highlighting the importance of training of development teams. This is not surprising because human resources are critical resources in software projects. Project success depends on team expertise. According to the European Union's Digital Skills and Job Platform, and more concretely the study (Dan et al. 2021), hiring and retaining the best experts with the know-how required for a project is considered one of the most important challenges for IT enterprises. Again, the position in the ranking of ST3 concurs that training is one of the "three major factors that do have significant impact on the success of an MDE project" (Vallecillo 2015).

The third ranked strategy, ST1 with a priority of 12.9%, is linked to the fact that any strategic change in a company, requires the commitment of all stakeholders involved in that change if conflicts and problems are to be avoided in adopting a different paradigm.

Note that these strategies of higher priority address managerial, economic and social aspects more than technical issues.

Our intention in this research was not to obtain a definitive answer about which strategies are the best for MBSE adoption, and much less to suggest that these strategies can be applied in every scenario. Rather than that, we aimed to propose a tool potentially capable of helping decision makers in the MBSE adoption process and to provide a good example of its use. However, it is important to highlight possible threats to the validity of the results obtained in our survey. The first is that, in the SWOT and TOWS matrices, we selected the factors and strategies most relevant for MBSE adoption according to the literature. This, however, was undoubtedly a subjective choice. To mitigate this subjectivity, we had our decisions reviewed by four experts who provided some very useful comments and suggestions. The second limitation, which may have strongly influenced the results, was the understanding of each question in the survey. Since we did not conduct personal interviews during the survey, any misunderstanding of a question could corrupt the results. Even though we used a two-round approach to improve the survey, this continued to be a possibility. Some remedies to these (and other) problems are proposed in Section 7.

## 6. Related Work

For the sake of clarity this section is divided into two parts: one describing existing work on SWOT analysis and the other describing the existing quantitative surveys related to MBSE.

### 6.1. Related Work on SWOT Analysis

As previously mentioned, although SWOT analysis is relatively popular in IT, particularly for assessing "new" technologies, it has seldom been used in combination with quantitative methods, as a tool for decision makers. However, many interesting examples can be found of SWOT matrices to summarise the characteristics of a technology. In (Lotz et al. 2019), for example, the authors present a thorough analysis based on a SWOT matrix to evaluate the use of microservices architectures in vehicular software. In (Ghaffari et al. 2014) (Penzel et al. 2015) (Tang et al. 2013) the authors present different views (rooted in SWOT analysis) of the present and future of cloud computing. In (Shahir et al. 2008) a SWOT analysis is proposed to identify strategies for Agile software development. The strategies described are similar to ours, although they are not elaborated upon. Finally, and closer to our specific field (Vallecillo et al. 2007) presented an initial SWOT matrix for model-driven web engineering. Not surprisingly many of the factors cited are similar to those presented here and elsewhere in the literature. A SWOT analysis on the OCL is outlined in (Cabot et al. 2021) .

The use of SWOT in combination with AHP is popular in different fields. Besides the previously mentioned work by Kurttila (Kurttila et al. 2000), there have been some remarkable studies in areas related to IT. (Kahraman et al. 2007), for example, used this approach to evaluate alternative strategies for e-Government applications at national level in Turkey, while (Taleai et al. 2009) used it to assess the challenges of using GIS systems in developing countries. In (Mehmood et al. 2014) an analysis was carried out to assess the factors relevant to the adoption of near field communication (NFC) in Italy.

What is common in these works, and in many similar studies is that the combination of a quantitative technique (in this case AHP), a qualitative analysis (like SWOT), and information obtained from a survey conducted among experts, provides an objective indicator of the impact of different strategies when a complex decision (such as adopting or changing some critical technology) has to be made. Many researchers point out, however, that a post-decision (and post-adoption) analysis may be necessary to subsequently assess the impact of the adoption. One important difference between the studies mentioned above and our own work is that we have complemented the SWOT factors with the TOWS approach focusing on the comparison and prioritisation of alternative strategies with the AHP technique.

### 6.2. Quantitative Studies Related to MBSE

Different authors have assessed the benefits and problems of MBSE quantitatively, but without using a multiple-criteria decision approach.

One interesting outcome of the previously mentioned survey in (Störrle 2017) has to do with how each of the survey respondents endorse the well-known advantages of modelling; such as higher quality software delivery, savings in time and effort, the generation of complete applications from models, etc.

(Broy et al. 2012) presents an extensive survey on the development of embedded software in the car industry in which respondents indicate positive and negative experiences and identify factors that have influence on costs, time and quality, due to model-based development. One of the conclusions of the study is that MBSE can offer significant advantages (during development and maintenance) only when using a well-selected development process: in other cases, it might lead to greater costs.

(Teppola et al. 2009) reports the results of a survey on companies using model-driven development highlighting the impact of organisation size on usage of the approach and on the associated benefits and challenges. The most severe challenge detected related to the lack of modelling experts in the organisations.

(Hutchinson et al. 2014) conducted a large online survey on MDE deployment that provides meaningful quantitative information about practices in industry. While this study produced many interesting results on the type of diagrams regularly used for modelling, and different kinds of model usage (e.g., for understanding, communication, code generation, etc.), the most valuable insight comes from a set of paired questions aimed to assessing the balance between positive and negative aspects of MDE. In an earlier paper (Hutchinson et al. 2011), the same authors have carried out a complementary study to measure (using paired questions) how different impact factors (in productivity, portability and maintenance) are influenced positively or negatively when using MDE. In this regards (the use of paired questions), both studies are somewhat similar to our research.

(Forward & Lethbridge 2008) compares code-centric vs. model-centric approaches. Their survey focuses on which tasks are performed better in each of the two approaches. The survey responses also helped identify the problems in a model-centric approach (some of which appear as “weaknesses” in our SWOT analysis). Another interesting outcome here is the relationship between sub-samples of participants and their answers; people classified as “programmers”, for example, consider modelling tools as “heavyweight”; while people that model are less likely to agree than models become outdated or inconsistent with code. Finally a set of challenges, is outlined for both approaches.

A more recent, and very interesting survey (Liebel et al. 2018) addresses the state of practice of MBSE in the embedded system domain. The authors analyse what is needed to introduce MBSE, together with the positive and negative effects of MBSE and its shortcomings.

(Whittle et al. 2013) addressed the importance of tools as a barrier in the adoption of MDE techniques. Using a thorough set of interviews the authors categorised issues related to tools (features, applicability, complexity, etc.), internal organisation (processes, culture and skills), external organisation (e.g., commercial aspects) and social aspects. They conclude that although tools are indeed a problem, processes and organisational issues are also critical.

(Amorim et al. 2019) assessed 18 best practices for tackling MBSE adoption challenges in the embedded systems industry. These best practices were defined by means of a set of semi-structured interviews with experts from embedded systems organisations and, later validated and prioritised with the help of an online questionnaire answered by MBSE practitioners. This paper, however, focused exclusively on embedded systems organisations.

## 7. Concluding Remarks

In this paper we have presented a quantitative analysis for the adoption of model-based software engineering. The analysis is based on the combination of a SWOT (Strengths, Weaknesses, Opportunities, Threats) analysis and the application of the Analytical Hierarchy Process (AHP). We mediated both analysis tools with a TOWS matrix designed to define strategic options from an external-internal analysis. Our survey was answered by 21 experts in software engineering, most of whom have more than 10 years of experience working in industry or (non-academic) public organisations. They confirmed certain well-known assumptions regarding MBSE adoption and confirmed the potential of the analytical process itself (the combination of SWOT-TOWS-AHP) as a powerful tool to help decision makers in the field of software technology (a field where quantitative analysis is seldom used).

This is the initial step of a more ambitious study. First, the survey needs to be sent to a much larger panel of experts to consolidate (or not) the results. Another possibility is a sensitivity analysis to determine how the effect of any change in the importance of the main factors will affect the priorities of the strategies. Detailed pairwise comparison could also be used at the SWOT factor level, and to compare factors' efficiency for each strategy.

Another interesting initiative for future research would be to use the methodology for a specific project involving all the stakeholders in a company. This would also provide the opportunity to perform a post-adoption analysis.

In order to bring our work closer to business reality, we need to facilitate the use of our SWOT-TOWS-AHP analytical tool. The idea is to create a library of strengths, weaknesses, opportunities and threats, and possible strategies. Thus, the companies, in particular, specific types of companies, could start from these lists, at least partially, and build their own SWOT without to go over the full process.

Finally, we are planning to compare the use of AHP with other multiple-criteria decision-making analytical tools.

## 8. Acknowledgements

We would like to thank Antonio Vallecillo for having inspired our research on model-driven engineering with discussions, co-authoring and organizing events on this topic. We would like to thank all the experts who provided feedback on the draft version of the SWOT matrix and those who participated in the pairwise survey of the factors and strategies. Their judgement made it possible to validate our approach. This publication is part of the project PID2019-105455GB-C31, funded by MCIN/AEI/10.13039/501100011033/ and by the European Union and by the project P20/00644, funded by Junta de Andalucia.

## References

Abrahão, S., Bourdeleau, F., Cheng, B. H. C., Kokaly, S., Paige, R. F., Störrle, H., & Whittle, J. (2017). User Experience for Model-Driven Engineering: Challenges and Future Directions. In *20th ACM/IEEE International Conference on Model Driven Engineering Languages and Systems, MODELS 2017, Austin, TX, USA, September 17-22, 2017* (pp. 229–236). IEEE Computer Society.

Ameller, D., Franch, X., Gómez, C., Martínez-Fernández, S., Araújo, J., Biffl, S., … others (2019). Dealing with non-functional requirements in model-driven development: A survey. *IEEE Transactions on Software Engineering*, *47*(4), 818–835.

Amorim, T., Vogelsang, A., Pudlitz, F., Gersing, P., & Philipps, J. (2019). Strategies and best practices for model-based systems engineering adoption in embedded systems industry. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)* (pp. 203–212).

Brambilla, M., Cabot, J., & Wimmer, M. (2017). *Model-Driven Software Engineering in Practice, Second Edition*. Morgan & Claypool Publishers.

Brown, A. W. (2004). Model driven architecture: Principles and practice. *Softw. Syst. Model.*, *3*(4), 314–327.

Broy, M., Kirstan, S., Krcmar, H., & Schätz, B. (2012). What is the benefit of a model-based design of embedded software systems in the car industry? In *Emerging technologies for the evolution and maintenance of software models* (pp. 343–369). IGI global.

Bucchiarone, A., Cabot, J., Paige, R. F., & Pierantonio, A. (2020). Grand challenges in model-driven engineering: An analysis of the state of the research. *Softw. Syst. Model.*, *19*(1), 5–13.

Bucchiarone, A., Ciccozzi, F., Lambers, L., Pierantonio, A., Tichy, M., Tisi, M., … Zaytsev, V. (2021). What Is the Future of Modeling? *IEEE Softw.*, *38*(2), 119–127.

Burgueño, L., Ciccozzi, F., Famelis, M., Kappel, G., Lambers, L., Mosser, S., … Wimmer, M. (2019). Contents for a Model-Based Software Engineering Body of Knowledge. *Softw. Syst. Model.*, *18*(6), 3193–3205.

Cabot, J. (2020). Positioning of the low-code movement within the field of model-driven engineering. In E. Guerra & L. Iovino (Eds.), *MODELS '20: ACM/IEEE 23rd International Conference on Model Driven Engineering Languages and Systems, Virtual Event, Canada, 18-23 October, 2020, Companion Proceedings* (pp. 76:1–76:3). ACM.

Cabot, J., Calegari, D., Clarisó, R., Gogolla, M., Vallecillo, A., & Willink, E. D. (2021). A SWOT Analysis of the Object Constraint Language. In L. Iovino & L. M. Kristensen (Eds.), *STAF 2021 Workshop Proceedings: Software Technologies: Applications and Foundations, Federation of Conferences (STAF 2021), Bergen, Norway, June 21-25, 2021* (Vol. 2999, pp. 178–184). CEUR-WS.org.

Ciccozzi, F., Famelis, M., Kappel, G., Lambers, L., Mosser, S., Paige, R. F., … Wimmer, M. (2018). How do we teach modelling and model-driven engineering?: A survey. In Ö. Babur et al. (Eds.), *Proceedings of the 21st ACM/IEEE International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings, MODELS 2018, Copenhagen, Denmark, October 14-19* (pp. 122–129). ACM.

Dan, S., Ivana, D., Zaharie, M., Metz, D., & Drăgan, M. (2021). Attracting, Developing and Retaining Digital Talent: Empirical Evidence. In *Digital Talent Management* (pp. 47–60). Springer.

Dedehayir, O., & Steinert, M. (2016). The hype cycle model: A review and future directions. *Technological Forecasting and Social Change*, *108*, 28–41.

Di Ruscio, D., Kolovos, D., de Lara, J., Pierantonio, A., Tisi, M., & Wimmer, M. (2022). Low-code development and model-driven engineering: Two sides of the same coin? *Software and Systems Modeling*, 1–10.

Escalona, M. J., Parcus de Koch, N., & Rossi, G. (2022a). *Answers to questionnaire on pairwise comparison of strategies.* https://nora.fam-koch.com/publications/survey2022/answers-MBSE.pdf. (published 30.04.2022)

Escalona, M. J., Parcus de Koch, N., & Rossi, G. (2022b). *Questionnaire on pairwise comparison of strategies.* https://nora.fam-koch.com/publications/survey2022/questionnaire-MBSE.pdf. (published 30.04.2022)

Forward, A., & Lethbridge, T. C. (2008). Problems and opportunities for model-centric versus code-centric software development: A survey of software professionals. In J. M. Atlee et al. (Eds.), *International Workshop on Modeling in Software Engineering, MiSE 2008, Leipzig, Germany, May 10-11, 2008* (pp. 27–32). ACM.

García-Borgoñon, L., García-García, J. A., Alba, M., & Escalona, M. J. (2013). Software process management: A model-based approach. In *Building Sustainable Information Systems* (pp. 167–178). Springer.

Ghaffari, K., Delgosha, M. S., & Abdolvand, N. (2014). Towards Cloud Computing: A SWOT Analysis on its Adoption in SMEs. *CoRR*, *abs/1405.1932*. Retrieved from http://arxiv.org/abs/1405.1932

Gogolla, M., & Selic, B. (2020). On teaching descriptive and prescriptive modeling. In E. Guerra & L. Iovino (Eds.), *MODELS '20: ACM/IEEE 23rd International Conference on Model Driven Engineering Languages and Systems, Virtual Event, Canada, 18-23 October, 2020, Companion Proceedings* (pp. 23:1–23:9). ACM.

Gottfried, O., De Clercq, D., Blair, E., Weng, X., & Wang, C. (2018). SWOT-AHP-TOWS analysis of private investment behavior in the Chinese biogas sector. *Journal of Cleaner Production*, *184*, 632–647.

Gregory, J., Berthoud, L., Tryfonas, T., Rossignol, A., & Faure, L. (2020). The long and winding road: MBSE adoption for functional avionics of spacecraft. *J. Syst. Softw.*, *160*.

Gürel, E., & Tat, M. (2017). SWOT Analysis: A Theoretical Review. *The Journal of International Social Research*, *10*, 994–1006.

Hutchinson, J. E., Whittle, J., & Rouncefield, M. (2014). Model-driven engineering practices in industry: Social, organizational and managerial factors that lead to success or failure. *Sci. Comput. Program.*, *89*, 144–161.

Hutchinson, J. E., Whittle, J., Rouncefield, M., & Kristoffersen, S. (2011). Empirical assessment of MDE in industry. In R. N. Taylor, H. C. Gall, & N. Medvidovic (Eds.), *Proceedings of the 33rd International Conference on Software Engineering, ICSE 2011, Waikiki, Honolulu , HI, USA, May 21-28, 2011* (pp. 471–480). ACM.

*ISO/IEC 25010:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models* (Standard). (2011). Retrieved from https://www.iso.org/standard/35733.html

Kahraman, C., Demirel, N. Ç., & Demirel, T. (2007). Prioritization of e-Government strategies using a SWOT-AHP analysis: the case of Turkey. *European Journal of Information Systems*, *16*(3), 284–298.

Khandoker, A., Sint, S., Gessl, G., Zeman, K., Jungreitmayr, F., Wahl, H., ... Kretschmer, R. (2022). Towards a logical framework for ideal mbse tool selection based on discipline specific requirements. *Journal of Systems and Software*, *189*, 111306.

Kurttila, M., Pesonen, M., Kangas, J., & Kajanus, M. (2000). Utilizing the analytic hierarchy process (AHP) in SWOT analysis — A hybrid method and its application to a forest-certification case. *Forest Policy and Economics*, *1*(1), 41-52.

Liebel, G., Marko, N., Tichy, M., Leitner, A., & Hansson, J. (2018). Model-based engineering in the embedded systems domain: an industrial survey on the state-of-practice. *Softw. Syst. Model.*, *17*(1), 91–113.

Lopez, V., & Akundi, A. (2022). A conceptual model-based systems engineering (mbse) approach to develop digital twins. In *2022 ieee international systems conference (syscon)* (p. 1-5).

Lotz, J., Vogelsang, A., Benderius, O., & Berger, C. (2019). Microservice architectures for advanced driver assistance systems: A case-study. In *2019 IEEE International Conference on Software Architecture Companion (ICSA-C)* (pp. 45–52).

Luna, E. R., Sánchez-Begínes, J. M., Rivero, J. M., Morales-Trujillo, L., Enríquez, J. G., & Rossi, G. (2018). Challenges for the Adoption of Model-Driven Web Engineering Approaches in Industry. *J. Web Eng.*, *17*(3&4), 183–205.

Mehmood, F., Hassannezhad, M., & Abbas, T. (2014). Analytical investigation of mobile NFC adaption with SWOT-AHP approach: A case of Italian Telecom. *Procedia technology*, *12*, 535–541.

Morin, B., Harrand, N., & Fleurey, F. (2017). Model-Based Software Engineering to Tame the IoT Jungle. *IEEE Softw.*, *34*(1), 30–36.

NTTDATA. (2021). *Low-Code Analysis - New paradigm in software development.* Retrieved from https://es.nttdata.com/documents/en_low_code.pdf. (accessed 04.05.2022)

Panach, J. I., Dieste, O., Marín, B., España, S., Vegas, S., Pastor, O., & Juristo, N. (2021). Evaluating Model-Driven Development Claims with Respect to Quality: A Family of Experiments. *IEEE Trans. Software Eng.*, *47*(1), 130–145.

Penzel, D., Kryvinska, N., Strauss, C., & Gregu, M. (2015). The future of cloud computing: A swot analysis and predictions of development. In *2015 3rd International Conference on*

*Future Internet of Things and Cloud* (pp. 391–397).
Saaty, T. L. (1990). How to make a decision: The analytic hierarchy process. *European journal of operational research*, *48*(1), 9–26.
Saaty, T. L., & Vargas, L. G. (2012). The seven pillars of the analytic hierarchy process. In *Models, methods, concepts & applications of the analytic hierarchy process* (pp. 23–40). Springer.
Selic, B. (2008). Personal reflections on automation, programming culture, and model-based software engineering. *Autom. Softw. Eng.*, *15*(3-4), 379–391.
Selic, B. (2016). Programming ⊂ Modeling ⊂ Engineering. In T. Margaria & B. Steffen (Eds.), *Leveraging Applications of Formal Methods, Verification and Validation: Discussion, Dissemination, Applications - 7th International Symposium, ISoLA 2016, Imperial, Corfu, Greece, October 10-14, 2016, Proceedings, Part II* (Vol. 9953, pp. 11–26).
Shahir, H. Y., Daneshpajouh, S., & Ramsin, R. (2008). Improvement strategies for agile processes: a SWOT analysis approach. In *2008 Sixth International Conference on Software Engineering Research, Management and Applications* (pp. 221–228).
Störrle, H. (2017). How are Conceptual Models used in Industrial Software Development?: A Descriptive Survey. In E. Mendes, S. Counsell, & K. Petersen (Eds.), *Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering, EASE 2017, Karlskrona, Sweden, June 15-16, 2017* (pp. 160–169). ACM.
Taleai, M., Mansourian, A., & Sharifi, A. (2009). Surveying general prospects and challenges of GIS implementation in developing countries: A SWOT–AHP approach. *Journal of Geographical Systems*, *11*(3), 291–310.
Tang, N.-H., Shao, J., & Lee, Y.-C. (2013). The SWOT-AHP Framework for the Enterprise Cloud Computing Strategy. *Journal of Information Systems*, *22*, 85-104.
Teppola, S., Parviainen, P., & Takalo, J. (2009). Challenges in Deployment of Model Driven Development. In K. Boness, J. M. Fernandes, J. G. Hall, R. J. Machado, & R. Oberhauser (Eds.), *The Fourth International Conference on Software Engineering Advances, ICSEA 2009, 20-25 September 2009, Porto, Portugal* (pp. 15–20). IEEE Computer Society.
Vallecillo, A. (2015). On the industrial adoption of model driven engineering. Is your company ready for MDE? *International Journal of Information Systems and Software Engineering for Big Companies*, *1*(1), 52–68.
Vallecillo, A., Koch, N., Cachero, C., Comai, S., Fraternali, P., Garrigós, I., … Zhang, G. (2007). MDWEnet: A Practical Approach to Achieving Interoperability of Model-Driven Web Engineering Methods. In N. Koch, A. Vallecillo, & G. Houben (Eds.), *Proceedings of the 3rd International Workshop on Model-Driven Web Engineering MDWE 2007, Como, Italy, July 17, 2007* (Vol. 261). CEUR-WS.org.
Weihrich, H. (1982). The TOWS Matrix—A tool for Situational Analysis. *Long Range Planning*, *15*, 54–66.
Whittle, J., Hutchinson, J. E., Rouncefield, M., Burden, H., & Heldal, R. (2013). Industrial Adoption of Model-Driven Engineering: Are the Tools Really the Problem? In A. Moreira, B. Schätz, J. Gray, A. Vallecillo, & P. J. Clarke (Eds.), *Model-Driven Engineering Languages and Systems - 16th International Conference, MODELS 2013, Miami, FL, USA, September 29 - October 4, 2013. Proceedings* (Vol. 8107, pp. 1–17). Springer.
Wymore, A. W. (1993). *Model-Based Systems Engineering*. CRC Press.

## About the authors

**Maria Jose Escalona** is a full professor in the Department of Computer Languages and Systems at the University of Seville (Spain). She manages the Engineering and Science for Software Systems research group. Her current research interests include the areas of requirements engineering, web system development, model-driven engineering, early testing and quality assurance. She has extensive experience in research and also in research transferred to industry. You can contact the author at mjescalona@us.es.

**Nora Parcus de Koch** is a collaborator of the University of Seville, Spain. She formerly worked at the Ludwig-Maximilians-Universität (LMU) München and Fortiss (TUM), and in parallel as a consultant and project manager at F.A.S.T. and NTT DATA GmbH, Germany. Her main research interests include meta-modelling, requirements engineering, model-based software development and web engineering. You can contact the author at norakoch@us.es.

**Gustavo Rossi** is full professor at Facultad de Informática, in the Universidad Nacional de La Plata and a researcher at CONICET, Argentina. He is a member of the editorial board of several journals (IEEE Internet Computing, IEEE IT Professional, Journal of Web Engineering, and others). His current research interests include requirements engineering, web engineering and human-computer interaction. You can contact him at gustavo@lifia.info.unlp.edu.ar.